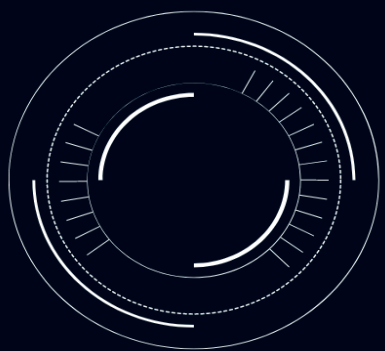
PROGRESS IN POLISH ARTIFICIAL INTELLIGENCE RESEARCH 4

Edited by:
Adam Wojciechowski and Piotr Lipiński



Lodz University of Technology Monograph
Łódź 2023

Lodz University of Technology, Poland
Faculty of Technical Physics, Information Technology and Applied Mathematics
Institute of Information Technology

Progress in Polish Artificial Intelligence Research 4

Edited by:
Adam Wojciechowski and Piotr Lipiński

Lodz University of Technology Monograph
Łódź 2023

Editors:
Adam Wojciechowski
Piotr Lipiński

Reproduced from materials supplied by the Editors.

ISBN: 978-83-66741-92-8

DOI: 10.34658/9788366741928

Publikacja dofinansowana przez Ministerstwo Edukacji i Nauki w ramach programu: "Doskonała Nauka II – Wsparcie Konferencji Naukowych".

Łódź University of Technology Press
93-005 Łódź, 223 Wólczańska St.
E-mail: zamowienia@info.p.lodz.pl
www.wydawnictwo.p.lodz.pl

Monografie Politechniki Łódzkiej, Nr 2437

Reviewed by:

Weronika Adrian
Michał Baczynski
Janusz Będkowski
Dominik Belter
Urszula Bentkowska
Szymon Bobek
Aleksander Byrski
Leszek Chmielewski
Michal Choraś
Sebastian Cygert
Ireneusz Czarnowski
Kamil Deja
Rafał Doroz
Paweł Drygaś
Włodzisław Duch
Agnieszka Duraj
Maksym Figat
Paweł Forczmanski
Dariusz Frejlichowski
Krzysztof Gajowniczek
Maciej Grzenda
Andrzej Janusz
Agnieszka Jastrzębska
Joanna Jaworek-Korjakowska
Przemysław Juszczuk
Jan Karwowski
Włodzimierz Kasprzak
Jakub Klikowski
Krzysztof Kluza
Anna Kobusińska
Jan Kocoń
Joanna Komorniczak
Tomasz Kornuta
Jan Kozak
Marek Kraft
Dariusz Król
Tomasz Kryjak
Paweł Ksieniewicz
Marcin Kurdziel
Krzysztof Kutt
Bogdan Kwolek
Agnieszka Ławrynowicz
Agnieszka Lazarowska
Krzysztof Lichy
Piotr Lipiński
Marcin Luckner
Ewa Łukasik
Tomasz Łukaszewski
Karol Majek
Andrzej Materka
Magdalena Mazur-Milecka
Konrad Miazga
Agnieszka Mikołajczyk-Bareła
Piotr Milczarski
Katarzyna Miś
Wojciech Moczulski
Mateusz Modrzejewski
Michał Morawski
Teresa Mroczek
Grzegorz J. Nalepa
Piotr Napieralski
Mariusz Nieniewski
Adam Niewiadomski
Mariusz Nowak
Przemysław Nowak
Robert Nowak
Agnieszka Nowak-Brzezińska
Michał Nowicki
Joanna Ochelska-Mierzejewska
Krzysztof Okarma
Michał Okulewicz
Karol Opara
Tomasz Orczyk
Mete Ozay
Maciej Patan
Barbara Pekala
Paweł Pełka
Piotr Pęzik
Maciej Piasecki
Leszek Podsędkowski
Jedrzej Potoniec
Marcin Przewięźlikowski
Dariusz Puchała
Przemyslaw Rokita
Jan Sawicki
Rafal Scherer
Marek Sikora
Paweł Skruch
Piotr Skrzypczyński
Maria Skublewska-Paszkowska
Dominik Ślęzak
Andrzej Śluzek
Bogdan Smołka
Bartłomiej Stasiak
Maciej Stefańczyk
Jerzy Stefanowski
Kamil Stokfiszewski
Michał Strzelecki
Maciej Świechowski
Dominik Szajerman
Marta Szarmach
Piotr Szczepaniak
Marcin Szeląg
Julian Szymański
Piotr Szymański
Arkadiusz Tomczyk
Tomasz Trzciński
Krzysztof Walas
Jakub Walczak
Tomasz Walkowiak
Jarosław Wąs
Piotr Wasilewski
Dawid Wiśniewski
Konrad Wojciechowski
Adam Wojciechowski
Szymon Wojciechowski
Konrad Wojtasik
Agnieszka Wosiak
Michal Woźniak
Teresa Zielińska
Cezary Zieliński
Beata Zielosko
Paweł Zyblewski
Adam Żychowski

Technical editor:

Dominik Szajerman

Preface

The multi-author monograph entitled "Progress in Polish Artificial Intelligence Research 4" is a comprehensive overview of research results, related to the broadly defined artificial intelligence, of the academic community and business community affiliated with Polish universities and selected foreign research centres. The thematic scope of the monograph fits perfectly into the strategic areas defined in the Regulation of the European Parliament and the Council of Europe establishing the Digital Europe program for 2021-2027:

- the development of research on artificial intelligence, including the issue of public data and its effective use as a raw material for the development of AI in the public and private sectors, the application of AI in key areas such as medicine, environmental sciences, transportation, security, and the development of the latest technologies (including deep learning, robotization and cobotization) in conjunction with AI;
- development of advanced digital competencies among students, graduates and academics in artificial intelligence and cyber security.

The dynamic development of artificial intelligence has resulted not only in the specialization of research, but also in an above-average interest in the topic by researchers from non-IT disciplines. The present monograph acts as a bridge integrating the achievements of the Polish community focused on the development of various branches of artificial intelligence and opening a dialogue between AI specialists and other scientific fields and disciplines.

Such a cross-sectional collection of achievements, collected in the monograph, is due to many people and institutions. It should be noted that the inspiration for the monograph came from the Polish Alliance for the Development of Artificial Intelligence (PP-RAI), which is the collaboration of the Polish Artificial Intelligence Society, the Polish Neural Networks Society, the Polish Special Interest Group on Machine Learning, the Polish Chapter of the IEEE Systems, Man, and Cybernetics Society, and the Polish Chapter of the IEEE Computational Intelligence Society. The PP-RAI Steering Committee, comprising Prof. Ireneusz Czarnowski, Prof. Włodzisław Duch, Prof. Janusz Kacprzyk, Prof. Jacek Koronacki, Prof. Halina Kwaśnicka, Prof. Jacek Mańdziuk, Prof. Grzegorz J. Nalepa, Prof. Leszek Rutkowski, Prof. Rafał Scherer, Prof. Jerzy Stefanowski, Prof. Dominik Ślęzak and Prof. Michał Woźniak, is very much credited in this field.

Due to its thematic scope, the monograph has been divided into 9 chapters on the following 11 domains:

- Computer Vision (domain edited by Prof. Leszek Chmielewski, Prof. Krzysztof Gajowniczek);
- Data Mining and Machine Learning (domain edited by Prof. Jerzy Stefanowski, Prof. Michal Woźniak, Prof. Ireneusz Czarnowski);
- Interdisciplinary Applications of Artificial Intelligence (domain edited by Prof. Jacek Mańdziuk, Prof. Agnieszka Ławrynowicz, Prof. Jarosław Wąs, Prof. Adam Wojciechowski);

- Knowledge Engineering (domain edited by Prof. Agnieszka Ławrynowicz, Prof. Dariusz Król, Prof. Grzegorz J. Nalepa);
- Medical Applications of Artificial Intelligence (domain edited by Prof. Włodzisław Duch, Prof. Julian Szymański, Dr. Marian Bubak);
- Natural Language Processing, Automatic Speech Recognition, Conversational AI (domain edited by Prof. Maciej Piasecki, Prof. Agnieszka Mykowiecka, Prof. Piotr Pęzik);
- Neural Network and Deep Learning Systems (domain edited by Prof. Aleksander Byrski, Prof. Marcin Kurdziel);
- Problem Solving and Optimization (domain edited by Prof. Jarosław Arabas, Prof. Karol Opara, Prof. Robert Nowak);
- Robotics and Autonomous Systems (domain edited by Prof. Piotr Skrzypczyński, Prof. Piotr Lipiński);
- Uncertainty in Artificial Intelligence (domain edited by Prof. Dominik Ślęzak, Prof. Beata Zielosko, Dr. Piotr Wasilewski);
- Young.AI (domain edited by Dr. Arkadiusz Tomczyk, Dr. Jakub Walczak, and Stanisław Kaźmierczak, MSc.).

Chapters cover the latest achievements of both young and experienced researchers, scientific teams and consortia. Each chapter is composed of subsections (sections) supervised by dedicated groups of authorities, further enhancing the diversity of viewpoints of each research area. Some of the topics have been the subject of scientific research projects funded by external sources or research assignments for industry, which may cause a paucity of details among readers, but is due to the protection of intellectual property enforced by the need to commercialize achievements.

In summary, we offer you an unique monograph that brings together, as in a lens, the problems undertaken by Polish and foreign scientists and entrepreneurs, as well as the results of research that can be an inspiration for students, doctoral students or scientists with an academic or business background.

Contents

Chapter 1

Computer Vision

Domain Editors:

1. Leszek Chmielewski, Warsaw University of Life Sciences
2. Krzysztof Gajowniczek, Warsaw University of Life Sciences

3D Reconstruction of Non-Visible Surfaces of Objects from a Single Depth View – Comparative Study

Rafał Staszak[1][0000−0002−5235−4201],
Piotr Michałek[1][0009−0003−9139−4665],
Jakub Chudziński[1][0000−0001−8228−0197],
Marek Kopicki[1][0000−0002−0769−0556],
Dominik Belter[1][0000−0003−3002−9747]

[1]*Poznan University of Technology*
Institute of Robotics and Machine Intelligence
ul. Piotrowo 3A, 60-965 Poznań, Poland
name.surname@put.poznan.pl

DOI:10.34658/9788366741928.1

Abstract. *Scene and object reconstruction is an important problem in robotics, in particular in planning collision-free trajectories or in object manipulation. This paper compares two strategies for the reconstruction of non-visible parts of the object surface from a single RGB-D camera view. The first method, named DeepSDF predicts the Signed Distance Transform to the object surface for a given point in 3D space. The second method, named MirrorNet reconstructs the occluded objects' parts by generating images from the other side of the observed object. Experiments performed with objects from the ShapeNet dataset, show that the view-dependent MirrorNet is faster and has smaller reconstruction errors in most categories.*
Keywords: *robotics, scene reconstruction, neural scene representation*

1. Introduction

Robots observing the scene utilize onboard RGB-D cameras to collect information about the shape of the objects. However, the full geometry of the scene cannot be registered from a single view due to occlusions. Some methods for grasping objects deal with incomplete data and perform well even though the full 3D model is unknown [1]. In this research, we are focused on the solutions that directly reconstruct the entities on the scene. The example object reconstruction scenario is presented in Fig. 1. The object reconstruction method that can be applied in robotics should be capable of reconstructing a full model of an entity

observed from a single RGB-D camera view. The features extracted by the considered solutions are potentially valuable for other robotics tasks e.g. grasping [2].

Multiple scene reconstruction techniques utilize 3D grids [3, 4] but these methods suffer from resource consumption growth when the resolution of the model is increased. Recently the Neural Scene Representation model based on Radiance Fields (NeRF) has been proposed [5]. This solution has superior accuracy but is designed for generating images from various viewpoints for static scenes. This property limits the possible applications in robotics. In contrast, DeepSDF [6] can be used to reconstruct various objects from a single view and represent them as a Signed Distance Transform (SDF). Other methods are designed to generate depth images of the observed objects from various viewpoints [7]. Thus, the obtained images are used to reconstruct a full 3D model of the entity. In this paper, we compare the view-dependent approach based on image generation named MirrorNet [7] with the SDF-based neural representation operating directly in the continuous 3D space [6].

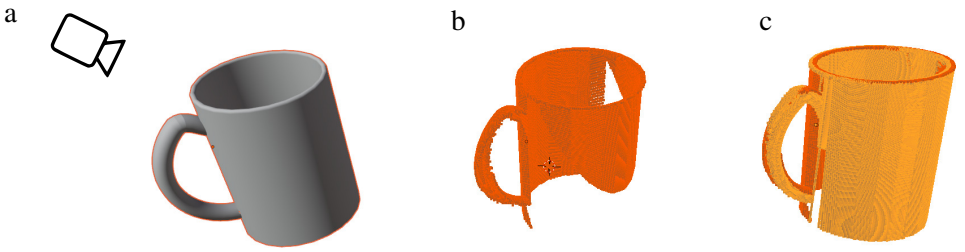


Figure 1. Example application scenario of the objects reconstruction system: the robot observes a 3D object from a single view (a). The incomplete model of the object (point cloud) (b) is provided to the input of the neural network to obtain a full model of the object (c). Source: own work.

2. Comparison between DeepSDF and MirrorNet

In this paper, we compare two representative approaches for scene reconstruction. The DeepSDF method [6] utilizes a fully-connected neural network to predict the Signed Distance Transform to the surface of the object for the given point in the 3D space. During the inference, the latent space, that describes the object, is estimated from the partial view. Then, the object is reconstructed by sampling the 3D space and generating new points. The second method proposed in [7] reconstructs the occluded parts of the objects by generating images from the other side of the observed object. It utilizes the depth image from the given position of the camera and the depth image obtained by projecting the input point cloud to the virtual position of the camera. In this case, a Convolutional Neural Network is

Table 1. Comparison of the reconstructed results obtained for the view-dependent model (MirrorNet) [7] and DeepSDF [6].

method	metric	bottle	can	helmet	jar	laptop	mug
MirrorNet	d_{C} [m]	**0.06836**	**0.1402**	0.1156	**0.0749**	0.1967	0.1843
	d_{H} [m]	**0.0997**	**0.1647**	**0.1673**	**0.1257**	**0.2606**	**0.2176**
DeepSDF	d_{C} [m]	0.09764	0.1486	**0.0724**	0.0803	**0.1437**	**0.1125**
	d_{H} [m]	0.3143	0.4162	0.3098	0.3071	0.4062	0.3596

used to generate depth images.

Both methods are designed to operate in slightly different conditions. DeepSDF requires depth observations in the canonical shape frame of reference. The MirrorNet does not have this limitation but directly utilizes noisy depth camera images from the real robot [7]. To compare both methods in the same condition, we select 6 representative categories of graspable entities from the ShapeNet dataset [8]. For each category, we selected 30 instances of objects to prepare the training datasets. To train the DeepSDF, we utilize a 3D mesh model of the objects located in the global frame and scaled to fit the unit sphere. To generate training data for the MirrorNet, we collect images generated for random positions around the object. For testing, we use randomly generated views of objects for another 10 instances of objects from the categories that were used for training.

3. Results

The obtained results are presented in Tab. 1. We utilize Chamfer d_C and Hausdorff d_H distances [9] to quantitatively evaluate the reconstruction results in 3D space. Both systems return similar results regarding the Chamfer distance d_C. Despite the visually better reconstruction of the objects by the MirrorNet, the obtained 3D model does not cover the whole surface which results in worse numerical results for the Chamfer distance. When the Hausdorff distance d_H is compared the model returned by the MirrorNet is 2-3 times better than the model given by the DeepSDF.

The example reconstruction results obtained from the MirrorNet and DeepSDF are presented in Fig. 2. The first visible difference is the number of points generated by the algorithms which is much smaller for the MirrorNet. The DeepSDF can generate higher number of points but the generation time significantly increases to 15.04 seconds per object (using GPU RTX 3060). For better results with DeepSDF, we increased the number of points drawn in preprocessing from 250,000 to 5 million and introduced a threshold for negative SDF values. The introduction of the threshold was necessary because the algorithm had problems calculating SDF values for an incomplete object mesh from a single view. DeepSDF fails to recover the

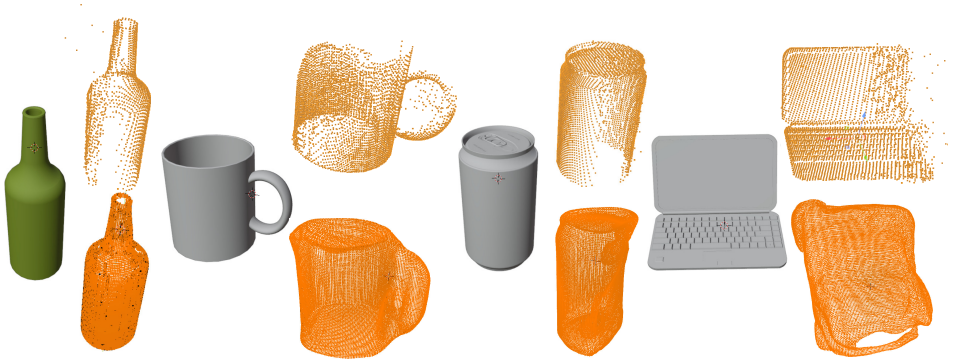


Figure 2. Example reconstruction results (point clouds) obtained for the view-dependent MirrorNet (top row) and DeepSDF (bottom row) compared to the ground truth models: bottle, mug, can, and laptop. Source: own work.

handle of the mug and the laptop. Also, DeepSDF does not preserve the rounded shape of the can. The view-dependent MirrorNet is fully convolutional so it generates images in about 22 milliseconds. This method preserves the shape of the objects but some surfaces of the objects are not reconstructed because they are not observed from the input and generated camera view. Also, MirrorNet generates random sparse points between the reconstructed surface and the camera pose. However, these points can be easily removed using voxel-based filters.

4. Conclusions

In this paper, we compare two neural network-based models that reconstruct the model of the objects from a single camera image. We have chosen the MirrorNet which generates the depth image of the object from the opposite pose of the camera and DeepSDF which operates directly in the 3D space. Our experiments show that the view-dependent approach returns more accurate reconstruction results. Moreover, the view-dependent approach is significantly faster than DeepSDF (22 ms for inference using MirrorNet and 15000 ms for DeepSDF) which requires optimization and 3D space sampling during the inference.

In the future, we are going to extract the features from the view-dependent models of the objects and use them for efficient grasping and manipulating objects from a single camera view.

Acknowledgment

This research is part of the project No. 2021/43/P/ST6/01921 co-funded by the National Science Centre and the European Union Framework Programme for

Research and Innovation Horizon 2020 under the Marie Skłodowska-Curie grant agreement No. 945339. R. Staszak and D. Belter were supported by the National Science Centre, Poland, under research project no UMO-2019/35/D/ST6/03959.

References

[1] Kopicki M.S., Belter D., Wyatt J.L., *Learning better generative models for dexterous, single-view grasping of novel objects*, *The International Journal of Robotics Research*, 2019, vol. 38, no 10–11, pp. 1246–1267, doi: 10.1177/0278364919865338.

[2] Weng T., Held D., Meier F., Mukadam M., *Neural grasp distance fields for robot manipulation*, 2022, doi: 10.48550/ARXIV.2211.02647.

[3] Choy C.B., Xu D., Gwak J., Chen K., Savarese S., *3D-R2N2: A unified approach for single and multi-view 3D object reconstruction*, [In:] B. Leibe, J. Matas, N. Sebe, M. Welling (eds.), *Computer Vision – ECCV 2016*, Springer International Publishing, Cham, pp. 628–644.

[4] Popov S., Bauszat P., Ferrari V., *Corenet: Coherent 3d scene reconstruction from a single RGB image*, [In:] A. Vedaldi, H. Bischof, T. Brox, J.M. Frahm (eds.), *Computer Vision – ECCV 2020*, Springer International Publishing, Cham, pp. 366–383.

[5] Mildenhall B., Srinivasan P.P., Tancik M., Barron J.T., Ramamoorthi R., Ng R., *NeRF: Representing scenes as neural radiance fields for view synthesis*, [In:] A. Vedaldi, H. Bischof, T. Brox, J.M. Frahm (eds.), *Computer Vision – ECCV 2020*, Springer International Publishing, Cham, pp. 405–421.

[6] Park J.J., Florence P., Straub J., Newcombe R., Lovegrove S., *DeepSDF: Learning continuous signed distance functions for shape representation*, [In:] *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 165–174.

[7] Staszak R., Kulecki B., Sempruch W., Belter D., *What's on the other side? a single-view 3D scene reconstruction*, [In:] *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 173–180.

[8] Chang A.X., Funkhouser T., Guibas L., Hanrahan P., Huang Q., Li Z., Savarese S., Savva M., Song S., Su H., Xiao J., Yi L., Yu F., *ShapeNet: An information-rich 3D model repository*, *ArXiv*, 2015, vol. abs/1512.03012.

[9] Lebrat L., Cruz R.S., Fookes C., Salvado O., *Mongenet: Efficient sampler for geometric deep learning*, [In:] *2021 IEEE/CVF Conference on Computer*

Vision and Pattern Recognition (CVPR), pp. 16659–16668, doi: 10.1109/CVPR46437.2021.01639.

Challenges of Crop Classification from Satellite Imagery with Eurocrops Dataset

Przemysław Aszkowski[0000−0003−0388−8546], **Marek Kraft**[1][0000−0001−6483−2357]

Faculty of Control, Robotics and Electrical, Engineering,
Institute of Robotics and Machine Intelligence
Poznań University of Technology,
60-965 Poznań, Poland
przemyslaw.aszkowski@doctorate.put.poznan.pl,
marek.kraft@put.poznan.pl

DOI:10.34658/9788366741928.2

Abstract. *Crops monitoring and classification on a nationwide level provide important information for sustainable agricultural management, food security, and policy-making. Recent technological advancements, followed by Earth observation programmes like Copernicus, have provided plenty of publicly available multispectral data. Combining these data with field annotations allows for continuous crop monitoring from publicly available data. In this paper, we present a solution for crop classification to determine crop type from Sentinel-2 multispectral data, utilizing machine learning techniques. Apart from presenting initial results, we discuss the challenges of crop classification on a Eurocrops dataset and further research directions.*
Keywords: *computer vision, multispectral imaging, remote sensing, crop classification*

1. Introduction

Crop classification using satellite imagery has been gaining more research attention recently, mainly utilizing publicly available data from Sentinel satellites and Landsat program [1]. Due to the massive amount of multispectral data, as well as the background of researchers, manually engineered features with phenological information are especially popular, replacing raw multispectral bands data. Authors in [2] combine data from heterogeneous sources: radar images from Sentinel-1 and multispectral images from Sentinel-2, with non-weighted accuracy reaching 0.85 for 23 crop types.

The research usually concentrates on a single country or even a small region, as the labelled data is scarce and diverse. Moreover, it usually covers only a small subset of cultivated crops that are especially relevant to the authors [3]. In this

paper, a publicly available Eurocrops [4] dataset is used, and its usefulness for crop classification is analysed. The Eurocrops dataset provides field annotations for European countries along with ready-to-use multispectral data for each parcel. Figure 1 shows a small part of the dataset fields visualised in QGIS.

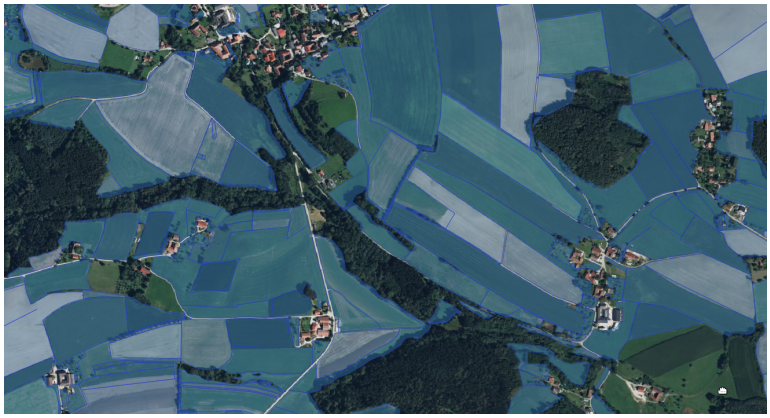


Figure 1. Fields from Eurocrops dataset with boundaries, visualized as semi-transparent blue polygons in QGIS on Bing Aerial Maps. Source: own work.

2. Materials and Methods

2.1. Dataset

Although the Sentinel-2 satellite has a resolution of 10 m and a revisits time of 10 days, the Eurocrops dataset provides for each field only a single representative pixel for each date, with 13 multispectral bands. Data are provided for each country individually. In this paper, due to the high amount of data, the training and testing focused on Austria, as data for this country were available at first, with 396600 annotated parcels in the training set, covering 44 crop types. The testing dataset is not a random subset of the training dataset, but it is a distinct geographic region of the country, specified in the Eurocrops dataset. The data provided by Eurocrops are from Sentinel-2 L1C products, which introduces significant bias due to the weather at a specific time point [5].

2.2. Data preprocessing

Dates when the images were taken are not consistent among the dataset. Therefore, before further processing, the data were resampled to common dates by taking the nearest data point for each date. Moreover, due to weather conditions, data are often obscured by clouds. Such data points could be potentially removed and

resampled, e.g. with spline interpolation through time. Apart from using the raw pixel data, a simple yet effective vegetation indicator NDVI (Normalized Difference Vegetation Index) [6] was utilized, which is calculated based on near-infrared and red band values (bands 8 and 4 respectively). Example NDVI curves for a few selected crop types are presented in Figure 2. Using this index alone does not contain all information required for classification but seems to usually improve the model results.

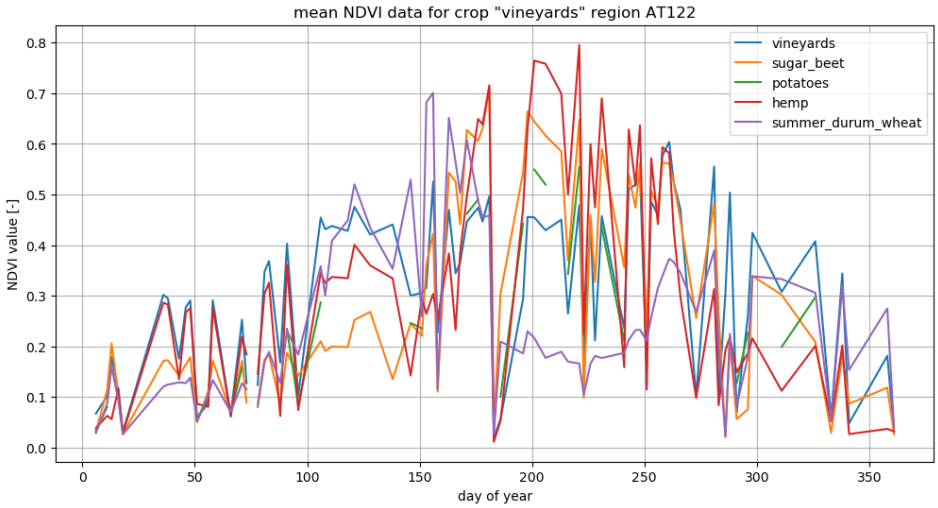


Figure 2. Average NDVI values through time for different crop types. One of the features used as model input, apart from the raw pixel values. Source: own work.

2.3. Algorithms

As there is no spatial information for a single parcel, the application potential for modern neural networks, including CNNs and transformers, is quite limited. Therefore, FCNN (Fully Connected Neural Network), as well as classical machine learning techniques, including SVM (Support Vector Machine) and random forest, were used. The input data consists of resampled and normalized pixel values with an additional NDVI feature. As it is a classification problem, the output data classes are different crop types. Due to high class imbalance, data were weighted by the number of samples in the training dataset.

2.4. Results

The best results on the testing set were achieved for FCNN, though the other methods were not significantly inferior. The best variant of the tested FCNN consists of four fully connected layers, with ReLU activation and batch normalization,

and trained with cross-entropy loss. Table 1) shows the metrics achieved by the compared methods.

Table 1. Metrics achieved by the FCNN model on the test dataset for Austria, for all 44 crop types

Method	**Accuracy**	**Weighted accuracy**	**Precision**	**Recall**
FCNN	0.715	0.620	0.715	0.715
SVM	0.691	0.613	0.691	0.691
Random forest	0.694	0.431	0.692	0.692

Most of the crop types can be easily distinguished, but some crops (e.g. nuts and leguminous plants) have an accuracy not better than a random guess. Fig. 3 shows the confusion matrix trained for a few randomly selected crop types, which yields significantly better results.

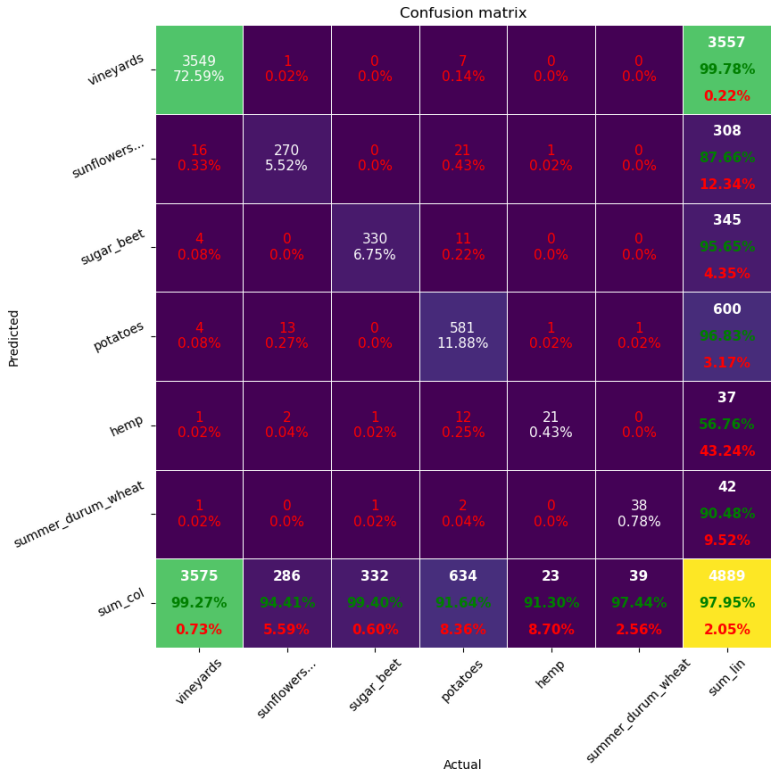


Figure 3. Confusion matrix for test dataset of crop classification, with a model trained only on a few selected crop types. To show high-class imbalance, the percentage of total predictions is shown in each cell (apart from the last row and column). Source: own work.

3. Conclusions

As presented in this paper, crop classification is achievable with the publicly available Eurocrops dataset, with weighted accuracy of 0.620 while training and testing on one country for all 44 crop types. Testing on fields from a different country than the training data yields significantly lower accuracy, possibly due to differences in climate, diverse vegetation processes, uncorrelated weather or even different subspecies for a specific crop, which is not covered by the dataset. Apart from including data from different countries in training, domain transfer with unlabeled data can be considered to tune the model to a different geographical region. The interoperability and accuracy of the model most probably could be improved by using Sentinel-2 L2A products with atmospheric correction and without cloudy data. Also, training the model to work only on a subset of crop types significantly improves the model's accuracy. Future work could involve analyzing all pixels within a parcel and not only a single representative pixel, allowing for spatial analysis. Another interesting research direction is the prediction from partial data without the full growing period.

Code available at https://github.com/PUTvision/crop_classification_ eurocrops.

References

[1] Asam S., Gessner U., Almengor González R., Wenzl M., Kriese J., Kuenzer C., *Mapping crop types of germany by combining temporal statistical metrics of sentinel-1 and sentinel-2 time series with lpis data*, *Remote Sensing*, 2022, vol. 14, no 13, ISSN 2072-4292, doi: 10.3390/rs14132981.

[2] Mleczko M., Slesiński P., Milewski T., Łączyński A., Miziołek D., Woźniak E., Bojanowski J., *Satelitarne rozpoznawanie upraw i szacowanie ich powierzchni w ramach systemu satmirol*, *Wiadomości Statystyczne. The Polish Statistician*, 2022, vol. 67, doi: 10.5604/01.3001.0015.9863.

[3] Sarvia F., Xausa E., De Petris S., Cantamessa G., Borgogno-Mondino E., *A possible role of copernicus sentinel-2 data to support common agricultural policy controls in agriculture*, *Agronomy*, 2021, vol. 11, no 1, ISSN 2073-4395, doi: 10.3390/agronomy11010110.

[4] Schneider M., Marchington C., Körner M., *Challenges and opportunities of large transnational datasets: A case study on european administrative crop data*, *Workshop on Broadening Research Collaborations in ML, NeurIPS*, 2022.

[5] Schneider M., Broszeit A., Körner M., *Eurocrops: A pan-european dataset for time series crop type classification*, *Proc. of the 2021 conference on Big Data from Space (BiDS21)*, 2021, pp. 125–128, doi: 10.2760/125905.

[6] Xue J., Su B., *Significant remote sensing vegetation indices: A review of developments and applications*, *Journal of sensors*, 2017, doi: 10.1155/2017/1353691.

Do We Always Need AI for Image Colorization?

Andrzej Śluzek

Warsaw University of Life Sciences-SGGW
Institute of Information Technology
Nowoursynowska 159, 02-776 Warsaw, Poland
andrzej_sluzek@sggw.edu.pl

DOI:10.34658/9788366741928.3

Abstract. *Monochrome image (re-)colorization is a problem where human guidance is considered indispensable (even if AI is used). We challenge this approach by presenting a method for converting grayscale images into credible color counterparts with no priors provided/learned on image content, semantics, etc. In contrast to our recent works (focusing on images from non-visual domains) this paper discusses mainly colorization of real-world images for which certain coloristic expectations exist. We argue (and show on selected examples) that a simple technique combining non-deterministic heuristics and randomized flood-fill can deliver colored outputs which are visually convincing and (occasionally) similar to the ground-truth images.*
Keywords: *image coloring, de-colorization, color models, flood-fill*

1. Introduction and Motivation

Monochrome image colorization is a topic of certain practical and commercial significance (e.g. restoration of legacy photos [1]). In general, development of colorization techniques is a process of incorporating more and more human knowledge/expectations into the algorithms [2]. Earlier, reference color images were provided [3], or important fragments manually *scribbled* [4]. Recently, AI-based techniques dominate, with architectures designed to learn color patterns suitable for specific domains, semantics and/or contents, e.g. [5, 6]. Recognition/learning the image domain (or specific objects) can further improve the results, e.g. [7, 8].

Some works consider inter-domain transfer learning [9] or produce alternative colorizations by learning probability densities [10].

Thus, the main idea of this paper seems untoward, as we attempt to produce convincing colorizations without any knowledge on domain, content, semantics, etc. of grayscale images (such an operation is, by definition, considered ill-posed).

Our recent papers [11, 12] focus on colorized insights into "gray worlds", i.e. images for which the color counterparts do not exist (e.g. IR or MRI domains). In here, the real-world images (where human experience/knowledge define certain

coloristic expectations) are the main topic of interest. It is shown that even in this domain the proposed "blind" colorization can produce realistically-looking results.

Section 2 briefly summarizes the background concepts and their specific use in the discussed problem. Section 3 provides exemplary experimental results with the relevant comments and explanations. Section 4 contains the concluding remarks.

2. Background Facts

The method is based on four pivotal concepts (fully described in [11, 12]):

1. It is assumed that b/w images are obtained from hypothetical color images by a predefined de-colorization (*rgb-to-gray*) model $I = k_R R + k_G G + k_B B$. The model can freely deviate from standard YUV values $[0.299, 0.587, 0.114]$, which results in diversified b/w images. Correspondingly, a b/w image is a projection of many (equivalent, though differently colored) images, Fig.1.

2. Given finite numbers of intensities and colors, each intensity corresponds to a number of colors (depending on how they satisfy the adopted *rgb-to-gray* model). The pools of colors are largest for mid-range intensities, but dwindle to single choices for extreme intensities (see Fig.2), i.e. the extreme-valued pixels would be uniquely colorized (or from very few options).

3. If an uncolored pixel of I intensity is neighbored by one or more already colored pixel(s), its color is randomly selected from the pool available for I, additionally considering the color(s) of neighbor(s) by using a straightforward heuristic: *The larger the intensity difference between adjacent pixels, the greater the chance that their colors will differ more.*

4. From **Point 2**, we get a number of uniquely colored pixels, which are considered the initial queue for *flood-fill* method. The remaining pixels are colored using a randomized version of *flood-fill*, i.e. the next pixel to be processed is randomly selected from the que of active pixels. Colors are assigned randomly, with the probabilities shaped by the **Point 3** heuristic.

(a) (b) (c) (d) (e) (f) (g)

Figure 1. Three b/w versions (b,c,d) of a color image (a) for various *rgb-to-gray* models. Three (hypothetically perfect) re-colorizations of (d) image are in (e,f,g). Source: own work.

Despite the random factors, the method produces surprisingly repeatable and visually attractive results (especially if means from a few runs of colorization are

used, to suppress local coloristic spikes). Additionally, the rendered colors can be projected onto the corresponding YUV planes $I = 0.299R + 0.587G + 0.114B$ so that the human perception of brightness is preserved, see Fig.3.

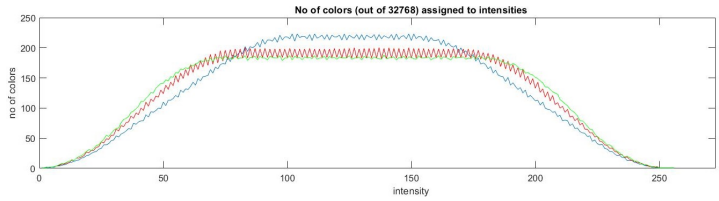


Figure 2. Numbers of colors assigned to 256 intensities for $[0.299, 0.587, 0.114]$, $[0.67, 0.12, 0.21]$ and $[0.13, 0.172, 0.698]$ *rgb-to-gray* models. Source: own work.

Obviously, the results depend on the adopted *rgb-to-gray* model. In Fig.3 example, the model is irrelevant since the image is IR, and its "true" colors simply do not exist (i.e. any plausible results can be accepted). For real-world images, however, we would expect results that agree with our "coloristic experiences".

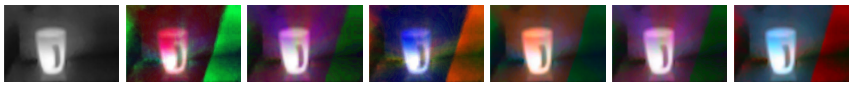


Figure 3. An IR monochrome image and its colorizations by three *rgb-to-gray* models. The results are either before (left) or after (right) the YUV projections. Source: own work.

3. Experiments with Real-world Images

The brute-force approach is to test a large number of *rgb-to-gray* models. Then, the most plausible outcomes can be identified (e.g. by visual inspection). We, therefore, sampled the $[k_R, k_G, k_B]$ space from $[0.05, 0.05, 0.9]$ (with 0.05 increment) to $[0.9, 0.05, 0.05]$, so that 171 *rgb-to-gray* models are built.

Given b/w images and their "true" colors (we select SUN dataset [13] used in other related works, e.g. [5]) the best *rgb-to-gray* models can be identified by the minimum RMSE (and overall visual credibility). Examples are shown in Fig.4.

If "true" colors are not known (which is the typical case), visual inspection of all 171 images is (technically) the only way to find the most plausible colorization. However, we have found that in most tested examples the best results are within 10% of images with the lowest *colorfulness* (details of this metric are in [2]). Thus,

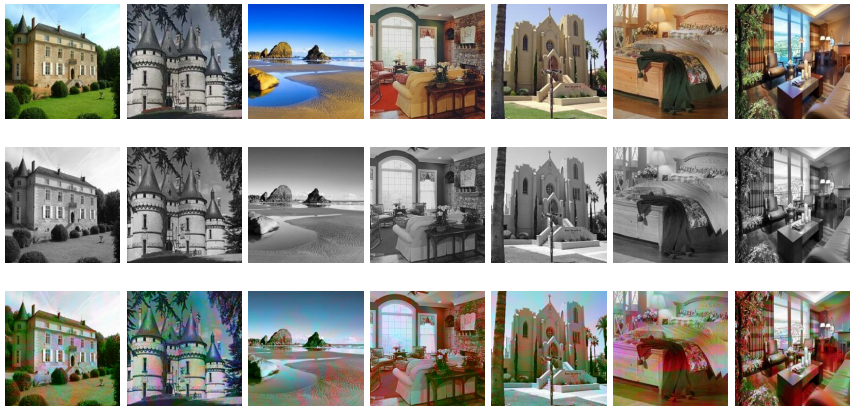


Figure 4. Exemplary original images (first row), their grayscale versions (second row) and the most plausible re-colorizations by the proposed method (last row). Source: own work.

the recommended colorization can be found by inspecting just 17 – 18 images. Selected results (compared to *state-of-the-art* results by [1]) are shown in Fig.5.

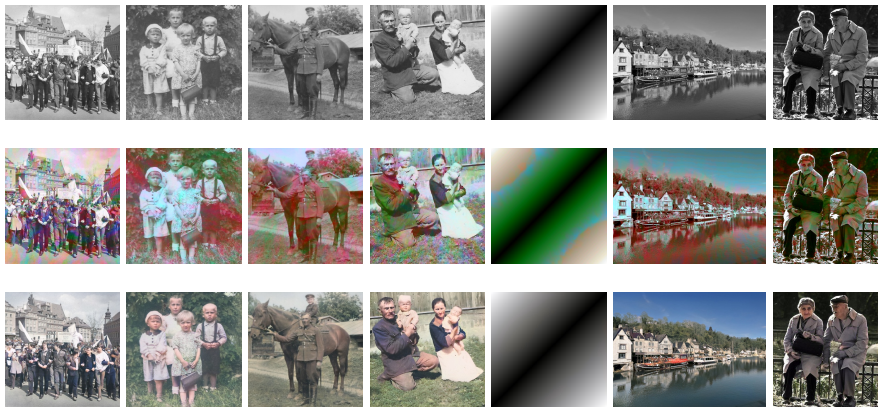


Figure 5. B/w images (top), plausible results by our method (middle), and results by DeOldify, https://deepai.org/machine-learning-model/colorizer (bottom row). Source: own work.

4. Concluding Remarks

In this paper, a colorization method is presented which combines just *one-pixel-wide* context and simple heuristics. No whatsoever learning or knowledge on image domains or contents is required. It is shown that such a basic approach can in some respects compete with *state-of-the-art* DL architectures trained on tens of millions of images. Thus, the question can be asked again: *Do we always need AI in monochrome image colorization*?

NOTE: ***All figures are best viewed in color and high resolution***.

References

[1] Salmona A., Bouza L., Delon J., *Deoldify: A review and implementation of an automatic colorization method*, *Image Processing On Line*, 2022, vol. 12, pp. 347–368, doi: 10.5201/ipol.2022.403.

[2] Zeger I., Grgic S., Vukovic J., Sisul G., *Grayscale image colorization methods: Overview and evaluation*, *IEEE Access*, 2021, vol. 9, pp. 113326–113346, doi: 10.1109/ACCESS.2021.3104515.

[3] Irony R., Cohen-Or D., Lischinski D., *Colorization by example*, [In:] *Eurographics Symposium on Rendering (2005)*, The Eurographics Association, ISBN 3-905673-23-1, ISSN 1727-3463, doi: 10.2312/EGWR/EGSR05/201-210.

[4] Levin A., Lischinski D., Weiss Y., *Colorization using optimization*, *ACM Transactions on Graphics*, 2004, vol. 23, pp. 689–694, doi: 10.1145/1015706.1015780.

[5] Zhang R., Isola P., Efros A., *Colorful image colorization*, [In:] *Computer Vision – ECCV 2016*, Springer, pp. 649–666, doi: 10.1007/978-3-319-46487-9_40.

[6] Farella E., Malek S., Remondino F., *Colorizing the past: Deep learning for the automatic colorization of historical aerial images*, *Journal of Imaging*, 2022, vol. 8, p. 269, doi: 10.3390/jimaging8100269.

[7] Iizuka S., Simo-Serra E., Ishikawa H., *Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification*, *ACM Transactions on Graphics*, 2016, vol. 35, pp. 1–11, doi: 10.1145/2897824.2925974.

[8] Su J., Chu H., Huang J., *Instance-aware image colorization*, [In:] *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7965–7974, doi: 10.1109/CVPR42600.2020.00799.

[9] Lee H., Kim D., Lee D., Kim J., Lee J., *Bridging the domain gap towards generalization in automatic colorization*, [In:] *Computer Vision – ECCV 2022*, Springer, pp. 527–543, doi: 10.1007/978-3-031-19790-1_32.

[10] Royer A., Kolesnikov A., Lampert C., *Probabilistic image colorization*, [In:] *Proc. British Machine Vision Conference (BMVC)*, BMVA Press, pp. 85.1–85.12, doi: 10.5244/C.31.85.

[11] Śluzek A., *On unguided automatic colorization of monochrome images*, submitted to WSCG'2023.

[12] Śluzek A., *Monochrome image colorization using de-colorization models*, submitted to ICIP'2023.

[13] Xiao J., Hays J., Ehinger K., Oliva A., Torralba A., *Sun database: Large-scale scene recognition from abbey to zoo*, [In:] *2010 IEEE Conference CVPR*, pp. 3485–3492, doi: 10.1109/CVPR.2010.5539970.

Fault Diagnosis in a Squirrel-Cage Induction Motor Using Thermal Imaging

Mateusz Piechocki[1][0000−0002−3479−0237], **Marek Kraft**[1][0000−0001−6483−2357], **Tomasz Pajchrowski**[1][0000−0002−0002−7161]

[1]*Poznan University of Technology*
Institute of Robotics and Machine Intelligence
Piotrowo 3A, 60-965 Poznań, Poland
{mateusz.piechocki, marek.kraft, tomasz.pajchrowski}@put.poznan.pl

DOI:10.34658/9788366741928.4

Abstract.
Fault diagnosis is a vivid topic in industrial applications or intelligent building solutions. One of the well-established techniques involves the measurement and analysis of current signals. However, this method has several significant drawbacks, such as the inability to inspect during machinery operation or the lack of precise information on the malfunction location. This article proposes a non-invasive method for squirrel-cage induction motor's state classification and fault diagnosis. The approach is based on thermal image analysis that utilizes a compact convolution neural network. In addition, the gathered and annotated image set, which consists of thermal images with 640 x 512 pixels resolution, is presented.
Keywords: *thermal imaging, fault diagnosis, squirrel-cage induction motor, deep learning, interpretability*

1. Introduction

Induction motors are commonly utilized in industrial applications for their straightforward construction and reliability. However, despite their simple design, as in other mechanical machinery, faults occur due to thermal expansion, human error, wear and tear deformation, or imprecise assembly. The following can be mentioned as the most frequent defects – misalignment when two shafts are not parallel and broken rotor bars, mainly caused by excessive thermal stress. Currently, the most widely used methods for anomaly detection in induction motors are based on current signals analysis [1, 2]. Even though it is a well-studied approach, it has several disadvantages. Current signals analysis requires direct access to the tested device, which is often impossible or hard to perform due to the continual machine operation. Furthermore, the examination of current signals does not give valuable spatial information about the malfunction location and does not

distinguish between different types of defects. Therefore, a non-invasive, thermal imaging-based method of fault diagnosis and operating anomalies detection was proposed in this paper. For this purpose, due to a lack of publicly available data, the new dataset was collected under laboratory conditions.

2. Dataset

The image set was gathered using Workswell WIC 640 InfraRed Camera, which has 640 by 512 pixels. The data collection consists of 5653 thermal images captured during the 42 series of measurements. Each series lasted between 20 and 30 seconds and was characterized by a different alignment and load. The data division was done based on splitting the experiments so that samples from one series of measurements could only be in one subset. Table 1 shows the assumed distribution of data, whereas Figure 1 depicts sample images from the dataset.

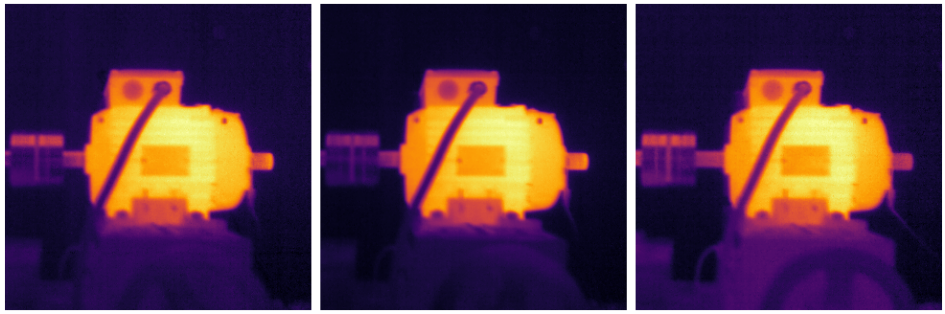


Figure 1. Sample thermal images captured by Workswell WIC 640 InfraRed Camera. (left) properly functioning system; (center) misalignment between two shafts; (right) motor with a broken rotor. Source: own work.

Table 1. Dataset distribution for 5-fold cross-validation.

	Fold					
Class	0	1	2	3	4	**Total**
Healthy	186	879	560	306	313	2244
Misalignment	311	314	417	381	376	1799
Broken rotor	335	502	211	279	283	1610

3. Experiments

The presented work is oriented to a solution for non-invasive, contactless, real-time measurement and analysis feasible on the edge device. Thus as a starting point, ResNet18 [3] with proven broad generalization capabilities, and additionally, ResNet10 [4], which is tailored to resource-constrained platforms with low computing power. Moreover, the MNASNet [5] in the variant with Squeeze-and-Excitation (SEMNASNet) alongside PP-LCNet [6] have been tested due to their high efficiency on the CPU-based hardware. The aforementioned neural network architectures were implemented in PyTorch and utilized through the PyTorch Image Models library [7]. In each experiment, thermal images were normalized and used as single-channel input of the convolutional neural networks mentioned above. The achieved results are presented in Table 2.

Table 2. Achieved metrics for benchmarked neural network architectures.

Model Architecture	Accuracy	F1 Score
lcnet_050	0.937 ± 0.049	0.943 ± 0.047
lcnet_075	**0.942 ± 0.064**	**0.948 ± 0.056**
lcnet_100	0.933 ± 0.055	0.947 ± 0.050
semnasnet_050	0.897 ± 0.095	0.919 ± 0.082
semnasnet_075	0.891 ± 0.096	0.919 ± 0.084
semnasnet_100	0.888 ± 0.132	0.920 ± 0.090
resnet10t	0.875 ± 0.223	0.897 ± 0.177
resnet18	0.905 ± 0.080	0.924 ± 0.059

3.1. Verification

The collected images look very similar at first view, regardless of class. Therefore, the neural network verification is a crucial part of this work to not treat the trained model as a black box and to ensure that the model focuses on the appropriate components of the induction motor rather than the background. For this task, the Grad-CAM [8] and Saliency Map [9] model interpretability methods were utilized from Captum [10] library. Figure 2 presents results obtained employing interpretability methods with the best-performing classification model from Table 2 for a random sample from each class.

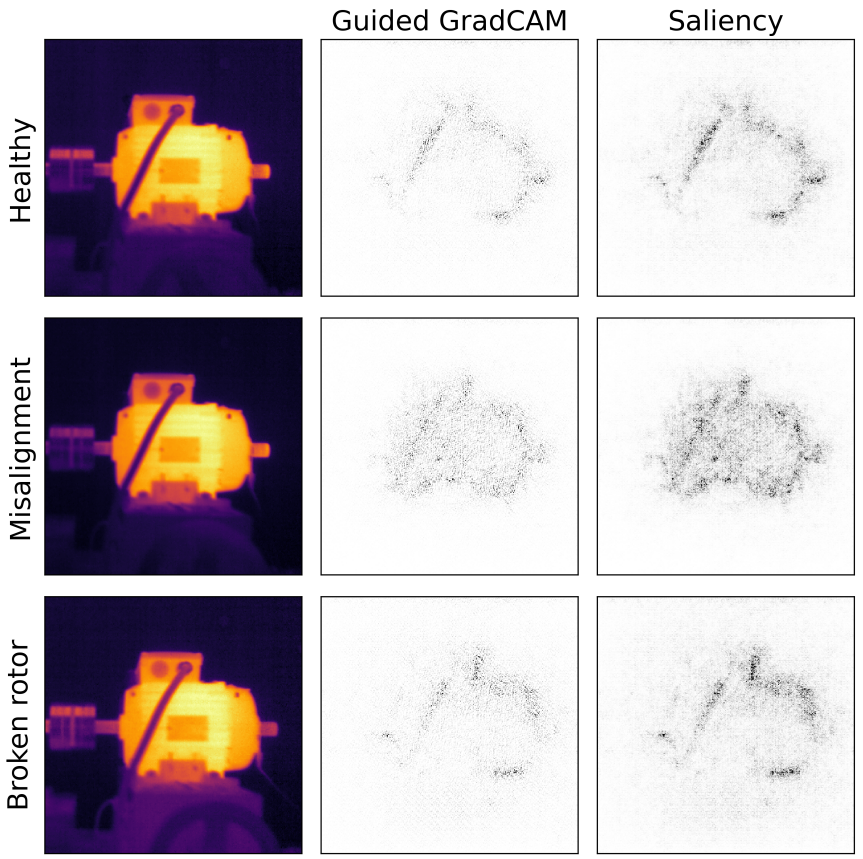


Figure 2. The visualization of interpretability methods' outcomes for trained lcnet_075 model. Source: own work.

4. Conclusions

The paper presents a novel dataset for anomaly detection and fault diagnosis in a squirrel-cage induction motor. The collected dataset is available in the project's repository at `https://github.com/MatPiech/motor-fault-diagnosis`. It contains 5653 thermal images gathered during 42 consecutive experiments with varied settings and loads applied. Moreover, in order to validate the proposed approach and demonstrate the dataset's feasibility, several neural network architectures were tested and benchmarked. The performance tests have shown that the presented solution achieves excellent results under laboratory conditions, accurately classifying the states of the tested induction motor. Moreover, it demonstrates potential application as an alternative non-invasive form of device inspection. Nevertheless, it is worth emphasizing that despite encouraging outcomes to develop a robust, well-generalizing algorithm based on thermal imaging, signifi-

cantly more data from various working environments and under diverse conditions is required.

Acknowledgment

The authors would like to thank Marcin Wolkiewicz and Paweł Ewert from Wrocław University of Science and Technology for test stand preparation and their help with data collection.

References

[1] Orlowska-Kowalska T., Wolkiewicz M., Pietrzak P., Skowron M., Ewert P., Tarchala G., Krzysztofiak M., Kowalski C.T., *Fault diagnosis and fault-tolerant control of pmsm drives–state of the art and future challenges*, *IEEE Access*, 2022, vol. 10, pp. 59979–60024, doi: 10.1109/ACCESS.2022.3180153.

[2] Halder S., Bhat S., Zychma D., Sowa P., *Broken rotor bar fault diagnosis techniques based on motor current signature analysis for induction motor – a review*, *Energies*, 2022, vol. 15, no 22, doi: 10.3390/en15228569.

[3] He K., Zhang X., Ren S., Sun J., *Deep residual learning for image recognition*, Computer Vision Foundation, 2015, doi: 10.48550/ARXIV.1512.03385.

[4] Gong J., Liu W., Pei M., Wu C., Guo L., *Resnet10: A lightweight residual network for remote sensing image classification*, [In:] *2022 14th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pp. 975–978, doi: 10.1109/ICMTMA54903.2022.00197.

[5] Tan M., Chen B., Pang R., Vasudevan V., Sandler M., Howard A., Le Q.V., *Mnasnet: Platform-aware neural architecture search for mobile*, 2018, doi: 10.48550/ARXIV.1807.11626.

[6] Cui C., Gao T., Wei S., Du Y., Guo R., Dong S., Lu B., Zhou Y., Lv X., Liu Q., Hu X., Yu D., Ma Y., *Pp-lcnet: A lightweight cpu convolutional neural network*, 2021, doi: 10.48550/ARXIV.2109.15099.

[7] Wightman R., *Pytorch image models*, 2019, doi: 10.5281/zenodo.4414861. https://github.com/rwightman/pytorch-image-models

[8] Selvaraju R.R., Cogswell M., Das A., Vedantam R., Parikh D., Batra D., *Grad-CAM: Visual explanations from deep networks via gradient-based localization*, *International Journal of Computer Vision*, 2019, vol. 128, no 2, pp. 336–359, doi: 10.1007/s11263-019-01228-7.

[9] Simonyan K., Vedaldi A., Zisserman A., *Deep inside convolutional networks: Visualising image classification models and saliency maps*, 2013, doi: 10.48550/ARXIV.1312.6034.

[10] Kokhlikyan N., Miglani V., Martin M., Wang E., Alsallakh B., Reynolds J., Melnikov A., Kliushkina N., Araya C., Yan S., Reblitz-Richardson O., *Captum: A unified and generic model interpretability library for pytorch*, 2020, doi: 10.48550/ARXIV.2009.07896.

Objective Hybrid Quality Assessment of Binary Images with the Use of Shallow Neural Networks

Mateusz Kopytek, Krzysztof Okarma[0000−0002−6721−3241]

West Pomeranian University of Technology in Szczecin
Department of Signal Processing and Multimedia Engineering
26 Kwietnia 10, 71-126 Szczecin, Poland
{km46880, okarma}@zut.edu.pl

DOI:10.34658/9788366741928.5

Abstract. *The state-of-the-art image quality assessment methods designed for binary images are not highly correlated with subjective evaluation results, therefore one of the efficient methods to improve their performance is the application of shallow neural networks. In such an approach each elementary metric is used as the input of the network and the network is trained with subjective quality scores used as the goal function. The obtained correlation with subjective scores depends not only on the number of elementary metrics and their choice but also on the training algorithm and the network's structure as presented in the paper.*
Keywords: *image quality assessment, binary images, neural networks*

1. Introduction

The most typical application of objective image quality assessment (IQA) metrics is the evaluation of grayscale or color images and many so-called general-purpose IQA methods have been developed for these purposes. They may be divided into full-reference (FR) methods, based on a comparison with an available reference image without any distortions, reduced-reference (RR) methods where only a partial reference information is available, and "blind"/no-reference (NR) metrics that may be applied when a reference image is unavailable. Since many such IQA methods, referred to as elementary metrics, are not always equally sensitive to various kinds of image distortions, and in many cases poorly correlated with their subjective perception, the idea of combined/hybrid metrics has been proposed [1, 2, 3, 4] that utilizes various methods of combination of elementary metrics with optimized weighting coefficients.

Assuming the availability of subjective scores provided in the form of Mean Opinion Scores (MOS) or Differental MOS (DMOS) values in the IQA databases containing numerous images subject to various kinds of distortions, a correlation of such elementary or combined objective metrics with them may be computed for

each dataset. Therefore, Pearson Linear Correlation Coefficient (PLCC), Spearman Rank-Order Correlation Coefficient (SROCC) or Kendall Rank-Order Correlation Coefficient (KROCC) with MOS values are typically used as the goal functions during the weights optimization in combined metrics. Nevertheless, another interesting method for the development of combined metrics is the use of shallow neural networks with elementary metrics used as inputs [5], considered in this paper. Currently, deep neural networks are commonly used to extract image features, on the basis of which the image quality is then estimated. This approach is widely used in the development of general-purpose IQA metrics, allowing for the creation of both FR and NR metrics. Nevertheless, such an approach requires the use of large-scale image datasets for network training.

Unfortunately, the availability of image datasets containing MOS values and highly correlated IQA metrics is limited for binary images. Therefore, the idea of the combined metrics, particularly constructed using neural networks, has not been fully explored for binary images. Nevertheless, binary images are commonly used in practice, for example for defect detection tasks on manufactured products, optical character recognition (OCR), as well as in devices with limited computing power and embedded systems. Therefore, a development of better IQA methods for binary images is important from a practical point of view. During the conducted experiments the currently available Bilevel Image Similarity Ground Truth Archive [6], being the only possibility of the verification of objective metrics with subjective quality scores dataset, has been utilized. This dataset contains 7 binary images and their 315 distorted versions subjectively assessed by independent observers in the normalized scale where 0 denotes the lowest quality and 1 corresponds to an image identical to the reference image.

2. The proposed approach and experimental results

Since the number of IQA metrics designed for the evaluation of binary images is relatively small, some metrics typically used as classification metrics may be used for this purpose. Therefore, such metrics as Precision, Recall (Sensitivity), F-Measure, Specificity, pseudo-Precision, pseudo-Recall, pseudo-F-measure, Accuracy, GAccuracy, or SFMeasure were used in the experiments. Some other similar metrics are: Balanced Classification Rate (BCR), Balanced Error Ratio (BER), and Negative Rate Metric (NRM). Although these metrics are typically used as the classification measures using image datasets, in this paper they are applied for comparisons of pixels from two binary images.

Additionally, some other metrics used for the evaluation of image binarization results, such as PSNR, Distance Reciprocal Distortion (DRD) [7], and Misclassification Penalty Metric (MPM) [8], were also applied. The only metric designed exclusively for the quality assessment of binary images, considered in the paper,

was the Border Distance [9] which was computed in three versions utilizing three various distance measures: city-block (D4), Euclidean, and chessboard (D8), leading to three variants of the BDPSNR metric. As the additional features, the masked local entropy and the masked local variance were used.

The starting point in the conducted research was the use of 21 elementary metrics as the network inputs for two types of fully connected shallow networks (feed-forward and cascade-forward) with experimentally selected number of 10 neurons in the hidden layer to minimize the risk of overtraining. For the process of training the artificial neural network, the test set of 301 binary images (14 images identical to the reference images were excluded) was randomly divided into two subsets, of which the first (70% of the set) contained the images used in the training process, whereas the second (30% of the set) was used for the validation and testing. Since the assumed input data for the popular Convolutional Neural Networks (CNN) are images or image patches, not the values of image quality metrics, they have not been used in this study.

Considering the MOS values as the target values, four different training methods were used applying the Mean Squared Error between the subjective and objective scores as the loss function: Levenberg-Marquardt (LM), BFGS Quasi-Newton (BFG), Resilient Backpropagation (RP), and Scaled Conjugate Gradient (SCG). Then, the number of inputs were reduced one by one. In each of the reduction steps, the "least important" of N metrics was determined by the calculation of N Pearson correlations for N various optimized (trained) combinations of $N-1$ elementary metrics used as inputs. Then the set of $N-1$ metrics with the highest PLCC value was selected as the result of the reduction step. This procedure was repeated until the combination of two elementary metrics. The final goal of such ablation was to reduce the number of inputs to select the combination that gives the best results with the smallest possible number of elementary metrics.

The PLCC values achieved during the ablation study, limiting the number of inputs are depicted in Fig. 1, whereas the results obtained for various training methods for three limited sets of elementary metrics are presented in Table 1. The first set of inputs (set no. 1) consisted of of two measures: BDPSNR (Euclidean type) and the local variance (for 5×5 pixels mask), the set no. 2 was extended by two additional metrics: local entropy (for 5×5 pixels mask) and pseudo-F-Measure, whereas the PSNR and NRM were added to the set no. 3. The choice of metrics and the order in which they were added were in accordance with the conducted ablation (reduction) process described above with results shown in Fig. 1.

The results of the conducted ablation presented in Fig. 1 show that the elementary BDPSNR (Euclidean type) metric has the highest correlation with subjective assessments, however, added local entropy and local variance significantly improve the quality prediction. A further increase of performance may be achieved adding three next metrics, however, the use of more than six elementary metrics does not increase the obtained PLCC values significantly. Therefore, the results

Table 1. The obtained correlation values for various training algorithms and network structures with average training time data.

Method	PLCC	SROCC	KROCC	Average training time	Average no. of epochs
Set no. 1 – feed-forward network					
LM	0.9042	0.9020	0.7244	0.1007	20
BFG	0.9062	0.9036	0.7266	0.0542	22
RP	0.9042	0.9041	0.7358	0.0519	19
SCG	0.9051	0.9042	0.7275	0.0531	19
Set no. 1 – cascade-forward network					
LM	0.9033	0.9048	0.7469	0.0531	18
BFG	0.9041	0.9019	0.7253	0.0558	17
RP	0.9042	0.9078	0.7393	0.0550	19
SCG	0.9040	0.9043	0.7362	0.0539	18
Set no. 2 – feed-forward network					
LM	0.9166	0.9199	0.7572	0.1116	21
BFG	0.9236	0.9245	0.7570	0.1895	24
RP	0.9176	0.9194	0.7520	0.0998	22
SCG	0.9144	0.9159	0.7449	0.0901	22
Set no. 2 – cascade-forward network					
LM	0.9219	0.9270	0.7815	0.0925	24
BFG	0.9178	0.9156	0.7458	0.1070	20
RP	0.9156	0.9181	0.7492	0.0964	20
SCG	0.9169	0.9185	0.7473	0.0880	21
Set no. 3 – feed-forward network					
LM	0.9338	0.9312	0.7696	0.1195	24
BFG	0.9391	0.9359	0.7832	0.0991	25
RP	0.9403	0.9409	0.7886	0.1013	26
SCG	0.9314	0.9326	0.7701	0.0961	22
Set no. 3 – cascade-forward network					
LM	0.9298	0.9274	0.7722	0.0998	23
BFG	0.9423	0.9366	0.8044	0.1216	24
RP	0.9366	0.9409	0.7850	0.1002	21
SCG	0.9339	0.9381	0.7790	0.1070	23

of the analysis of the influence of the network structure and the training algorithm presented in Table 1, were obtained for the reduced numbers of network inputs.

3. Conclusions

The conducted experiments concerning the choice of the network structure, and the selection of the training function, showed that the proposed metric based on neural networks is resistant to these modifications. Therefore, simpler algorithms and structures may be used for the implementation of the developed combined

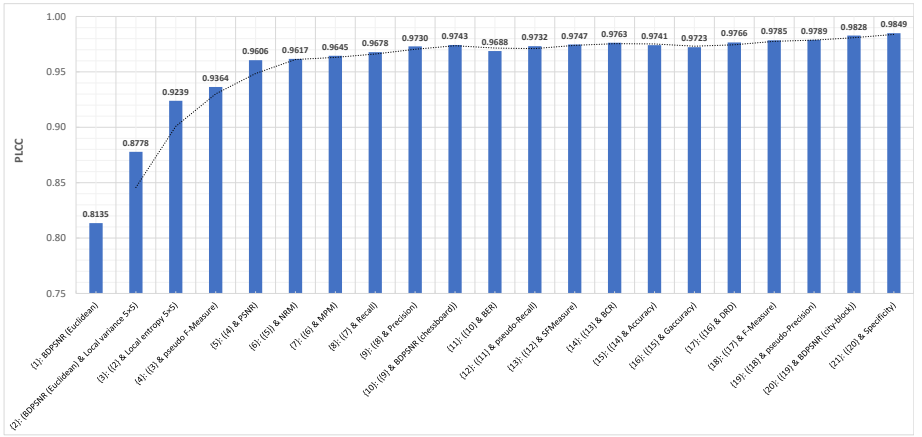


Figure 1. The PLCC values obtained during the ablation process. Source: own work.

metric for practical applications. The key information was provided by the network ablation process, as a result of which the best complementary elementary metrics were selected. The result of limiting the size of the network to 6 "best" metrics was a linear correlation equal to 0.9617 (compared to PLCC = 0.9849 for 21 input metrics).

Acknowledgment

This research is partially supported by the ZUT Highfliers School (Szkoła Orłów ZUT) project within the framework of the program of the Minister of Education and Science (Grant No. MNiSW/2019/391/DIR/KH, POWR.03.01.00-00-P015/18), co-financed by the European Social Fund.

References

[1] Okarma K., *Combined full-reference image quality metric linearly correlated with subjective assessment*, [In:] L. Rutkowski, R. Scherer, R. Tadeusiewicz, L. Zadeh, J. Zurada (eds.), *Artificial Intelligence and Soft Computing*, *LNCS*, vol. 6113, Springer, 2010, pp. 539–546.

[2] Okarma K., *Combined image similarity index*, *Optical Review*, 2012, vol. 19, no 5, pp. 349–354, doi: 10.1007/s10043-012-0055-1.

[3] Liu T.J., Lin W., Kuo C.C.J., *Image quality assessment using multi-method fusion*, *IEEE Trans. Image Processing*, 2013, vol. 22, no 5, pp. 1793–1807, ISSN 1057-7149, 1941-0042, doi: 10.1109/TIP.2012.2236343.

[4] Oszust M., *Decision fusion for image quality assessment using an optimization approach*, *IEEE Signal Processing Lett.*, 2016, vol. 23, no 1, pp. 65–69, doi: 10.1109/lsp.2015.2500819.

[5] Ieremeiev O., Lukin V., Ponomarenko N., Egiazarian K., *Combined no-reference IQA metric and its performance analysis*, *Electronic Imaging*, 2019, vol. 31, no 11, pp. 260–1–260–7, doi: 10.2352/issn.2470-1173.2019.11.ipas-260.

[6] Zhai Y., Neuhoff D.L., *Similarity of scenic bilevel images*, *IEEE Trans. Image Processing*, 2016, vol. 25, no 11, pp. 5063–5076, doi: 10.1109/tip.2016.2598493.

[7] Lu H., Kot A., Shi Y., *Distance-reciprocal distortion measure for binary document images*, *IEEE Signal Processing Lett.*, 2004, vol. 11, no 2, pp. 228–231, doi: 10.1109/lsp.2003.821748.

[8] Young D., Ferryman J., *PETS metrics: On-line performance evaluation service*, [In:] *Proc. 2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, IEEE, Beijing, China, pp. 317–324, doi: 10.1109/vspets.2005.1570931.

[9] Zhang F., Cao K., Zhang J.L., *A simple quality evaluation method of binary images based on border distance*, *Optik*, 2011, vol. 122, no 14, pp. 1236–1239, doi: 10.1016/j.ijleo.2010.07.030.

One-point Hough Transform with Centred Accumulator

Leszek J. Chmielewski[0000−0002−9725−2479],
Marcin Bator[0000−0002−6881−3695],
Krzysztof Gajowniczek[0000−0001−6953−8907]

Warsaw University of Life Sciences – SGGW
Institute of Information Technology
Nowoursynowska 159, 02-776 Warszawa, Poland
leszek_chmielewski@sggw.edu.pl

DOI:10.34658/9788366741928.6

Abstract. *A novel approach for improving the accuracy of the Hough transform by centering the accumulator in the middle of the image is proposed. This improves the results in this crucial image region and optimizes the utilization of the accumulator space. The information on the direction as well as on the sense of edgels is accumulated, which makes it possible to effectively group the edgels into meaningful continuous edges.*
Keywords: *Hough transform, centred accumulator, one-point, directional*

1. Introduction

The method for detecting straight lines in images invented by Paul V. C. Hough in 1959 [1] in the application to analysis of experimental results in high energy physics was not mentioned in the literature for ten years, which could be due to the famous patent filed in 1960 and granted in 1962. The publication by Duda and Hart of a new formulation of the Hough transform (HT) in 1972 [2] has gained extreme popularity. The last survey dedicated solely to HT is probably [3]; further surveys were dedicated to general problems rather than particular methods, e.g. [4]. In 2022 alone the number of papers related to HT was about 13 200. New concepts still emerge, like for example the *Cartesian* HT, with interesting properties, published in 2002 [5]. This was close to the 60-th anniversary of the patent.

The proposition made in this paper is one more modification of HT among many existing solutions. It caters for an improvement of the detection accuracy and of the information content of the results of the Hough transform.

2. Motivation by a practical problem

The study towards extending the informative content of the accumulator in the HT emerged as an answer to the needs in the project on monitoring the behaviour of plants watered daily in a greenhouse. As the measure of the state of water needs in plants their turgor pressure can be used. Its loss in the cells of a plant manifests itself macroscopically with lowering the leaves with respect to the stem. The proper moment of watering is just before the turgor pressure becomes too weak. This phenomenon can be monitored by measuring the changes of angles formed by twigs and leaves and the stem of a plant in a series of images.

The shape primitives suitable for finding angles are straight line elements, possibly merged into longer edges. Images taken in everyday practice in a greenhouse make finding such primitives feasible. The analysis of plants will not be presented in this paper, but images of plants will be used to illustrate the considerations.

It must be stated that the methods other than HT are now frequently used to find lines, circles and ellipses (see e.g. [6]). Robustness, accuracy and lack of training are in our opinion the reasons why the HT is still a method with great potential.

3. Hough transform and its versions

The lines considered in this paper are actually *edges* in which the gradient is known. The edges are found with the directional second derivative zero-crossing detector and the colour image is transformed into greenness intensity by linearly scaling the distance of the H component of the HSV representation from the green hue, from the $\langle 0, 180\rangle$ interval (periodicity considered) into the $\langle 255, 0\rangle$ interval.

Let us consider the geometry shown in Fig. 1. A pixel P_i in which the image intensity gradient is $\vec{G}_i$ lies on a line l_i. Such a pixel is called *edgel*. The line can be defined by the distance R, called radius, between its foot point F_i and the origin of the coordinate system $O(0, 0)$, and the angle φ_i between the x axis and the line OF_i [2] (in fact F_i univocally defines the line [5]; however, accuracies go down as F_i approaches O). The equation of line l going through pixel $P(x, y)$ is

$$x \cos\varphi + y \sin\varphi - R = 0\,. \tag{1}$$

Each edgel $P_i(x_i, y_i)$ with φ_i defined by the direction of its gradient casts a vote in a single point (φ_i, R_i) in the accumulator, where R_i is found from (1) for x_i, y_i, φ_i.

In the classical approach [2], if from (1) it follows that $R < 0$, then $R := -R$ and $\varphi := \lfloor \varphi - \pi \rfloor$, hence $R = |\vec{R}|$ and the sense of edgels is forgotten ($\lfloor\cdot\rfloor$ means here bringing to the interval $(-\pi, \pi\rangle$). Then, if the image of diagonal D is located in the first quadrant of Oxy, then $\varphi \in (-\pi/2, \pi\rangle$ and $R \in [0, D]$.

The first well known improvement is to extend the range of R to negative values, so that not only the direction, but also the sense of edgels, i.e., of $\vec{G}$, is accumulated. Conventionally, the origin is located in the upper left corner of the image;

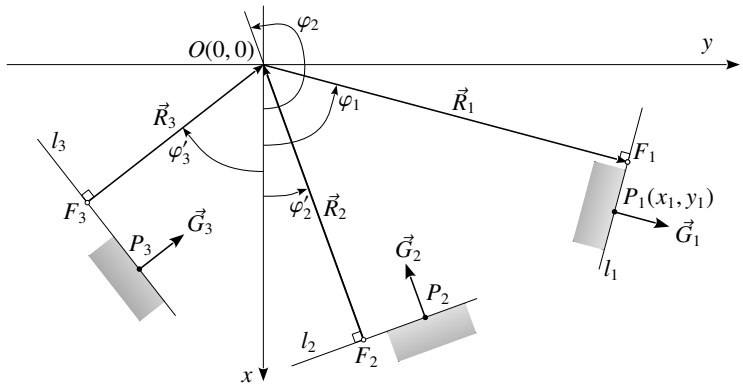


Figure 1. Geometry of the Hough transforms discussed. $P_i(x_i, y_i)$ – pixels of edges which form lines l_i; $\vec{G}_i$ – respective image brightness gradients; $\vec{R}_i$ – HT directed radii in pixels P_i, $\vec{R}_i \parallel \vec{G}_i$; φ_i – HT angles; $\varphi_i' = \varphi_i - \pi$; $i = 1, 2, 3$. F_i – foot of line l_i. Grey rectangles symbolize image intensity gradients. Source: own work.

edges of green objects in Fig. 2a were found in this setting. It can be seen that the density of pixels found as belonging to lines goes down with the distance from the origin. The same appears if the origin is placed in the opposite corner (Fig. 2b).

Therefore, we propose to place the coordinate system origin in the centre of the image. The result can be seen in Fig. 2c. This simple operation seems not to have been proposed until now, to our best knowledge. We shall name this new version of HT the One-Point Hough Transform with Centred Accumulator: OPHT-CA.

Let us consider the contents of the accumulator in these three approaches. An ideal source image for this would be such that there are edgels *everywhere* and in *all* directions; our image of Fig. 3d is its humble approximation. The accumulators for the three HT versions considered are shown in Fig. 3. In the proposed OPHT-CA the volume of the accumulator is utilized the best (no unused black regions).

In the accumulator of PPHT-CA the weak fuzzification with a paraboloidal fuzzifying function is applied [7] (paraboloids can be seen around strong maxima).

It is important that the scanning of the image is done twice: once for accumulation, and a second time for assigning pixels to maxima. A pixel the vote of which falls inside an elliptic neighbourhood of a maximum is assigned the direction related to this maximum, and the intensity being a product of maximum value and edgel gradient modulus. Neighbourhood size corresponds to the scale of the fuzzifying function. A pixel with the vote not assigned to any maximum is dismissed. This reduces the number of potential edgels making edge aggregation effective.

The reduced accuracy of the HT far from the coordinate system origin results from that the inaccuracies related to roundings in the indexing functions (which bind accumulator indices with radius and angle) are growing with the distance

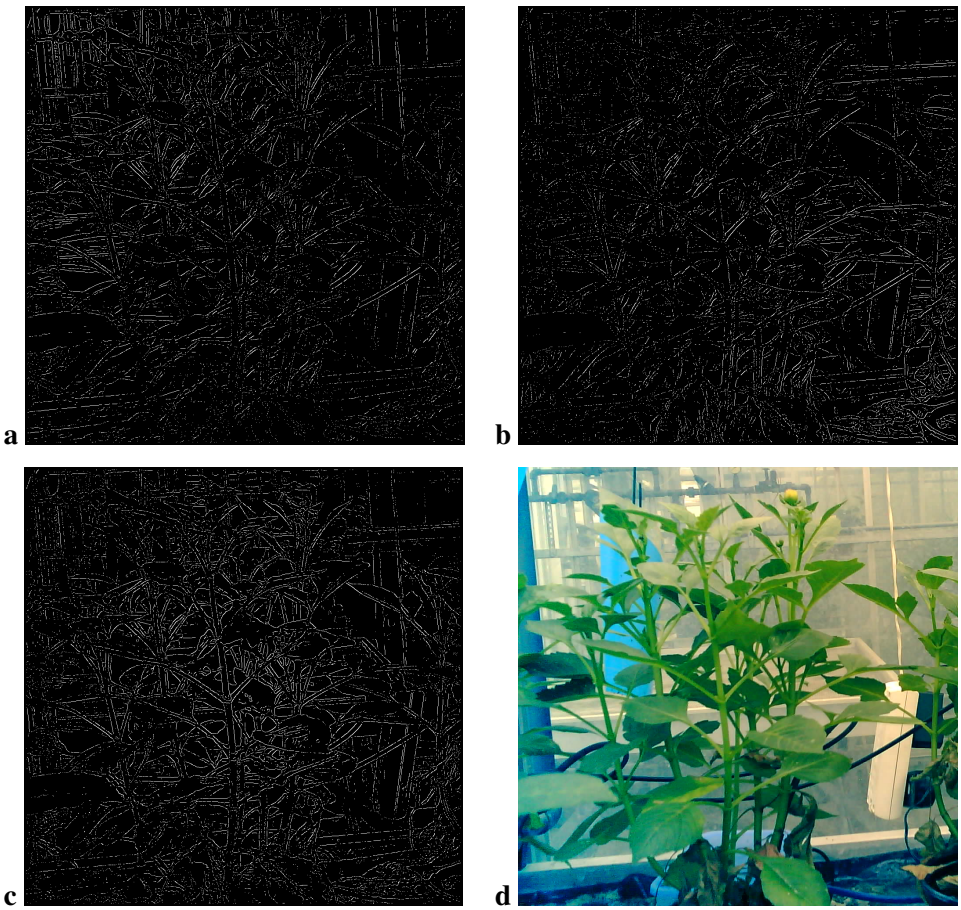


Figure 2. Pixels belonging to lines, origin of the coordinate system $OR\varphi$ located in: (**a**) upper left corner; (**b**) lower right corner; (**c**) centre of the image. (**d**) Source. Source: own work.

from this origin. Discretizing the accumulator more densely to achieve accuracy gain implies the necessity of increasing the scales of the fuzzifying function; both these operations bring longer calculations. So, a compromise between accuracy and time is needed. Accuracy could be tested quantitatively with an image similar to that of Fig. 3d but such tests fall beyond the scope of this short communication. Moving the origin to the image center improves the accuracy at no computational cost, and brings memory reduction by optimizing the use of the accumulator space.

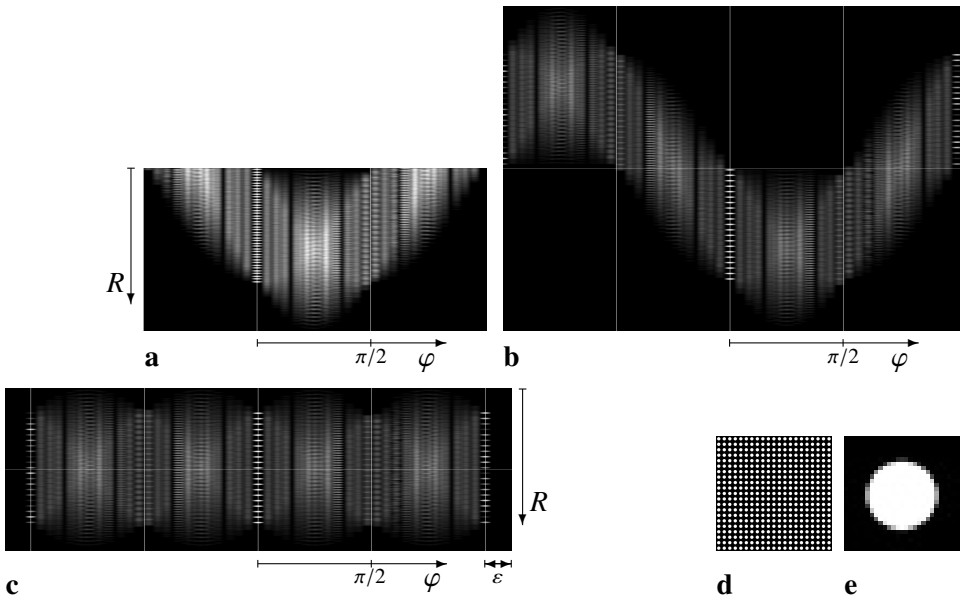


Figure 3. Accumulator in three versions of HT for image **d** with diagonal D. (**a**) Classic version, $R \in [0, D]$, $\varphi \in [-\pi/2, \pi]$; edgel sense forgotten. (**b**) Version with $R \in [-D, D]$, $\varphi \in [-\pi, \pi]$. (**c**) Version with centred origin; $R \in [-D/2, D/2]$, $\varphi \in [-\pi - \varepsilon, \pi + \varepsilon]$, where ε – margin useful in implementation of the accumulator, which is cyclic. (**d**) Source image of diagonal D formed of 21×21 images (**e**). Source: own work.

4. Conclusion

A modification of the Hough transform based on its most classic version has been proposed. It consists in placing the centre of the accumulator in the region in which the detection accuracy should be the best, that is, in the centre of the image. This not only improves the accuracy of results, but also leads to optimal use of the accumulator volume. Accumulating the sense of edgels together with their direction, and performing a second scan of the image to assign pixels to maxima in the accumulator, provides for grouping these edgels into contiguous, meaningful edges with a consistent direction. Quantitative analysis of the benefits and costs of these improvements will be carried out in further publications.

It is planned to use the proposed One-Point HT with Centred Accumulator – OPHT-CA – in measurements of plant movements in the process of their watering.

References

[1] Hough P.V.C., *Machine analysis of bubble chamber pictures*, [In:] *Proc. 2nd Int. Conf. on High Energy Accelerators and Instrumentation HEACC '59*, CERN, Geneva, Switzerland, pp. 554–558.

[2] Duda R.D., Hart P.E., *Use of the Hough transformation to detect lines and curves in pictures*, *Comm. Assoc. for Computing Machinery*, 1972, vol. 15, pp. 11–15, doi: 10.1145/361237.361242.

[3] Mukhopadhyay P., Chaudhuri B.B., *A survey of Hough transform*, *Pattern Recognition*, 2015, vol. 48, no 3, pp. 993–1010, doi: 10.1016/j.patcog.2014.08.027.

[4] Jha R., Sonune S., Shahid M.T., Gudadhe S., *A relative study on object and lane detection*, [In:] A.D. Thakare, S.U. Bhandari (eds.), *Artificial Intelligence Applications and Reconfigurable Architectures*, chap. 9, John Wiley & Sons, 2023, pp. 167–185, doi: 10.1002/9781119857891.ch9.

[5] Yang G., Hu J., Hou Z., Zhang G., Wang W., *A new Hough transform operated in a bounded Cartesian coordinate parameter space*, *IET Image Processing*, 2022, vol. 16, no 8, pp. 2282–2295, doi: 10.1049/ipr2.12489.

[6] Małaszek M., Zembrzuski A., Gajowniczek K., *ForestTaxator: A tool for detection and approximation of cross-sectional area of trees in a cloud of 3D points*, *Machine Graphics and Vision*, 2022, vol. 31, no 1/4, pp. 19–48, doi: 10.22630/MGV.2022.31.1.2.

[7] Chmielewski L.J., *Fuzzy histograms, weak fuzzification and accumulation of periodic quantities. Application in two accumulation-based image processing methods*, *Pattern Analysis & Applications*, 2006, vol. 9, no 2-3, pp. 189–210, doi: 10.1007/s10044-006-0037-7.

Pedestrian Detection with High-resolution Event Camera

Piotr Wzorek[0000−0003−3885−600X], **Tomasz Kryjak**[0000−0001−6798−4444]

Embedded Vision Systems Group, Computer Vision Laboratory
Department of Automatic Control and Robotics
AGH University of Krakow, Poland
{pwzorek,tomasz.kryjak}@agh.edu.pl

DOI:10.34658/9788366741928.7

Abstract. *Despite the dynamic development of computer vision algorithms, the implementation of perception and control systems for autonomous vehicles such as drones and self-driving cars still poses many challenges. A video stream captured by traditional cameras is often prone to problems such as motion blur or degraded image quality caused due to challenging lighting conditions. In addition, the frame rate – typically 30 or 60 frames per second – can be a limiting factor in certain scenarios. Event cameras (DVS – Dynamic Vision Sensor) are a potentially interesting technology to address the above mentioned problems. In this paper, we compare two methods of processing event data by means of deep learning for the task of pedestrian detection. We used a representation in the form of video frames, convolutional neural networks and asynchronous sparse convolutional neural networks. The results obtained illustrate the potential of event cameras and allow the evaluation of the effectiveness and efficiency of the methods used for high-resolution (1280 x 720 pixels) footage.*
Keywords: *pedestrian detection, event camera, convolutional neural networks, sparse convolutional neural networks*

1. Introduction

Event cameras are neuromorphic vision sensors inspired by the structure and behaviour of the human eye [1]. They are increasingly being used in computer vision. They respond to changes in the brightness of the observed scene independently for each pixel. Each so-called "event" is generated when the change in the logarithm of the brightness sensed by a given pixel reaches a certain threshold. A single event is described by four values: $e = \{t, x, y, p\}$, where: t is the timestamp of the event (in microseconds), x and y are the coordinates of the pixel registering the event, and p is its polarity in the form of a value of 1 (positive change in brightness) or -1 (negative change).

The growing popularity of event cameras is the result of their ability to be used in conditions of rapid movement of the object relative to the sensor (temporal resolution of microseconds) and unfavourable lighting conditions (dynamic range of 120 dB, relatively good performance in low light). The event camera only records changes in the scene. This means that redundant information is largely eliminated and the number of events generated depends on the dynamics of the scene, camera ergo-motion or lighting changes. As a result, the power consumption of the sensor is low (in typical situations) and the processing of the most important information can be efficient.

However, there are significant challenges in applying known computer vision algorithms to event data. These type of data differ significantly from traditional video frames in that they form a sparse spatio-temporal cloud. Meanwhile, state-of-the-art object detection solutions use traditional video frames as input to deep convolutional neural networks.

In this work, we have compared different methods for applying deep learning algorithms to event data. We considered using the representation of event data in the form of frames generated by accumulating them over a defined time window and using as input to a CNN, as well as the use of asynchronous sparse neural networks to optimise energy consumption by reducing the number of operations performed. We focused on the problem of pedestrian detection as a key issue for different types of autonomous vehicles – both drones and self-driving cars. The main contribution of this paper is the evaluation of selected methods available in the literature for a high resolution dataset and their comparison in terms of accuracy and efficiency.

The remainder of this paper is organised as follows. Section 2 describes the different methods considered in the literature for applying deep learning with event data to the task of object detection. Section 3 describes the research we have conducted. A summary and considerations for further development plans for the implemented systems conclude the paper.

2. Object detection with event cameras

State-of-the-art object detection algorithms use deep convolutional neural networks adapted to process data represented as two- or three-dimensional matrices. The unusual nature of event data requires both the use of so-called event data representations and the introduction of modifications to the algorithms used. Many approaches to this problem have been proposed in the literature.

The simplest solution is to accumulate the event data in a matrix, analogous to the frames recorded by classical cameras, and then use deep convolutional neural networks. The paper [2] proposes a method in which each pixel is assigned the polarity value of the last event recorded. An extension of this idea is the expo-

nentially decaying time surface, where the time of occurrence of an event is also taken into account. Also popular are methods that take into account the frequency of occurrence of an event for a given pixel [3], or the so-called leaky surface [4], where the memory of previous events is retained.

An extension of the idea of using deep convolutional neural networks to process event data is the use of asynchronous sparse convolutional neural networks (ASCNN). The authors of [5] exploit the sparsity of event data to reduce the computational complexity and energy consumption of the detection system, using networks in which only the convolution results for changing input values are updated. The use of ASCNNs allows to perform detection efficiently and asynchronously (updating the necessary values for each incoming pixel).

The literature also considers methods based on the fusion of event data and traditional RGB frames, the use of transformer-based architectures, recurrent architectures, Graph Neural Networks (GNNs) or Spiking Neural Networks (SNNs).

3. The analysed pedestrian detection methods

For our research, we used the Prophesee 1 Megapixel automotive detection dataset [6] containing sequences recorded with a 1280x720 pixel resolution event camera in road conditions with labelled objects such as pedestrians, traffic signs, traffic lights or cars. The dataset was filtered to select only fragments containing pedestrians. This yielded more than one million 10ms sequences on which pedestrians were labelled. For some experiments, a subset of 100,000 10ms sequences was used to speed up the neural network training process.

The first method considered was to accumulate event data in time windows of 10ms and use them as input to a deep convolutional neural network. To maximise the amount of information in the representation, we used a fusion of representations using different characteristics of the data – polarity, temporal resolution and frequency of occurrence, analogous to [7]. As a detector, we used the YOLOv7 architecture of [8], which represents the state of the art in terms of execution time and accuracy. The detector trained on the described dataset using the transfer learning mechanism achieved a detection accuracy of 67.7%mAP@0.5 (38%mAP@.5:.95). The YOLOv7 model is characterised by a value of 104.7 GFLOPs (floating point operations). We use this metric to evaluate and compare the computational efficiency between multiple models.

As an alternative to the detection system described above, we also decided to test ASCNNs. Analogous to [5], we used the event histogram as event data representation, the VGG-16 network as feature extraction and the YOLO output layer. This model has a reported computational efficiency of 205 MFLOPs, which is a significant reduction compared to the YOLOv7-based detector.

However, we were unable to achieve satisfactory accuracy scores for a ASC-

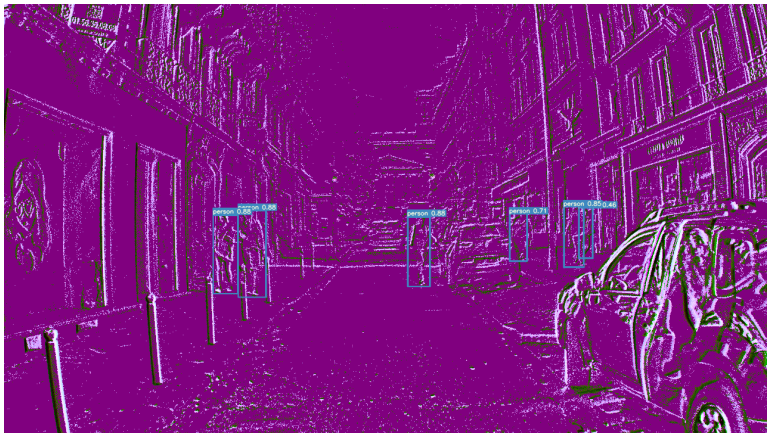


Figure 1. Example of correct detection on the fused representations. Source: own work.

NNs after numerous experiments on a reduced data set (100,000 samples). The maximum result was 0.04%mAP. To increase the network input, we also tried to extend the model by two convolution layers and a max-pooling layer. For the transformed model, we obtained a maximum result of 0.1%mAP. As a comparison, for the same subset of dataset we obtained 47%mAP@0.5 (19.8%mAP@.5:.95) for YOLOv7 model.

4. Summary

In this work, we investigated the accuracy and computational efficiency of event camera-based detection systems with deep convolutional network YOLOv7 and ASCNNs. The accuracy score achieved for the YOLOv7 detector (67.7% mAP@0.5, 38%mAP@.5:.95) are satisfactory and comparable to the RED network accuracy (43%mAP) reported in [6], which is the highest result achieved for this dataset. The accumulation of data in a 10ms window provides enough information to perform detection with a large YOLOv7 model and a diverse and efficient representation. However, the event histogram representations generated in the same time windows and the use of a relatively small network model (VGG16 with YOLO output) do not allow for satisfactory performance results for ASCNNs.

The conducted research makes it possible to plan the direction of further work on detection systems for high-resolution event data, taking into account their accuracy, computational complexity and the possibility of hardware implementation using SoC FPGA or eGPU platforms. We plan to further explore ASCNNs architectures to improve the accuracy of detection systems. We will conduct further experiments considering other event data accumulation times (to increase the

amount of information) and larger sparse network models. The significant reduction in computational complexity in sparse models is a motivation for further work on this type of network. Another approach that we also want to test is the use of binary networks. A relatively simplified event representation should work well with this type of model and allow for improved computational efficiency. We are also considering various methods to accelerate the mentioned networks using the FPGA platform. Another type of network that we plan to evaluate and study are spiking neural networks (SNN), which allow direct processing of events. The use of Dynamic Vision Sensors and neuromorphic platforms such as BrainChip or Loihi would enable the task of pedestrian detection to be realised effectively and efficiently.

Acknowledgements

The work presented in this paper was supported by the programme "Excellence initiative – research university" for the AGH University of Krakow. We gratefully acknowledge Poland's high-performance computing infrastructure PLGrid (HPC Centers: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016130.

References

[1] Gallego G., Delbrück T., Orchard G., Bartolozzi C., Taba B., Censi A., Leutenegger S., Davison A.J., Conradt J., Daniilidis K., et al., *Event-based vision: A survey*, *IEEE transactions on pattern analysis and machine intelligence*, 2020, vol. 44, no 1, pp. 154–180.

[2] Afshar S., Ralph N., Xu Y., Tapson J., Schaik A.v., Cohen G., *Event-based feature extraction using adaptive selection thresholds*, *Sensors*, 2020, vol. 20, no 6, p. 1600.

[3] Chen N.F., *Pseudo-labels for supervised learning on dynamic vision sensor data, applied to object detection under ego-motion*, [In:] *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 644–653.

[4] Cannici M., Ciccone M., Romanoni A., Matteucci M., *Asynchronous convolutional networks for object detection in neuromorphic cameras*, [In:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0.

[5] Messikommer N., Gehrig D., Loquercio A., Scaramuzza D., *Event-based asynchronous sparse convolutional networks*, [In:] *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, Springer, pp. 415–431.

[6] Perot E., De Tournemire P., Nitti D., Masci J., Sironi A., *Learning to detect objects with a 1 megapixel event camera*, *Advances in Neural Information Processing Systems*, 2020, vol. 33, pp. 16639–16652.

[7] Wzorek P., Kryjak T., *Traffic sign detection with event cameras and dcnn*, [In:] *2022 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, IEEE, pp. 86–91.

[8] Wang C.Y., Bochkovskiy A., Liao H.Y.M., *Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*, *arXiv preprint arXiv:2207.02696*, 2022.

Recognition of Shoplifting Activities in CCTV Footage Using the Combined CNN-RNN Model

Lyudmyla Kirichenko[1,2][0000−0002−2780−7993],
Oksana Pichugina[3,4][0000−0002−7099−8967],
Bohdan Sydorenko[1][0000−0002−5963−5911],
Sergiy Yakovlev[3,5][0000−0001−6736−371X]

[1]*Kharkiv National University of Radio Electronics*
14 Nauki Avenue, 61166 Kharkiv, Ukraine
lyudmyla.kirichenko@nure.ua
[2]*Wroclaw University of Science and Technology*
27 Wyspianskiego, 50-370 Wroclaw, Poland
[3]*National Aerospace University "Kharkiv Aviation Institute"*
17 Chkalova Street, 61070 Kharkiv, Ukraine
o.pichugina@khai.edu
[4]*University of Toronto*
27 King's College Circle, M5S 1A1 Toronto, Canada
[5]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
s.yakovlev@khai.edu

DOI:10.34658/9788366741928.8

Abstract. *The recognition of human activities through surveillance has numerous applications across various fields. This article presents a proposed approach to identify shoplifting in camera-recorded video data using a neural classifier that combines two neural networks, specifically, convolutional and recurrent networks. The hybrid architecture consists of two parallel streams: initial and processed video fragments (histogram of oriented gradients and optical flow). The convolutional network extracts features from each frame of the video fragment, while the recurrent network processes the temporal information from sequences of frames as features to classify the activity.*

Keywords: *human activity recognition, surveillance, shoplifting, convolutional neural network, recurrent neural network, features extraction, histogram of oriented gradients, optical flow*

1. Introduction and Literature Review

Recognition of human actions is an important task in modern video surveillance. Over the years, the number of video cameras in public places has complicated the task of video monitoring. CCTV (Closed Circuit Television) camera networks generate and transmit huge amounts of data, which makes automatically processing all the information crucial.

Video surveillance processing is an important tool for detecting shoplifting, as video analytics can automatically analyze large amounts of video data, detect illegal activities, and send real-time alerts to security guards.

For this, various Machine Learning (ML) algorithms are used. Such ML models are trained on comprehensive datasets of shoplifting, allowing them to identify thief patterns and classify their actions based on certain features.

Convolutional Neural Networks (CNNs) are a powerful tool for image classification and have significantly advanced video processing in recent years. However, it should be noted that video classification requires considering both spatial and temporal characteristics of objects.

In [1], the authors use a pre-trained 3D CNN model to extract video features. Then they use a fully connected neural network to build a regression predicting whether an action was "normal" or "abnormal". The authors tested its model on videos of thefts, fights, and traffic accidents on the UCF-Crime dataset [2]. In [3], the authors presented an approach to real-time anomaly detection using 3D CNN. In the paper [4], the authors used 3D CNN to extract features from video data and classify some events.

Another approach to detecting anomalies in video surveillance is a combination of convolutional and recurrent neural networks (RNNs). It allows the creation of models for extracting spatial and temporal features from a video sequence.

In [5], the authors analyze the existing video classification methods and found that the combination of CNN and RNN works better than methods that use only CNN. For example, in [6], authors use 3D CNN to analyze the presence of violence in surveillance video. To solve this problem, applying only CNN may not be sufficient, so the authors utilize RNN in the model to encode relevant temporal information.

Similarly, in [7], authors use 3D CNN to extract spatial features from video data and LSTM (Long Short-Term Memory) to classify human actions.

Authors of [8] apply a convolutional neural network to analyze typical thief movement features and LSTM for training-derived features.

In [9, 10], a hybrid neural network detects shoplifting. It consists of convolutional and recurrent neural networks. This model uses a CNN to extract important features from video frames. In the recurrent network, gated recurrent units were utilized.

This paper aims to create an effective real-time store theft detection model based on video data processing that utilizes the Combined CNN-RNN Model.

2. Methods and materials

Our study uses the UCF-Crime Dataset [2] as input for our experiments. The dataset includes 1900 videos of varying lengths, totalling 128 hours of actual criminal acts, such as abuse, arrest, arson, assault, traffic accidents, burglary, explosion, fight, robbery, shooting, shoplifting, and vandalism. In particular, the dataset includes 28 videos from a retail store and video surveillance cameras containing shoplifting.

For training our neural network, we artificially increased the number of instances by dividing each video into 32 fragments of 3-second duration. The resulting dataset of 896 video fragments was divided into two classes: 155 videos with shoplifting and 741 videos without shoplifting.

Since video recordings contain information about time and space, both types of information need consideration when analyzing video fragments. We believe convolutional and recurrent neural networks are the best architectures for accomplishing the task. Therefore, we chose a combination of these networks to classify the videos.

Classified video clips of equal duration and many video frames were used as input data. Each frame sequence was marked as either "0" (not shoplifting) or "1" (shoplifting). The marked set of frame sequences was used as a training sample. The features were obtained for each object through a hybrid neural network to train a classifier, which was then used to classify new objects.

In order to improve the quality of the model, we decided to use preprocessing of video fragments. Namely, we used a combination of histograms of oriented gradients and optical flows.

A fast and highly promising way to represent images for classification is by utilizing HOG features. These features were extensively used in pedestrian identification and have remained a reliable technique for feature extraction. HOG features rely on the distribution of gradient angles and magnitudes, making them resistant to minor shifts in lighting and colour variations in visual data [11].

The apparent motion of objects in a visual scene, caused by the motion of a camera or object or both, can be described as optical flow. When a camera records a scene over a certain time, the resulting image sequence can be represented as a function of gray values at the pixel position (x, y) and time t. If the camera or an object within the scene moves, it causes a time-varying shift in the gray values of the image sequence. The optical flow field in the image domain is the resulting two-dimensional pattern of apparent motion [12].

3. Computational Experiment and Results

We develop an algorithm to detect shoplifting, which can be seen as a classification problem. In order to achieve a sufficiently high level of accuracy for our classifier, we conducted model tunning, including extensive research and experiments, including selecting the video classification method, searching for a suitable data set, determining optimal data processing, and configuring neural networks and their parameters.

We describe our main experiment below. Due to the small size of the dataset (only 310 instances) for the non-trivial task of human action classification, we artificially enlarged the dataset. Each video fragment was horizontally mirrored to achieve this, resulting in 620 instances. Additionally, two more copies were generated from each of the 620 fragments, rotated 5 degrees to the left and right, respectively, which resulted in a total of 1860 video fragments.

To enhance the dataset for our purposes, a combination of a histogram of directional gradients and optical flow was applied as preprocessing to the initial set of video fragments. As a result, two sets of 1860 video fragments were obtained: one initial and one preprocessed. The model processed these two sets in parallel, and the results were combined by averaging.

For feature extraction, MobileNetV3Large was used as a convolutional neural network. The 'imagenet' weights for the model are stored in Keras. The values of calculated accuracy, recall, and F1-scores for each of the classes are presented in Table 1. Thus, the accuracy of the conducted classification is 92%. Since the sample was balanced and considering the presented values of the metrics, this value fully characterizes this result of classification.

Table 1. Accuracy, precision, recall and F1-score values

metric	**precision**	**recall**	**F1-score**
Not Shoplifting	0.90	0.95	0.93
Shoplifting	0.95	0.90	0.92
Accuracy			0.92

4. Conclusions

This study aimed to develop a classifier for identifying shoplifting cases in video data from security cameras. A hybrid neural network classifier involving convolutional and recurrent neural networks was offered to achieve the goal.

The UCF-Crime dataset was chosen as the training dataset, containing videos depicting shoplifting incidents. The major class was under-sampled to address

the issue of unbalancing the dataset, while the video data set was artificially enlarged. Additional experiments were conducted using a pre-trained CNN. A neural network with gated recurrent units was utilized for the sequence classification of video clips.

The classifier exhibited a high classification accuracy of 92%, which is several percent higher than the accuracy of models presented in previous relevant studies. Furthermore, our trained classifier demonstrates high performance, enabling its use in real-time applications. Future research will focus on the practical implementation of the proposed model in shopping malls.

Acknowledgment

The work was supported in part by Beethoven Grant No. DFG-NCN 2016/23/G/ST1/04083.

References

[1] Nasaruddin N., Muchtar K., Afdhal A., Dwiyantoro A.P.J., *Deep anomaly detection through visual attention in surveillance videos*, vol. 7, no 1, p. 87, ISSN 2196-1115, doi: 10.1186/s40537-020-00365-y.

[2] Sultani W., Chen. C., Mubarak S., *Papers with code – UCF-crime dataset*. https://paperswithcode.com/dataset/ucf-crime

[3] Sultani W., Chen C., Shah M., *Real-world anomaly detection in surveillance videos*, [In:] *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, doi: 10.1109/CVPR.2018.00678.

[4] Martínez-Mascorro G.A., Abreu-Pederzini J.R., Ortiz-Bayliss J.C., Garcia-Collantes A., Terashima-Marín H., *Criminal intention detection at early stages of shoplifting cases by using 3d convolutional neural networks*, vol. 9, no 2, p. 24, ISSN 2079-3197, doi: 10.3390/computation9020024.

[5] Islam M.S., Sultana S., Roy U.K., Mahmud J.A., *A review on video classification with methods, findings, performance, challenges, limitations and future work*, *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 2021, vol. 6, no 2, pp. 47–57, doi: 10.26555/jiteki.v6i2.18978.

[6] Li J., Jiang X., Sun T., Xu K., *Efficient violence detection using 3d convolutional neural networks*, [In:] *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–8, doi: 10.1109/AVSS.2019.8909883.

[7] Alfaifi R., Artoli A.M., *Human action prediction with 3d-CNN*, *SN Computer Science*, 2020, vol. 1, no 5, p. 286, doi: 10.1007/s42979-020-00293-x.

[8] Ansari M.A., Singh D.K., *ESAR, an expert shoplifting activity recognition system*, *Cybernetics and Information Technologies*, 2022, vol. 22, no 1, pp. 190–200, doi: 10.2478/cait-2022-0012.

[9] Kirichenko L., Radivilova T., Sydorenko B., Yakovlev S., *Detection of shoplifting on video using a hybrid network*, vol. 10, no 11, p. 199, ISSN 2079-3197, doi: 10.3390/computation10110199.

[10] Kirichenko L., Sydorenko B., Radivilova T., Zinchenko P., *Video surveillance shoplifting recognition based on a hybrid neural network*, [In:] *2022 IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT)*, pp. 44–47, doi: 10.1109/CSIT56902.2022.10000545.

[11] Torrione P.A., Morton K.D., Sakaguchi R., Collins L.M., *Histograms of oriented gradients for landmine detection in ground-penetrating radar data*, *IEEE Transactions on Geoscience and Remote Sensing*, 2014, vol. 52, no 3, pp. 1539–1550, doi: 10.1109/TGRS.2013.2252016.

[12] Shah S.T.H., Xuezhi X., *Traditional and modern strategies for optical flow: an investigation*, *SN Applied Sciences*, 2021, vol. 3, no 3, p. 289, doi: 10.1007/s42452-021-04227-x.

Spotting Advertisements from Above: Billboard Detection and Segmentation in UAV Imagery

Bartosz Ptak[0000−0003−1601−6560], **Jan Dominiak**[0009−0007−7914−1859], **Marek Kraft**[0000−0001−6483−2357]

Poznań University of Technology
Institute of Robotics and Machine Intelligence
Piotrowo 3A, 60-695 Poznań, Poland
bartosz.ptak@doctorate.put.poznan.pl

DOI:10.34658/9788366741928.9

Abstract. *In this work, deep-learning methods were researched for billboard detection in urban environments. Billboards are one of the adversarial visual pollutants occurring in cities, causing over-saturation of visual stimulation. Due to this, we develop an algorithm that helps in the analysis and management of urban space. We utilise near real-time object detection methods to detect and segment them on images registered by unmanned aerial vehicles (UAVs). Research is based on recent algorithms from the YOLO family with modified heads for the instance segmentation task. We gathered images and prepared hand-annotated labels for training and evaluation purposes of deep learning approaches. We reached the mAP@0.5 metric of 0.61 for detection and 0.60 for segmentation, enabling us to develop smart city applications.*

Keywords: *object detection, segmentation, deep learning, YOLO, UAV*

1. Introduction

With the still growth and evolution of cities, visual pollution from advertising has become an increasingly pressing issue in urban environments. One specific type of urban visual pollution is billboards located along roads and on the walls of buildings. These structures often overpower a city's natural and architectural beauty, leading to an over-saturation of visual stimuli and a decline in overall quality of life [1] and driver safety [2].

Since visual pollution is a subjective issue, its identification and assessment are relatively complex. However, to handle this issue, many solutions were proposed. In [3], a web system and smartphone application were created to estimate pollution from advertisement boards. Next, in recent years, deep learning-based methods have been developed. Authors of [4] introduce a machine learning-based classification model for automatic visual pollutant identification. Moreover, in [5],

authors employ an object detection algorithm to recognise textile-based visual pollutants automatically. Nevertheless, the quality analysis of each advertised surface is still a challenging task and requires improving methods and developing new ways of data acquisition.

This paper presented deep-learning models for billboard detection and instance segmentation based on near real-time methods. The task is performed on a dataset collected in an urban environment from Unmanned Aerial Vehicle (UAV). The algorithm can be helpful not only for large-scale billboard pollution analysis in smart cities but also for an inventory of the correctness of the placement of billboards.

2. Methods

To reach our requirements – high-quality instance segmentation and online processing, we based our approach on state-of-the-art methods for near real-time object detection tasks. We employ Convolutional Neural Network (CNN) models from the You Only Look Once (YOLO) family with an additional head for instance segmentation. The extension generates a probability mask that for each pixel of the image contains the probability of belonging to each declared class. Then scale interpolation is applied to the mask to obtain the same resolution as an input image. Finally, having bounding boxes from the classical YOLO head, the mask is processed in each bounding box area to obtain final segmentation. Mask values outside bounding boxes are ignored.

In benchmark, we compare YOLOv5 [6] and YOLOv7 [7]. Releasing of YOLOv5 introduced several improvements over the previous versions of YOLO, including faster inference times, improved accuracy, and better handling of small objects. YOLOv5 achieved state-of-the-art results on several benchmark datasets and quickly became a popular choice for object detection tasks. One major change introduced by YOLOv7 is the use of a new backbone architecture, which allows for better feature extraction and improves the accuracy of object detection. Additionally, YOLOv7 introduces new training techniques and post-processing methods, which further improve the performance of the algorithm.

To compare the models' accuracy, mean Average Precision (mAP) was applied. It is a widely used metric to evaluate the performance of object detection and segmentation models. mAP computes the average precision (AP) over different Intersection over Union (IoU) thresholds (from 0.5 to 0.95 with the step of 0.05) and then takes the mean over all the classes. In the object detection case, bounding boxes are compared, while in segmentation tasks, mAP is used to evaluate the accuracy of predicting object boundaries.

In the research, we use a self-collected billboard dataset. It is a collection of 1404 images captured by drones in an urban environment. It contains three classes: freestanding billboards, wall-mounted billboards, and road-sign. Please

Figure 1. Example image from the dataset with hand-labelled masks and bounding boxes. Source: own work.

note the last class is an additional option that helps distinguish large road signs from billboards, as found in the initial tests. The images were hand-annotated with bounding boxes and segmentation masks. In experiments, we perform some pre-processing steps for images. Due to the substantial image perspective distortions, we cut them from the top to improve accuracy by removing distant objects. After that, images are cropped to two square regions. Furthermore, we split data to train, validate, and test subsets clustering based on their GPS locations. This is due to small variations between image frames at a given location, the presence of which in different subsets would cause the results to be biased. Sample images from the dataset are presented in Figure 1.

3. Results

In our experiments, we measure mean Average Precision (mAP) for both detection and instant segmentation tasks. In the object detection case, the YOLOv5 algorithm achieves better metrics only for the road sign class, outperforming YOLOv7 by 0.02. In other classes and overall YOLOv7 obtained higher metrics. As shown in Table 1, the higher $mAP@0.5$ is 0.61, while $mAP@0.5-0.95$ is 0.47. In the case of the instance segmentation task, version 7 of YOLO achieved the highest scores for all classes. They are 0.60 and 0.42, respectively, for $mAP@0.5$ and $mAP@0.5-0.95$ (Table 2). This demonstrates that while YOLOv5 may have a slight advantage for certain classes in object detection, overall, YOLOv7 outperforms YOLOv5 in both object detection and instance segmentation tasks, as evidenced by the higher mAP scores achieved in the experiments.

Table 1. Metrics achieved for the object detection task.

Metric	**YOLOv5-Detection**		**YOLOv7-Detection**	
	mAP@0.5	mAP@0.5-0.95	mAP@0.5	mAP@0.5-0.95
Freestanding billboard	0.70	0.52	0.74	**0.57**
Wall-mounted billboard	0.44	0.26	0.46	**0.32**
Road sign (additional class)	0.66	**0.54**	0.63	0.52
All together	0.60	0.44	0.61	**0.47**

Table 2. Results obtained for instance segmentation task.

Metric	**YOLOv5-Segmentation**		**YOLOv7-Segmentation**	
	mAP@0.5	mAP@0.5-0.95	mAP@0.5	mAP@0.5-0.95
Freestanding billboard	0.70	0.49	0.74	**0.53**
Wall-mounted billboard	0.42	0.24	0.44	**0.28**
Road sign (additional class)	0.66	0.40	0.63	**0.45**
All together	0.59	0.38	0.60	**0.42**

Additionally, a visual comparison was performed. The effect of it is included in Fig. 2. We can observe, that YOLOv7 usually generate more certain predictions. To sum up, both these algorithms show robustness in billboard detection. The visual comparison of the two algorithms further confirms the superiority of YOLOv7 over YOLOv5 in terms of generating more certain and accurate predictions. This, coupled with the high mAP scores achieved in the experiments, highlights the robustness of both algorithms for billboard detection.

Figure 2. Visual comparison of results. On left side YOLOv5, on right – YOLOv7. Source: own work.

4. Conclusions

In this work, we have shown it is possible to detect and perform instance segmentation of billboards in an urban environment, based on real-time deep learning algorithms. The researched methods were trained and validated on our own dataset captured from UAV-view. Further experiments will focus on feature aggregation between frames. In particular, we are interested in the application of deep neural networks able to re-identify the same billboards on other recordings. Overall, this work paves the way for more advanced and efficient billboard detection systems. Future research could explore the potential of using deep neural networks to geo-locate the billboards based on GPS data and re-identify the same billboards in different recordings, enabling more accurate and reliable analysis of billboard advertising effectiveness over time.

Acknowledgment

This work was supported by the Polish Ministry of Science and Higher Education under Grant 0214/SBAD/0244.

References

[1] Kelishadi R., *Environmental pollution: health effects and operational implications for pollutants removal*, *Journal of Environmental and Public health*, 2012, vol. 2012, no 341637, doi: 10.1155/2012/341637.

[2] Jian A., *Road Traffic Safety Management of Visual Pollution By Outdoor Advertisements*, Ph.D. thesis, Universiti Teknologi Malaysia Skudai, Malaysia, 2020.

[3] Chmielewski S., Samulowska M., Lupa M., Lee D., Zagajewski B., *Citizen science and webgis for outdoor advertisement visual pollution assessment*, *Computers, Environment and Urban Systems*, 2018, vol. 67, pp. 97–109, ISSN 0198-9715, doi: https://doi.org/10.1016/j.compenvurbsys.2017.09.001.

[4] Ahmed N., Islam M.N., Tuba A.S., Mahdy M., Sujauddin M., *Solving visual pollution with deep learning: A new nexus in environmental management*, *Journal of environmental management*, 2019, vol. 248, p. 109253.

[5] Tasnim N.H., Afrin S., Biswas B., Anye A.A., Khan R., *Automatic classification of textile visual pollutants using deep learning networks*, *Alexandria Engineering Journal*, 2023, vol. 62, pp. 391–402, ISSN 1110-0168, doi: https://doi.org/10.1016/j.aej.2022.07.039.

[6] Jocher G., *ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation*, 2022, doi: 10.5281/zenodo.7347926.

[7] Wang C.Y., Bochkovskiy A., Liao H.Y.M., *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*, *arXiv preprint arXiv:2207.02696*, 2022, doi: 10.48550/arXiv.2207.02696.

Transformers Neural Networks Applications in Different Computer Vision Tasks

Andrzej Brodzicki[0000−0001−7713−526*X*],
Michał Piekarski[0000−0001−9391−4263],
Aleksander Kostuch[0000−0003−1242−9851],
Filip Noworolnik[0000−0000−0000−0000],
Maciej Aleksandrowicz[0000−0003−3388−5653],
Anna Wójcicka[0000−0001−8060−2009],
Joanna Jaworek-Korjakowska[0000−0003−0146−8652],

Akademia Górniczo-Hutnicza im. Stanisława Staszica
Department of Automatic Control and Robotics
al. Mickiewicza 30, 30-059 Kraków, Poland
brodzicki@agh.edu.pl

DOI:10.34658/9788366741928.10

Abstract. *Transformers architectures are one of the latest inventions in the field of deep learning. Originally dedicated to NLP, they begin to find use in computer vision too. In this paper, we briefly describe the idea behind vision transformers and present a few examples, where we utilised them in our research, focusing on the field of medical images and autonomous driving. We show, that vision transformers can be used in various tasks, such as detection or classification, as well as explain how some of their drawbacks can be mitigated with a transfer learning approach.*
Keywords: *transformers, neural networks, computer vision, classification, detection, segmentation*

1. Introduction

Deep neural networks are a rapidly growing field in which many research groups are constantly working on not only improving existing models but also on developing new architectures. There is growing interest in enhancing models' capabilities from the research community and engineers alike as they solve tasks from various, sometimes very distant areas of life. One such large field is computer vision (CV), where convolutional neural networks (CNNs) have so far been the first choice model for visual data. Recent work has shown however that vision transformer models (ViT) can achieve comparable or even superior performance on image classification and recognition tasks due to their ability to capture long-range dependencies within an image. In this work, we present the results of the

latest work of our research group which focuses on employing ViT to solve tasks in various fields such as medical images and autonomous driving.

This paper is organized as follows: in Section 2 we provide a brief description of vision transformer models and attention mechanisms, in Section 3 we present the examples of how we apply them to different computer vision problems.

2. Vision Transformers

2.1. Related works

Transformers were introduced in 2017 by a team at Google, primarily to be used in the field of natural language processing (NLP). A.Vaswani et al. proposed a sequence transduction model based entirely on attention, replacing the recurrent layers most commonly used in encoder-decoder architectures with multi-headed self-attention [1]. Inspired by the major success of transformer architectures in the field of NLP, researchers have recently applied transformer to computer vision tasks [2]. The vision transformer model applies a pure transformer directly to sequences of image patches to classify the full image (Figure 1). Dosovitskiy et al. achieved state-of-the-art performance on multiple image recognition benchmarks [3]. In addition to image classification, transformer has been utilized to address a variety of other vision problems, including object detection [4], semantic segmentation [5], image processing [6], and video understanding [7]. Nowadays we observe a rapid increase in the number of transformer-based vision models, like the detection transformer (DETR) proposed by Carion et al. [8] redesigning the framework of object detection or MaX-DeepLab – the first end-to-end model for panoptic segmentation with mask transformers [9].

Figure 1. In vision transformers we calculate attention between individual patches. Source: own work.

2.2. Attention Mechanisms

The main idea behind Transformers models is an attention mechanism. The network maps relations between different words (in NLP), by checking all the possible combinations between them. Those vectors go through a block called multiple head attention, where we calculate the correlation between words, repeat it "from the point of view" of all the other words and average it – creating attention vectors for every single word. In practice, popular libraries, such as Keras, have those mechanisms implemented as dedicated layers (MultiHeadAttention).

In vision transformers, the network behaves similarly, but when the texts consisted of individual words, the images have to be converted into patches (see Fig. 1). They are mapped with positional embedding and multi-head attention just like text data. The final feature vectors can then be processed like in any other network.

One of the benefits of transformers is that they are easily interpretable. Their basic structure is based on attention mechanisms, so they already have the attention calculated. Transformers look at the whole image and map the relations, instead of focusing on the small neighbourhood like the CNNs. It is more intuitive than the CNNs hierarchical representation. In fact, this is what humans do.

2.3. Transfer learning

It is worth mentioning that transformers require a very large amount of data to beat CNNs. If we don't have those, it is usually better to keep using basic CNNs for most of the tasks. However, one of the well-established ways to solve the problem of small datasets is a transfer learning methodology. Similar to CNN, we can use an already pre-trained transformer model and retrain some layers for our task.

There are currently no pre-trained transformers available in popular python libraries (like keras.applications), but many online communities offer a way to share weight files for state-of-the-art architectures. One such community is Hugging Face, with hundreds of pre-trained weights available for different types of transformers, both for NLP and vision ones.

3. Applications

In this section, we present three examples of our research, where we have used vision transformers for different image processing tasks, such as classification or detection, in the field of medical imaging and autonomous driving.

3.1. Diabetic Foot Ulcer Detection

While in recent years, Vision Transformers (ViT) [3] have become a popular choice for image classification, their use in object detection and semantic segmen-

tation is still gaining traction. These tasks also benefit from the transformers' ability to capture global dependencies in an image, however, they require the model to output not only a class label but also a bounding box or a segmentation mask. To achieve this, the architecture require additional output heads. Additionally, modifications to the loss function has to be made to optimize for the specific task.

One of the most promising detection methods based on transformers is DETR (DEtection TRansformer) [4]. It is an end-to-end framework, which utilizes a CNN backbone, encoder-decoder architecture, and bipartite matching between the ground-truths and the detections. DETR is especially efficient for objects with varying sizes, thanks to replacing anchor-based processing with bipartite matching.

We compared the performance of the CNN-based YOLOv4 network with DETR in our work for the Diabetic Foot Ulcer Segmentation Challenge 2022 [10]. We have integrated the ensemble of these methods with U-Net segmentation. The challenge dataset consisted of a train and test set of 2000 images each. The sizes of ulcers varied between 0.04% and 35.04% of the image's total size. This makes traditional anchor-based methods either not efficient or not accurate. We used a pre-trained DETR model, which we fine-tuned on the challenge dataset. Examples in Fig. 2 show that DETR was able to handle different sizes of ulcers. As can be seen in Tab.1, DETR outperformed YOLOv4 in almost every metric.

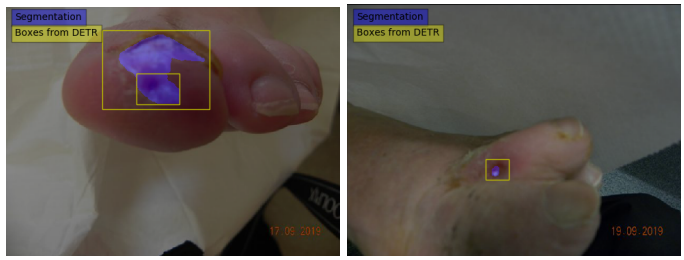


Figure 2. Ulcer examples from the challenge validation set that were missed by YOLOv4 and detected by DETR [10].

Table 1. Comparison of results obtained on the validation set from the challenge.

Method	MeanOverlap	UnionOverlap	DiceCoefficient	VolumeSimilarity	FalseNegativeError	FalsePositiveError	JaccardCoefficient
YOLOv4	0.5556	0.4746	0.5556	**0.2473**	0.3975	0.3931	0.4746
DETR	**0.5808**	**0.4790**	**0.5808**	0.0139	**0.3622**	**0.3293**	**0.4790**

3.2. Vehicles and Skin Lesion Classification

Vehicle make and model recognition (VMMR) is a key task for automated vehicular surveillance (AVS) and various intelligent transport system (ITS) applications. Cars seem to be a complex object to recognize due to their diverse

construction and different perspectives. The Stanford Cars dataset contains 16185 images, split equally between train and test subset, that have been divided into 196 car classes. Given the size of the dataset, we used a model based on the original Vision Transformer (ViT) architecture [3]. Its input are images of size 224*224 and patch sizes are 16. An accuracy of 82.37% was achieved on the test set. Compared to the best CNN solutions, achieving around 93%, this is a satisfactory result taking into account the advantages of a transformer-type network. There are still a few ways to improve it: most notably data augmentation and extensive pre-processing.

To test the performance of a vision transformer on medical data we have chosen dermoscopic images, as melanoma is one of the deadliest skin cancers, with many computer-aided diagnostic solutions and a large public database. We used a ViT-B/32 model pre-trained on imagenet21k dataset. We have fine-tuned it for the task of 7-class classification, from the ISIC 2018 challenge. We used 4482 train images, and 1492 for both validation and testing. We achieved 0.827 accuracy, which is far from from state-of-the-art, but good enough for such a small subset, with no image augmentation.

Figure 3. Vehicles and melanoma examples from the sets used in experiments.

4. Conclusion

In this paper we have presented three applications of vision transformers in different computer vision tasks, such as classification and detection, in different domains. Our experiences coincide with a well established view that transformers are very effective models, but require a lot of data to achieve good results. It is highly recomended to use transfer learning whenever it is possible. These applications were one of the first uses of vision transformers in those tasks.

Acknowledgment

Research project partly supported by program “Excellence initiative – research university” for the AGH University of Science and Technology.

References

[1] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I., *Attention is all you need*, 2017, doi: 10.48550/arXiv.1706.03762.

[2] Han K., Wang Y., Chen H., Chen X., Guo J., Liu Z., Tang Y., Xiao A., Xu C., Xu Y., Yang Z., Zhang Y., Tao D., *A survey on vision transformer*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023, vol. 45, no 1, pp. 87–110, doi: 10.1109/TPAMI.2022.3152247.

[3] Dosovitskiy A., Beyer L., Kolesnikov A., Weissenborn D., Zhai X., Unterthiner T., Dehghani M., Minderer M., Heigold G., Gelly S., Uszkoreit J., Houlsby N., *An image is worth 16x16 words: Transformers for image recognition at scale*, 2020, doi: 10.48550/arXiv.2010.11929.

[4] Zhu X., Su W., Lu L., Li B., Wang X., Dai J., *Deformable detr: Deformable transformers for end-to-end object detection*, 2021, doi: 10.48550/arXiv.2010.04159.

[5] Zheng S., Lu J., Zhao H., Zhu X., Luo Z., Wang Y., Fu Y., Feng J., Xiang T., Torr P.H.S., Zhang L., *Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers*, 2021, doi: 10.48550/arXiv.2012.15840.

[6] Chen H., Wang Y., Guo T., Xu C., Deng Y., Liu Z., Ma S., Xu C., Xu C., Gao W., *Pre-trained image processing transformer*, 2021, doi: 10.48550/arXiv.2012.00364.

[7] Zhou L., Zhou Y., Corso J.J., Socher R., Xiong C., *End-to-end dense video captioning with masked transformer*, 2018, doi: 10.48550/arXiv.1804.00819.

[8] Carion N., Massa F., Synnaeve G., Usunier N., Kirillov A., Zagoruyko S., *End-to-end object detection with transformers*, [In:] *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, Springer, pp. 213–229, doi: 10.48550/arXiv.2005.12872.

[9] Wang H., Zhu Y., Adam H., Yuille A., Chen L.C., *Max-deeplab: End-to-end panoptic segmentation with mask transformers*, 2021, doi: https://doi.org/10.48550/arXiv.2012.00759.

[10] Kucharski D., Kostuch A., Noworolnik F., Brodzicki A., Jaworek-Korjakowska J., *DFU-ens: End-to-end diabetic foot ulcer segmentation framework with vision transformer based detection*, [In:] M.H. Yap, C. Kendrick, B. Cassidy (eds.), *Diabetic Foot Ulcers Grand Challenge*, Springer International Publishing, Cham, pp. 101–112, doi: 10.1007/978-3-031-26354-5_9.

Weak Supervision in Enemy Detection Based on Computer Game Output Video Stream

Jakub Rajtar, Dominik Szajerman[0000−0002−4316−5310]

Lodz University of Technology
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
dominik.szajerman@p.lodz.pl

DOI:10.34658/9788366741928.11

Abstract. *This work contains a solution for image classification and enemy detection in the output video stream of a computer game. Weak supervision was used to achieve the goal. It shows that an image dataset with a certain number of incorrect classification labels can be used to correctly build a classification model that distinguishes between images containing and not containing an enemy. Based on the results of such classification stage and the use of class activation maps, a method for detecting enemies on positively classified images was proposed. The tedious process of image labeling, which is necessary for supervised learning, does not occur here.*
Keywords: *computer game testing, computer vision, weak supervision*

1. Introduction

Testing games is a time-consuming and therefore expensive stage throughout their production. That is why machine learning is increasingly used in the task of automatic game testing. A programmed agent that would imitate the player's behavior should use the same mechanisms of interaction with the game as the player (input and output systems). In this paper, the tasks of classifying and detecting an enemy are considered. Solving the first one allows for determine which picture frames contain the enemy or enemies. After solving the first, the second problem becomes possible to solve. The proposed method based on weak supervision allows the training with use of a dataset that may contain some degree of inaccurate data indicating the presence of an enemy and at the same time with no labels indicating its position on the screen.

2. Related work

Object detection is one of the fundamental issues of computer vision, recently it has been using machine learning extensively, in particular the convolutional neural networks employing supervised learning with labeled datasets. Such solutions

are precise and efficient, unfortunately the preparation of labeled data sets can be expensive and difficult, and even impossible for some application areas. Weak supervision could be the answer fot these drawbacks. The labels for the training set are available to a limited extent or have some imperfections. In the object detection problem, this usually means that the training set has classes labels, but does not contain information about the location of objects. An example of the use of weak supervised learning can be the use of a CNN with an architecture containing a special layer that performs the max pooling [1] at the network output, which determines the location of detected objects [2].

3. Method

The proposed method consists of two stages. The first is the classification of images. It provides two possible results: the image **contains** the enemy or it **does not**. The training is based on inaccurate training dataset. The second stage is for the enemy detection. As its result it provides the enemy's location given as bounding box. The classic computer game "Doom" was chosen as the research environment for the developed solution. It is a First Person Shooter game in three-dimensional space.

The training set was based on the video game stream recording. It consisted of two classes: 3,000 images labeled as containing the enemy (positive) and 3,000 images labeled as containing no enemy (negative). The labeling process was automatic and therefore inaccurate. Thanks to this, the presented method could be tested in the context of weak supervision. To assess the quality of the dataset and its suitability for weak supervision, a subset of 60 images for each class was randomly selected and manually classified. Then its results were compared with the original classification. The average quality for the positive class was 90% and for the negative class it was 80%. Such dataset was used for training for the classification. In addition, the dataset contained 100 manually labeled images for each of the two classes. It has been made sure that the selected frames of video stream are not ambiguous in any way. They were later used as a test set.

The presented dataset was the basis for **training a binary classifier**, whose task was to determine whether there is an enemy in the image presented to him. The classifier was built on the basis of CNN [3]. It consists of three sets of two layers each: a convolution one and a pooling one. The data is then transformed into a one-dimensional form by a flattening layer so that it can be further processed by a fully connected hidden layer (Fig. 1).

The result of the proposed CNN is only binary information about whether there is an enemy in the image, therefore an additional mechanism was needed to determine on the basis of which image features the network made a decision about classification. The next step of the detection pipeline were **the class activation maps**

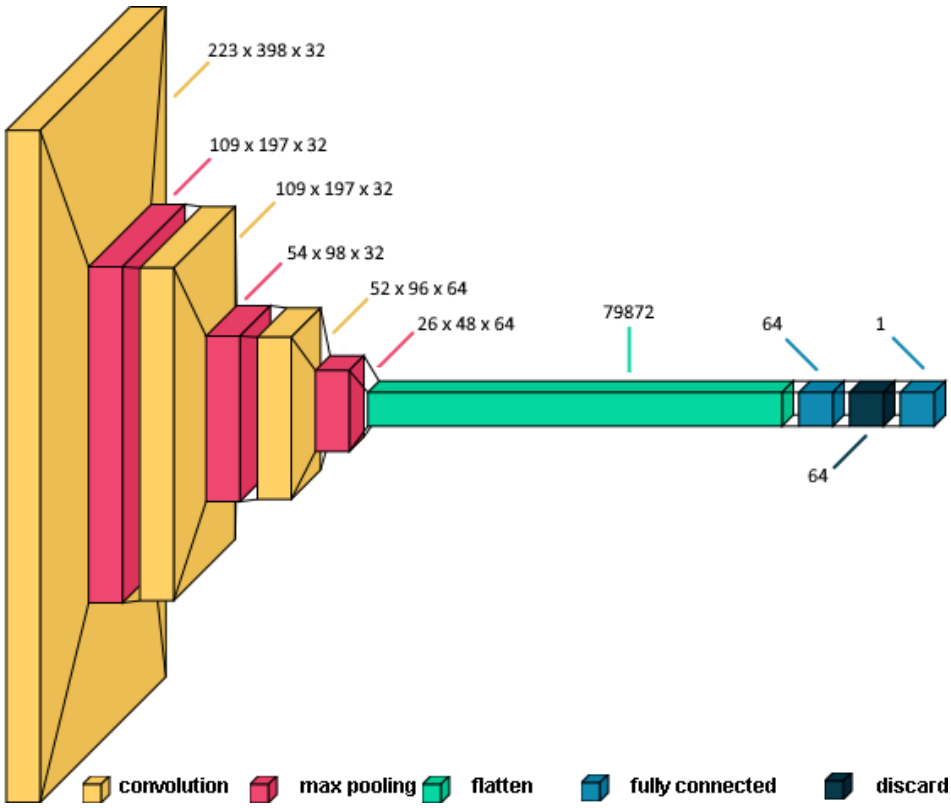


Figure 1: Architecture of the convolutional neural network used, taking into account the type of layer and the size of its output. Source: own work.

[4]. The map has the form of a two-dimensional grid describing numerically the significance of each pixel of the input image for the recognition of a specific class. In the presented solution, to create such a map, the Gradient-weighted Class Activation Mapping (GradCAM) algorithm was used [5].The map is then trasformed to grayscale and after that, using the thresholding operation it is transformed into a binary form. In the next step, around all regions of the received binary image that have not been set to 0, the contours are drawn using the Border Following [6] method. From the polygons thus obtained, the one containing the most intense point of the input heat map was selected. A rectangle was circumscribed on the selected polygon – an enemy position bounding-box.

4. Experiments, results, and discussion

In order to evaluate the part of the process responsible for classifying images into containing enemies and not containing them, the following two neural net-

work models for classification were trained over 30 training epochs: 1000s_30e – network model with 2000 images, 1000 for each positive and negative class and 3000s_30e – network model with 6000 images, 3000 for each class.

The evaluation of the detection of enemies was carried out on the basis of a set of 100 positive class images prepared earlier for the classifier test. To assess the accuracy of the location of enemies determined by the detector, each of the images from the set has been enriched with labels containing the coordinates of the rectangles circumscribed on all enemies in the test images. Then, using the 1000s_30e and 3000s_30e models, the detection of enemies was carried out on positively classified images from the test set. The rectangular bounding boxes obtained in this way were compared with the bounding boxes defined manually by the labels. On this basis, for each such set of data, the value of Intersection over Union [7] was calculated.

Based on the created set of 100 images for each of the classes, a test was performed for both trained models. The test results are listed in Table 1. It can be

Table 1: Binary classification test results for two trained models.

Model	PP	PN	FP	FN	Accuracy	Precision	Sensitivity	Specificity
1000s_30e	89	99	1	11	94%	99%	89%	99%
3000s_30e	95	100	0	5	97.5%	100%	95%	100%

concluded that both models are characterized by high accuracy and precision, exceeding 90%. In both cases, the number of false negatives is significantly higher than the false positives, which means that both models are conservative and are more likely to classify with a negative result in difficult cases. It is also worth noting that the 3000s_30e model, trained on a training set three times greater than the 1000s_30e model, achieved noticeably better results. Model 3000s_30e is able to correctly classify the enemy, even in the pictures, where he is hardly noticeable due to the large distance from the player, or the colors blending in with the surroundings.

The object detection test for two trained models, was carried out on the basis of the test set consisting of 100 examples of the positive class, enriched with labels indicating the correct position bounding boxes of the enemies. The test results are shown in Table 2.

The achieved results clearly show significant differences in the method of detecting enemies by both tested models. The 1000s_30e model attempted to locate the enemy for 89 of the 100 test images, the remaining images falsely classified as negative were not involved in the detection process. The potentially correct position of the enemies has been established for 66 images, of which for 33 it was a position defined by a bounding box with an $IoU \geq 0.5$. That means a relatively high accuracy of the location information. The detector in the second model made

Table 2: Detection test results for two trained models.

Model	Number of detections where $IoU > 0$	$IoU \geq 0.5$	$IoU = 0$	there is no bounding box	$I\bar{o}U_{max}$	$I\bar{o}U_{max}$ for $IoU > 0$	$\bar{S}$
1000s_30e	66	33	22	1	0.33	0.44	583
3000s_30e	90	8	2	3	0.25	0.26	235

an attempt to locate the enemy for as many as 95 of the 100 test images, after rejecting 5 for false negative. It can be noticed that compared to the 1000s_30e model, it is over ten times less detections with the IoU equal to 0. Potentially correct bounding boxes could be found for as many as 90 out of 95 images, but only for 8 frames the IoU value was greater than or equal to 0.5 suggesting that most of the bounding box found were significantly different from the labels.

5. Conclusions

In this paper, a method was developed that that allow the use of weak supervision, convolutional neural networks, and used by them class activation maps to drastically reduce the expenses needed to build the system of classification and detection of enemies that can be used in the game testing process. It was shown that the development of a working object detector is possible without an extensive, manually prepared database of labeled training data. The database building process can therefore be automated based on less accurate solutions. An image classifier, while trained on an imperfect dataset with incomplete and sometimes misleading labels, can be effective and offer satisfactory classification accuracy. Weak supervision learning makes it possible to create a functioning detector well at locating objects in images despite the complete absence of labels describing the position of objects in the training dataset.

References

[1] Chollet F., *Deep Learning with Python*, chap. 5.1.2 *The max-pooling operation*, Manning, ISBN 9781617294433, 2017.

[2] Oquab M., Bottou L., Laptev I., Sivic J., *Is object localization for free? – weakly-supervised learning with convolutional neural networks*, [In:] *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE.

[3] Goodfellow I., Bengio Y., Courville A., *Deep Learning*, MIT Press, 2016. http://www.deeplearningbook.org

[4] Zhou B., Khosla A., Lapedriza A., Oliva A., Torralba A., *Learning deep features for discriminative localization*, [In:] *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, doi: 10.1109/cvpr.2016.319.

[5] Selvaraju R.R., Cogswell M., Das A., Vedantam R., Parikh D., Batra D., *Grad-CAM: Visual explanations from deep networks via gradient-based localization*, [In:] *2017 IEEE International Conference on Computer Vision (ICCV)*, IEEE, doi: 10.1109/iccv.2017.74.

[6] Suzuki S., Abe K., *Topological structural analysis of digitized binary images by border following*, *Computer Vision, Graphics, and Image Processing*, 1985, vol. 29, no 3, p. 396, doi: 10.1016/0734-189x(85)90136-7.

[7] Padilla R., Netto S.L., da Silva E.A.B., *A survey on performance metrics for object-detection algorithms*, [In:] *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, doi: 10.1109/iwssip48289.2020.9145130.

Chapter 2

Data Mining and Machine Learning

Domain Editors:

1. Jerzy Stefanowski, Poznan University of Technology
2. Michał Woźniak, Wrocław University of Science and Technology
3. Ireneusz Czarnowski, Gdynia Maritime University

A Comparison of Shallow Explainable Artificial Intelligence Methods against Grammatical Evolution Approach

Dominik Sepioło[1][0000−0001−7746−3781], **Antoni Ligęza**[1][0000−0002−6573−4246]

[1]*AGH University of Science and Technology*
Department of Applied Computer Science
al. Mickiewicza 30, 30-059 Kraków, Poland
sepiolo@agh.edu.pl, ligeza@agh.edu.pl

DOI:10.34658/9788366741928.12

Abstract. *This paper reports on an ongoing, innovative research in the area of eXplainable Artificial Intelligence (XAI). An XAI task is considered as finding an explanation of the model generated via Machine Learning by identifying the most influential variables for local decision-making. The proposed approach moves the explanatory process to a new, deeper-level dimension. It is oriented towards Model Discovery, i.e. the internal structure and functions of the components. An experiment on Function Discovery via Grammatical Evolution is reported in brief.*
Keywords: *explainable artificial intelligence, grammatical evolution, structural regression, model-based explainable artificial intelligence*

1. Introduction

Machine Learning (ML) is nowadays a well-matured discipline of research with a large set of problem-solving tools and a vast area of practical applications. Many successful projects and implementations in complex domains such as biomedical data analysis, natural language processing or large scale technological systems are reported, and significant progress w.r.t. variety of ML algorithms and tools is observed. However, it seems that the classical ML paradigm as stated itself, consisting in finding a finite-set decision-making classification or value prediction model, remains a bit too restricted, conservative, where little or no progress is observed concerning *problem formulation*. Practically all the ML data repositories are built according to the same simple scheme of *attributive decision tables*, with no other *knowledge-based components*. Especially when encountering some of the most challenging AI issues of today, concerning real understanding of how

intelligent systems work: Model-Discovery for Model-Based Reasoning [1], eXplainable Artificial Intelligence (XAI) [2, 3], Trustworthy Decision-Making, Interpretable and Explainable AI[1], Model-Based Reasoning [1] and others. In order to make a step towards building Model-Driven XAI, incorporation of Knowledge-Based components and more advanced function identification methods (e.g. symbolic regression, grammatical evolution) seems to be necessary and promising.

This short paper is structured as follows: Sect. 2 describes state-of-the-Art in XAI and research motivation. Theoretical aspects of Grammatical Evolution are described in Sect. 3. This is followed by a description of an experiment with function identification in Sect. 4. Concluding remarks are presented in Sect. 5.

2. State-of-the-Art in XAI and Motivation

Explainable Artificial Intelligence is focused on providing solutions, decisions and predictions that can be understood by humans. Majority of current ML techniques (e.g. Deep Learning) are based on *black-box* models. In order to assure an appropriate level of transparency, and further justifiability and trustability, man must be aware of the underlying *rules of the game*.

The current approaches to develop XAI tends towards *shallow models*. By sacrificing accuracy, a simple but interpretable model is built upon the one generated with ML technologies. Its work is demonstrated on a subset of the original input data, and there is no way to incorporate auxiliary knowledge. In [2, 3] a vast, representative selection of the proposed approaches and tools is presented.

Local Interpretable Model-Agnostic Explanations (LIME) are the most prominent example of explanations by simplification. LIME algorithm generates an explanation for an individual prediction by creating a simple, interpretable, linear model that approximates the behavior of the opaque model in the neighborhood of the prediction. SHapley Additive exPlanations (SHAP) are another benchmark technique of explainability. SHAP utilizes a game theoretic approach (*Shapley values*) in order to create an explanation that shows feature importance for each prediction. Other methods include visualization techniques and explanations by example. In [4] we provided a comprehensive, critical overview of the current shallow approaches to XAI.

The aforementioned approaches are shallow because they provide explanations that are at a high level of abstraction and do not involve a deep understanding of the model underlying principles and external declarative knowledge. A born-in feature of such shallow methods is the inability to capture the full underlying complexity of the model. It is important that they should be used only together with other, deep and more transparent, techniques for a more complete understanding of the behavior and structure of the model.

[1] https://www.bmc.com/blogs/machine-learning-interpretability-vs-explainability/

3. A Note on Grammatical Evolution

Grammatical evolution (GE) is an Evolutionary Algorithm (EA) that takes inspiration from the biological evolutionary process to search for solutions to problems [5]. Unlike classical EA, GE combines genetic algorithms with formal language theory. User-defined context-free grammars constrain the structure and syntax of the genome, which allows generating solutions with a well-defined structure. Each solution is evaluated using a fitness value for a given objective function. While GE is used primarily to generate variable-length linear genome encoding of computer programs, it can also be employed for symbolic regression tasks and identification of functional dependencies. There are numerous implementations of GE algorithms; including PyNeurGen, PonyGE and PonyGE2 for Python, GEVA and ECJ for Java, and GELab for Matlab, gramEvol for R and others[2].

4. An Experiment with Different Explainability Methods

Body Mass Index (BMI) is a simple numeric medical parameter that indicates whether the body weight of a person of a given height is within the healthy range. It can be calculated by dividing a person's weight in kilograms by the square of their height in meters.

A series of experiments was performed in order to create models that calculate BMI function. Besides standard ML methods such as Decision Trees and Random Forest models, Grammatical Evolution was applied. For each method we tried to learn the BMI function using initially 10, then 50, and finally 100 observations.

The first applied technique was Decision Tree. For 10 observations the resulting tree had only one node: the root which returned the mean BMI value from input data. The model had 85% accuracy, however, it was overfitted and performed much worse on test data. The increase in training data (50 observations) did not increase accuracy, though the model was more prone to overfitting. Decision tree created with 100 observations performed at around 90% accuracy. Although the advantage of the model was a simple, interpretable structure presented in Fig. 1, the prediction accuracy was disenchanting.

Secondly, Random Forest algorithm was used for BMI calculation. For 10 and 50 observations model accuracy was around 95% but both of the models were overfitted. For 100 training instances, the training accuracy was 97.5% and 93% for test data. As Random Forest is black-box model, we applied LIME and SHAP methods to generate explanations. For some predictions, explanations proposed by those techniques were significantly different. One example of such discrepancy is presented in Fig. 2. LIME explanation for a given instance shows that the person's height (160–169 cm) increases BMI value, while the person's weight

[2]https://en.wikipedia.org/wiki/Grammatical_evolution\#Implementations

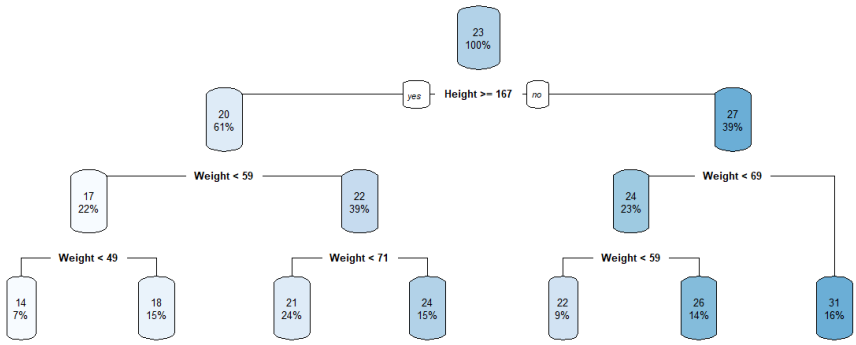


Figure 1. Resulting decision tree. Source: own work.

(61.5–74.6 kg) decreases it. The influence of the Height variable on the predicted value is seven times higher than the influence of the Weight variable. In contrast to LIME method, SHAP explanation for the same instance exhibits that both Weight and Height have a negative contribution to the predicted BMI value. Moreover, the Weight variable contributes significantly more to the result than the Height. Another disadvantage of local explanations technique for BMI prediction is that it assumes that for each prediction BMI coefficient was generated using another formula, while all predictions can be calculated using one simple function.

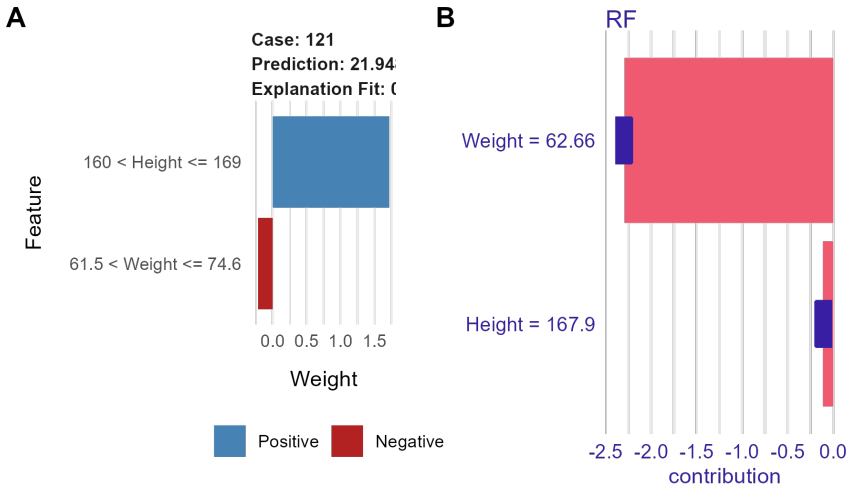


Figure 2. LIME (A) and SHAP (B) explanation. Source: own work.

As there were critical differences in explanations of the model behavior generated by LIME and SHAP techniques for a given prediction, the need for a deeper

understanding of the problem's structure emerged. For this reason, we decided to implement grammatical evolution approach in order to find causality and functional dependencies in data and create a deep, transparent model.

The definition of the proposed context-free grammar for BMI function identification is presented below:

```
<expr> ::= <op>(<expr>, <expr>) | <func>(<expr>) | <var>
<func> ::= `log` | `sqrt`
<op>   ::= "+" | "-" | "*" | "/" | "^"
<var>  ::= Weight | Height | <n>
<n>    ::= -3 | -2 | -1 | 0 | 1 | 2 | 3
```

We allowed basic functions and arithmetic operations and variables: Weight, Height and $n \in \mathbb{Z}$ from $\langle -3, 3 \rangle$ interval as structure elements of final expression. Only 10 observations were enough in order to correctly identify the formula for BMI coefficient. The result of grammatical evolution operations is shown below:

```
BestExpression: Weight * Height ^-2
```

The resulting expression (function) is coherent with the BMI formula. The discovered functional model assures 100% accuracy for every instance of input data and outperforms ML solutions. Furthermore, it provides information that enables understanding the model behavior and structure.

5. Conclusions

The presented experiment shows that using only shallow explainability techniques can lead to inconsistent and misleading explanations for the same prediction. Explanation discrepancies point out the limitations of the abstractive model-agnostic approach. Simple models and explanation techniques that lead to a *deep understanding* of the model behavior and structure should be preferred if possible.

Grammatical Evolution seems to be a promising technology for extension and further developments in identifying interpretable functional structure. It can be successfully applied to the identification of functional dependencies in data. The reported experiment shows advantages of an accurate model generated with a very limited amount of training data over other shallow methods.

References

[1] Magnani L., Bertolotti T., *Springer handbook of model-based science*, Springer, 2017.

[2] Arrieta A.B., Díaz-Rodríguez N., Del Ser J., Bennetot A., Tabik S., Barbado A., García S., Gil-López S., Molina D., Benjamins R., et al., *Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible AI*, *Information Fusion*, 2019.

[3] Guidotti R., Monreale A., Ruggieri S., Turini F., Giannotti F., Pedreschi D., *A survey of methods for explaining black box models*, *ACM Computing Surveys*, 2019, vol. 51, no 5, p. 1–42, doi: 10.1145/3236009.

[4] Sepioło D., Ligęza A., *Towards explainability of tree-based ensemble models. A critical overview*, [In:] W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, J. Kacprzyk (eds.), *New Advances in Dependability of Networks and Systems*, Springer International Publishing, Cham, 2022, pp. 287–296.

[5] Ryan C., O'Neill M., Collins J.J. (eds.), *Handbook of Grammatical Evolution*, Springer, 2018, ISBN 978-3-319-78716-9, doi: 10.1007/978-3-319-78717-6.

Clustering Dilemmas – A Study of the Request of Homogenicity within Clusters Versus Diversity Between Clusters

Mieczysław Alojzy Kłopotek[0000−0003−4685−7045]

Institute of Computer Science
Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
klopotek@ipipan.waw.pl

DOI:10.34658/9788366741928.13

Abstract. *An interplay between the requirements of within-cluster-homogenicity and between-clusters-diversity is investigated. It is shown that taking the requirements of homogenicity and diversity makes the clustering an easy task, but these requirements are rarely matched in the practise.*
Keywords: *clustering, artificial intelligence*

1. Introduction

Clustering is most frequently understood as splitting data into several subsets such that each of these clusters consists of data objects with high intra-cluster-similarity and low inter-cluster-similarity. This is the general assumption, as visible in majority of papers see e.g. [1], [2].

If we take this assumption seriously, then the distance of a datapoint to other datapoints of the same cluster should be lower than the distance of the same datapoint to datapoints of other clusters. This assumption fits the definition of so-called *nice clustering* [3]. If we push further and require that the minimum separation between clusters be larger than the maximum cluster diameter, then we speak about *perfect clustering* [3]. If we extend this requirement not only to datapoints but to the entire hyperballs centered at cluster gravity center and encompassing all cluster datapoints, then we speak about *perfect-ball-clustering* [4]. Provably, incremental k-means algorithm introduced in [3], can discover the perfect ball clustering if it exists [4] and if we know the correct number of clusters in advance. It shall be stressed, however, that incremental k-means does not seek to optimize the traditional k-means cluster quality function.

This paper presents a study of k-means properties when dealing with this most simplistic case of clusters. The merit of studying such a trivial case of clustering is two-fold. First, the concept of a cluster is not so clear in general, as one

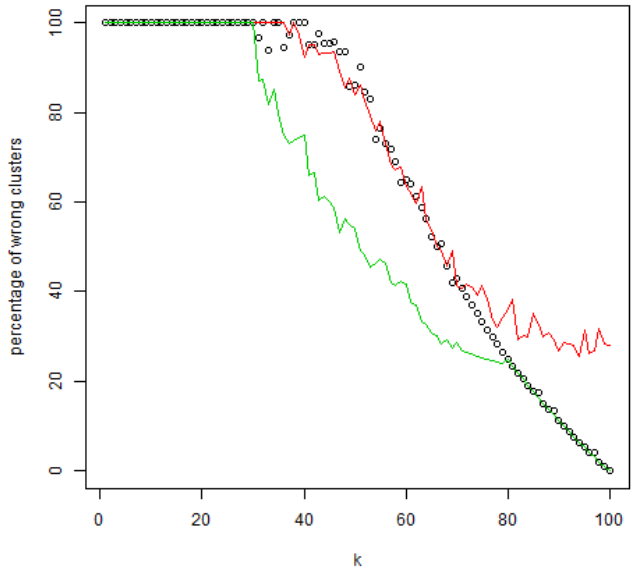


Figure 1: Search for the true number of clusters, starting with k=2. Source: own work.

may expect. So the study of clustering should start with conceptual investigation of the case when the concept seems to be clear-cut. Second, it turns out that the well-studied and widely applied k-means algorithm may fail under these ideal circumstances. We point at an algorithm based on k-means that is well suited under such circumstances. We hope that it may constitute a hint for elaboration of some in-between versions for leaning k-means towards human cluster expectations.

2. The discovery of k clusters for predefined k

A cluster shall contain points that are similar to one another and dissimilar from points in other clusters. What does it mean for distance based clustering? Distances within the cluster shall be smaller than distances to outside elements. So the nice clustering is a must. In the context of centric clusterings, amorphisity should be assumed. This implies that the clusters should be enclosed into hyperballs centered at their center and the distances between these hyperballs shall be bigger than the distances within the balls (that is diameters).

Let us investigate the case when the clusters are obtained by a uniform sampling a ball with radius R, and various balls have distances between centers of $4R$

Table 1: Percentage of erroneously detected clusters by k-means on a $n_1 \times n_2$ grid with $k = n_1 \cdot n_2$.

n_1/n_2	2	3	4	5	6	7	8	9	10
2	0.000	25.000	0.000	15.000	12.500	21.430	9.375	16.670	22.500
3	25.000	0.000	12.500	20.000	8.335	21.430	18.750	22.220	20.000
4	0.000	12.500	18.750	15.000	16.665	21.430	23.435	26.390	26.250
5	15.000	20.000	15.000	12.000	25.000	21.425	25.000	18.890	25.000
6	12.500	8.335	16.665	25.000	30.560	23.810	27.085	24.075	29.165
7	21.430	21.430	21.430	21.425	23.810	22.450	25.000	23.810	28.570
8	9.375	18.750	23.435	25.000	27.085	25.000	31.250	29.165	30.000
9	16.670	22.220	26.390	18.890	24.075	23.810	29.165	27.160	28.335
10	22.500	20.000	26.250	25.000	29.165	28.570	30.000	28.335	29.000

at least. In particular we will study such balls in 2d with centers forming a regular grid with dimensions $n_1 \times n_2$.

It should be an ideal case for k-means application as k-means is generally claimed to detect ball-shaped clusters.

Furthermore, it is known that k-means has the tendency to split clusters with higher cardinality. Therefore we assume that the clusters are of the same size m.

We performed a simulation study of capability of k-means-random to detect clusters under such ideal conditions. We assumed that $k = n_1 \cdot n_2$, distance between ball centers $4.9 \cdot R$ and that R package version of k-means is used with k restarts. $m = 80$ is assumed. The table 1 shows what percentage of clusters is identified incorrectly. We see that k-means-random is not well suited for discovery of such clear-cut clusters.

As a remedy we propose initialization of k-means with "most distant points", k-means-mxdst, which was shown to be nearly as effective as k-means++ [5]. At each iteration step the point is selected as next seed that is most distant from the closest previous seed. In this case the number of errors was zero for each n_1, n_2 ranging from 0 to 10.

3. Discovering the number of clusters

A major handicap of k-means is that the number of clusters has to be known apriori. As k-means-mxdst works well for discovering the clusters when k is known apriori, one can try the following procedure: for k in some range perform k-means-mxdst clustering and check if the discovered clustering matches the criterion of enclose in balls with appropriate distance.

Figure 1 illustrates the run of this algorithm. The Figure presents the number of wrong clusters that occurred during the runs of three k-means based algorithms. The red line refers to the standard k-means-random implemented in R. The green line shows the results of the (unrealistic) algorithm k-means-truecenter

that is seeded with true cluster centers. The black squares are the results of k-means-mxdst algorithm. The true number of clusters in this experiment was 100. As we see, initially all the algorithms identify no correct cluster. But soon the unrealistic k-means-truecenter starts outperforming the other two which behave similarly. But in the final stage k-means-mxdst performs similarly to the unrealistic one while the standard k-means-random has worse performance and does not get close to the intrinsic clustering which is detected at the end by k-means-mxdst and the k-means-truecenter.

Finally, one could ask why not to use k-means-mxdst as the standard clustering algorithm. We performed an experiment of running clustering with $k = 25$ (on the grid $n_1 = 5, n_2 = 5$) but with addition of noise. Noise was generated as uniformly spread data points over the grid area. The percentage of noise means how much noisy data points were added compared to the original data in clusters. The Figure 2 presents results averaged over 100 runs. As visible, k-means-random improves its performance with increase of noise, while k-means-mxdst deteriorates, and with 30% of noise both are comparable. It is worth noting that the unrealistic k-means-truecenters outperforms definitely both of them. It may be a hit for further research that it is important to hit high density areas. This may be indicative that proposal of [6] is worth deeper investigation.

4. Conclusions

The claim that clustering should discover subsets of data that are homogeneous inside and there is a diversity between them, is repeated in many publications. It means in case of distance based algorithms that there should be gaps between areas occupied by the identified clusters. However, the actually most frequently used algorithms from the k-means family do not care about gaps between the clusters as is visible from the defining cost functions. The relationship between gaps and k-means clusters remains a mystery, though a number of studies has been already performed. For example, [7] proposes conditions under which k-means quality criterion coincides with split into clusters separated by gaps. There is a practical problem there, because gaps are really large.

Therefore, in this study, we tried to answer the question whether or not there exist possibilities to use versions of k-means such that they will detect clusters under significantly smaller gaps between clusters. It turns out that the distance maximizing initialization would be the proper choice and even the number of clusters could be discovered under such favourable conditions. However, cluster number discovery is impossible with this algorithm in case of even slight noise and with high noise the standard k-means with random initialization performs equally well with discovering clusters given the number of clusters is known apriori.

A hint requiring further investigation is the behaviour of unrealistic algorithm

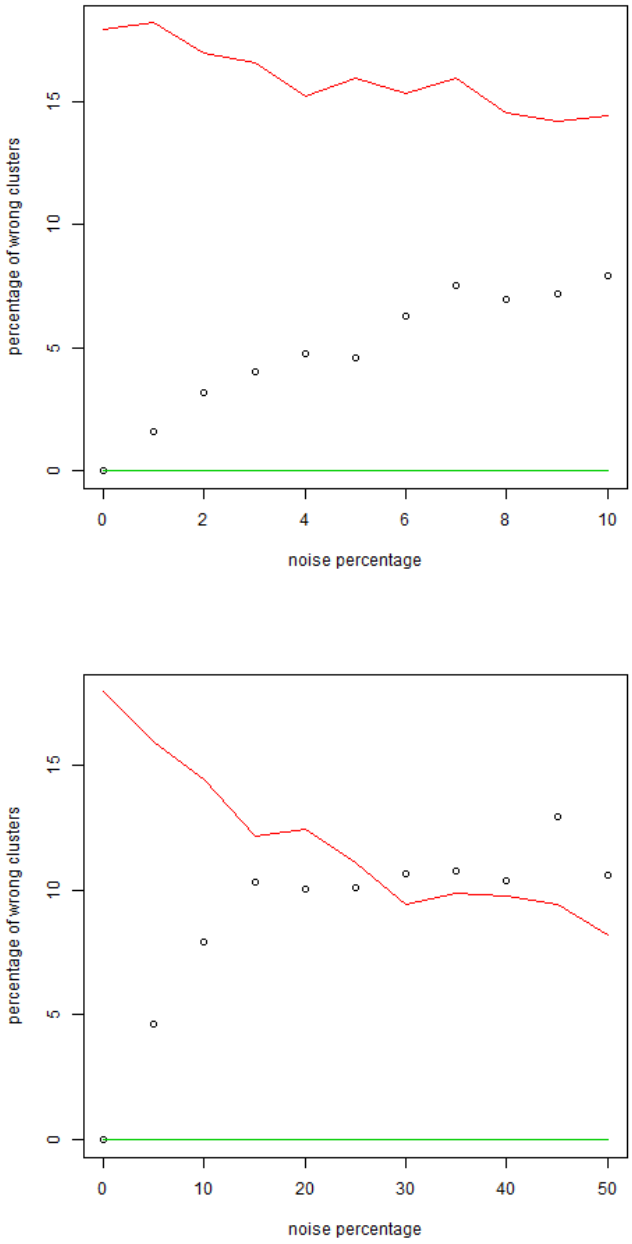


Figure 2: Clustering with $k = 25$ true clusters and noise. Top picture in the range of 0-10% of noise, bottom – 0-50% of noise. Source: own work.

that is initialized with true cluster centers. It suggests that combining k-means with some density based clustering methods may be a promising direction for future research.

References

[1] Nikhare N.B., Prasad P.S., *A review on inter-cluster and intra-cluster similarity using bisected fuzzy c-mean technique via outward statistical testing*, [In:] *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, pp. 215–217, doi: 10.1109/ICISC.2018.8399066.

[2] Grace G.H., Desikan K., *Experimental estimation of number of clusters based on cluster quality*, *Journal of mathematics and computer science*, 2014, vol. 12, pp. 304–315.

[3] Ackerman M., Dasgupta S., *Incremental clustering: The case for extra clusters*, *CoRR*, 2014, vol. abs/1406.6398, doi: 10.48550/arXiv.1406.6398.

[4] Kłopotek R.A., Kłopotek M.A., *Solving inconsistencies of the perfect clustering concept*, [In:] *Proc. of PP-RAI'2019 Congress*, Wrocław, 2019, pp. 273–276.

[5] Li Y., *Generalization of k-means related algorithms*, *CoRR*, 2019, vol. abs/1903.10025, doi: 10.48550/arXiv.1903.10025.

[6] Hamerly G., Elkan C., *Learning the k in k-means*, [In:] S. Thrun, L. Saul, B. Schölkopf (eds.), *Advances in Neural Information Processing Systems*, vol. 16, MIT Press, Cambridge, 2003.

[7] Kłopotek M., Kłopotek R., *Issues in clustering algorithm consistency in fixed dimensional spaces. some solutions for k-means*, *Journal of Intelligent Information Systems*, 2021, vol. 57, pp. 509–530.

Contextual ES-adRNN with Attention Mechanisms for Forecasting

Sławek Smyl[1][0000−0003−2548−6695],
Grzegorz Dudek[2][0000−0002−2285−0327],
Paweł Pełka[2][0000−0002−2609−811X]

[1]*slawek.smyl@gmail.com*
[2]*Czestochowa University of Technology*
Electrical Engineering Faculty
Al. AK 17, 42-200 Czestochowa, Poland
{grzegorz.dudek,pawel.pelka}@pcz.pl

DOI:10.34658/9788366741928.14

Abstract. *In this study, we propose a hybrid contextual forecasting model with attention mechanisms for generating context information. The model combines exponential smoothing and recurrent neural network to extract and synthesize information at both the individual series and collective dataset levels. The model is composed of two simultaneously trained tracks: context track and main track. The main track generates forecasts and predictive intervals, while the context track generates additional inputs for the main track based on representative time series. Attention mechanisms are integrated into the model in six different variations to adjust the context information to the forecasted series and so increase the predictive power of the model.*
Keywords: *hybrid forecasting models, recurrent neural networks, attention mechanism, contextual forecasting*

1. Introduction

Forecasting time series with multiple seasonality, nonlinear trends, and varying variances is a difficult task. Neural networks (NNs) are capable of modeling complex nonlinear relationships and reflecting process variability in uncertain dynamic environments. Recurrent neural networks (RNNs) are especially effective for time series forecasting [1], as they can capture both short- and long-term dynamics due to their internal memory and gating mechanism. Combining RNNs with other forecasting models, ensembling, and introducing context information can further improve the predictive power of the model.

Attention mechanism is a powerful technique used in NNs that enables the model to selectively focus on the most informative parts of the input data when making predictions. Essentially, it allows the network to learn which parts of the

input are most relevant and to weigh them more heavily when generating the output. This helps to improve the accuracy and interpretability of the model, especially in tasks where certain parts of the input are more important than others.

In this study, we propose a hybrid model consisting of two tracks that enables to incorporate context information. We use different types of attention mechanisms to adjust the context information for each individual time series forecasted by the main track.

2. Model

The forecasting problem is to predict h successive values of a time series based on its M historical values $\{z_\tau\}_{\tau=1}^{M}$. To provide a concrete example, we consider the task of short-term electrical load forecasting (STLF) which incorporates triple seasonality (see [2] for more information). Specifically, our objective is to predict the 24-hour daily load profile for the next day based on historical load data.

A forecasting model is a modified version of contextually enhanced ES-dRNN with dynamic attention proposed in [2]. It is a hybrid of exponential smoothing (ES) and RNN, with two tracks that are trained simultaneously: a context track and a main track. The main track generates point forecasts for a given horizon h, as well as predictive intervals. The context track generates additional inputs for the main track based on representative time series. Fig. 1 illustrates the block diagram of the model with a new component: an attention mechanism.

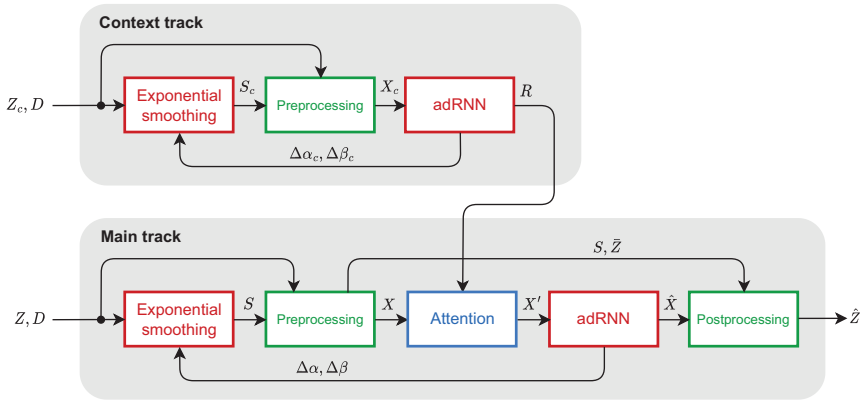


Figure 1. Block diagram of the model. Source: own work.

The model has the following characteristics: (i) A hybrid architecture that leverages ES for on-the-fly preprocessing and attentive dilated RNN (adRNN) for capturing temporal dependencies in the time series. (ii) A context track that provides global information to the main adRNN. (iii) New recurrent cells that incor-

porate dilation and attention mechanisms (in addition to the attention mechanisms proposed in this study), enabling adRNN to model long-term and seasonal dependencies while selecting relevant input information. adRNNs in both tracks are composed of three layers dilated 2, 4 and 7, respectively (see [2]). (iv) A dynamic ES model, which parameters are adjusted in each recurrent cycle by adRNN. (v) Cross-learning, which allows the model to capture shared features across individual series and prevent overfitting. (vi) A quantile loss function that enables the model to produce both point forecasts and predictive intervals, and helps to reduce forecast bias. (vii) Ensembling and regularization to prevent overfitting.

A context track generates additional inputs for the main track based on K representative time series. It produces K u-dimensional context vectors $\mathbf{r}_t$ in each recursive cycle. To adjust these vectors to individual series forecasted by the main track, in [2] we introduced per series parameters collected in learnable modulation vectors. In this study, we propose different approach based on attention mechanisms. The context vectors generated by the context track for each context series are adjusted dynamically to the series collected in the main batch using attention operations. They employ keys and values related to the context vectors, while queries related to the input vectors and hidden states of the main adRNN (in our implementation there are two types of hidden states: recent and delayed).

3. Attention Mechanisms

Attention mechanisms are usually deployed in time axis or sequence steps. This is not needed here, as we use as a main forecasting NN a dilated RNN, with several cells of different dilation. This mechanism is faster and produces equally or more accurate results. Instead, we attend output from the context RNN, across context time series.

To produce informative context vector for the main track, we examine six types of attention mechanisms. They combine input vectors, hidden states and context vectors, and produce modified context vectors.

Single-head transformer-style attention (Att1) [3]. We define a query, key and value as follows:

$$\mathbf{q}_t = \text{Linear}(\text{Concat}(\mathbf{x}_t^{in}, \mathbf{h}_{t-1}^{(3)}, \mathbf{h}_{t-d}^{(3)})),\ \mathbf{k}_t = \text{Linear}(\mathbf{r}_t),\ \mathbf{v}_t = \text{Linear}(\mathbf{r}_t) \quad (1)$$

where $\mathbf{x}_t^{in}$ is the input vector, $\mathbf{h}_{t-1}^{(3)}$ and $\mathbf{h}_{t-d}^{(3)}$ are the recent and delayed hidden states of the third (top) recurrent layer, respectively (the total size of the concatenated three vectors is n), and Linear(.) is a learnable linear transformation into u-dimensional space.

If the batch size of the main track is J, number of the context series is K, and

the size of the context vector is u, the attention is defined as follows:

$$\text{Attention}(\mathbf{Q}_t, \mathbf{K}_t, \mathbf{V}_t) = \text{softmax}\left(\frac{\mathbf{Q}_t \mathbf{K}_t^T}{\sqrt{u}}\right)\mathbf{V}_t \in \mathbb{R}^{J\times u} \tag{2}$$

where $\mathbf{Q}_t \in \mathbb{R}^{J\times u}$, and $\mathbf{K}_t, \mathbf{V}_t \in \mathbb{R}^{K\times u}$.

The attention result represents modified context vectors for each series in the main batch, which are packed together into one matrix $\mathbf{R}'_t = \text{Attention}(\mathbf{Q}_t, \mathbf{K}_t, \mathbf{V}_t)$.

Single-head transformer-style attention, simplified version 1 (Att2). This solution is similar to Att1, except that $\mathbf{r}_t$ is not transformed linearly to get the value vector, just $\mathbf{v}_t = \mathbf{r}_t$.

Single-head transformer-style attention, simplified version 2 (Att3). In this variant of Att1, a query is defined using (1) while a key and value are the context vectors without transformation, $\mathbf{k}_t = \mathbf{v}_t = \mathbf{r}_t$.

Multi-head transformer-style attention (Att4) [3]. It produces the queries, keys, and values using independently learned linear projections. These queries, keys, and values are then fed into attention pooling in parallel, producing multiple outputs. These outputs, or "heads", are then concatenated to produce the final output. The advantage of this approach is that each head can attend to different parts of the input sequence, allowing the model to capture different types of relationships and patterns in the data. This approach for our case is expressed as follows:

$$\mathbf{R}'_t = \text{MHAttention}(\mathbf{Q}_t, \mathbf{K}_t, \mathbf{V}_t) = \text{Concat}\left(\mathbf{R'}_t^{(1)}, \ldots, \mathbf{R'}_t^{(h)}\right) \in \mathbb{R}^{J\times mu} \tag{3}$$

where m is a number of heads, $\mathbf{R'}_t^{(i)} = \text{Attention}\left(\mathbf{Q}_t^{(i)}, \mathbf{K}_t^{(i)}, \mathbf{V}_t^{(i)}\right)$, and $\mathbf{Q}_t^{(i)}, \mathbf{K}_t^{(i)}, \mathbf{V}_t^{(i)}$ are determined using (1) with different linear projections for $i = 1, \ldots, m$.

Bahdanau additive attention (Att5) [4]. This mechanism in our version first produces a query and key according to (1), then it calculates an additive attention between each query vector and each key vector:

$$\text{a}(\mathbf{q}_t, \mathbf{k}_t) = \text{Linear}\left(\tanh\left(\mathbf{q}_t + \mathbf{k}_t\right)\right) \in \mathbb{R} \tag{4}$$

The modified context matrix is determined as follows:

$$\mathbf{R}'_t = \text{softmax}\left(\mathbf{A}_t\right)\mathbf{V}_t \in \mathbb{R}^{J\times u} \tag{5}$$

where $\mathbf{A}_t \in \mathbb{R}^{J\times K}$ is a matrix of additive attentions determined using (4).

Dynamically adjusted Gaussian kernel attention (Att6). A query is defined using (1), while $\mathbf{k}_t = \mathbf{v}_t = \mathbf{r}_t$. An attention between a query and key is defined using Gaussian kernel:

$$\text{a}(\mathbf{q}_t, \mathbf{k}_t) = \exp\left(-\sum_{i=1}^{u}\left(\frac{q_{t,i} - k_{t,i}}{\exp\left(\sigma_{t,i}\right)}\right)^2\right) \in \mathbb{R} \tag{6}$$

where $\sigma_{t,i}$ is a learnable bandwidth parameter controlling width of the kernel.

Note that $\sigma_{t,i}$ is adjusted dynamically for each dimension of the context. A modified context matrix is determined using (5), where $\mathbf{A}_t$ is composed of individual attentions (6).

4. Experimental Study

We evaluate the performance of our proposed model with different attention mechanisms using the ENTSO-E electricity demand dataset (www.entsoe.eu). The dataset contains hourly electricity loads from 2006 to 2018 for 35 European countries. We optimized the model on data from 2006 to 2017 and evaluated its performance on data from 2018.

We use similar training and optimization setup as in [2]. The values of the key hyperparameters selected based on experimentation were: size of the context vector – $u = 30$ for Att1 and Att5, $u = 20$ for Att2, Att3 and Att6, and $u = 10$ for Att4; number of heads in Att4 – $m = 3$; size of the c-state – 150; size of the h-state – 70; quantiles used by the loss function – $q^* = 0.525$, $\underline{q} = 0.045$, and $\overline{q} = 0.975$; number of ensemble members – 5.

Table 1 presents the forecast quality metrics. The results demonstrate that all attention methods produced very similar outcomes. Therefore, the selection of the most suitable method should be based on an evaluation of their computational complexity. The last column of Table 1 displays the number of learnable parameters for each mechanism. Notably, multi-head Att4 has the most parameters, whereas Att3 has the least.

Table 1. Quality metrics of the models.

Model	MAPE	MdAPE	RMSE	MPE	StdPE	#parameters
Att1	2.03	1.82	276.64	-0.26	3.39	$u(2u+n)$
Att2	2.04	1.83	277.70	-0.27	3.40	$u(u+n)$
Att3	2.02	1.82	275.54	-0.26	3.37	un
Att4	2.03	1.83	276.91	-0.25	3.39	$mu(2u+n)$
Att5	2.03	1.82	276.59	-0.26	3.39	$u(u+n+1)$
Att6	2.03	1.83	276.50	-0.29	3.39	$u(n+1)$

5. Conclusions

In this study, we incorporate the attention mechanism into cES-adRNN by six different schemes to adjust the context information to the time series properties. Experimental studies have shown very similar results for all tested attention

mechanisms. Therefore, the least computationally complex ones, i.e. simplified single-head transformer-style attention Att3 and dynamically adjusted Gaussian kernel attention Att6, are recommended.

References

[1] Hewamalage H., Bergmeir C., Bandara K., *Recurrent neural networks for time series forecasting: Current status and future directions*, *International Journal of Forecasting*, 2021, vol. 37, no 1, pp. 388–427.

[2] Smyl S., Dudek G., Pełka P., *Contextually enhanced ES-dRNN with dynamic attention for short-term load forecasting*, 2022, doi: 10.48550/ARXIV.2212.09030.

[3] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I., *Attention is all you need*, doi: 10.48550/arXiv.1706.03762.

[4] Bahdanau D., Cho K., Bengio Y., *Neural machine translation by jointly learning to align and translate*, 2014, doi: 10.48550/ARXIV.1409.0473.

Graph-Supported Preparation of GIS Machine Learning Datasets

Sebastian Ernst[1][0000−0001−8983−480X]

[1]*AGH University of Science and Technology*
Department of Applied Computer Science
Al. Mickiewicza 30, 30-059 Kraków, Poland
ernst@agh.edu.pl

DOI:10.34658/9788366741928.15

Abstract. *The paper presents an approach to preparing spatial (GIS) datasets for machine learning models, and using graph structure to materialise and utilise the results. The presented work is based on the Spatially-Triggered Graph Transformations (STGT) methodology, previously used for many real-world applications, e.g. in the area of* smart cities*. A workflow using OSM data is presented, aimed at improving the granularity and semantic annotation of map features.*
Keywords: *graph transformations, ML, classification, GIS, smart cities*

1. Introduction

Spatial (GIS) data plays an increasingly important role in many areas of research and industry. It is the foundation of some fields, such as transportation and urban planning, and because many non-GIS datasets include location information, it is valuable for analysis and understanding e.g. of social media activity [1]. In the field of *smart cities*, the results of such research can have very tangible impact: with rapidly-growing costs of energy, any reduction of its consumption translates to significant savings in the cities' operating expenses, provided that it is executed in an optimal and efficient way [2]. Spatial datasets have been an important part of several R&D projects led by our group, including a large-scale pilot of intelligent lighting in the city of Kraków (covering almost 4,000 lamps), a smart city project in Siechnice (Lower Silesian Voivodeship), as well as bulk optimisation of street lighting (Washington DC, Tbilisi).

2. Background and Motivation

Experiences in the aforementioned projects have shown that there are several problems related to their analysis:

1. The datasets often have no attribute links between them, i.e. the only existing relationships (albeit significant ones) result from their geographic locations and positions.

2. The granularity of objects often does not reflect their "natural", intuitive granularity in the real world.

3. Shape characteristics, obvious to the human eye, are barely perceptible by computational algorithms.

The first issue, related to the integration of separate GIS datasets, can be difficult, error-prone, and challenging in terms of complexity [3]. Graph transformations, an efficient method of data analysis and synchronisation, can be used to solve these issues. However, they rely on data which has already been modelled as a graph, and in GIS datasets, the relationships (vicinity, intersection) are implicit. Therefore, a new concept, *Spatially-Triggered Graph Transformations* (STGT), has been developed [4]. In essence, STGT takes a subset of the graph elements, analyses it using external tool spatial analysis tools, and *materialises* the results (detected spatial relationships along with their attributes, e.g. distance between objects) by introducing new elements into the graph. This allows for further analysis using rules defined as graph productions.

The result from the way spatial data is typically modelled. In GIS-related applications, it is common practice to use general-purpose map datasets, such as OpenStreetMap[1] (OSM), as a "common denominator" for other, application-specific datasets. While there have been numerous attempts to assess the quality of OSM data [5] or make it suitable for other applications, such as smoothening the geometries it for autonomous vehicles [6].

However, while these methods cover some aspects of GIS data quality improvement, they do not address latter two of the problems. Granularity and "semantic" description of shapes are crucial for some applications, such as design of street lighting installations and/or strategic planning of modernisation projects, which requires quick and accurate estimation of possible energy gains for each street fragment.

3. Proposed Approach

The presented methodology is aimed at *preparing* spatial datasets for machine learning processes, as well as gathering their results in the form of a graph and, finally, obtaining (and applying) the optimal set of map data modification or reclassification actions. The objective of the presented approach is therefore to modify the granularity of the dataset so that it matches the analytic requirements, and

[1] https://www.openstreetmap.org/

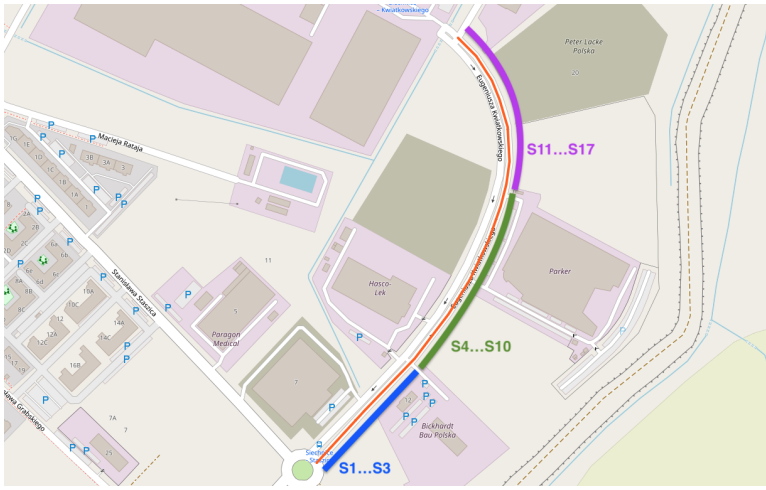


Figure 1: An example OpenStreetMap *way* modelling a street. Source: own work.

to detect elements of the road network (e.g. roundabouts), which have not been explicitly distinguished in the source data.

In OSM, road and street data are modelled as *way* objects, with each *way* being an ordered sequence of *nodes*. However, the granularity of OSM *ways* depends largely on the person modelling them: some *ways* cover large stretches with multiple intersections along the way, while in other locations, a short stretch of road between two intersections can be composed of multiple *ways*.

3.1. Data Preparation

Let us assume that OpenStreetMap data has already been imported into a graph structure. Due to space limitations, we will only focus on the parts used for further analysis, namely nodes labelled as S, modelling way *segments* – part of a *way* consisting of two subsequent *nodes*.

The first step involves extracting all connected subgraphs (up to a given node count) from the graph, thus generating a set of *samples* – map extracts of different sizes. These extracts can then be submitted to an ML classifier, trained to detect objects such as straights, curves (bends), intersections or roundabouts. Let us consider an example OpenStreetMap way shown in fig. 1, which consists of 18 nodes, i.e. 17 segments modelling its curvature. The procedure will therefore generate a series of *linestrings* (polygonal chains), which are subsequences of the indicated shape.

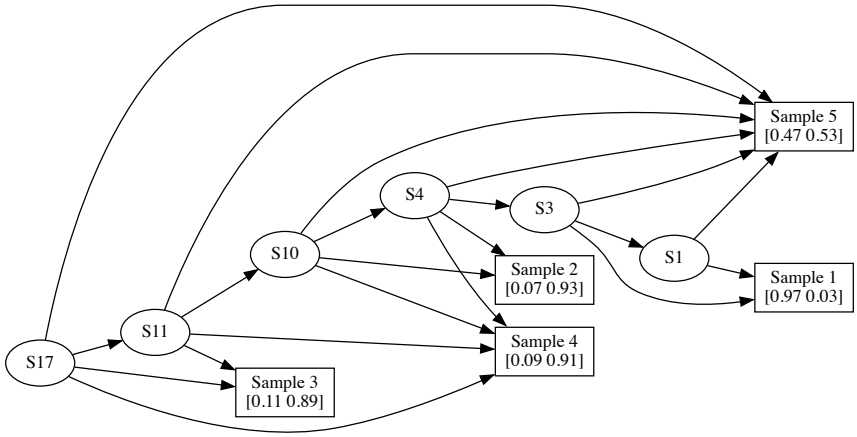


Figure 2: Example results of classification. Source: own work.

3.2. Classification procedure

Having prepared the dataset, it can be submitted to a pre-trained[2] multi-class[3] classifier model, thus obtaining the probabilities that the objects belongs to one of the classes. The results therefore form vectors of length n, where n is the number of classes.

Please note that this paper is focused on data preparation and utilisation of results, and therefore the classifier model is, to some extent, treated as a "black box". As for the model itself, an intuitive choice would be to use a shape-aware algorithm, such as a simple convolutional neural network (CNN). Other parameters can also be computed from the shape and extracted from OSM attributes (e.g. road type, lane count), which makes an ensemble model seem like a viable solution.

3.3. Results Collection and Processing

As a result of the classification procedure, for each sample, we will obtain the probabilities of that map fragment representing an object of the defined classes. These results can be stored back in the graph as nodes labelled *Sample*, with class probabilities stored as a node attribute.

Example classification results for the street shown in fig. 1 are presented in fig. 2. Please note that as the street includes 18 nodes (and therefore, 17 segments),

[2]Due to volume limitations, the training process has not been described in detail. Using supervised learning is straightforward in this case, but obviously requires a labelled training model first. OpenStreetMap data is a good candidate for this task, as some object instances in given classes (e.g. roundabouts) are labelled with appropriate tags, while others are not.

[3]In the presented example, for simplicity, only two classes – curve and straight road – are distinguished. Naturally, in real-life applications, the number of classes would be higher, to include objects such as roundabouts, turning circles, as well as different types of curves (e.g. hairpins).

for clarity, not all possible samples (subgraphs) have been shown here. For the same reasons, the example assumes two classes (*straight*, *curve*), and the presented vectors contain the respective probabilities for each sample.

The final stage of processing is the selection of a subset of samples which satisfied the following conditions: (a) the selection maximises the classification quality, which can be expressed e.g. as arg $\max_V$, where V is the probability vector, thus preferring unambiguously classified samples; (b) the selection minimises the total number of included samples; (c) there are no intersections between samples, and (d) every street segment S is covered by a sample (and, obviously, appears only once).

4. Conclusions and Future Work

The presented, preliminary work, presents an ML-based variant of the STGT methodology, which was originally based on GIS analysis tools and used to support integration of GIS datasets. Inclusion of ML allows for development of more elaborate analysis procedures, which would be difficult and costly to implement as deterministic, expert-driven algorithms. The result of the procedure presented in sec. 3 is a "best-guess" assignment of map features to the defined classes, along with re-segmentation of the dataset, which results in its granulation being adjusted to more accurately represent intuitive real-world objects.

A practical application of these results, one in line with the problem of street lighting design referenced throughout this paper, is using the newly-defined objects as input for the process. This, in turn may simplify it, by allowing reuse of lighting designs existing for similar parts of the street network (e.g. curves). It will also be used to support a rapid energy requirement estimation tool being developed as part of an ongoing research and development project.

Finally, we should also mention the possibility of automatically introducing added value to the map dataset itself. Results of the presented analyses can be used to add more semantic information to OSM objects, which may, in turn, broaden the spectrum of its applications, e.g. to finding roads with certain geometric characteristics.

References

[1] Sufi F.K., Khalil I., *Automated disaster monitoring from social media posts using AI-based location intelligence and sentiment analysis*, *IEEE Transactions on Computational Social Systems*, 2022, pp. 1–11, doi: 10.1109/tcss.2022.3157142.

[2] Campisi D., Gitto S., Morea D., *Economic feasibility of energy efficiency improvements in street lighting systems in rome*, *Journal of Cleaner Production*, 2018, vol. 175, pp. 190–198, doi: 10.1016/j.jclepro.2017.12.063.

[3] Beeri C., Doytsher Y., Kanza Y., Safra E., Sagiv Y., *Finding corresponding objects when integrating several geo-spatial datasets*, [In:] *Proceedings of the 13th annual ACM international workshop on Geographic information systems*, ACM, doi: 10.1145/1097064.1097078.

[4] Ernst S., Kotulski L., *Estimation of road lighting power efficiency using graph-controlled spatial data interpretation*, [In:] *Computational Science – ICCS 2021*, Springer International Publishing, 2021, pp. 585–598, doi: 10.1007/978-3-030-77961-0_47.

[5] Girres J.F., Touya G., *Quality assessment of the french OpenStreetMap dataset*, *Transactions in GIS*, 2010, vol. 14, no 4, pp. 435–459, doi: 10.1111/j.1467-9671.2010.01203.x.

[6] Artunedo A., Godoy J., Villagra J., *Smooth path planning for urban autonomous driving using OpenStreetMaps*, [In:] *2017 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, doi: 10.1109/ivs.2017.7995820.

Hashtag Similarity Based on Laplacian Eigenvalue Spectrum

Bartłomiej Starosta [0000−0002−5554−4596],
Mieczysław A. Kłopotek[0000−0003−4685−7045],
Sławomir T. Wierzchoń[0000−0001−8860−392X]

Institute of Computer Science, Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
barstar,klopotek,stw@ipipan.waw.pl

DOI:10.34658/9788366741928.16

Abstract. *Hashtags play nowadays an important role in the current social media world. They are usually deemed to represent topics of e.g. tweets. As the number of hashtags is growing, an overview of the information flow requires some method of grouping these hashtags. The grouping requires a similarity measure. In this paper we propose a novel measure of similarity between hashtags based on the Graph Spectral Analysis.*
Keywords: *Graph Spectral Analysis, combinatorial Graph Laplacian, eigenvalue spectrogram based similarity, artificial intelligence*

1. Introduction

The so called Graph Spectral Analysis (GSA) represents a novel way of looking into relationships between data objects that are characterized by mutual similarity measures, and hence can be best described by a graph with weights equal to these similarities. The similarity matrix is transformed to e.g. combinatorial Laplacian, which in turn is subject to eigen-decomposition. Eigenvectors constitute a new coordinate system into which the data objects are embedded and thus may be subject of distance-based data clustering or data classification methods [1, 2], also with hashtags [3]. The main stream of research concentrates on usage of a carefully selected subset of eigenvalues and corresponding eigenvectors.

Our experiments (in Sect.3) have shown, however, that there exists a possibility to use the entire eigenvalue spectrogram as a way to characterize classes within the aforementioned weighted graph of objects and consequently, the similarity of the spectrograms is usually related to the similarity of the classes themselves. The paper explains our methodology in brief in Sect.2. It is based on the observation that spectra of combinatorial Laplacian of random subsamples of the same class can be down-scaled to overlap, while those from different classes do not.

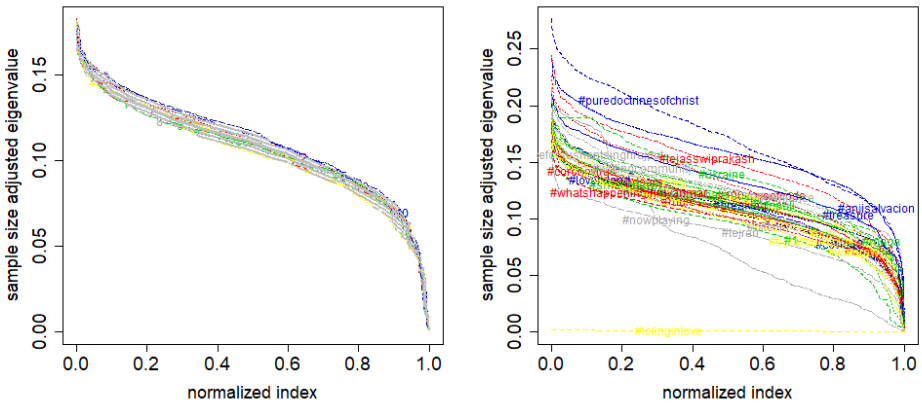


Figure 1. Normalized spectrograms for samples of (left:) one single hashtag, (right:) various hashtags. Source: own work.

2. The Method

Let S be a similarity matrix between pairs of items (e.g. tweets). It induces a graph whose nodes correspond to the items. A(n unnormalised) or combinatorial Laplacian L corresponding to this matrix is defined as

$$L = D - S \ , \tag{1}$$

where D is the diagonal matrix with $d_{jj} = \sum_{k=1}^{n} s_{jk}$ for each $j \in [n]$.

Let its eigenvalues be in non-decreasing order $0 = \lambda_1 \leq \ldots \leq \lambda_n$.

We proposed a function $\lambda_{CLSSAL} : [0, 1] \to \mathbb{R}$ in such a way that

$$\lambda_{CLSSAL}\Big(\frac{n-i}{n-1}\Big) = \frac{\lambda_i}{n} \ . \tag{2}$$

The linear interpolation is applied in-between.

Based on the above assumption, we can compute a "distance" between a given new sample and the elements of a class as the area between the λ_{CLSSAL} curves. So if the first subgraph $G1$ is characterized by $\lambda_{CLSSAL,G1}$ curve, and the second subgraph $G2$ is characterized by $\lambda_{CLSSAL,G2}$ curve, then the dissimilarity is computed as

$$dissim(\lambda_{CLSSAL,G2}, \lambda_{CLSSAL,G1}) = \int_0^1 |\lambda_{CLSSAL,G2}(x) - \lambda_{CLSSAL,G1}(x)|dx \tag{3}$$

Table 1. Closeness of hashtags based on eigenvalue spectrum

hashtag	s.hashtag	min.dist	avg.dist	std.dist	subsamp.dist	subsamp.err	rel.subsamp.dist
#1	#tejran	0.008004704	0.02757048	0.02010759	0.003701717	0.002201124	0.1342638
#100daysofcode	#treasure	0.006955492	0.02583407	0.02107681	0.005498188	0.002859725	0.212827
#90dayfiance	#maga	0.005161067	0.02306229	0.02111358	0.003688091	0.001489804	0.1599187
#aewdynamite	#demdebate	0.001579806	0.01779384	0.02137026	0.003271367	0.001303339	0.1838483
#anjisalvacion	#tejasswiprakash	0.01082191	0.04816793	0.02440739	0.003400781	0.00184204	0.07060261
#auspol	#coronavirus	0.001122648	0.01758666	0.02167454	0.002911198	0.002186825	0.1655344
#bbnaija	#whatsh...mar	0.004470872	0.02170779	0.02088629	0.001556021	0.0008715957	0.07168028
#bitcoin	#whatsh...mar	0.001994543	0.02048817	0.02139435	0.002381747	0.00131487	0.1162499
#blacklivesmatter	#demdebate	0.001858698	0.01791834	0.02139373	0.002099182	0.001207379	0.1171527
#breaking	#just...ajput	0.001608261	0.01806659	0.02094577	0.005108423	0.003956473	0.2827553
#cdnpoli	#covid_19	0.002422555	0.01907016	0.02120027	0.003914232	0.002502598	0.2052542
#coronavirus	#auspol	0.001122648	0.0176662	0.02168422	0.001548446	0.0005699945	0.08765021
#covid	#coronavirus	0.001857122	0.01755118	0.0215299	0.00517836	0.002254856	0.2950434
#covid_19	#cdnpoli	0.002422555	0.01823082	0.02124761	0.00580463	0.004045009	0.3183966
#covid19	#wweraw	0.001685603	0.01865511	0.02165203	0.002037235	0.0009840167	0.1092052
#demdebate	#aewdynamite	0.001579806	0.01805983	0.02133996	0.002643427	0.001205658	0.1463705
#endsars	#blacklivesmatter	0.002633596	0.01757168	0.02135928	0.003110856	0.001541862	0.1770381
#justice...put	#breaking	0.001608261	0.01802846	0.0209234	0.004319812	0.00216075	0.2396107
#lolinginlove	#nowplaying	0.08095599	0.1135585	0.01869101	0.0003761939	1.161857e-05	0.003312778
#loveisland	#demdebate	0.003870494	0.01846769	0.02116395	0.001906628	0.0009974423	0.1032413
#maga	#cdnpoli	0.004050012	0.0207609	0.02121451	0.006509118	0.003850841	0.3135278
#mufc	#coronavirus	0.001592204	0.01805593	0.02168119	0.002320167	0.0006874776	0.1284989
#nowplaying	#1	0.03017543	0.04539396	0.01372244	0.01041429	0.004262152	0.2294201
#nufc	#mufc	0.001667688	0.01831197	0.02156489	0.003933591	0.0008943253	0.2148098
#puredoct...christ	#anjisalvacion	0.01333359	0.06026725	0.02505434	0.004094367	0.0016304	0.06793685
#smackdown	#auspol	0.002635276	0.01869505	0.02131051	0.004017606	0.001469838	0.2149021
#tejasswiprakash	#ukraine	0.009322752	0.03850018	0.02322012	0.004882715	0.002870357	0.1268232
#tejran	#1	0.008004704	0.03201366	0.01932727	0.003926262	0.0015748	0.1226433
#tigraygenocide	#bitcoin	0.003517509	0.02141553	0.02096667	0.004367564	0.002388737	0.2039438
#treasure	#ukraine	0.00570973	0.02789071	0.02178642	0.003076978	0.0009072463	0.1103227
#ukraine	#treasure	0.00570973	0.03094697	0.02215523	0.005446961	0.001946473	0.1760095
#whatsh...myanmar	#bitcoin	0.001994543	0.02059761	0.02131954	0.003152089	0.00101502	0.1530318
#writingcommunity	#maga	0.004232268	0.02220214	0.02099251	0.00568054	0.003211025	0.2558555
#wweraw	#covid19	0.001685603	0.01941004	0.02155251	0.005705186	0.003549209	0.2939297

3. Experiments

We investigated this phenomenon for a small collection of hashtags extracted from Twitter tweets. Their names are listed in the first column of the Table 1. We constructed a graph of tweets having only one hashtag from this list, where the weights of the tweets are computed as cosine measure in the bag-of-words vector space.

We investigated two types of subgraphs of this graph: subgraphs that include all objects of the same hashtag and subgraphs of such graphs.

For each of the subgraph we computed the combinatorial Laplacian according to equation (1). Then the function $\lambda_{CLSSAL}()$ was created for each subgraph based on the equation (2). Finally, the dissimilarity between the spectrograms was computed according to equation (3).

Fig.1, left, represents overlapped diagrams of functions $\lambda_{CLSSAL}()$ of ten samples of tweets belonging to the same hashtag. It turns out that the spectrograms of the subsets of the same hashtags are quite close to one another.

Fig.1, right, represents overlapped diagrams of functions $\lambda_{CLSSAL}()$ of 34 samples of tweets belonging to the various hashtags listed in the first column of the table 1. It turns out that the spectrograms of the subsets related to different hashtags may differ even substantially.

Table 1 shows more details of dissimilarities between the chosen tags. The column `avg.dist` presents the average dissimilarity of the given hashtag from the remaining ones, while `std.dist` shows the standard deviation of dissimilarity. The column `s.hashtag` represents the closest hashtag, with `min.dist` being the dissimilarity to it. As a contrast, `subsamp.dist` represents the average dissimilarity to 100 samples from the same hashtag, `subsamp.err` being the standard deviation of this measure. `rel.subsamp.dist` is the quotient of `subsamp.dist` / `avg.dist`.

`rel.subsamp.dist` demonstrates that in fact the samples from the same hashtag are closer to one another than to other hashtags.

The hashtag *#lolinginlove* seems to be most distant from all the other hashtags on average, while *#blacklivesmatter* seems to be close to many other hashtags from the list. The hashtag *#puredoctrinesofchrist* seems also to be distant from the other, though it is quite near to *#anjisalvacion*. *#covid* has a characteristic quite similar to *#coronavirus*.

Based on the dissimilarity matrix, most dissimilar hashtags were identified as follows: The first one was that with the highest sum of dissimilarities to other hashtags. The other were added with the highest sum of dissimilarities to those already chosen. The following list of hashtags was obtained in this way: *#lolinginlove*, *#puredoctrinesofchrist*, *#anjisalvacion*, *#nowplaying*, *#tejran*, *#tejasswiprakash*, *#1*, *#ukraine*, *#bbnaija*, *#90dayfiance*, *#tigraygenocide*, *#treasure*, *#whatshappeninginmyanmar*, *#100daysofcode*, *#bitcoin*, *#writingcommunity*, *#smackdown*,

Table 2. Classification errors and F1 measure for most distant hashtags.

no. of hashtags	2	3	4	5	6	7	8	9	10	11
error %	0.00	1.33	1.75	0.60	4.83	7.00	9.75	9.22	8.20	9.82
F1*100	100.00	98.67	98.25	99.40	95.18	93.03	89.99	90.53	91.71	90.03
no. of hashtags	12	13	14	15	16	17	18	19	20	21
error %	14.17	16.77	19.07	18.47	20.75	23.53	26.22	30.47	26.90	29.38
F1*100	85.74	83.18	80.86	81.00	78.90	76.04	73.61	69.37	72.89	70.60

#maga, *#wweraw*, *#loveisland*, *#cdnpoli*, For each hashtag 100 samples from 30% of its tweets were drawn and classification via the smallest dissimilarity to the hashtag spectra was performed. The computations were performed with increasing number of hashtags from this list. The results are shown in Table 2. For first two hashtags were taken (*#lolinginlove*, *#puredoctrinesofchrist*), no classification error was made. When the third was included (*#anjisalvacion*), 1.3% error was observed. With 11 hashtags, 9.8% classification error was observed. The F1 measure is also reported in this table.

4. Conclusions

We have elaborated a new characterization of topical groups of objects, like tweets, via a characteristic spectrum of combinatorial Laplacian. It appears to be quite a stable descriptor of samples from the same population, while discriminating different populations. Potential applications seem be as ingredients in classification and clustering tasks as well as data visualization and hashtag recommendation [4, 5].

It requires further research as to what causes this spectral behavior for similar and different hashtags.

References

[1] Sevi H., Jonckheere M., Kalogeratos A., *Generalized spectral clustering for directed and undirected graphs*, *CoRR*, 2022, doi: 10.48550/arxiv.2203.03221.

[2] Wierzchoń S., Kłopotek M., *Modern Clustering Algorithms*, *Studies in Big Data*, vol. 34, Springer Verlag, 2018.

[3] Schmidt A., et al., *Using spectral clustering of hashtag adoptions to find interest-based communities*, [In:] *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, doi: 10.1109/ICC.2018.8422244.

[4] Jeon M., Jun S., Hwang E., *Hashtag recommendation based on user tweet and hashtag classification on twitter*, [In:] *Web-Age Information Management*, Springer International Publishing, Cham, pp. 325–336, doi: 10.1007/978-3-319-11538-2_30.

[5] Gupta V., Hewett R., *Unleashing the power of hashtags in tweet analytics with distributed framework on Apache Storm*, *CoRR*, 2018, doi: 10.1109/bigdata.2018.8622302.

Improvement of Attention Mechanism Explainability in Prediction of Chemical Molecules' Properties

Bartosz Durys, Arkadiusz Tomczyk[0000−0001−9840−6209]

Lodz University of Technology
Institute of Information Technology
al. Politechniki 8, 93-590 Łódź, Poland
bartdurys@gmail.com, arkadiusz.tomczyk@p.lodz.pl

DOI:10.34658/9788366741928.17

Abstract. *In this paper, the analysis of selected graph neural network operators is presented. The classic Graph Convolutional Network (GCN) was compared with methods containing trainable attention coefficients: Graph Attention Network (GAT) and Graph Transformer (GT). Moreover, which is an original contribution of this work, training of GT was modified with an additional loss function component enabling easier explainability of the produced model. The experiments were conducted using datasets with chemical molecules where both classification and regression tasks are considered. The results show that additional constraint not only does not make the results worse but, in some cases, it improves predictions.*
Keywords: *attention mechanism, graph transformer, graph neural network, explainability, chemical molecules*

1. Introduction

Graphs for a long time were an uncommon data type in machine learning solutions. Lately, many new methods for graph prediction tasks have been proposed, including those utilizing attention-based graph neural network operators. Interpretability of the attention mechanism in natural language processing problems is a well-researched subject. In this work, we investigate it in the context of the graph data structures. We evaluate the performance of the chosen operators on selected benchmark graph datasets containing chemical molecules. An original contribution of this work is a mechanism that forces attention coefficients to be more precise in indicating, which neighbouring nodes, and consequently which relations, are particularly important for prediction. The obtained outcomes reveal that, although the used mechanism imposes additional constraints on the trained neural network, it surprisingly does not aggravate the prediction results increasing, at the same time, model explainability.

2. Method

Three Graph Neural Network (GNN) operators were selected: a classic Graph Convolutional Network [1] (GCN), a more sophisticated Graph Attention Network [2] (GAT) and a Transformer's adaptation called Graph Transformer [3] (GT). The last two of them use the attention mechanism. The working principle (transformation of node embeddings $\mathbf{h}$ in layer t) for above operators can be summarized as:

$$\mathbf{h}_i^{t+1} = \sigma\left(\alpha_{ii}(\mathbf{W}_1\mathbf{h}_i^t + \mathbf{b}_1) + \sum_{j\in\mathcal{N}(i)} \alpha_{ij}(\mathbf{W}_2\mathbf{h}_j^t + \mathbf{b}_2)\right) \tag{1}$$

which, although is not the most general formulation for all GNN operators, is sufficient for further considerations. In this formula σ represents non-linear activation function, matrices $\mathbf{W}_1$, $\mathbf{W}_2$ as well as vectors $\mathbf{b}_1$, $\mathbf{b}_2$ are (if present) directly trainable parameters and $\mathcal{N}(i)$ denotes set of nodes connected with given node i. Coefficients α are fixed in GCN and depend on graph structure only. In GAT and GT these are indirectly trainable attention coefficients that take into account embeddings of connected nodes. In both cases for a given node i those coefficients are normalized with softmax function, which means that $\alpha_{ij} \in [0, 1]$ for $j \in \mathcal{N}(i)$ and their sum is equal to 1.

Training GT model it can be frequently observed that for a given node i coefficients α_{ij} tend to have similar values. It means that all the neighbouring nodes have similar influence on the calculated embedding of the node i and, consequently, it does not allow to draw any conclusions explaining the final predictions. To make those attention coefficients more interpretable we have introduced a new loss function component:

$$\mathcal{L}^{explain} = \sum_i \left(1 - \max_{j\in\mathcal{N}(i)} \alpha_{ij}\right) \tag{2}$$

It forces the model to direct its attention to only one neighbour while aggregating embeddings from each and every node. This component utilizes the softmax normalization of α_{ij} for a given i since optimally there should be only one 1 value among α_{ij} and the rest of them should be equal to 0. The final loss function used during training was the following:

$$\mathcal{L} = \mathcal{L}^{prediction} + \lambda \cdot \mathcal{L}^{explain} \tag{3}$$

where the first component was dependent on the considered task and it was MSE for regression and cross-entropy for classification. The parameter λ controls trade-off between those components, and it has been experimentally set to 0.1.

For experiments we have chosen two widely-used sources of chemical graph data – MoleculeNet [4] for graph-level regression tasks and TUDataset [5] for graph-level classification tasks. From each data source, we have selected three

Table 1: Quality metrics for graph-level regression tasks from MoleculeNet.

Dataset	Operator	Validation set			Test set		
		MSE	Standard deviation	Best	MSE	Standard deviation	Best
ESOL	GCN	35.24	9.11	8.15	35.24	8.80	8.11
	GAT	**11.76**	**5.09**	4.62	**11.01**	**5.04**	**4.46**
	GT	32.07	18.40	5.61	31.58	17.80	5.41
	GT with $\mathcal{L}^{explain}$	17.87	11.29	**4.53**	17.25	11.11	**4.46**
FreeSolv	GCN	78.64	38.21	17.42	75.11	34.37	17.01
	GAT	36.02	16.94	14.42	**33.95**	**15.02**	14.17
	GT	44.30	26.35	**13.91**	41.87	25.38	14.01
	GT with $\mathcal{L}^{explain}$	**34.71**	**11.35**	14.14	34.11	17.25	**13.86**
Lipophilicity	GCN	12.65	6.75	2.34	12.69	6.84	2.35
	GAT	10.67	4.54	2.12	10.64	4.58	2.16
	GT	12.51	9.19	2.44	12.44	9.15	2.46
	GT with $\mathcal{L}^{explain}$	**2.62**	**0.80**	**1.62**	**2.56**	**0.77**	**1.64**

datasets: ESOL (prediction of water solubility), FreeSolv (estimation of hydration free energy) and Lipophilicity (finding octanol/water distribution coefficient) from MoleculeNet and AIDS (identification of molecule's activity against HIV), ENZYMES (assign a molecule to one of the six Enzyme Commission top-level classes) and PROTEINS (prediction if a protein is an enzyme) from TUDataset. In MoleculeNet feature vectors for each dataset contained nine numerical features describing atoms, e.g. its hybridization, while in TUDataset it was a one-hot encoded representation of node class (chemical element in AIDS or secondary structure element in ENZYMES and PROTEINS).

To train and evaluate the performance of GNN operators on datasets, we have split each dataset into training, validation, and test sets using an 80/10/10 proportion. To ensure a fair comparison and easier interpretability, we have limited our research to only single-headed attention mechanisms. For each operator, we have selected two GNN layers with batch normalization, dropout and ReLU as activation function σ. After that, the global average pooling was used to aggregate the calculated hidden node embeddings. Finally, we have used an MLP to generate our final predictions. We have repeated every experiment 50 times with 500 epochs per repeat, and averaged the results using the best epoch on the validation set. This approach allowed us to obtain reliable and robust results for each dataset and operator combination.

3. Results

Starting the analysis of the results from MoleculeNet's datasets in the Table 1, we can observe several interesting phenomena. First, which is expected, we can see that operators with the attention mechanism perform better than simple GCN. However, the GT operator suffers from a high standard deviation value, which in-

Table 2: Quality metrics for graph-level classification tasks from TUDataset.

Dataset	Operator	Validation set			Test set		
		Accuracy	Standard deviation	Best	Accuracy	Standard deviation	Best
AIDS	GCN	80.22	2.81	**88.50**	79.84	**2.27**	84.00
	GAT	79.84	**2.74**	85.50	79.91	2.55	86.00
	GT	**81.05**	2.80	86.00	**80.49**	2.46	**87.00**
	GT with $\mathcal{L}^{explain}$	79.75	2.76	86.00	80.03	2.42	86.00
ENZYMES	GCN	30.90	4.14	43.33	20.87	5.69	35.00
	GAT	32.03	4.79	**46.67**	21.43	**5.56**	35.00
	GT	35.17	**3.45**	43.33	24.03	5.80	**40.00**
	GT with $\mathcal{L}^{explain}$	**36.77**	3.81	45.00	**24.33**	5.74	35.00
PROTEINS	GCN	73.69	3.99	81.08	68.88	5.27	78.57
	GAT	73.39	**3.43**	81.08	68.55	4.63	76.79
	GT	**74.29**	3.97	**81.98**	**69.20**	5.04	**79.46**
	GT with $\mathcal{L}^{explain}$	74.22	3.64	**81.98**	68.66	**4.50**	78.57

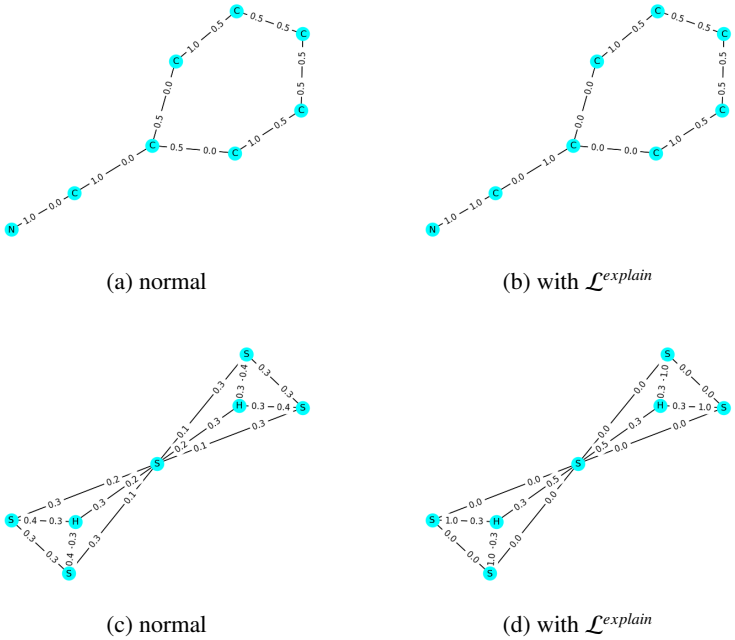
dicates that it is difficult to train, much like the original Transformer. Surprisingly, adding the $\mathcal{L}^{explain}$ function significantly improves the training of the model.

When it comes to the evaluation of TUDataset, our results are close in value to each other. As shown in the Table 2, the GT operator performs slightly better overall. The biggest difference can be seen in the ENZYMES dataset, which has six classes, whereas the other problems are binary classifications. These results suggest that the GT operator may be a better choice for classifications with a large number of classes.

To show the impact of $\mathcal{L}^{explain}$ component, we have prepared visualisations of attention coefficients, which are shown in Figure 1. Values near each node represent weights during the aggregation of its neighbours. In the first figure, we can observe that the proposed modification made the nitrogen atom more important. In the second figure, the model focused more on helices rather than sheets elements. This behaviour exhibited by the model could be a valuable source of information for explainability. In organic compounds, there are many chemical substituents that can have an impact on the molecules' properties. By utilizing a modified loss function, we may be able to better represent and understand their effects, ultimately leading to improved results, as demonstrated in this paper.

4. Conclusions

In this work, a modification of the training loss function for attention-based models was proposed. Its goal is to improve the interpretability of attention coefficients. Outcomes reveal that indeed it works correctly and, what is more, it does not worsen prediction results. An explanation of this phenomenon can be the fact that network architectures are frequently overdesigned and the proposed constraint allows to select the model with desired properties out of many equivalent (similarly predicting) solutions. The quality of the discussed method was assessed



(a) normal (b) with $\mathcal{L}^{explain}$

(c) normal (d) with $\mathcal{L}^{explain}$

Figure 1: Attention visualisation: (a), (b) – benzonitrile from the FreeSolv set with GT normalization coefficients from the first layer. C – carbon, N – nitrogen, (c), (d) – enzyme from the ENZYMES set with GT normalization coefficients from the first layer. H – helix, S – sheet. Source: own work.

using datasets with chemical molecules. It should be, however, emphasized that it can be of use in any task where an attention mechanism is used, leading to better explainability of model behaviour.

References

[1] Kipf T.N., Welling M., *Semi-supervised classification with graph convolutional networks*, *CoRR*, 2017, doi: 10.48550/arXiv.1609.02907.

[2] Veličković P., Cucurull G., Casanova A., Romero A., Liò P., Bengio Y., *Graph attention networks*, 2017, doi: 10.48550/arxiv.1710.10903.

[3] Shi Y., Huang Z., Wang W., Zhong H., Feng S., Sun Y., *Masked label prediction: Unified massage passing model for semi-supervised classification*, *CoRR*, 2020, doi: 10.48550/arxiv.2009.03509.

[4] Wu Z., Ramsundar B., Feinberg E., Gomes J., Geniesse C., Pappu A.S., Leswing K., Pande V., *Moleculenet: a benchmark for molecular machine learning*, *Chem. Sci.*, 2018, vol. 9, pp. 513–530, doi: 10.1039/C7SC02664A.

[5] Morris C., Kriege N.M., Bause F., Kersting K., Mutzel P., Neumann M., *Tu-dataset: A collection of benchmark datasets for learning with graphs*, *CoRR*, 2020, doi: 10.48550/arXiv.2007.08663.

On Usefulness of Dominance Relation for Selecting Counterfactuals from the Ensemble of Explainers

Ignacy Stępka[0009−0004−4575−0689], **Mateusz Lango**[0000−0003−2881−5642], **Jerzy Stefanowski**[0000−0002−4949−8271]

Poznan University of Technology
Institute of Computing Science
Piotrowo 2, 60-965 Poznań, Poland
{mlango,jstefanowski}@cs.put.poznan.pl
ignacy.stepka@put.poznan.pl

DOI:10.34658/9788366741928.18

Abstract. *Counterfactual explanations are widely used to explain ML model predictions by providing alternative scenarios. However, choosing the most appropriate explanation method and one of generated counterfactuals is not an easy task. In this paper, we propose an approach that filters out a large set of counterfactuals generated by a set of diverse algorithms through a multi-criteria subset selection problem solved using the dominance relation. Experiments show that exploiting the dominance relation results in a concise set of counterfactual explanations.*
Keywords: *Explainable AI, Counterfactual explanations, Multiple evaluation criteria, Dominance relation*

1. Introduction

The current development of machine learning (ML) has led to a growing interest in explaining how predictive models work; for the review of different methods see e.g. [1]. In this paper, we focus our interest on counterfactuals. Unlike other explanation methods, which attempt to answer why the prediction is made, *counterfactual explanations* (briefly counterfactuals) provide recommendations on how to change values of some attributes describing the considered instance in order to get the more desired predictions. Typical real-life examples include indicating changes to a rejected loan application that may lead to its approval or assessing possible minor adjustments to an apartment that may increase its sale price.

Counterfactuals are an appreciated explanation type as they are quite intuitive and may provide guidance to individuals on how to achieve their desired outcomes.

There are also psychological justifications for humans considering questions such as „what would have happened if. . . ” and looking for appropriate interventions [2].

Several methods for the generation of counterfactuals, based on different approaches to the perturbations of attribute values, have been proposed; for their review see [3, 4]. However, they may produce considerably different counterfactuals for the instance predicted by the black box model. As the evaluation of these counterfactuals with current measures may be still ambiguous, then the selection of one explanation method is a nontrivial task.

Instead of looking for this single explanation method, we advocate for providing richer counterfactual information for the given instance. Therefore, we propose to use an *ensemble* of different explanation methods (briefly explainers) in order to generate a diverse set of alternative counterfactuals.

In our opinion, the user should receive support in the analysis of this set of counterfactuals from the *multiple criteria* point of view. This leads us to the choice of these criteria and investigation of how many *non-dominated solutions on the Pareto front* can be obtained using the currently popular counterfactual explainers. Recall that the *dominance relation* for two alternatives x and y means that the solution x is rated no worse on each criterion than the alternative y and it is better on at least one of them. The application of the dominance relation is beneficial in this context due to its ability to identify the optimal alternatives in multi-criteria situations without relying on the feature space. This property ensures that the approach remains impartial towards specific attributes and this relation provides objective information in the space of many criteria, while it does not require specialized methods of modeling the preferences of decision makers.

Another important issue for the selection of good counterfactuals is to address the need to impose certain *constraints* to forbid changes on some attributes [5]. Sensible attributes (e.g. race) should not be altered as they cannot be acted upon. While there may be cases where allowing changes to sensitive attributes could be beneficial (e.g. when the goal of explanations is to expose hidden biases in the model), these situations are not realistic for counterfactual scenarios

To sum up, the aim of our paper is to experimentally investigate to what extent the use of the above-mentioned constraints and the dominance relation in the multi-criteria analysis of counterfactuals from the ensemble of explainers methods can lead to the selection of a diverse and concise subset of explanations.

2. Counterfactual explanations – properties and evaluation measures

The literature review shows that in order to meet human expectations, counterfactuals should have several properties [3, 4]. We discuss the most relevant ones

below[1]. *Validity*: counterfactual explanations are expected to actually change the prediction of the classification ML model f to the desired one ($f(x') \rightarrow y'$). *Proximity*: generated counterfactual x' should be as similar as possible to the original instance x in terms of the considered distance measure. *Sparsity*: a counterfactual should change as few attributes as possible because people find it easier to understand short explanations [2]. *Feasibility* (*Plausibility*): counterfactuals should be located in such positions to ensure their plausibility (as some methods may produce out-of-distribution, unrealistic attribute-value distributions). A counterfactual should be *unambiguous*, i.e. it cannot be a borderline case. *Actionability*: counterfactuals should not alter immutable features (e.g. age, race) choice of which might be an important subject of the user's (hidden) preference or restricted with respect to background knowledge. In [5] the more extended set of constraints was also considered e.g. *monotonicity* – the direction of attribute value changes (e.g. only an increase in age is allowed) and an appropriate *encoding of nominal attributes*.

These properties are referred to the following quality measures, some of which will be selected as criteria in our approach.

Proximity is measured as the attribute-wise distance between the counterfactual explanation and its original instance $d(x', x)$. *Attribute sparsity* – the number of changed attributes. *Feasibility* (out-of-distribution measure) is defined as the average distance to k real examples in the data X, i.e. $\min_k \forall_{z \in X \setminus \{x\}} d(x', z)$. *Discriminative power* of x' as the reclassification rate of its k nearest neighbors in training data (i.e. how many of them predict the same class as x' – which is the postulate of unambiguous). *Actionability* will be restricted in our further experiments as the binary indicator of whether any attribute change between x and x' violates any constraint from the set of non-actionable constraints C defined for the considered dataset.

3. Experiments

In order to obtain a set of counterfactual explanations for a given testing instance, we considered Python implementations of selected methods from libraries CARLA, CFEC, DICE-ML and ALIBI. Our analysis focused on assessing the usability of the implementation, the compatibility of data representations and explanations, and the potential for integration in a coherent software environment. As a result, we chose the following 9 methods (Dice, Cadex, Fimap, Wachter optimization, CEM, CFProto, Growing Spheres – GS, ActionableRecourse – AR, FACE)[2] Where possible, we forced generations of up to 20 counterfactuals for each of the

[1] In the following considerations we will use the following notation: x' is a counterfactual of the original instance x and its predicted class $f(x) = y$, obtained by perturbations of attributes in x which lead to the desired change in the prediction $y' = f(x') \neq y$.

[2] The description of these methods can be found, e.g., in [3, 4]

implemented methods.

For the evaluation of the generated counterfactuals, we chose the following 3 criteria: Proximity, Feasibility (with the number of neighbors k equal to 3) and Discrimination Power (with $k = 9$). HEOM was used as a distance measure in each of these criteria. We abandoned consideration of the sparsity measure because preliminary experiments showed its strong correlation with the proximity measure.

As a black box classifier, the neural network consisting of 3 linear layers, each of which consists of 128 neurons and ReLU activation function, was used.

Two datasets, Adult and German Credit, were chosen for the experiments due to their frequent use in the evaluation of counterfactuals in the related literature. Recall that the Adult collection contains a description of 32,000 US residents described by 14 attributes and assigned to two wealth classes. From this dataset, 250 examples (balanced sizes of both classes) were randomly selected for the test set. Constraints (not allowed changes) were defined for three attributes (race, gender and native country). In the case of the German credit set (consisting of 1,000 credit applications assigned to two classes), the test set included 50 examples from each class. One attribute was blocked as the constraint.

The analysis of generated counterfactual explanations consists of the following steps: (1) validity – checking whether the counterfactual leads to a change of the model decision; (2) actionability – checking whether the received counterfactual does not violate the constraints; (at both steps, the solutions that did not meet these requirements were removed); (3) applying dominance relations for pairs of solutions to reject dominated counterfactuals and create a Pareto front.

Table 1. The average number of retrieved counterfactual explanations per instance in the test datasets, along with their standard deviation.

Dataset	**All**	**Valid**	**Actionable**	**P-Front**	**Time**
Adult	81.4 ± 2.8	67.1 ± 9.2	60.2 ± 8	7.6 ± 3.7	$106.6s \pm 9.5s$
German	85 ± 3.8	66.5 ± 7	65.4 ± 7.3	11.9 ± 3.9	$89.1s \pm 7.9s$

Table 1 presents the average numbers of counterfactuals generated by the ensemble per single test instance in both datasets (with detailed numbers after each processing steps). Analogous results for each of the considered explanation methods are presented in Table 2.

4. Discussion and final remarks

Based on the experimental results presented in Table 1, our proposed ensemble of explainers produces, on average, over 80 counterfactual explanations per in-

Table 2. A comparison of methods in the ensemble of explainers by the average number of generated counterfactuals per instance in the test datasets, with the same columns as Table 1.

Dataset	**Adult**				**German**			
	Any	**Valid**	**Act.**	**Front.**	**Any**	**Valid**	**Act.**	**Front.**
Dice	20	20	20	0.5	20	20	20	2.22
Cadex	6.53	6.38	6.38	0.96	11.9	10.87	10.87	1.34
Fimap	6	4.87	4.87	0.37	6	3.38	3.38	0.84
Wachter	9.96	3.40	3.40	1.69	10	2.33	2.33	2.71
CEM	1	0.5	0.5	0.4	1	0.21	0.21	0.41
CFProto	7.66	7.63	2.26	0.14	5.76	5.71	4.81	0.3
GS	20	14.34	14.34	0.16	20	13.85	13.85	0.24
AR	0.29	0.15	0.15	0.1	0.34	0.17	0.17	0.15
FACE	10	9.78	8.28	3.27	10	9.98	9.98	3.74

stance. It should be noted that a relatively large number of these explanations fail to meet the validity and actionability requirements, and are subsequently removed from further consideration. By evaluating the resulting set of counterfactuals in a multi-criteria setting with the dominance relation, we are able to identify a concise Pareto front, resulting in a sufficient number of alternatives (8 – 12) to be further considered by the user. An analysis of the considered methods (as shown in Table 2) reveals that some methods, such as Dice, Cadex, FACE, and Wachter, perform better than others, such as CFProto, GS, and AR. Nonetheless, all methods are capable of generating nondominated counterfactuals, i.e. they can make a varied contribution to the proposed ensemble of explainers.

In further research, we will consider the next stage of multi-criteria support for the decision maker in the final choice of a compromise solution – a counterfactual – from the Pareto Front.

Acknowledgment

The research has been partially supported by 0311/SBAD/0743 PUT University grant and TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA no. 952215.

References

[1] Bodria F., Giannotti F., Guidotti R., Naretto F., Pedreschi D., Rinzivillo S., *Benchmarking and survey of explanation methods for black box models*, 2021, doi: 10.48550/arXiv.2102.13076.

[2] Miller T., *Explanation in artificial intelligence: Insights from the social sciences*, *CoRR*, 2017, doi: 10.48550/arXiv.1706.07269.

[3] Guidotti R., *Counterfactual explanations and how to find them: literature review and benchmarking*, *Data Mining and Knowledge Discovery*, 2022, doi: 10.1007/s10618-022-00831-6.

[4] Verma S., Boonsanong V., Hoang M., Hines K.E., Dickerson J.P., Shah C., *Counterfactual explanations and algorithmic recourses for machine learning: A review*, 2020, doi: 10.48550/arXiv.2010.10596.

[5] Falbogowski M., Stefanowski J., Trafas Z., Wojciechowski A., *The impact of using constraints on counterfactual explanations*, *Proceedings of the 3rd Polish Conference on Artificial Intelligence, PP-RAI 2022*, 2022, pp. 81–84.

Towards Detection of Unknown Polymorphic Patterns Using Prior Knowledge

Przemysław Kucharski[0000−0001−6051−2962],
Krzysztof Ślot[0000−0003−1228−0970]

Lodz University of Technology
Institute of Applied Computer Science
Stefanowskiego 18/22, 90-537 Łódź, Poland
pkuchars@iis.p.lodz.pl

DOI:10.34658/9788366741928.19

Abstract. *The presented paper proposes a novel approach for detecting unknown polymorphic patterns in sequences composed of random symbols and of known polymorphic patterns. We propose to represent rules that drive pattern generation as regular expressions. To detect unknown patterns, we first incorporate knowledge on known rules into a Convolutional Autoencoder (CAE), then we train the CAE with additional objective to prevent weights from learning the already known patterns. Analysis of training results provides statistically significant information on presence or absence of polymorphic patterns that were not previously known.*
Keywords: *polymorphic pattern detection, knowledge and learning integration, Convolutional Autoencoder.*

1. Introduction and Related Work

Recent trends in machine learning indicate an emerging need for methods and models capable of incorporating explicitly formulated knowledge. This is especially important when training data are scarce and shortage of available information needs to be compensated with other means. Motif detection is one of the most challenging tasks in data analysis, due to polymorphic nature of patterns that can encode a given information and scarcity of data, which impairs learning feasibility.

The presented paper is concerned with research on detection of polymorphic patterns in sequences, by utilizing prior knowledge to facilitate the considered, difficult task. Polymorphic sequences considered in the paper are sequences generated by regular expressions with flexible rules allowing high diversity of valid, alternative sequence instances.

A significant amount of research have been done so far on handling the problem of detecting patterns (motifs) in sequences, among which we can name methods that make the patterns recognized by convolutional neural networks more disentangled. Liang et al. [1] propose a training method for classification models that make the convolutional filters class-specific.

Koo et al. [2] examine representation of genomic motifs in CNNs. They searched for motifs in first layer convolutional filters, transforming them into position-weight matrices basing on the response of the filter to specific samples.

Zhang et al. [3] propose a method of updating a specific subfilter cascade chosen dynamically during training to produce more diverse convolutional filters and reduce overlap in representation. More general examination of this problem founds the solutions like structuring the network to resemble the knowledge base, which can be done either manually [4] or generated in the training process [5].

2. The proposed methodology

An objective of the proposed approach is to train a network that analyzes input using a cascade of convolutional filters, where a part of this structure is preset to encode knowledge on known polymorphic sequences (we refer to it as a Fixed Convolutional Module – FCM) and the remaining part is expected to learn any previously unknown regularities that exist in data (we refer to it as the Learnable Convolutional Module – LCM). We adopt a Convolutional Autoencoder (CAE) to be a framework for filter weight training, as it enables monitoring of a pattern learning process.

Convolutional filters can be seen as pattern-detecting operators that produce the maximum output for inputs that match filters' weights. To provide flexible representation of polymorphic patterns of arbitrary length it is reasonable to use a cascade of simple filters, arranged in a conventional convolutional layers. The first layer filters could be designed to capture different short patterns that comply to local rules defined by a given regular expression, whereas the purpose of subsequent layers could be to combine these short chunks into longer strings, that are compatible with the considered expressions. An ease in defining filter cascades that specialize in detecting specific input patterns enables simple incorporation of initial knowledge into a structure of convolutional neural networks.

The second layer filters that are to merge short segments detected by first-layer filters into the longer ones, need to have a depth that enables integration of all relevant first-layer detectors.

The proposed knowledge injection method can be considered universal, albeit in the presented experimental scenario several constraints were introduced in the filters. The method can be scaled both in terms of the number and size of patterns, as well as in terms of rule-complexity, by adding new layers of filters, and be used

in numerous applications, such as bioinformatics or anomaly detection. It should be emphasised that the size of filters does not define the exact length of the pattern, but only constraints its maximum length – as the proposed methodology allows for patterns that contain any character at each position, including the ends. Therefore, shorter patterns can be injected or detected by filling the remaining positions with expression allowing any character.

To search for unknown polymorphic patterns that are embedded in sequences comprising runs of random symbols together with known, possibly polymorphic strings, we initialize our algorithm with knowledge provided in a form of a cascade of appropriately preset filters.

The reconstruction follows the scheme provided in Equation 1, which involves weight normalization, aimed at converting learned weights into probabilities (transformation of R into R"), followed by thresholding that is expected to produce unambiguous basis for reconstruction of the detected regular expression. Here, the symbols U and W denote position-wise minimum of R and position-wise sum of R', respectively.

$$\begin{aligned} R' &= R + 2 * |U| \quad \text{where} \quad U_i = \min_{j \in J} R_{ij},\ i = 1, 2, \ldots, n \\ R'' &= R'/W \quad \text{where} \quad W_i = \sum_{j \in J} R'_{ij},\ i = 1, 2, \ldots, n \\ R''' &= f(R'') \quad \text{where} \quad f(r_{ij}) = \begin{cases} 1 & r_{ij} > th_{upper} \\ -1 & r_{ij} < th_{lower} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{1}$$

The reconstructed regular expression RE is defined as a tree, build of shorter chunks RS, encoded by the first layer filters:

$$\begin{gathered} \forall_{s_i \in S} \quad s_i \in R_i \\ R \in RE \\ RE = \{RS\} \diamond \{\vee, \wedge\} \end{gathered} \tag{2}$$

where $\diamond$ denotes a recursive tree operator, s_i is a symbol at i^{th} position of a string S that is admissible at this position of the regular expression R (i.e. R_i), which is a specific instance of an expression RE.

We expect that throughout learning, any new polymorphic, unknown patterns present in input sequences, will get learned by learnable weights of the convolutional module. Learning of rules that underlie new polymorphic patterns might be impaired by influence of patterns that are already known to the network. Therefore, we consider an additional learning scenario, where learnable convolutional

filters are discouraged to discover knowledge that has already been injected to the network via FCM filters. To measure similarity of rules that generate patterns, we use a mean Levenshtein distance applied to pairs of sequences produced by the considered regular expressions.

3. Experiments

The proposed polymorphic pattern detection procedure has been trained on a datasets made up of 100 40-element long sequences. This small number of samples is motivated by the scarcity of genomic data representing rare patterns. Human genome data (GRCh38.p14 assembly) was used in the experiment. Short samples with set of 2 or 3 patterns in each were extracted, Labeling was performed with pattern detection using full matching.

Convolutional Autoencoder has been used as a computational framework for polymorphic pattern detection experiments.

Every network was trained for 500 epochs with the use of Mean Square Error as loss function and RMSProp as optimizer with learning rate of 0.001

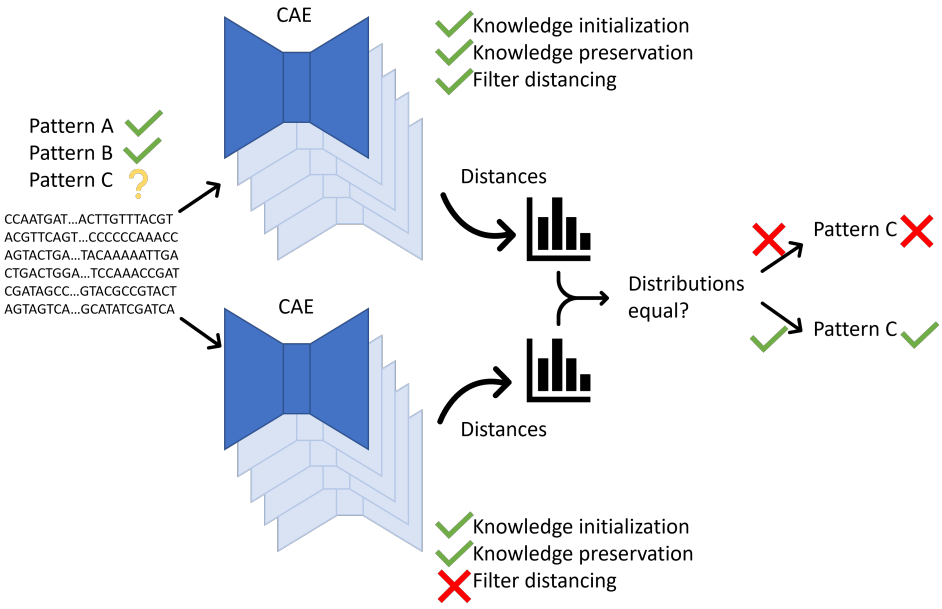


Figure 1. The flowchart of the proposed testing method. The purpose is detection of existence of unknown pattern C. Source: own work.

4. Results

The target characteristics to be quantified at the completion of training was a mean distance between the learned filter cascades and the fixed, knowledge-representing cascades. We were interested, whether there exist statistically significant differences among results produced for four different experimental scenarios (Figure 1):

1. Unknown pattern present and filter similarity discouragement turned on.
2. Unknown pattern not present and filter similarity discouragement turned on.
3. Unknown pattern present, no filter similarity discouragement.
4. Unknown pattern not present, no filter similarity discouragement.

For each of the scenarios, the resulting mean distances between pairs of regular expressions represented by LCM filters and FCM filters were evaluated. The results of Levene test and Fligner tests, for which the null hypothesis is that samples are drawn from the same distribution, show significant outcome when group 2 is tested against group 4, For groups 1 and 3, test results give no basis for rejecting the null hypothesis – in both training regimes, rules that are similarly distant from the preset ones are learned (Table 1).

Table 1. Results of statistical tests.

Unknown regex present in data	Levene test		Fligner test	
	Statistics	p-value	Statistics	p-value
True	1.89	0.09	3.74	0.24
False	3.56	0.02	9.57	0.01

5. Conclusions

The proposed method for unknown polymorphic pattern detection introduces several novel elements. Firstly, we show how prior knowledge on rules, which generate some of the patterns that could be found in input sequences, can be incorporated into a network and preserved during training. Another contribution is concerned with the proposal of measuring a distance between pattern-generating rules by evaluation of Levenshtein distances between sequences generated using these rules. The proposed network architecture is designed in a way that enables injection of complex knowledge. It is also worth noting that the presented problem

is complex and difficult to be solved by traditional approaches. As it can be seen from the results, the proposed data processing pipeline built upon the introduced methodology is capable of answering the question whether new, unknown patterns are present in the data.

References

[1] Liang H., Ouyang Z., Zeng Y., Su H., He Z., Xia S.T., Zhu J., Zhang B., *Training Interpretable Convolutional Neural Networks by Differentiating Class-Specific Filters*, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, vol. 12347 LNCS, pp. 622–638, doi: 10.1007/978-3-030-58536-5-37.

[2] Koo P.K., Eddy S.R., *Representation learning of genomic sequence motifs with convolutional neural networks*, *PLoS Computational Biology*, 2019, vol. 15, no 12, pp. 1–17, doi: 10.1371/journal.pcbi.1007560.

[3] Zhang D., He L., Luo M., Xu Z., He F., *Weight asynchronous update: Improving the diversity of filters in a deep convolutional network*, *Computational Visual Media*, 2020, vol. 6, no 4, pp. 455–466, doi: 10.1007/s41095-020-0185-5.

[4] Towell G.G., Shavlik J.W., *Knowledge-based artificial neural networks*, *Artificial Intelligence*, 1994, vol. 70, no 1-2, pp. 119–165, ISSN 00043702, doi: 10.1016/0004-3702(94)90105-8.

[5] Gaier A., Ha D., *Weight agnostic neural networks*, [In:] H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (eds.), *Advances in Neural Information Processing Systems*, pp. 5364–5378, doi: 10.48550/arXiv.1906.04358.

Chapter 3

Interdisciplinary Applications of Artificial Intelligence

Domain Editors:

1. Jacek Mańdziuk, Warsaw University of Technology
2. Agnieszka Ławrynowicz, Poznań University of Technology
3. Jarosław Wąs, AGH University of Science and Technology
4. Adam Wojciechowski, Lodz University of Technology

AI-driven Ecodriving and ETA Solutions for Truck Transport

**Piotr Lipiński[1], Michał Morawski[1], Piotr Napieralski[1],
Paweł Nowok[2], Bartosz Zawiślak[2], Leszek Hojdys[2], Marcin Lazar[2]
Przemysław Lazarek[2], Norbert Zając[2], Sylwester Pizoń[2]
Rafał Jakubiec[2], Jacek Sienkiewicz[2], Sebastian Gołąbek[2]
Mateusz Kabocik[2], Szymon Fedrizzi[2], Michał Kuliga[2]
Mateusz Frączkiewicz[2], Mirosław Malarz[2], Jarosław Puchalski[2]
Ewa Danysz[2], Maciej Grajcarek[2]**

[1]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
piotr.napieralski@p.lodz.pl
[2]*INELO Polska sp. z o.o.*
ul. Karpacka 24/U2b
43-300 Bielsko-Biała,

DOI:10.34658/9788366741928.20

1. Introduction

The transport industry is facing challenges due to rising costs, labor shortages, environmental pressure, and increasing customer demands for timely delivery. In response, Inelo Polska Sp. z o.o. has developed an innovative product called Ecodriving and Estimated Time of Arrival (ETA) solutions. The AI-driven system aims to optimize the economy of driving and predict the delivery time to reduce transportation costs, increase delivery efficiency[1, 2], improve workforce quality, and reduce environmental impact. The project aimed to develop AI algorithms that could optimize driving economy and predict delivery time to enable users to make real-time adjustments. The project team integrated Ecodriving and ETA solutions into Inelo's Intelligent Transport Management System (ITMS). The ITMS enables users to manage fleets and transport orders, monitor driver behavior, and track vehicle locations in real-time. The developed AI-driven Ecodriving and ETA solutions help transport companies optimize their operations by reducing fuel consumption, increasing delivery efficiency, and improving driver behavior. The system provides drivers with real-time feedback and recommendations to optimize their driving style, reducing fuel consumption, and CO2 emissions[3]. The AI

algorithms also predict the estimated time of arrival, helping companies manage transport orders, reduce delays, and enhance customer satisfaction.

2. Integration of Multi-Criteria Optimization Algorithms for Ecodriving and ETA

Based on previous research, an effective method for integrating research conducted in earlier stages has been developed. However, the main difficulty is constructing a superior multi-criteria optimization algorithm [4] in the ISZT module, and building a system that allows for the simultaneous optimization of both travel time and cost, which is an extremely complex issue due to the apparent contradiction between these variables. The proposed algorithm allows for the development of a version of the system that integrates previously developed solutions. The system will be installed in selected vehicles of INELO's clients in the future, and dispatchers (logisticians) will have access to the ETA test version. The study will determine whether the parameters obtained under laboratory conditions are achievable in operational conditions, particularly whether the developed models still function with the expected precision (travel time estimation error < 5%, fuel savings> 5%) for many different models and versions of vehicles with varying degrees of use, different drivers, and different dispatchers (logisticians). The research will also encompass ease of use, clarity and accuracy of messages (Eco-driving).

All of these elements will only be possible after designing a theoretical model in laboratory conditions, which will enable integration. The work at this stage leads to the development of intelligent multi-criteria optimization algorithms that consider simultaneous optimization of travel costs through the Eco-driving module and travel time through the ETA system, as well as data provided by users (such as costs and commercial conditions of orders). To better estimate while considering eco-driving, optimization planning and minimizing the total distance traveled by vehicles with driver stops and rest are proposed.

The developed method proposes a multi-objective optimal intelligent planning algorithm. It is essential to introduce the model to solve the path planning problem considering all the factors described in the previous reports. The proposed multi-objective path planning algorithm uses a stochastic algorithm and variable probability individual optimization of local exchange search methods. The tested algorithm's solution and optimal solution deviation are minimal, and its efficiency and effectiveness are better than existing solutions. The significant advantage of the proposed solution is its adaptability and the possibility of further algorithm improvements and applications.

At this stage, we have identified the preliminary concept of the technology (using artificial intelligence algorithms to analyze factors affecting estimated driving time and cost efficiency and transmitting this information in real-time) and its

future applications (calculating work time – ETA – and optimizing travel costs – Eco-driving – and their integration in the ISZT).

This project employs an approach that incorporates multi-criteria optimization algorithms designed for creating AI-powered Ecodriving and ETA (Estimated Time of Arrival) solutions specific to truck transportation. These solutions were integrated into an Intelligent Transport Management System (ITMS)[5], a platform enabling real-time supervision and coordination of aspects like fleet status, transportation orders, driver behavior, and vehicular positioning. The algorithms deployed take into account simultaneous optimization of travel duration and expense via the Ecodriving and ETA systems, alongside the incorporation of user-provided data.

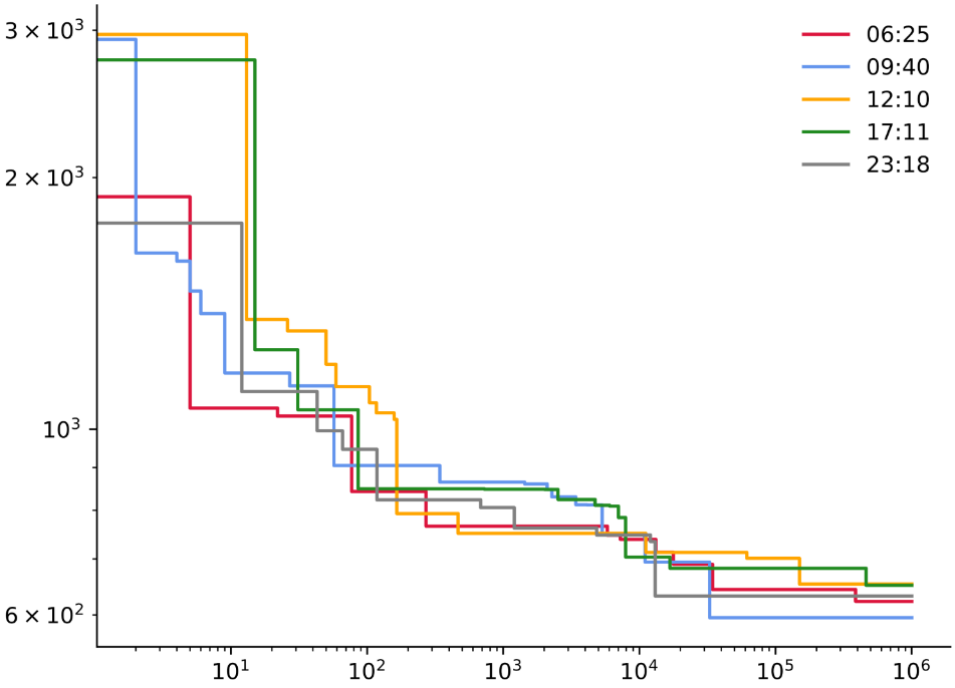


Figure 1. the dependence of the obtained quality index J on the number of iterations (epochs), for different optimisation problems, with differences due to different departure times. Source: own work.

The project team has developed a multi-objective path planning algorithm, utilizing a stochastic algorithm and the individual optimization of local exchange search methodologies with variable probability. The discrepancy between the solution generated by this algorithm and the optimal solution is minimal, demonstrating that its performance and effectiveness surpass those of existing solutions.

The objective function used in this project can be expressed as:

$$\min_x f(x) = \min_x \{c_1(x) + c_2(x)\} \tag{1}$$

where $f(x)$ is the objective function, $c_1(x)$ represents the cost of eco-driving, and $c_2(x)$ represents the estimated time of arrival. The variable x represents the various factors that affect the optimization, such as driver behavior, vehicle performance, and road conditions. The algorithm seeks to minimize $f(x)$, which is the sum of $c_1(x)$ and $c_2(x)$, to simultaneously optimize both travel time and cost. The algorithm seeks to minimize both Cost and Time simultaneously, offering an optimal solution for eco-driving and ETA.

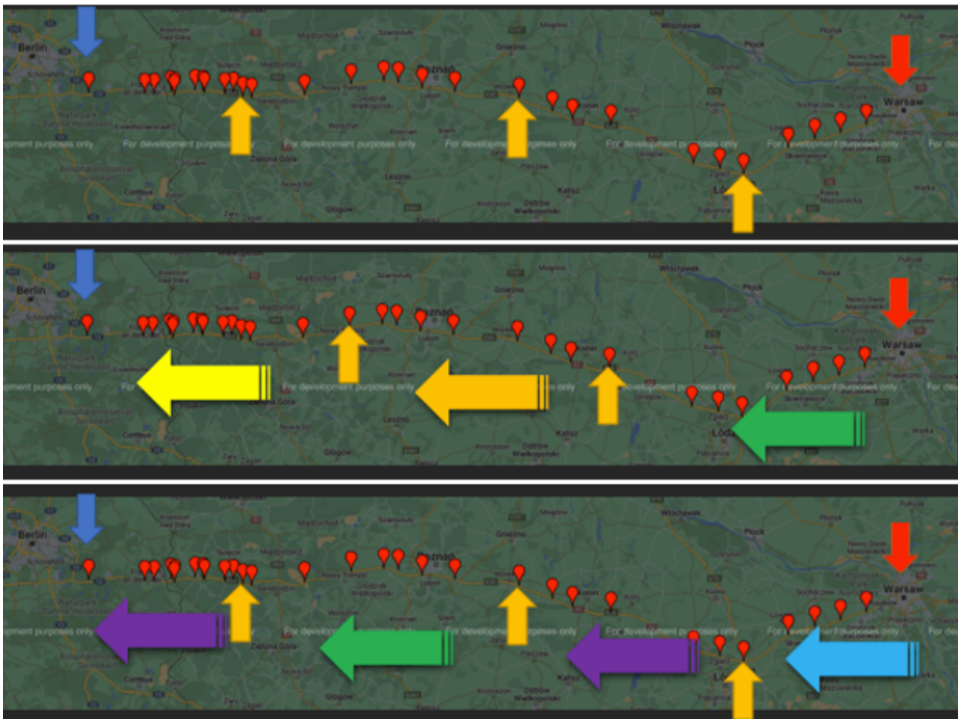


Figure 2. Method of determining stopping places and optimising journeys. Source: own work.

3. Conclusions

In this project, a multi-objective optimization algorithm was developed to address the challenges facing the transport industry. The algorithm was designed to optimize travel time and cost simultaneously by considering factors such as driving

behavior, vehicle efficiency, and delivery requirements. The algorithm was tested in realistic conditions and showed close results to the optimal solution.

Integrating intelligent multi-criteria optimization algorithms for eco-driving and ETA systems is a challenging task, but with a well-designed algorithm and a suitable system, it can be possible to achieve simultaneous optimization of travel time and cost. The proposed solutions have significant potential for application in real-world scenarios, and further research and development are essential to realize the full potential of these solutions. The AI-driven Ecodriving and ETA solutions developed by Inelo Polska Sp. z o.o. offer an innovative and effective solution for the challenges facing the transport industry. The system provides real-time feedback, recommendations, and ETA predictions, enabling users to make real-time adjustments, reduce costs, and enhance environmental performance. The system's benefits extend to transport companies, drivers, and customers, making it an essential tool for the transport industry's future.

Acknowledgment

The research presented in this article was made possible through the financial support of "Zastosowanie sztucznej inteligencji w rozwiązaniach optymalizacyjnych dla transportu ciężarowego nr proj.: POIR.01.01.01-00-1283/19".

References

[1] Javaid M., Haleem A., Singh R.P., Suman R., Gonzalez E.S., *Understanding the adoption of industry 4.0 technologies in improving environmental sustainability*, *Sustainable Operations and Computers*, 2022, vol. 3, pp. 203–217, ISSN 2666-4127, doi: https://doi.org/10.1016/j.susoc.2022.01.008.

[2] Garcia-Musila F.E., Gonzalez-Sanchez R., Ferrari A.M., Volpi L., Pini M., Siligardi C., Settembre-Blundo D., *Identifying the equilibrium point between sustainability goals and circular economy practices in an industry 4.0 manufacturing context using eco-design*, *Social Sciences*, 2019, vol. 8, no 8, ISSN 2076-0760, doi: 10.3390/socsci8080241.

[3] Hyeon E., Ersal T., Kim Y., Stefanopoulou A.G., *Loss function design for data-driven predictors to enhance the energy efficiency of connected and automated vehicles*, *IEEE Transactions on Intelligent Transportation Systems*, 2023, vol. 24, no 1, pp. 827–837, doi: 10.1109/TITS.2022.3216748.

[4] Liu L., Mu H., Yang J., Li X., Wu F., *A simulated annealing for multi-criteria optimization problem: Dbmosa*, *Swarm and Evolutionary Computation*, 2014, vol. 14, pp. 48–65, ISSN 2210-6502, doi: https://doi.org/10.1016/j.swevo.2013.09.001.

[5] Nigam N., Singh D.P., Choudhary J., *A review of different components of the intelligent traffic management system (itms)*, *Symmetry*, 2023, vol. 15, no 3, ISSN 2073-8994, doi: 10.3390/sym15030583.

Analysis of Surface EMG Signals to Control of a Bionic Hand Prototype

Adam Pieprzycki[1][0000−0001−7059−2118],
Daniel Król[1][0000−0002−8611−0838],
Piotr Wawryka[2][0000−0002−7096−1275],
Katarzyna Łachut[2][0000−0001−7543−0257],
Mateusz Hamera[0000−0003−3090−7634],
Bartosz Srebro[3][0000−0002−5345−7800]

[1]*Department of Computer Science*
University of Applied Sciences in Tarnów
Mickiewicza 8, 33-100 Tarnów, Poland
a_pieprzycki@anstar.edu.pl, d_krol@anstar.edu.pl
[2]*Scientific Club of Computer Science*
University of Applied Sciences in Tarnów
[3]*Department of Automation and Robotics*
University of Applied Sciences in Tarnów
Mickiewicza 8, 33-100 Tarnów, Poland
m_srebro@anstar.edu.p

DOI:10.34658/9788366741928.21

Abstract. *The aim of the presented project is to develop a comprehensive system for acquiring surface EMG data and carry out time-frequency analysis to determine useful parameters for subsequent gesture classification for a simple bionic hand prosthesis. This system is expected to assist in controlling both the prosthetic hand and the robotic hand in making precise gestures with the fingers on the hand. The article presents the methods for acquiring and processing multi-channel EMG signals and feature extraction for gesture recognition by an artificial neural network (ANN).*
Keywords: *Bionic hand, surface EMG, sEMG, neural network, NN, Fast Fourier Transform, FFT, Hilbert-Huang Transform, HHT, blind source separation, BSS*

1. Introduction

Electromyographic (EMG) signals are electro-physiological signals originating in muscles in response to muscle stimulation by electric potential from the nervous system. The most important muscles in the context of this study are the

superficial flexor muscle of the fingers (FDS – m. flexor digitorum superficialis), the deep flexor muscle of the fingers (FDP – m. flexor digitorum profundus), and the extensor joint muscle of the fingers (EDC – m. extensor digitorum communis). EMG analysis provides information about these muscles' electrical activity during different movement phases. The Electromyographic signals were recorded following the guidelines of SENIAM (The European Recommendations for Surface Electromyography).

2. Materials and Methods

A prototypical LPCXpresso LPC1347 board was used to acquire the electromyographic signals. The system is equipped with a 32-bit ARM Cortex-M3 core microprocessor clocked at 72 MHz and a 12-bit SAR-type ADC converter with an 8-channel multiplexer and a hardware sequencer that switches the measuring channels with the DFRobot Gravity OYMotion SEN 0240 type (3.5 x 2.2 cm) surface EMG sensors.

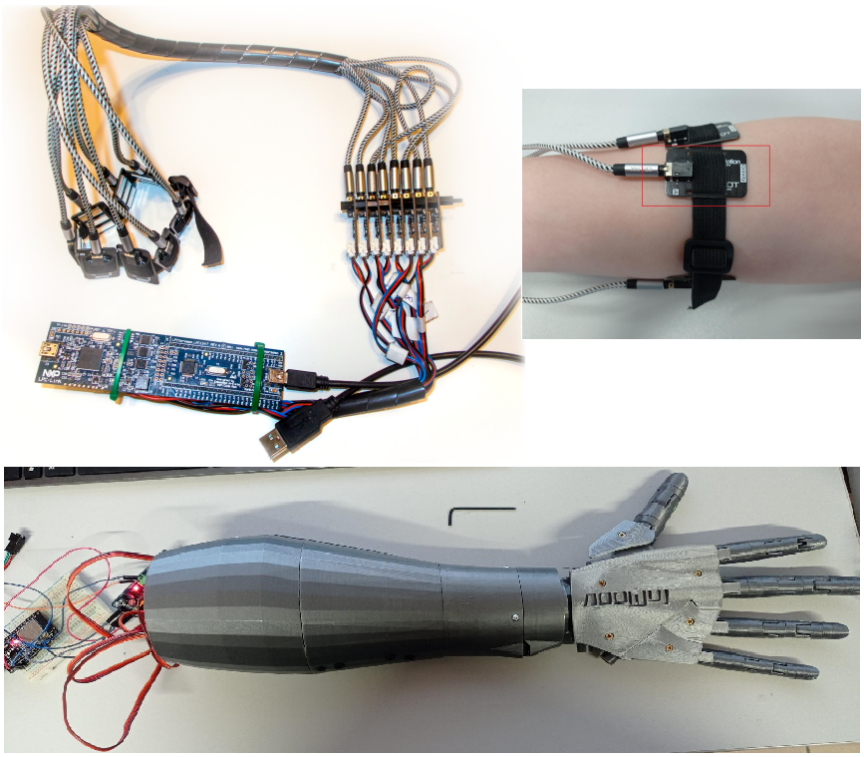


Figure 1. Prototype installation for the analysis of electromyographic signals and positioning of the first electrode and printed bionic hand. Source: own work.

The following gestures were arbitrarily selected for sEMG analysis: straight fingers, clenched hand, victory, three middle fingers straight, and the OK gesture. A software program for data acquisition to multi-channel WAVE file [1] was written in C++ Qt Creator.

Fourier and Hilbert-Huang transform [2, 3] were applied using Matlab environments to extract selected features of the EMG signal [1]. A neural network (NN) was implemented and trained using the same computational environment (Matlab with Machine Learning Toolbox).

The assumption was made to identify for 109 healthy volunteers four gestures of the right hand. Electrode no. 1 was placed on the brachioradialis muscle (BR), in order of channels towards the outside, looking from the volunteer's side.

3. Preprocessing EMG signal features

Time-domain (t), frequency-domain (f), and time-frequency (tf) analyses of the signal, as well as Blind Source Separation (BSS) [4] and smoothing of the sEMG signal [5], was conducted.

The time-frequency (tf) analysis of the Hilbert-Huang transformation method (HHT) [2] considered in this article uses an Empirical Mode Decomposition (EMD) algorithm [2], which is adaptive enough to separate the Intrinsic Mode Function (IMF) components from each other both in time and frequency. The transform makes it possible to determine the distribution of energy of its frequencies in the time domain, and it resembles a wavelet transform.

Each of the signals received by any of the channels $x(t)$ is decomposed accordingly to the algorithm introduced by Huang and realized as:

$$x(t) = \sum_{i=1}^{K} h_i(t) + r_K(t), \tag{1}$$

where $h_i(t)$ is a function that might be a component of the IMF, and $r_K(t)$ is the residual signal.

The process of finding an IMF component is approached as follows: the signal of the local mean average $m_k(t)$ (the mean average of the upper and lower envelopes of the signal) is subtracted k-times until an IMF function is found, validated by two criteria [6]:

- the number of extrema and the number of times the signal crosses zero must be equal, or the difference between them cannot be greater than 1,
- the mean values of the envelope interpolating the local maxima and the envelope interpolating the local minima amount to 0.

The following parameters can be analyzed in all channels: [7]: RMS (Root Mean Square), ENT (Entropy), ENE (Energy), MEDIAN frequency (frequency domain) after FFT, AR (autoregressive model) 1st or 4th order it incorporates the linear combination of the last four samples of each IMF component.

4. Results and Discussion

Calculations were performed for G=4+1 (reference) gestures and neural network-based recognition for 109 healthy volunteers. The data were smoothed over 5 samples using a moving average filter with detection, eliminating unwanted noise, and finding, filling, and removing outliers. It was a linear interpolation of neighboring, nonoutlier values, for 8 sEMG channels and window 128 samples and other parameters specified as: BSS (Blind Source Separation) time delayed with 2, 3, 4 samples, number of IMF (nIMF) – 1, 2, 3 or 4 components, sampling frequency was 2048 Hz and signal acquisition/extraction time was 5 s /1 s.

Table 1. Comparison of selected gesture recognition performance

No	Parameters: (domain)features/BSS/nIMF/HLS/IVS	RA %
1	(t)RMS, /-/-/- /37/128	37.89
2	(t)RMS, /2/-/128/128	39.53
3	(t)RMS, (f)MEDIAN, 2/-/230/256	58.55
4	(t)RMS, (f)ENE /2/-/166/256	67.99
5	(t)RMS,(tf)ENE/2/4/-/638/640	73.05
6	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)ENT,(tf)AR(1)/3/1/365/520	74.69
7	(t)RMS,(f)MEDIAN,(tf)ENE/2/4/642/768	75.24
8	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)ENT,(tf)AR(4)/4/1/489/544	75.51
9	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)AR(4)/2/4/634/896	78.25
10	(t)RMS,(tf)ENE,(tf)ENT/2/4/1093/1152	78.25
11	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)ENT/2/4/1207/1280	78.66
12	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)ENT,(tf)AR(4)/2/4/931/1408	79.75
13	(t)RMS,(f)MEDIAN,(tf)AR(1)/3/2/228/272	85.23
14	(t)RMS,(f)MEDIAN,(tf)ENE/2/1/379/384	85.49
15	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)AR(1)/2/1/262/392	86.27
16	(t)RMS,(f)MEDIAN,(tf)ENE/3/2/295/512	86.53
17	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)AR(1)/3/2/264/528	86.79
18	(t)RMS,(f)MEDIAN,(tf)ENE/2/4/642/768	87.56
19	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)AR(4)/3/2/168/576	88.08
20	(t)RMS,(f)MEDIAN,(tf)ENE,(tf)AR(4)/2/1/276/416	88.08

A feedforward neural network was used for gesture pattern recognition with hidden layer size (HLS) tested in the field of 1 to the size of the data input vector

(IVS), and a training function was scaled conjugate gradient and cross-entropy loss for classification tasks. Comparison of gesture recognition performance based on one-second middle samples were extracted from five-second samples of eight sEMG channels.

The influence of different characteristics of the signal at the neural network input on the accuracy of the recognition (RA) was tested. An analysis of gesture recognition accuracy was carried out for the different combinations of EMG signal parameters analyzed. BSS increased this accuracy, and its highest values were obtained with an input vector that took into account: RMS, median frequency, energy, and parameter values of the 4th-order autoregressive model (Table 1).

A detailed analysis of how the features were extracted contributed to implementing the control prosthetic hand [8]. The designed system is expected [9] to assist in controlling the prosthetic hand in making precise gestures with the fingers of the hand. The model of the bionic hand has been printed on a 3D printer with filament Spectrum PETG 1.75 mm (Fig. 1).

The bionic hand is controlled by six servo mechanisms Tower Pro MG996R with an angle range of 180° and attached to the main board with PCA 9685 Adafruit 815.

Acknowledgements

This work was supported by the university's internal grant BAD/09/2020 PWSZ/PRWRs/0700-8/PN-U/2020.

References

[1] Wawryka P., *EMG*, (access 26-02-2023). https://github.com/informacja/EMG

[2] Huang N.E., Shen S.S.P., *Hilbert-Huang Transform and Its Applications*, Singapore: World Scientific, 2005, doi: 10.1142/5862.

[3] Tan A., *Hilbert-huang transform*, (access: 26-02-2023). https://www.mathworks.com/matlabcentral/ fileexchange/ 19681-hilbert-huang-transform

[4] Parra L., Sajda P., *Blind source separation via generalized eigenvalue decomposition*, *J. Mach. Learn. Res.*, 2003, vol. 4, p. 1261–1269, ISSN 1532-4435.

[5] Kendall M.G., Stuart A., Ord J.K., *The Advanced Theory of Statistics, Vol. 3: Design and Analysis, and Time-Series*, vol. 3, Macmillan, London, 1983.

[6] Gawędzki W., Socha M., Sławik P., *Dekompozycja sygnałów eeg w dziedzinie czasu przy zastosowaniu transformacji hilberta-huanga hht (in polish)*, *Przegląd Elektrotechniczny*, 2015, , no 5, pp. 33–361, ISSN 0033-2097.

[7] Too J., Abdullah A.R., Mohd Saad N., Tee W., *Emg feature selection and classification using a pbest-guide binary particle swarm optimization*, *Computation*, 2019, vol. 7, no 1, doi: 10.3390/computation7010012.

[8] Yoo H.J., Park H.j., Lee B., *Myoelectric signal classification of targeted muscles using dictionary learning*, *Sensors*, 2019, vol. 19, no 10, doi: 10.3390/s19102370.

[9] Pieprzycki A., Król D., *General concept of the emg controlled bionic hand*, *Science, Technology and Innovation*, 2020, vol. 8, no 1, pp. 26–34, doi: https://doi.org/10.5604/01.3001.0014.1901.

Brief Overview of Selected Research Directions and Applications of Process Mining in KRaKEn Research Group

Krzysztof Kluza[1][0000−0003−1876−9603],
Mateusz Zaremba[1][0000−0002−3888−4783],
Dominik Sepioło[1][0000−0001−7746−3781],
Piotr Wiśniewski[1][0000−0003−3777−642X],
Weronika T. Adrian[1][0000−0002−1860−6989],
Maria Teresa Gaudio[2][0000−0003−1057−4836],
Paweł Jemioło[1][0000−0001−5962−4043],
Marek Adrian[1][0000−0002−0435−0994],
Krystian Jobczyk[1][0000−0001−6194−2737],
Mateusz Ślażyński[1][0000−0001−7269−8215],
Bernadetta Stachura-Terlecka[1][0000−0003−2887−5936],
Antoni Ligęza[1][0000−0002−6573−4246]

[1]*AGH University of Krakow,*
al. A. Mickiewicza 30, 30-059 Krakow, Poland
{kluza,mzaremba,sepiolo,wpiotr,wta,pawljmlo,
madrian,jobczyk,bstachur,mslaz,ligeza}@agh.edu.pl
[2]*Laboratory of Transport Phenomena and Biotechnology,*
Department of DIMES, University of Calabria,
Cubo-42a, 87036 Rende, CS, Italy
mariateresa.gaudio@unical.it

DOI:10.34658/9788366741928.22

Abstract. *Process mining allows for exploring processes using data from event logs. By providing insights into how processes are actually executed, rather than how they are supposed to be executed, process mining can be used for optimizing business processes and improving organizational efficiency. In this exploratory paper, we report on selected research threads related to process mining carried out within KRaKEn Research Group at AGH University of Science and Technology. We introduce a collection of initial ideas that require further exploration. Our research threads are concerned with the use of process mining techniques 1) for discovering processes from unstructured data, specifically text from e-mails, 2) for explaining black-box machine learning models, using process models as a global explanation, and*

3) for analyzing data from different food industry systems to identify inefficiencies and provide recommendations for improvement.
Keywords: *Process Mining, Explainability, Knowledge Engineering*

1. Introduction

Process mining techniques [1, 2] usually involve the extraction of data from event logs. The event logs can come from various sources, from manual data entries, through specialized software systems, to sensor data. In this paper, we review selected applications of process mining in the recent developments and research threads of the KRaKEn Research Group (https://kraken.edu.pl/).

The paper is structured as follows. In Section 2, we present selected applications of process mining for model discovery from unstructured data. Section 3 focuses on using process mining for explaining black-box machine learning models. Section 4 provides insight into our future works regarding using process mining for analyzing food industry data. We conclude our paper in Section 5.

2. Process Discovery from Unstructured Data

Process mining techniques can be applied to different types of data, even unstructured ones. Data is unstructured if it has not been structured in a predefined way. Such data usually contain a lot of text, such as open-ended responses in surveys or social media conversations, but also include emails, websites, images, videos, and audio. Because of a lack of structure manner, such data is difficult to analyze and process using traditional methods.

One of the most common forms of unstructured data is text. It can come from a variety of sources, such as social media posts, customer reviews, emails, and documents. In the case of process mining or modeling, e-mail data usually does not directly specify information about the process or its elements.

With the rise of machine learning methods, there exist advanced techniques for analyzing unstructured data. Such techniques include natural language processing for text analysis, computer vision for image analysis, and speech recognition for audio analysis. For the discovery of a process from e-mail data, in our research, we use both supervised and unsupervised machine learning methods to prepare the event log based on e-mails.

Our basic algorithm is based on [3] and for preparing an event log suitable for process mining from e-mail data, we use several steps that have to be performed to avoid data interpretation errors and provide meaningful results. A schema of this procedure is presented in Figure 1. It shows three blocks representing the operations performed and the input and extracted data.

First, the text from e-mails is extracted, normalized, encoded, and parsed. Next, the preprocessing methods are used for the parsed texts, suitable actions

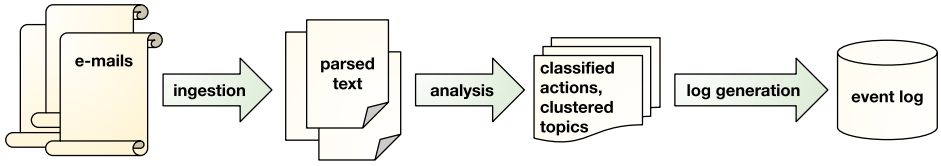


Figure 1. Algorithm scheme for creating an event log from e-mail data. Source: own work.

are classified, and topics are grouped. In the following phase, an event log is generated based on actions to activities matching, event construction, as well as case and process clustering. Based on such an event log, it is possible to use any process mining algorithm to discover the processes.

Although there are some e-mail bases with annotated data, based on which it is possible to use supervised learning, as presented in [4], in our approach, we explore unsupervised or semi-supervised machine learning methods for the same purpose, such as clustering, extractive and abstractive summarization.

3. Process Mining for Explainability

Visual explanations are a well-established and popular technique for explaining black-box machine learning models, as they are easily tractable and interpretable by human users. The most popular methods for generating model-agnostic explanations include LIME and SHAP explanations [5]. It is also possible to use more visual techniques, such as GBEx [6].

However, these techniques generate local explanations that allow the understanding of the behavior of the model for a single instance. Although SHAP explanations can be combined into a global explanation, aggregation is rather not used in practice.

Therefore, in order to grasp a big picture of the model's actions, a user needs to go through multiple explanations of single predictions, which can be considered impractical and time-consuming.

As process models provide a generalization for process instances, it is possible to treat the single explanations as instances and get a global explanation in the form of a process model (see Figure 2) using process mining methods.

In our research, we focus on transforming the explanations into a suitable form for the process mining algorithm, simplifying the model, and providing a suitable visual representation of a process explanation. Although generated explanations have an attractive visual form, the interpretation of the resulting graphs is not always straightforward.

Figure 2. Explanation using directly follows graph for the *Iris* dataset Source: own work.

4. Process Mining for Analyzing Food Industry Data

Business process models help organizations in visualizing and optimizing their processes, to achieve their business goals more effectively. By analyzing data from different food industry systems [7, 8], it is possible to construct event logs by identification of suitable entities from various data sources such as sensors, logistics systems, process automation systems, inventory management systems, as well as quality control databases (see Figure 3). These data sources are also distributed among different stakeholders or participants in the process, such as farmers, distributors, producers, dairy producers, end distributors, and retailers.

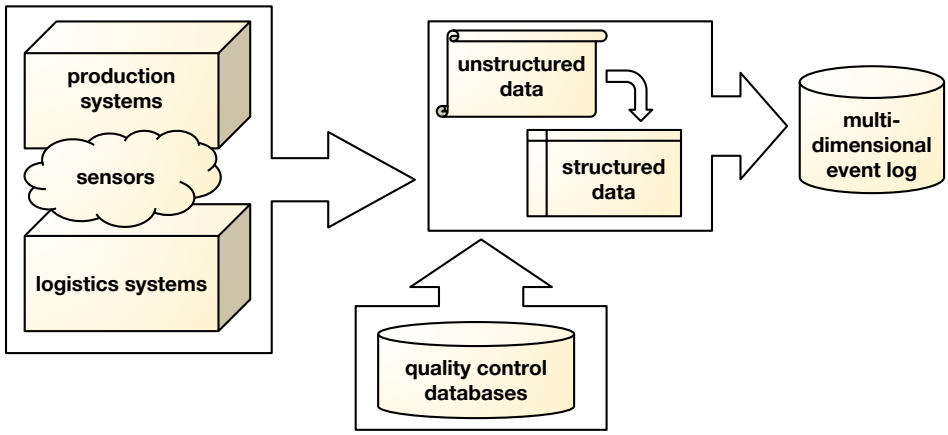


Figure 3. Constructing multi-dimensional event log for food industry processes. Source: own work.

Construction of a part of an event log where the data is textual might take advantage of the method used for unstructured data. In the case of sensor data, it is required to collect such data at a fine enough level of detail, preprocess it and preferably annotate sensor data in order to find key activities or use the sensor

data for the discovery of decision logic in the process. In the case of IT systems supporting the food industry, the logged events might be just transformed into a suitable representation.

Based on such constructed event logs, it is possible to use methods of multi-dimensional process mining that involve analyzing process data across multiple dimensions, such as control flow, time, location, and resources.

Process mining techniques can not only reveal bottlenecks, delays, and other inefficiencies in food industry processes but also provide recommendations for improvement. As the food industry is usually strictly regulated, process mining can also help to identify non-compliance issues and improve quality control measures.

5. Conclusions

In this paper, we give a brief overview of the selected research threads conducted in the KRaKEn Research Group at the AGH University of Science and Technology in Krakow concerning process modeling and mining methods. We show how process mining can be used for unstructured data processing and analysis, as well as to for providing the explainability method for black box machine learning models. We also outline the possibility of using multi-dimensional process mining for analyzing food industry data.

References

[1] van der Aalst W., *Process Mining: Data Science in Action*, Springer Berlin Heidelberg, 2016, doi: 10.1007/978-3-662-49851-4.

[2] van der Aalst W.M., Carmona J., *Process mining handbook*, Springer Nature, 2022, doi: 10.1007/978-3-031-08848-3.

[3] Shing L., Wollaber A., Chikkagoudar S., Yuen J., Alvino P., Chambers A., Allard T., *Extracting workflows from natural language documents: A first step*, [In:] *Business Process Management Workshops: BPM 2018 International Workshops, Sydney, NSW, Australia, September 9-14, 2018, Revised Papers 16*, Springer, pp. 294–300, doi: 10.1007/978-3-030-11641-5_23.

[4] Chambers A.J., Stringfellow A.M., Luo B.B., Underwood S.J., Allard T.G., Johnston I.A., Brockman S., Shing L., Wollaber A., VanDam C., *Automated business process discovery from unstructured natural-language documents*, [In:] *Business Process Management Workshops: BPM 2020 International Workshops, Seville, Spain, September 13–18, 2020, Revised Selected Papers 18*, Springer, pp. 232–243, doi: 10.1007/978-3-030-66498-5_18.

[5] Sepiolo D., Ligęza A., *Towards explainability of tree-based ensemble models. a critical overview*, [In:] *New Advances in Dependability of Networks and Systems: Proceedings of the Seventeenth International Conference on Dependability of Computer Systems DepCoS-RELCOMEX, June 27–July 1, 2022, Wrocław, Poland*, Springer, pp. 287–296, doi: 10.1007/978-3-031-06746-4_28.

[6] Mróz P., Quemy A., Ślażyński M., Kluza K., Jemioło P., *GBEx-towards graph-based explanations*, [In:] *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, pp. 112–117, doi: 10.1109/ICTAI50040.2020.00028.

[7] Gaudio M.T.G., Chakraborty S., Curcio S., *Advanced logistic in the food industry: a system engineering approach for a multi-layered solution*, [In:] *GRICU 2022: Centralità dell' Ingegneria Chimica in un Mondo che cambia, Ischia, (Italy), July 3-6, 2022*.

[8] Gaudio M.T., Chakraborty S., Curcio S., *Agri-food supply-chain traceability: a multi-layered solution*, [In:] *2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, IEEE, pp. 1–5, doi: 10.1109/DASC/PiCom/CBDCom/Cy55231.2022.9928017.

Carbon Footprint Reduction of a Petrochemical Process Supported by ML and Digital Twin modelling

Sławomir Kulikowski[1][0000−0002−2329−8474],
Andrzej Romanowski[2][0000−0001−5241−0405],
Artur Sierszeń[3][0000−0001−8466−4856]

[1]*Atos Poland R&D Sp. z o.o.*
Kraszewskiego 1, 85-240 Bydgoszcz, Poland
slawomir.kulikowski@atos.net
[2]*Lodz University of Technology*
Institute of Applied Computer Science
Stefanowskiego 18/22, 90-537 Łódź, Poland
andrzej.romanowski@p.lodz.pl
[3]*Lodz University of Technology*
Institute of Applied Computer Science
Stefanowskiego 18/22, 90-537 Łódź, Poland
artur.sierszen@p.lodz.pl

DOI:10.34658/9788366741928.23

Abstract. *This article aims to present a concept of an Artificial Intelligence application in the form of pre-trained Machine Learning modules to reduce the carbon footprint of a chemical recycling process.*

Chemical recycling of plastic is energy-consuming as it requires relatively high temperatures and calibration cycles based on a constantly changing structure of raw materials. Due to that fact, complex process parameters must be tuned to allow the production of the required fraction of gasoline. In general, the designed IoT system enables a massive collection of technology and environmental data and the processing of parameters to feed the Digital Twin of a petrochemical plant.

The scientific part of the project consists of Digital Twin modelling, experiments, simulations, and training of machine learning modules to predict the optimal set of production line parameters based on the specific structure of raw materials to reduce the number of calibrations and lower energy consumption indirectly which will lead to carbon footprint reduction. There is an here is an estimate that that deployed solution will allow reduction of energy consumption on a monthly level of 10-15% which could generate direct savings on a cost of energy but also savings in a field of carbon emission and

related credits. The project also includes the evaluation of predictions supported by machine learning modules, training techniques and comparison to expert settings to assess the quality of the application.
Keywords: *Internet of Things, Digital Twin, Machine Learning, Carbon Footprint*

1. Introduction

As an industrial environment, the petrochemical plan consists of many specialised systems which have never been integrated. Moreover, some areas are still managed by simple tools such as spreadsheets and document templates, which limits automation and seamless dataflow related to complex production processes. However, the proposed Smart Refinery solution integrates the most important ones.

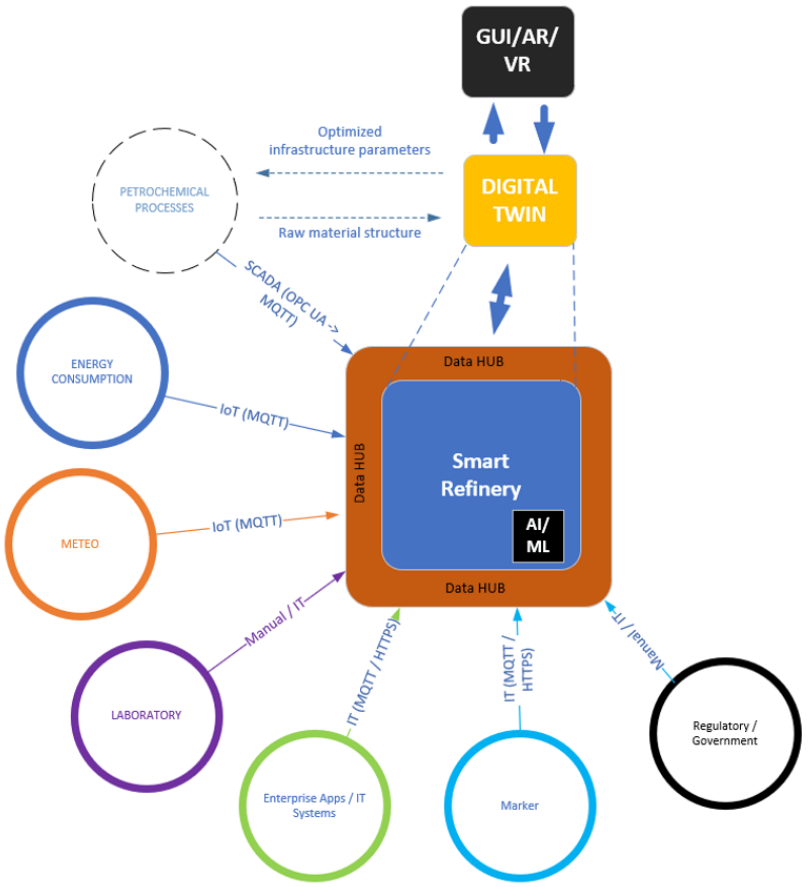


Figure 1. High-Level overview of data sources and integration considerations. Source: own work.

It enables data exchange, processing, storing and visualization to enable insights, improvements, optimization, and carbon footprint reduction through controlled and reduced energy consumption supported by Machine Learning modules. Increased monitoring, better control and optimization over energy consumption allow reduction on a monthly level of 10-15%. The following diagram presents a high-level view of data sources integrated into a Smart Refinery system which is in a designprototyping phase. Additionally, it indicates the simulation capabilities of a Digital Twin based on exact laboratory and hypothetical data. (Figure 1).

2. Data processing and ML application

Machine Learning and Digital Twins [1, 2] require a constant data feed through designed interfaces connected to a data hub and stored in a time-series database as metrics and events (Figure 2).

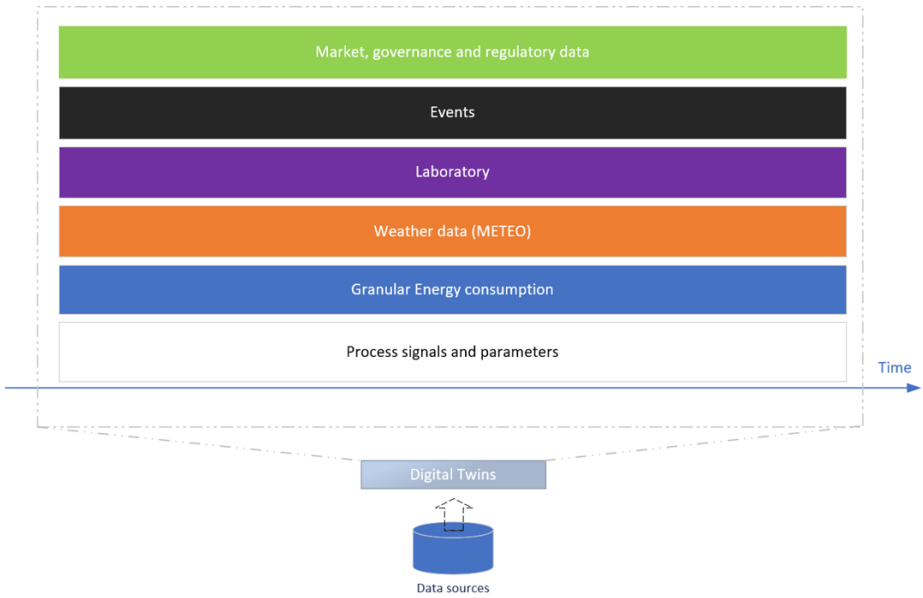


Figure 2. Multidimensional context of enterprise data in the form of stacked timelines. Source: own work.

Collected and prepared data is used to create a multidimensional analysis supported by pre-trained Machine Learning modules to identify patterns, trends, anomalies, and cross-dimensional contrast to find undiscovered dependencies and areas for optimisation [3, 4]. Each layer depicted below consists of a dataset enriched by time signatures which enables contextual research on the recycling process, its parameters and phases, the structure of raw materials, environmental data, and

events. In most cases, applied Machine Learning features will be based on the following algorithms:

- Linear and Logistic Regressions,
- Decision Tree,
- Gradient Boosting and AdaBoosting.

Yet another area of ML application is prediction of process parameters based on a current and forecasted weather data. In this specific case a petrochemical plant is large on-air installation of pipes, distillation towers and tanks which are sensitive on environmental conditions (Figure 3).

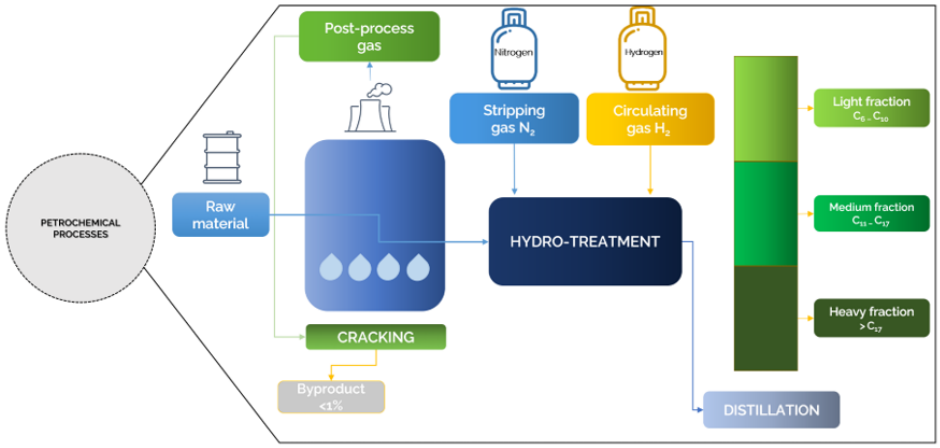


Figure 3. High level overview of petrochemical process. Source: own work.

Outdoor temperature, pressure and humidity have a direct impact on a process parameters such as flows, installation pressure and temperatures and overall processing time and differs on specific season of a year. Considered ML modules will adjust mentioned process parameters based on weather forecast collected from a weather API and current data from weather station. Historical process parameters and whether data will be used to create a dataset to train and evaluate a ML module in the cloud [5]. Once trained and evaluated specific ML module will be deployed on Edge in a form of web service as presented below (Figure 4).

3. Conclusions

The latest, the 4th Industrial Revolution, requires an efficient ability to convert data into information which is the only way to build market advantage, competitiveness, and sustainability. According to market research conducted by widely recognized consulting companies [6], and details described in article [7]:

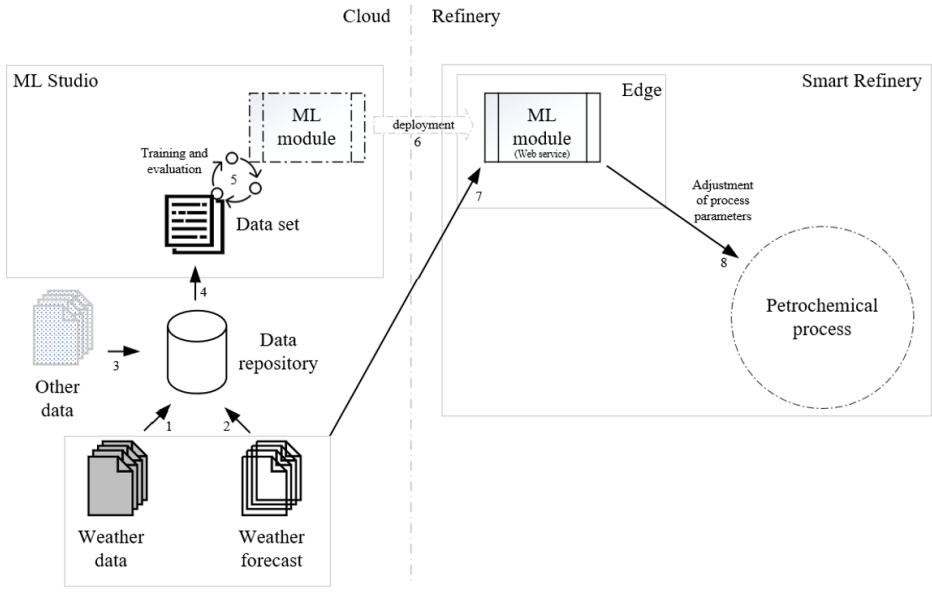


Figure 4. ML application example including training and deployment – weather conditions compensation ML module. Source: own work.

- almost every second organisation does not know how to use the collected data to manage the company better properly,
- on the other hand, 88% of corporate data is not used to manage the organisation better.

Above results lead to a statement that the amount of collected data makes human perception inefficient at that scale and complexity. Application of AI in a form of specialized Machine Learning modules is crucial to achieving expected goals related to 10-15% reduction of energy consumption in a monthly basis what's a field of studies, researches and evaluations defined on a project level.

Acknowledgment

This research and development are conducted as a part of the 6th edition of the Implementation Doctorate program financed by the Polish Ministry of Education and Science. As part of the research consortium project will be funded by Atos R&D partners and, in addition, by EU and NCBiR grants.

References

[1] Barricelli B.R., Casiraghi E., Fogli D., *A survey on digital twin: Definitions, characteristics, applications, and design implications*, *IEEE Access*, 2019, vol. 7, pp. 167653–167671, doi: 10.1109/ACCESS.2019.2953499.

[2] Mowbray M., Savage T., Wu C., Song Z., Anye Cho B., del Rio-Chanona E., Zhang D., *Machine learning for biochemical engineering: A review*, *Biochemical Engineering Journal*, 2021, doi: 10.1016/j.bej.2021.108054.

[3] Körner C., Waaijer K., *Mastering azure machine learning*, *Packt Publishing*, 2020.

[4] Kopper A., Karkare R., Paffenroth R.C., Apelian D., *Model selection and evaluation for machine learning: Deep learning in materials processing*, *INTEGRATING MATERIALS AND MANUFACTURING INNOVATION*, 2020, vol. 9, no 3, pp. 287–300, doi: 10.1007/s40192-020-00185-1.

[5] O'Donovan P., Gallagher C., Leahy K., O'Sullivan D.T., *A comparison of fog and cloud computing cyber-physical interfaces for industry 4.0 real-time embedded machine learning engineering applications*, *Computers in Industry*, 2019, vol. 110, pp. 12–35, doi: https://doi.org/10.1016/j.compind.2019.04.016.

[6] Linthwaite R., *Leverage data where it originates to drive substantial business benefits*, 2022.

[7] Bikmukhametov T., Jäschke J., *Combining machine learning and process engineering physics towards enhanced accuracy and explainability of data-driven models*, *Computers and Chemical Engineering*, 2020, vol. 138, p. 106834, doi: 10.1016/j.compchemeng.2020.106834.

Digital Twin for Training Set Generation for Unexploded Ordnance Classification

Piotr Ściegienka[1,2][0000−0003−1294−049X],
Marcin Blachnik[3][0000−0003−3336−4962]

[1]*Silesian University of Technology, Doctoral School*
ul. Akademicka 2A, 44-100 Gliwice, Poland
piotr.sciegienka@polsl.pl
[2]*SR Robotics Sp. z o.o.*
ul. Lwowska 38, 40-389 Katowice, Poland
[3]*Silesian University of Technology, Institute of Information Technology*
ul. Krasińskiego 8, 40-019 Katowice, Poland
marcin.blachnik@polsl.pl

DOI:10.34658/9788366741928.24

Abstract. *The use of machine learning methods for unexploded ordnance (UXO) detection and classification is very limited. This limitation derives from the lack of representative and enough large training data. To overcome this issue we propose a construction of a digital twin where UXO and non-UXO objects are represented using mathematical models in a simulated Earth magnetic field. The use of digital twins allows for simulating and collecting a large training set which can be used for training machine learning models. In the conducted research we discuss obtained results and point out several of the detected problems.*
Keywords: *machine learning, artificial intelligence, uxo*

1. Extended abstract

Underground and underwater unexploded ordnance (UXO) detection and classification is a serious problem, and to some extent, it could be supported by autonomous vehicles [1]. In that case, a system equipped with a magnetometers should be able to detect if the Earth's magnetic field distortion is caused by an UXO object. Manually defining such a set of rules is very complex due to the fact that the shape of magnetic field distortion depends on the position of the object. In addition, the magnetic field also changes depending on an object's geographical orientation. To overcome this limitation one of the options is the use of machine learning methods, although it requires a training set. The collection of such data

is very complex and time-consuming. A solution to this problem that we investigate is data generation using digital twin which allows easy object manipulation and data collection. In order to create digital twins we created a 3D numerical model of generic UXO and non-UXO models [2]. The numerical model of UXO object allows setting the caliber and length as well as material by defining it's μ_R. Non-UXO objects were simulated using a sheet of metal as well as a pipe similar to the UXO but to small or too long. All of the objects were discretized using gmesh software and later simulated using finite element methods with getdp software. The quality of the obtained simulation results were verified empirically on real objects. Later the numerical models were randomly rotated and the image of the magnetic field was collected by simulating the movement of the magnetometer over the tested object. The simulations of the autonomous vehicles were also recorded in various geographical directions and on various heights over the tested object. Next, the recorded signals were used to construct a training set. In total, the training set consisted of 40 000 simulated data which represented the movement of 3 independent magnetometers. This dataset was used to train various machine learning models including random forest, kNN and convolutional neural network (using 1D conv. neurons). The obtained results indicate that the highest accuracy can be obtained using CNN. When neglecting the initial object magnetization it allows for obtaining 96% accuracy.
Currently, an open issue is initial magnetization which can be caused by the production process or transportation. Additionally, long-term retention underground causes magnetization due to the influence of the earth's magnetic field. The initial magnetization of objects significantly reduces prediction performance, because the magnetic images start to be very similar between UXO and non-UXO objects.

References

[1] Bełdowski J., et al., *CHEMSEA Findings – Results from the CHEMSEA project (chemical munitions search and assessment)*, 2014.

[2] Maschler B., et al., *Transfer learning as an enabler of the intelligent digital twin*, *Procedia CIRP*, 2021, vol. 100, pp. 127–132, ISSN 2212-8271, doi: 10.1016/j.procir.2021.05.020.

Energy Dissipation Anomalies in Buildings

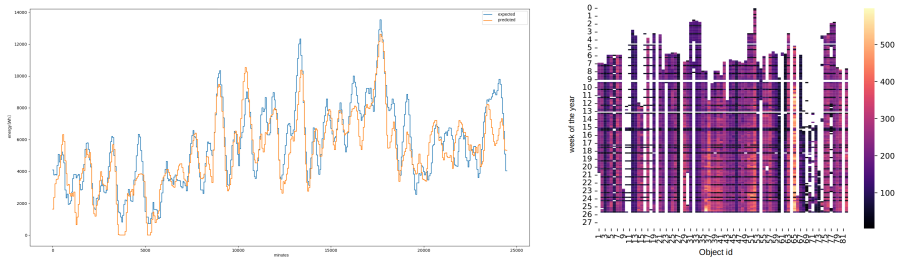
Michał Morawski[1 [0000−0002−8902−1259]], **Arkadiusz Tomczyk**[1 [0000−0001−9840−6209]], **Maciej Idaczyk**[2]

[1]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
michal.morawski@p.lodz.pl, arkadiusz.tomczyk@p.lodz.pl
[2]*IDANET*
Siewna 15/303 94-250 Łódź, Poland

DOI:10.34658/9788366741928.25

Buildings, or even their fragments constitute a complex system, where, taking into account their purpose, we expect specific usage conditions (specific temperature, humidity, air quality, etc.). To check if those conditions are provided, one can directly or indirectly measure the required parameters. Naturally, we cannot assume that those parameters will be in the expected ranges without any additional efforts since both building usage characteristics and external conditions (weather) change in time. Consequently, to keep them having acceptable values, we need to use devices that allow controlling imposed conditions (heating, cooling, air exchange, filtering, humidification, etc.). This results in additional energy usage, which we want to minimize. In this work, we focus on rationalization techniques involving the identification of anomalies in energy consumptions. Anomaly detection is possible if we have some reference data (best, acceptable, or simply typical distribution of energy consumption) in given internal and external conditions. For our experiments, we were collecting such data for buildings of different types (offices, shops, etc.), which was possible thanks to specially designed device and publicly available data from weather stations. To find the typical distribution of these data one can employ historical observations. There are, however, two problems with this approach: to cover the diversity of external conditions we would have to wait for a few years cycles, which usually is not acceptable, and the characteristic of building usage may significantly change over a longer period of time. That is why, in our research we have decided to use two alternative methodologies.

Firstly, assuming that building usage characteristic and weather condition do not change very quickly, we have used predictive models (neural networks) to estimate one day ahead energy consumption for: the boiler, chiller, and air conditioner (AC) used by the air handling unit (AHU). This prediction could be next compared with actual observation leading to detection of anomalies (e.g. open



(a) (b)

Figure 1: Anomaly detection: (a) comparison of actual and predicted energy consumption, (b) comparison of energy usage in different buildings with the same usage characteristic. Source: own work.

window, boiler control failure, etc.). Naturally, this process required careful selection of input variables. Our experiments revealed that for considered office buildings important were: the temperature and humidity of air leaving the building and external air as well as insolation. Sample prediction results are presented in Figure 1a. Secondly, having measurements from different buildings with similar usage characteristic (stores with the same products), we can compare them. However, despite similar use pattern, the particular buildings differ from each other. That is why, to credibly compare different constructions, the dissipated energy should be normalized e.g. by the area of flats, internal building volume, or the number of persons who benefit from the object. Figure 1b illustrates the difference between energy dissipated by objects, normalized by their area, and its evolution over time.

Acknowledgment

This research was financed by the National Center for Research and Development, Poland, grant no. POIR.01.01.01-00-0188/21-00, entitled *Predictive and adaptive METERNET-EnMS system for integrated control of environmental parameters and energy consumption in industrial and business processes (PA)*.

Identification of Damaged AIS Data Based on Clustering and Multi-Label Classification

Marta Szarmach[[0000−0002−3793−6641]],
Ireneusz Czarnowski[0000−0003−0867−3114]

Gdynia Maritime University
Morska 81-87, 81-225 Gdynia, Poland
m.szarmach@we.umg.edu.pl, i.czarnowski@umg.edu.pl

DOI:10.34658/9788366741928.26

Abstract. *Automatic Identification System (AIS) is a telecommunication system that allows ships to communicate with each other by sending information regarding their trajectory: location, speed, course and so on. Due to some technical issues, some parts of the transmitted data might be damaged or incomplete. In this paper, we propose a machine learning based approach for detecting AIS data that requires reconstruction. The general idea of the proposed approach, utilizing clustering and mutli-label classification algorithms, and its performance are discussed.*
Keywords: *AIS data analysis, anomaly detection, multilabel classification*

1. Introduction

Automatic Identification System (AIS) is a telecommunication system that allows ships to communicate by sending information regarding their trajectory: location, speed, course and so on [1]. It consists of two segments: terrestrial, operating (using Very High Frequency band of 161.975 and 162.025 MHz) on a relatively small range of 74 km, and satellite, with much greater range, but struggling with other technical limitations, e.g. packet collision due to the lack of synchronisation between terrestrial cells that a single satellite operates on [2]. Those issues lead to the damage or incompleteness of parts of the transmitted data. Therefore, a need for a reconstruction of damaged or incomplete AIS data occurs. This topic is still relevant from the research point of view – see literature in [3].

In this paper, we propose a machine learning based approach for detecting AIS data that requires further reconstruction. The proposed approach utilizes clustering and classification algorithms to decide whether the value of a given field from AIS message seems incorrect or not. It is a continuation of the previous work [3]. The key idea behind the approach is described in section 2. The computational experiment that examines the performance of this approach is presented in section 3, while section 4 concludes the research.

2. Proposed Approach

In the presented approach of AIS data reconstruction, the overall algorithm can be divided into 3 stages.

Clustering Stage. During the first stage, the collected data from ships from a given area and given period of time are clustered. The clustering algorithm is believed to distinguish individual trajectories, i.e. each cluster should contain messages from one and only one ship (regardless of whether the ship identification field (MMSI) is damaged or not) or create a separate, 1-element cluster for messages that do not resemble the rest (potential outliers).

Anomaly Detection Stage. The next stage is to analyse each cluster obtained earlier and decide which messages and their fields require further correction (i.e. are damaged). Since the damage may originate from not only interception during transmission, but also wrong measurement read (i.e. GPS drift), we cannot only rely on control sum calculated by AIS. The detection can be divided into 2 steps:

- in standalone, 1-element clusters – examined in [3],
- in proper, multi-element clusters – examined in this paper only.

During the latter step, the following fields were analysed: longitude, latitude, speed over ground and course over ground. The main idea was that each message from ith ship (with respect to nth field) from time t_m can be defined with vector $v_{in}^{t_m}$, consisting of: previously (t_{m-1}), currently and next (t_{m+1}) recorded speed and course, the differences between currently and previously/next recorded latitude and longitude, the differences between the consecutive messages timestamps (12 elements in total). Only for course over ground field, instead of speed information, results of trigonomectic function (arctan of differences of latitude and longitude) was added to help the classifier deal with those relations. To find damage in values from those fields, each field needs to have its own classifier that would learn (from correct $v_{in}^{t_m}$s) the relationships between those fields and decide whether they are preserved in each message from proper clusters. Since for each message a different number of labels can be assigned (not only one), indicating its fields that need to be inspected, this is a problem of so-called multi-label classification.

Prediciton Stage. The last stage would be the actual prediciton of the correct values for the fields of messages marked as damaged – yet to be researched.

3. Computational Experiment

3.1. Overview

The Aim of the Research. The computational experiment was supposed to evaluate the performance of the proposed approach of detecting damaged AIS data (corrupted fields) in messages that were inside proper clusters after the clustering stage. To do so, the following quality metrics were taken into consideration: recall (the percentage of detected damaged fields among all truly damaged) and precision (the percentage of truly damaged fields among all classified as damaged).

Data. Original data from AIS system were used in this experiment. The data was gathered in 3 different datasets (split later into training, validation and test sets):

1. 850 messages from 22 ships from Gulf of Gdansk,
2. 19999 messages from 387 ships from Baltic Sea,
3. 19999 messages from 524 ships from Gibraltar.

Environment. The computational experiment was conducted in Visual Studio Code using Python programming language with additional libraries, such as numpy, scikit-learn [4] and xgboost [5].

3.2. Hyperparameters Tuning

The first step, before entering the actual experiment, was to define the hyperparameters for classificators used for analysing the vectors $v_{in}^{t_m}$ described earlier. Two classification algorithms were used: Random Forest [6] and XGBoost [5], both having the following hyperparameters to optimize: *max_depth*, describing the complexity of a single decision tree, and *n_estimators* – the number of trees in a forest.

A dedicated datasets used solely for training field classifiers (one for each analysed field) were build, consisting of vectors $v_{in}^{t_m}$ describing messages from available datasets. Some messages were artificially damaged (having randomly chosen bit from the given field inverted) – those were labelled as 1 (“damaged”), some were not (labelled as 0 – “no damage”). Then, the set was additionally split into training (that the classifiers learn from) and validation (for hyperparameters tuning).

The optimal values for the two hyperparameters (between 2 and 100) were chosen as a trade-off between the best performance (specifically, in terms of the best average F1 score – the harmonic mean of recall and precision) on validation set and lack of clear overfitting to the training set. Different *max_depth* values were only chosen for models analysing the speed over ground field (since the vectors $v_{in}^{t_m}$

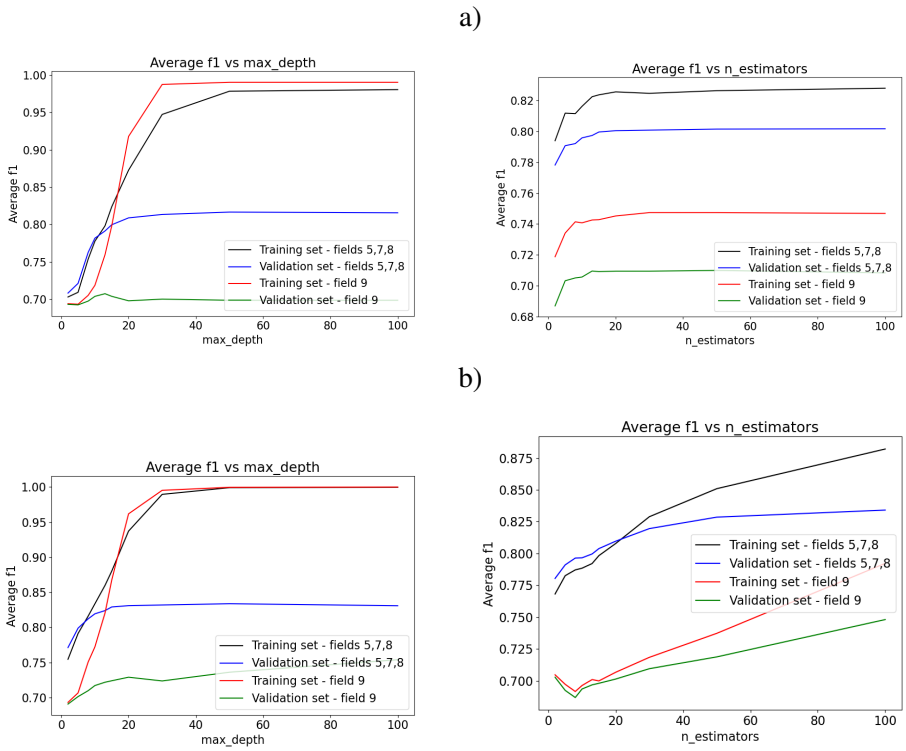


Figure 1. Hyperparameters optimisation: a) Random Forest, b) XGBoost. Source: own work.

differ here). According to Fig. 1, the *max_depth* for Random Forest was set to 12 (sog field) and 15 (other fields), for XGBoost it was set to 5 (sog) and 7 (others); the *n_estimators* for Random Forest was set to 15 and for XGBoost – 20.

3.3. Proposed Approach Performance Results

After establishing the optimal hyperparameter values, the actual performance of the proposed approach was evaluated. The course of the experiment was as follows: for each test dataset, 100 times a different message was randomly chosen and 1 or 2 its bits from analysed dynamic fields were inverted, then the dataset was clustered (using DBSCAN with hyperparameter *epsilon* set to 10 and *minpts* to 1 [3]) and finally, the anomaly detection algorithm was run and its performance (in terms of recall and precision of finding damaged AIS message fields) was examined.

The results are presented in Tab 1. The achieved recall ranges from 48% to 82,5% – we find this result promising, since the damage that was supposed to be detected must have been slight (because the clustering algorithm managed to

Table 1. Detection of damaged fields in AIS messages inside proper clusters results

Dam. bits	Algorithm	Metric	1. dataset	2. dataset	3. dataset
1	Random Forest	Recall	67.00%	73.00%	48.00%
		Precision	25.62%	30.91%	21.26%
	XGBooost	Recall	62.00%	76.00%	55.00%
		Precision	25.10%	32.17%	25.03%
2	Random Forest	Recall	69.00%	82.50%	51.50%
		Precision	47.97%	56.68%	37.39%
	XGBooost	Recall	73.00%	82.50%	52.50%
		Precision	52.73%	57.40%	38.08%

cluster the message together with other messages from the given vessel). The precision results were lower than the recall, the model seems to mark more fields as damaged as it should, but still it is more important to identify the incorrect ones (thus, not to miss parts of data that need to be reconstructed).

4. Conclusions

The proposed approach of finding damaged parts of AIS data that require correction, using clustering and multi-label classification, manages to correctly find most of the artifically damaged AIS message fields. However, the precision of the model could be improved. Future work will focus on determinig the optimal observation time from the perspective of the anomaly detecion stage, as well as designing the last reconstruction stage – predicition of the correct values.

Acknowledgment

The authors would like to thank Mr Marcin Waraksa and Prof. Jakub Montewka for sharing the raw data that the authors used in the experiment.

References

[1] *Technical characteristics for an automatic identification system using time division multiple access in the vhf maritime mobile frequency band, recommendation itu-r m.1371-5*, 2014, vol. 02, (access: 11-07-2023). https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.1371-5-201402-I!!PDF-E.pdf

[2] Swetha G., Natarajan S., Hemavathy K., *Overcome message collisions in satellite automatic id systems*, 2018, (access: 11-07-2023). https://www.mwrf.com/technologies/systems/article/21849164/overcome-message-collisions-in-satellite-automatic-id-systems

[3] Szarmach M., Czarnowski I., *Multi-label classification for ais data anomaly detection using wavelet transform*, *IEEE Access*, 2022, vol. 10, pp. 109119–109131, doi: 10.1109/ACCESS.2022.3214217.

[4] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay E., *Scikit-learn: Machine learning in Python*, *Journal of Machine Learning Research*, 2011, vol. 12, pp. 2825–2830.

[5] Chen T., Guestrin C., *XGBoost: A scalable tree boosting system*, [In:] *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, ACM, New York, NY, USA, ISBN 978-1-4503-4232-2, pp. 785–794, doi: 10.1145/2939672.2939785.

[6] Ho T.K., *The random subspace method for constructing decision forests*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, vol. 20, no 8, pp. 832–844, doi: 10.1109/34.709601.

Integrating Anomaly Detection for Enhanced Data Protection in Cloud-Based Applications

Konrad Czerkas[2], **Michał Drozd**[2[0009−0002−4577−3050]], **Agnieszka Duraj**[1], **Krzysztof Lichy**[1], **Piotr Lipiński**[1], **Michał Morawski**[1], **Piotr Napieralski**[1], **Dariusz Puchała**[1], **Marcin Kwapisz**[1], **Adrian Warcholiński**[1], **Michał Karbowańczyk**[1], **Piotr Wosiak**[1]

[1]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
piotr.napieralski@p.lodz.pl
[2]*LTC Sp. z o.o.*
Institute of Computer Science
Narutowicza 2, 98-300 Wieluń, Poland
kczerkas@finn.pl

DOI:10.34658/9788366741928.27

Abstract. *In this research, anomaly detection techniques and artificial neural networks were employed to address the issue of attacks on cluster computing systems. The study investigated the detection of Distributed Denial of Service (DDoS) and Partition attacks by monitoring metrics such as network latency, data transfer rate, and number of connections. Additionally, outlier detection algorithms, namely Local Outlier Factor (LOF) and COF, as well as ARIMA and SHESD models were tested for anomaly detection. Two types of neural network architectures, multi-layer perceptron (MLP) and recursive LSTM networks, were used to detect attacks by classifying events as "attack" or "no attack". The study underscores the importance of implementing proactive security measures to protect cluster computing systems from cyber threats.*
Keywords: *computer games, artificial intelligence*

1. Introduction

Cluster computing is a popular way to increase the computing power of systems. However, with this increase in power comes an increase in vulnerability to attacks. Two of the most common attacks on clusters are Distributed Denial of Service (DDoS) and Partition attacks [1, 2]. In order to detect these attacks, a series of experiments were conducted, where metrics were collected during normal

operation and then during anomalous behavior caused by external factors. The experiments showed that a solution was developed to detect DDoS and Partition attacks on clusters. The solution involved monitoring several key metrics, such as the number of connections, data transfer rate, and network latency. When these metrics exceeded a certain threshold, an alarm was triggered to alert the system administrator. The solution was able to accurately detect anomalies and distinguish between normal and anomalous behavior [3].

2. Anomaly Detection Techniques for Cluster Security

According to the commonly accepted definition of an outlier, it is a result of an observation that significantly differs from other results in a group. This difference suggests that this result is due to a different mechanism of generation. In this study, attacks were treated as anomalies or exceptions. The analysis of the stream began with determining the inner and outer lower and upper fences according to Tukey's method [4]. The extreme value exceeding the distribution limits was adopted as the definition of an outlier.

Two most popular algorithms for detecting outliers were also examined: the Local Outlier Factor[5](LOF) algorithm and COF. They define outliers based on the calculated coefficient. LOF creates so-called uniqueness ranks, while COF determines the isolation coefficient. It is assumed that an object (point) for which the LOF coefficient is approximately 1, e.g., $LOF \in (0.8; 1.2)$ belongs to the designated group of objects (belongs to the cluster). Objects for which the LOF coefficient changes abruptly relative to their local neighbors (upward and downward jumps can be observed) are called local objects (points) – local detected outliers. The COF isolation coefficient, on the other hand, determines how strongly a given object is isolated from the entire set. Two cases are considered, namely: the smaller the outlier index WW, the more objects the COF algorithm may indicate as outliers. If the outlier index equals 1, outliers will be those objects for which the isolation index is > 1. Two models directly related to anomaly detection in time series were also taken into account, namely the ARIMA (AutoRegressive Integrated Moving Average) model [6] and the Seasonal Hybrid Extreme Studentized Deviation (SHESD) model. SHESD detects one or more outliers in one-dimensional data streams that are approximately normally distributed. A necessary condition for detecting an outlier is to determine the upper limit of the predicted value of the given deviation. Unlike simulation studies, the experiments were conducted on a real cluster, which allowed for a reliable determination of the impact of request types on the collected load statistics. The cluster was built based on the OpenShift and Kubernetes systems, using an IBM rack server consisting of nodes. Instead, different types of loads were generated to observe the behavior of the cluster. The performance was also tested for the impact of the load balancer on response times

and error rates. Loads were generated from two computers, one of which simultaneously ran the DNS server and load balancer required for the cluster to function properly. Each computer had 6 CPU cores and 16 GB of RAM. The developed system detects anomalies to discover insider and outsider attacks from cloud centres. The proposed system has been evaluated using different datasets and its performance has been compared with several anomaly detection methods to determine its effectiveness when deployed on cloud data servers. The aim of the experimental research carried out was to test the effectiveness of neural networks of two selected types in detecting exceptions based on real data representing network traffic. Three parameters were selected as input data, representing the number of bytes in the system input, the average processor load per computational unit and the average response time per request, respectively. In addition, in order to determine the norms, data from consecutive days of individual months, which consequently form a yearly overview, were obtained and appropriately prepared, allowing good determination of daily, monthly or annual trends. Thus, properly analysed and processed data will constitute benchmark data which, in the best possible statistical sense, describe the behaviour of the system under normal operation. They thus constitute the 'ground-truth' for intelligent exception detection methods, i.e. situations that deviate from the norm, i.e. the normal observed operation of the system.

3. Artificial neural networks

Artificial neural networks are successfully applied to the task of attacks and anomalous behavior detection in computer systems and networks[7, 8]. For this reason, as the part of our research, we also considered artificial neural networks, focusing on the following two network architectures: multi-layer perceptron (MLP) and recursive LSTM networks. The task of attack detection was carried out in two schemes and both network architectures were used in both cases. In the first case, it was a problem of classifying events as "attack" or "no attack". The vectors of parameters (previously discussed) were fed to the input of a network, with the adjustable history window of size $L \in \{1, 2, 4, \ldots, 64\}$, and at the output of the network, we demanded a binary response indicating moments in time when an attack or an exceptional situation took place, which was further compared to the reference signal. This allowed to train neural networks in a supervised learning scheme using gradient techniques. In the second case, the task of attack detection was formulated as the prediction of parameters at a given time based on the history window ($L \in \{1, 2, 4, \ldots, 64\}$). Then depending on the accuracy of prediction and the specified threshold, it was decided whether the attack took place or not. Also here, both types of network architectures were tested. Based on the obtained results, we can conclude that in both considered cases, MLP and LSTM networks

enabled effective attack detection (understood as the distinct indication of a period of time when the attack took place). It should be noted that the predictive approach based on LSTM network allowed to detect attacks in the case of noisy data, when the signal-to-noise ratio was only 3dB.

4. Method

The proposed method for recognizing the current system state involves using a trained neural network model and a Python script that applies median filtering to the network's output. The input data is loaded from a CSV file containing the last 128 input data samples, which are then normalized based on coefficients and mean values obtained during the training process. The data is then prepared in packages of K samples, where K is a configurable parameter set in the INI file.

The trained model is loaded and applied to the prepared data, and the resulting outputs are compared with expected output vectors, which serve as reference points for decision making. The Euclidean distance between the actual output and each reference point is calculated, and the smallest distance indicates the system's current state. A median filter is then applied to the output to smooth out any dynamic changes, and a decision is made based on the final output.

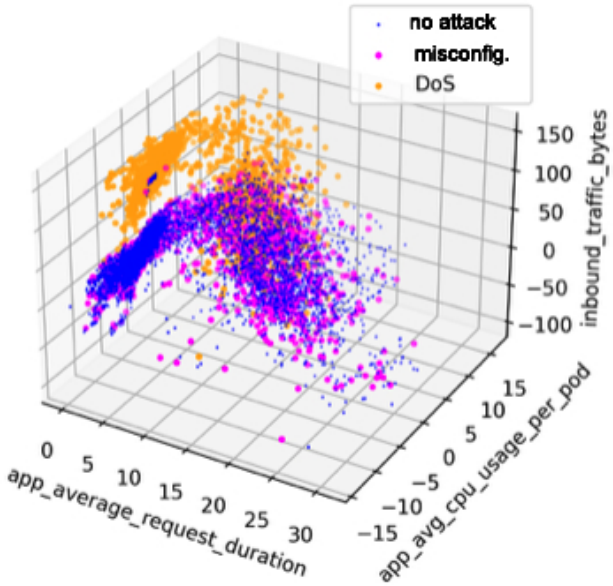
The following equation can summarize the method:

$$wdata = \text{fmedian}(\arg\min_{j \in 1,\ldots,4} \|z - vv_j\|) \tag{1}$$

where *wdata* is the final output, z is the output of the trained neural network model, *vv* is the expected output vector, and $fmedian$ is the median filter function.

Analysis of the results allows important conclusions to be drawn. Certainly, the network tends to activate in discrete moments where the attack has not taken place. Likewise, discrete moments indicating the absence of an attack may occur when an attack has occurred. Adding a median filter helps to smooth out such dynamic changes resulting in results closer to the expected ones. We can base the above intuition on the assumption that the attack lasts for a longer period. The results obtained from the neural network-based intrusion detection system showed promising performance in accurately detecting and classifying network attacks. The use of a median filter to reduce the impact of dynamic changes in the input data resulted in more stable and accurate outputs. The confusion matrix showed that the model achieved high accuracy in classifying different types of attacks, with only a small percentage of misclassifications. Overall, the developed system has the potential to provide reliable and effective network security for various types of organizations.

Values on the main diagonal indicate correct recognition. Values off the main diagonal represent misrecognition results. The sum of the elements lying outside the main diagonal is 7.9%. This is a good result.



	No Attack	DoS	Misconfiguration	Both Attacks
No Attack (predicted)	61.05%	1.3%	1.8%	0.2%
DoS (predicted)	0.9%	14.7%	0.2%	0.5%
Misconfiguration (predicted)	1.9%	0.2%	13.0%	0.2%
Both Attacks (predicted)	0.2%	0.4%	0.1%	2.8%

Figure 1. Visualisation of input data as a point cloud and Results as a confusion matrix. Source: own work.

It should be noted that the task of classifying the input data into the four classes considered is not a trivial issue. The complexity of the problem is best demonstrated by visualising the input data as a point cloud in three-dimensional space.

The analysis of the results suggests that the network tends to activate in discrete moments where no attack occurred, and conversely, may have moments indicating the absence of an attack during actual attacks. Using a median filter helps to alleviate these dynamic changes, leading to more accurate results. This can be attributed to the assumption that attacks usually last for a longer period of time. In conclusion, the study suggests that adding a median filter can improve the accuracy of IDS systems in detecting attacks.

5. Conclusions

The focus of this research is to emphasize the significance of identifying and mitigating attacks on cluster computing systems. The development of a solution

to detect Distributed Denial of Service (DDoS) and Partition attacks is a proactive approach for system administrators to safeguard their systems and avoid any possible downtime. The metrics utilized in this research can be utilized as a foundation for creating more advanced security solutions for cluster computing. This study underlines the need for continuous improvement and implementation of security measures to protect cluster computing systems from various cyber threats.

Acknowledgment

The research presented in this article was made possible through the financial support of "Środowisko budowy i eksploatacji bezpiecznych aplikacji działających w chmurze w oparciu o inteligentne wykrywanie anomalii w klastrach obliczeniowych oraz techniki kryptograficzne blockchain/DLT (CL) Nr Umowy z NCBR: POIR.01.01.01-00-0263/21-00".

References

[1] Nugraha B., Kulkarni N., Gopikrishnan A., *Detecting adversarial ddos attacks in software- defined networking using deep learning techniques and adversarial training*, [In:] *2021 IEEE International Conference on Cyber Security and Resilience (CSR)*, pp. 448–454, doi: 10.1109/CSR51186.2021.9527967.

[2] Wang B., Zheng Y., Lou W., Hou Y.T., *DDoS attack protection in the era of cloud computing and software-defined networking*, *Computer Networks*, 2015, vol. 81, pp. 308–319, ISSN 1389-1286, doi: https://doi.org/10.1016/j.comnet.2015.02.026.

[3] Yan Q., Yu F.R., Gong Q., Li J., *Software-defined networking (sdn) and distributed denial of service (ddos) attacks in cloud computing environments: A survey, some research issues, and challenges*, *IEEE Communications Surveys & Tutorials*, 2016, vol. 18, no 1, pp. 602–622, doi: 10.1109/COMST.2015.2487361.

[4] Tukey J.W., *Comparing individual means in the analysis of variance*, 1949, vol. 5, pp. 99–114, ISSN 0006-341X (print), 1541-0420 (electronic), doi: https://doi.org/10.2307/3001913.

[5] Breunig M.M., Kriegel H.P., Ng R.T., Sander J., *Lof: Identifying density-based local outliers*, *SIGMOD Rec.*, 2000, vol. 29, no 2, p. 93–104, ISSN 0163-5808.

[6] Alam T., *Predicting revenues and expenditures using artificial neural network and autoregressive integrated moving average*, [In:] *2020 International Conference on Computing and Information Technology (ICCIT-1441)*, pp. 1–4.

[7] de Campos Souza P.V., Guimarães A.J., Rezende T.S., Silva Araujo V.J., Araujo V.S., *Detection of anomalies in large-scale cyberattacks using fuzzy neural networks*, *AI*, 2020, vol. 1, no 1, pp. 92–116, ISSN 2673-2688.

[8] Bongiovanni W., Guelfi A.E., Pontes E., Silva A.A.A., Zhou F., Kofuji S.T., *Viterbi algorithm for detecting ddos attacks*, [In:] *2015 IEEE 40th Conference on Local Computer Networks (LCN)*, pp. 209–212, doi: 10.1109/LCN.2015.7366308.

Learning Non-Differentiable Graphs of Utility AI

Maciej Świechowski[12][0000−0002−8941−3199]

[1]*QED Software and* [2]*QED Games*
Miedziana 3A, 00-814 Warsaw, Poland
maciej.swiechowski@qed.pl

DOI:10.34658/9788366741928.28

Abstract. *Utility AI is an approach to modelling AI players in computer games. Its structure is a graph that computes the utility values of possible actions and chooses the one with the highest value. Currently, such graphs are created by experts manually. This paper presents the first attempts to create them automatically – through learning from data. The problem is similar to training neural networks except that the utility graphs are non-differentiable and contain various types of nodes (more complex than neurons). We present the most promising methods, preliminary experiments and results.*
Keywords: *computer games, utility theory, machine learning*

1. Introduction

Utility is a general term used in game theory to model the motivations and preferences of players. The value of utility is a real number assigned to a player in a state of the game that represents how much the player prefers the outcome of that state. In this paper, however, we are concerned with a specific approach to modelling players in multi-agent systems that goes by the following three names: "Utility AI", "Utility-Based AI" and "Utility System" [1, 2]. It is a mathematical model that provides the structure for AI designers to create complex behaviors. It is not a (declarative) goal-based approach. That means that human creators handcraft the system using its formalism (the development paradigm) and the available components to achieve the desired behavior of AI agents. Such a work-flow makes this technique particularly popular in the video games domain, wherein computer players must fulfill certain design goals and their strength must be carefully balanced, i.e. the AI needs to pose a challenge for humans, but not too hard to overcome.

The structure of a Utility AI is a feed-forward computational graph with the following possible types of nodes:

- **Considerations** – the input nodes that represent state variables, which are always single real values (equivalent to features in machine learning).

- **Utility curves** – transform any input by a curve that represents the utility (e.g. of an action) per given input. They are the focal point of the technique.
- **Aggregators** – take any number of inputs to perform aggregation operations such as *max*, *min*, *multiply*, *sum* and, after that, output the result.
- **Selector** – pairs the inputs with actions and chooses the one to play, usually by performing the *arg-max* operation (i.e. the highest utility wins).

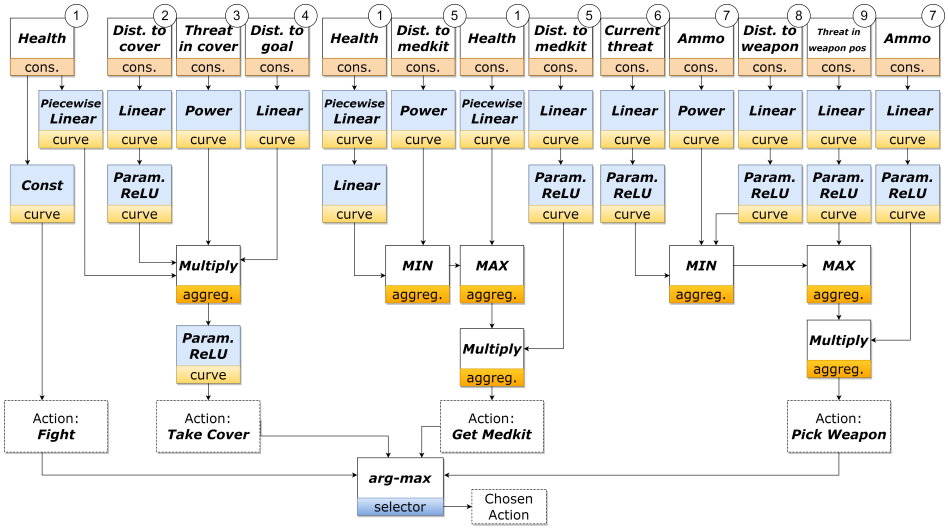


Figure 1. The Utility AI setup in *Grailbots* – the experimental environment. The topmost row contains considerations (1-9). Some of them are used more than once. Source: own work.

A sample graph is illustrated in Figure 1. However, as mentioned before, such graphs are currently created manually by human experts. In this paper, we investigate a novel approach to **training the Utility AI representation in an automatic fashion** from data. Then, such a representation can be further tweaked by human experts to their preferences. Our aim is to adapt machine learning (ML) ideas to Utility AI. Please note that this is different to just applying ML models such as *random forest* and *neural networks* to action selection. While such an approach is feasible, we would lose the advantages of Utility AI – designability, control, explainability, and the fact that game developers are already familiar with it.

2. Experimental Setup and Methodology

In our experiment, we used a game called *Grailbots*, which serves as a demonstration environment (a real use-case) for the commercial AI library called *Grail* [3].

Its Utility AI graph is presented in Figure 1. It uses 9 unique considerations. The number of possible actions is 4. Since this Utility AI model already exists and can be used to generate data, let us call it the *ground truth model (GTM)*.

Our research objective is to validate whether it is possible to train a new Utility AI model that produces the same decisions (actions) as *GTM*. For this purpose, we generated a dataset with 10 columns using *GTM*. The first 9 columns contain sampled values of considerations and the last column is the action chosen (the result of the last *arg-max* node). We believe that such an approach is more general than having exact utility values for each action (the inputs to the *arg-max*) in the training dataset. In practical applications, the input data to learning an empty Utility AI model would either come from human games (wherein only chosen actions would be present) or from time-consuming planning algorithms executed offline.

We divided the domains of each consideration into 6 values in such a way that each action appears as the *ground truth* label in 25% of the rows. The dataset has $6^9 = 10077696$ rows in total, which were randomly split 90-10 for training-testing.

In this early research, we made two simplifying assumptions. Firstly, we decided to only train the utility functions. In Fig. 1, they are marked by the *curve* subscript and colored background. For training, we model each utility function $f(x)$ as a **quadratic interpolating spline** with N control points (however, each function is interpolated individually), where N is a hyperparameter in the experiments. Each i-th control point has a weight w_i that controls its Y coordinate as follows: $y_i = y_{min} \cdot w_i + y_{max} \cdot (1 - w_i)$. The weights are learnable parameters similarly to weights in neural networks. Our chosen functions are differentiable both with respect to w (for training) and x (for applying the chain rule). Secondly, we use a fixed topology, which is the same as already present in *Grailbots*, rather than starting from an empty graph. The latter approach will be investigated in future work after we make sure that the simpler research objective can be achieved.

3. Methods and Results

Due to strict page limits, we only provide a high-level overview of each approach. For more details, we plan to release an extended version of this article.

Gumbel Softmax Approach – this approach uses the *Gumbel-Max trick* [4] to efficiently draw samples from categorical distribution in the forward pass. Next, the sampling idea is combined with a continuous differentiable function of *Softmax* instead of *arg-max*. This method has been successfully used to train neural networks when the training signal is non-differentiable [5]. However, in our problem, there are more non-differentiable nodes than just the last *arg-max* that signal goes through (the regular *max* and *min*). We use the *Gumbel Softmax* for the last node (that performs the *arg-max* – choosing the action with the highest utility) and

regular *Softmax* and *Softmin* for the internal *max* and *min* nodes, respectively. Optimized for the *temperature parameter* $\in [0.25, 0.5, 0.75, 0.9, 0.95]$.

Naive Stochastic Gradient Descent (SGD) – this approach is a regular *back-propagation* using *SGD* [6] with two tricks that make it possible to apply with non-differentiable nodes. First, we convert the training labels to one-hot encoding. For example, when the first action is the label, we convert the training signal to $[1, 0, 0, 0]$. Secondly, when encountered a *max* or *min* internal node, we only propagate the gradient along the one connection that determined this *max* or *min*, respectively, in the forward pass. We tested two types of losses in the experiments.

REINFORCE – in this approach we use the *REINFORCE* algorithm [7], which can empirically estimate gradient of any function through sampling (in a similar fashion to Monte Carlo methods). It takes "baby steps" into the direction of the steepest increase of the expected reward function. For the reward, we use root-mean-square error between the current vector of actions' utilities and one-hot encoding of *ground truth* as in the *SGD* method.

Hill Climbing – for each learnable parameter, i.e. weights w_i of each control point of each curve, we generate possible steps: $w_i \pm \eta$. The algorithm chooses the step that locally maximizes the reward function (which is the same as in *REINFORCE*).

We have chosen a safe number of 200 training epochs for the methods, i.e. each sample in the training dataset was used 200 times. Each method was optimized for each hyperparameter considering 5 values expertly chosen by us. The learning rates: $\eta \in [0.01, 0.02, 0.05, 0.1, 0.2]$ and *batch sizes* $\in [1, 10, 100, 1000, 10000]$. The results are presented in Table 1. For each method, the result shown was achieved by the best performing combination of hyperparameters. We can observe that **Gumbel Softmax** and **REINFORCE** achieve the best results and that there is no improvement of their performance past 50 control points per utility curve.

Table 1. **Accuracy** achieved by each implemented method on the testing dataset from the *Grailbots* game. Each result is a mean value from 100 repeats.

Approach \ Points per curve	**10**	**50**	**100**	**200**	StdDev
Gumbel Softmax	0.25	0.87	0.87	0.87	< 0.05
SGD with Hinge Loss	0.25	0.5	0.75	0.75	< 0.05
SGD with CrossEntropy Loss	0.36	0.53	0.75	0.75	< 0.05
REINFORCE	0.5	0.87	0.87	0.87	< 0.05
Hill Climbing	0.33	0.5	0.625	0.625	< 0.05

4. Discussion and Conclusions

This paper presented the first attempt to train the Utility AI model from data. Despite a relatively simple game and a fixed topology, which simplifies the problem to learning only the shapes of the utility curves, **none of the approaches were able to achieve the perfect accuracy score** of 1.0 and recreate the *ground truth* model. This might be caused by various reasons such as (1) convergence to a local optimum, which none of the methods can escape from; (2) approximation error introduced by using quadratic splines as utility curves; (3) limitations of the training methodology (e.g. how *ground truths* are used); (4) fixed topology – it is likely that a learning system should have some degree of flexibility in choosing the topology rather than using the same one that was used to generate the training data; (5) too many *max* and *min* nodes. A detailed investigation of the reasons will follow in future work. To tackle these (1)-(5) problems, we also plan to employ the evolutionary algorithms (EA) approach. EAs are particularly suitable for discrete global optimization when gradient of the optimizable function is infeasible to compute or does not exist. In addition, EA can evolve the graph topology in a similar fashion as neural network topology is constructed from scratch in the NEAT approach [8].

Acknowledgment

This research was co-funded by the Smart Growth Operational Programme 2014-2020, financed by the European Regional Development Fund under the project POIR.01.02.00-00-0150/16 granted to Silver Bullet Labs Sp. z o.o., operated by The National Centre for Research and Development (NCBiR) in Poland.

References

[1] Graham D., *An Introduction to Utility Theory*, [In:] S. Rabin (ed.), *Game AI Pro: Collected Wisdom of Game AI Professionals*, A K Peters/CRC Press, 2014, pp. 113–126, doi: 10.1201/b16725.

[2] Świechowski M., Lewiński D., Tyl R., *Combining Utility AI and MCTS Towards Creating Intelligent Agents in Video Games, with the Use Case of Tactical Troops: Anthracite Shift*, [In:] *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1–8, doi: 10.1109/SSCI50451.2021.9660170.

[3] Świechowski M., Ślęzak D., *Grail: A Framework for Adaptive and Believable AI in Video Games*, [In:] *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, IEEE, pp. 762–765.

[4] Gumbel E.J., *Statistical theory of extreme values and some practical applications: a series of lectures*, vol. 33, US Government Printing Office, 1954.

[5] Jang E., Gu S., Poole B., *Categorical Reparameterization with Gumbel-Softmax*, *arXiv preprint arXiv:1611.01144*, 2016.

[6] Amari S.i., *Backpropagation and stochastic gradient descent method*, *Neurocomputing*, 1993, vol. 5, no 4, pp. 185–196, doi: https://doi.org/10.1016/0925-2312(93)90006-O.

[7] Zhang J., Kim J., O'Donoghue B., Boyd S., *Sample efficient reinforcement learning with REINFORCE*, [In:] *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 10887–10895.

[8] Stanley K.O., Clune J., Lehman J., Miikkulainen R., *Designing neural networks through neuroevolution*, *Nature Machine Intelligence*, 2019, vol. 1, no 1, pp. 24–35.

Lessons Learned from a Smart City Project with Citizen Engagement

Sebastian Ernst[1[0000−0001−8983−480X]],
Konrad Zaworski[1[0000−0002−7157−6280]],
Piotr Sokołowski[1[0009−0005−8181−6275]], **Grzegorz Salwa**[2]

[1]*AGH University of Science and Technology*
Department of Applied Computer Science
Al. Mickiewicza 30, 30-059 Kraków, Poland
{ernst,zaworski}@agh.edu.pl, psokolowski@student.agh.edu.pl
[2]*Urząd Miejski w Siechnicach*
ul. Jana Pawła II 12, 55-011 Siechnice, Poland
gsalwa@umsiechnice.pl

DOI:10.34658/9788366741928.29

Abstract. *The paper discusses the experiences gained in a joint research project by AGH and the commune of Siechnice. Two main areas are discussed: collecting data from heterogenous sensor devices as well as input from citizens, and development of analytic procedures in a way which guarantees integration between day-to-day and research operations. The most prominent outcomes of the project include the development of a* living lab *as well as automation of multi-aspect inference, which would normally have to be carried out by a team of experts.*
Keywords: *smart cities, data analysis, citizen engagement*

1. Introduction

Most branches of Artificial Intelligence depend heavily on data. Their availability is key e.g. for most Machine Learning-based approaches, and insufficient data quality or quantity may hamper the expansion of AI to different branches of industry and governance. Hence, collection of high-quality, open data is a high priority. The Internet of Things (IoT) is an emanation of the concept of ubiquitous computing [1], where devices can cooperate and exchange data, is one technology which can bridge this gap. One area where IoT is being applied are urban environments, where sensors are being used to gather data on everything from traffic patterns to air quality.

This paper presents the experiences gained during a research project executed jointly from 2019 to 2022 by Siechnice is a town and municipality in the Lower

Poland Voivodeship, Poland, and AGH University of Science and Technology. Neighbouring with the city of Wrocław, the region capital, the commune of Siechnice is characterised by dynamic growthof population – over 80% over the last 17 years – and, as of 2023, is the youngest Polish municipality with regard to the citizen age.

Efficient data analysis demands that they are well-structured, and that the semantics and domains of the real-world objects (entities) as well as their properties (attributes) are precisely described. This imposed several requirements: data published by IoT systems needed to be normalised, reshaped, rescaled, while retaining any details and nuances. Also, we must not forget that citizens are the source of invaluable information and knowledge. However, it is often expressed in a way that is impossible to directly analyse. One approach may be to apply various techniques, such as sentiment analysis [2] e.g. to data collected in social media; another one is to *encourage* engagement in a way which produces data that is *structured* and *linked* in the first place. These issues have been presented in sec. 2.

Besides the data itself, it is also important to note that real added value comes from ongoing cooperation, not "one-off" analyses: authorities should aim at establishing an evolving relationship with research centres instead of just obtaining a static report. In other words, the actual *process* should be designed so that the result follows the guidelines of a *living lab* [3]. A key requirement to achieve this is appropriate management of the data pipeline, analytic code repository and group work infrastructure, as outlined in sec. 3.

2. Data collection and citizen engagement

The first challenge was related to collecting sensor data from heterogenous IoT devices, including street lighting, city cameras, and weather stations. To ensure the openness and high quality of the data, the model needed to be adaptable to any attribute schemas present in the source IoT system, while maintaining readability both by data retrieval procedures in the application and by analytic procedures, further described in sec. 3. This was achieved e.g. by procedures have been created to produce usable and self-descriptive Pandas structures automatically, based on the schema information stored in the central database.

Another significant practical outcomes of the project is the development of a web application that facilitates communication with residents, designed to encourage users to share their knowledge and opinions in a way which guarantees their analytic usability. To achieve this, the application was equipped with GIS database, including a comprehensive set of objects from OpenStreetMap, to enable semantic references for the data provided by the users, as well as easy generation of interactive maps to increase citizen engagement, as shown in fig. 2.

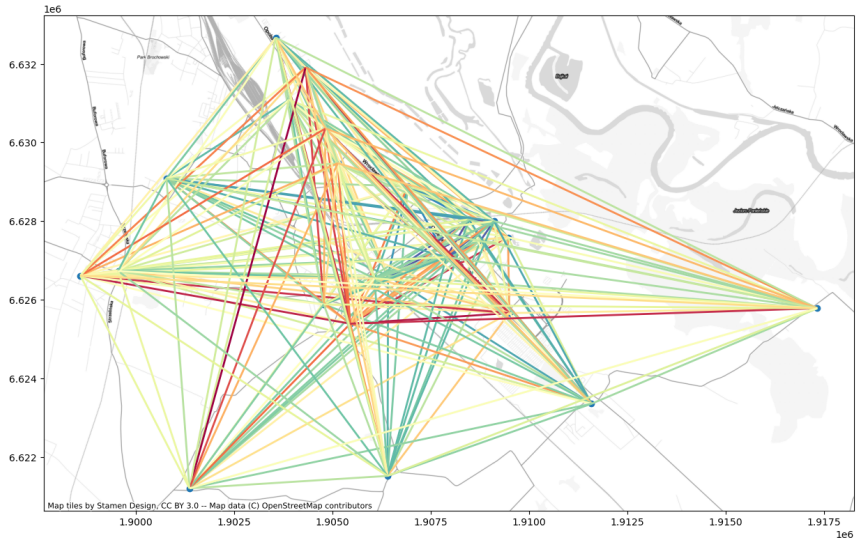


Figure 1: Visualization of correlations between readings from PM sensors in the municipality of Siechnice. Source: own work.

3. Data analysis

Consistent data collection was one of the leading enablers of introducing the concept of a *living lab* – an environment where researchers, businesses and officials can collaborate and obtain added value. In this case, the sensors in Siechnice provide a real-life testing ground for researchers to collect data and develop innovative solutions. In return, local government officials can benefit from recommendations originating from the analyses. However, good practices for managing data and analysing them are crucial for maintaining a successful partnership between the institutions.

The key elements here were maintaining order in the datasets, proper management and versioning of the source code, as well as providing a software environment, based on JupyterHub and JupyterLab, adaptable to the researchers' needs. Good practices and guidelines were developed for these purposes, including aspects such as code conventions, experiment separation, versioning and procedure testing.

However, one of the main contributions of the project involved further development of the Spatially-Triggered Graph Transformation (STGT) methodology, which allows for materialisation of relationships detected in the data [4].

Using this methodology, we observed relatively low correlation level of the sensor located at the intersection of DK 94 (Opolska Street) and Ciepłownicza Street with the other sensors (see fig. 1). Further analyses were carried out by

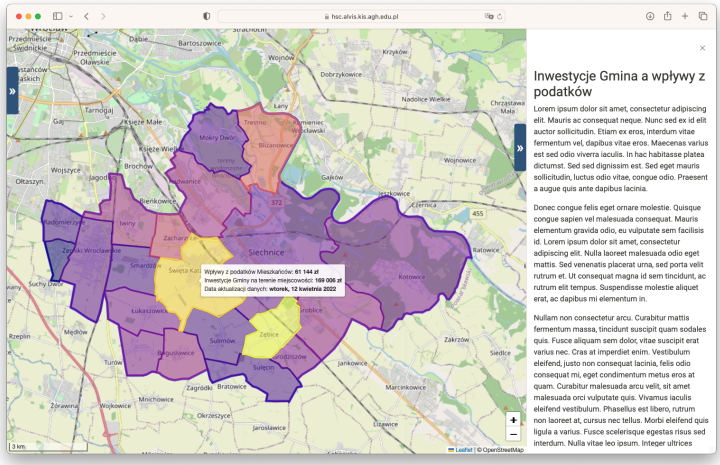


Figure 2: The published article based on data, along with an interactive map. Source: own work.

building a graph representing objects in this area and their relationships. A fragment of the model used in this case is shown in in fig. 3, where 5 classes of graph vertices were identified: *S* (street), *J* (intersection), T (street lights), *A* (air quality sensor), C (camera). The *NEAR* and *SEES* relationships, as well as the connection between lights and intersections, were generated using the aforementioned STGT analysis, which made it possible to detect proximity between objects using GIS tools and materialise it in the graph. Data on the location of street lights were directly fetched from the OSM database. By correlating the objects in the graph, and later by using the traffic intensity data[1], we discovered that traffic on these streets undergoes significant fluctuations during peak working hours. Further analysis showed that the traffic at the intersection is regulated by traffic lights, causing frequent engine idling of vehicles.

This example illustrates the benefits of the proposed holistic methods of data analysis compared to classical statistical methods. The ability to detect relationships both within one sensory system and between few of them (linking air quality data with traffic intensity and organization data) allows for easier and more consistent drawing of conclusions.These observations can be used to support current decision-making as well as to plan for further expansion of sensory networks – for example, determining which locations should be equipped with additional sensors to maximize the utility of investments.

[1]Since camera-based traffic intensity sensors were not present at this precise location, the traffic intensity in the junction under consideration was estimated using *virtual sensors* [5], which were configured using a separate graph-based procedure, not described here due to space limitations.

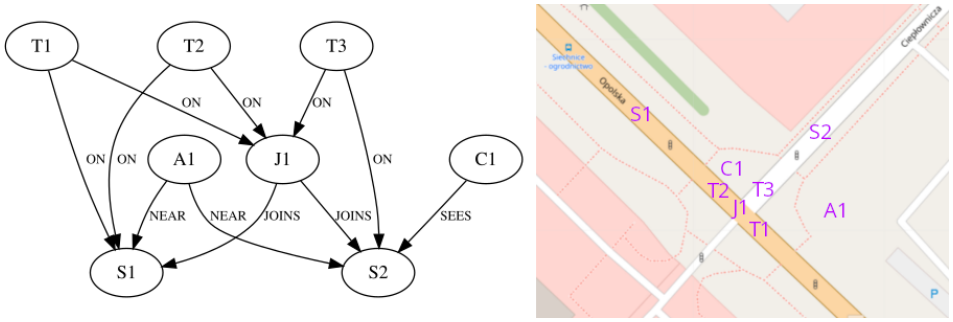


Figure 3: Graph representation of relationships between objects in Siechnice and map view showing their locations. Source: own work.

4. Conclusions

The paper presents the experience and knowledge acquired during a *smart city* project implemented jointly by local authorities in the commune of Siechnice and scientists from AGH University of Science and Technology. Focus was put on two aspects: collection of IoT data in a semantically-correct manner, as well as acquisition of usable knowledge from citizens, and automation of inference processes, which would normally be carried out by experts. Development and implementation of data management guidelines have greatly reduced the overhead usually related to performing analytic research tasks. These experiences have also supported development of a new course in the *Computer Science and Intelligent Systems* study programme at AGH, regarding the issues of data engineering.

Acknowledgments

The authors would like to thank Prof. Leszek Kotulski and Prof. Igor Wojnicki for their valuable work in the areas of data modelling and analysis, which formed the theoretical foundation of the developed solutions, as well as Władysław Sokołowski, an AGH student, for his contributions in the software development process.

References

[1] Weiser M., *The computer for the 21st century*, *Scientific American*, 1991, vol. 265, no 3, pp. 94–104, doi: 10.1038/scientificamerican0991-94.

[2] Yue L., Chen W., Li X., Zuo W., Yin M., *A survey of sentiment analysis in social media*, *Knowledge and Information Systems*, 2018, vol. 60, no 2, pp. 617–663, doi: 10.1007/s10115-018-1236-4.

[3] Cosgrave E., Arbuthnot K., Tryfonas T., *Living labs, innovation districts and information marketplaces: A systems approach for smart cities*, *Procedia Computer Science*, 2013, vol. 16, pp. 668–677, doi: 10.1016/j.procs.2013.01.070.

[4] Ernst S., Kotulski L., *Estimation of road lighting power efficiency using graph-controlled spatial data interpretation*, [In:] *Computational Science – ICCS 2021*, Springer International Publishing, 2021, pp. 585–598, doi: 10.1007/978-3-030-77961-0_47.

[5] Wojnicki I., Kotulski L., *Improving control efficiency of dynamic street lighting by utilizing the dual graph grammar concept*, *Energies*, 2018, vol. 11, no 2, p. 402, doi: 10.3390/en11020402.

Machine Learning for Water Leak Detection and Localization in the WaterPrime Project

Przemysław Głomb[1][0000−0002−0215−4674],
Michał Romaszewski[1][0000−0002−8227−929X],
Michał Cholewa[1][0000−0001−6549−1590],
Wojciech Koral[2][0000−0002−6316−1261],
Andrzej Madej[3], **Maciej Skrabski**[1][0000−0002−3631−9772],
Katarzyna Kołodziej[1]

[1]*Institute of Theoretical and Applied Informatics,*
Polish Academy of Sciences
Bałtycka 5, 44-100 Gliwice, Poland
{pglomb,mromaszewski,mcholewa,mskrabski,kkolodziej}@iitis.pl
[2]*Faculty of Energy And Environmental Engineering*
Silesian University of Technology
S. Konarskiego 18, 44-100 Gliwice, Poland
wkoral@aiut.com
[3]*AIUT Sp. z o.o.*
ul. Wyczółkowskiego 113, 44-109 Gliwice, Poland
andrzej.madej@aiut.com

DOI:10.34658/9788366741928.30

Abstract.*We present an integrated approach for water leak detection and localization developed for the WaterPrime project. Proposed method is based on telemetric monitoring of a District Metered Areas (DMA), using first an application of anomaly detection on sensors' data and then building a 'digital twin' of a DMA state using a combination of hydraulic simulator and machine learning algorithms. This approach leads to reduction of time of leak location estimation from the order of weeks/months to days, and provides a significant reduction in quantity of water lost, as was preliminary verified in two waterworks associated with the project.*
Keywords: *leak detection, leak localization, anomaly detection in time series, machine learning of a digital twin*

Water leaks, or loss of water through leakages and bursts in pipe networks occurring between water treatment and delivery to customers, are a significant economic and environmental issue. Water loss occurs in almost all water networks and starts from 3% to 7% in developed countries, rising to more than 50% in undeveloped ones. Water loss is both substantial economic burden (through lost

cost of water acquisition) and environmental issue (due to aridification of many areas and reduction of available clean water quantity). Main challenges associated with reducing the loss size are: (a) fast detection of anomalies, especially in case of 'increasing' leaks (pipe damage which grows exponentially over time) and (b) precise localization of leaks not visible on the surface.

Leak detection and localization is achieved through a combination of methods: monitoring water networks (e.g. inflow and consumption, pressure) to detect trend changes or anomalous situations; physical inspection using on-site measurements (e.g. geophones); creating a hydraulic model using GIS and monitoring data and analysis of possible leak locations; and many others. While a lot of hardware and software solutions are within reach of the waterworks, their integration and application with the reality of water networks is complex and requires considerable personnel and financial resources. Diversification of network structure, its unknown status (lying underground for many years), inexact documentation, measurement imprecisions or errors, and other issues raise the difficulty of practical leak management.

This study is presented within the framework of the WaterPrime project, a collaboration between AIUT sp. z o.o. and the ITAI PAS, aimed at developing an advanced IA (Intelligence Augmentation) system for water distribution network monitoring and leak detection. This project, co-financed with EU funds through the Polish National Centre for Research and Development, has started in early 2021 and quickly developed into a monitoring system for waterworks of two Polish cities, encompassing several thousands of individual clients across several monitoring zones. Analysis of several months of gathered data has allowed for an in-depth examination of leak patterns and their properties.

Our proposition is based on fast (within hours) detection of anomalies in sensor data, using an ensemble of detectors, including continual learning models, which single out key areas regarding operator's attention. Upon that, another set of machine learning tools is applied to build a hydraulic model – a 'digital twin' of a DMA state to investigate possible leak scenarios and narrow down inspection areas. To further reduce time of on-site inspection, an original solution of changing state of LoRa IoT network is proposed, which uses algorithmic optimization to get a temporary intensification of data collection. Individually, proposed methods have achieved very good results on real-world data benchmarks; together, they have been used within the networks of two waterworks associated with a project, achieving slow but steady reduction in water loss across numerous DMA zones.

Acknowledgment

This work has been partially supported by the Polish National Centre for Research and Development grant POIR.01.01.01-00-1414/20-00, 'Intelligence Augumentation Ecosystem for analysts of water distribution networks'.

Performance Analysis of Machine Learning Platforms Using Cloud Native Technology on Edge Devices

Konrad Cłapa[1][0000−0002−6939−6504],
Krzysztof Grudzień[2][0000−0003−4472−8100],
Artur Sierszeń[2][0000−0001−8466−4856]

[1]*Atos Poland R&D Sp. z o.o. Bydgoszcz, Poland*
konrad.clapa@atos.net
[2]*Lodz University of Technology, Institute of Applied Computer Science, Poland*
krzysztof.grudzien@p.lodz.pl, artur.sierszen@p.lodz.pl

DOI:10.34658/9788366741928.31

Abstract. *This article presents the results of an experiment performed on a machine learning edge computing platform composed of a virtualized environment with a K3s cluster and Kubeflow software. The study aimed to analyze the effectiveness of executing Kubeflow pipelines for simulated parallel executions. A benchmarking environment was developed for the experiment to allow system performance measurements based on parameters, including the number of pipelines and nodes. The results demonstrate the impact of the number of cluster nodes on computational time, revealing insights that could inform future decisions regarding increasing the effectiveness of running machine learning pipelines on edge devices.*
Keywords: *Machine learning, Artificial intelligence, Cloud computing, Edge computing, Internet of Things*

1. Introduction

Machine learning (ML) and artificial intelligence (AI) on edge computing devices are among the most dynamically developing research areas. The importance of the research results is directly related to the development of deep learning methods, the architecture of data processing hardware modules and the increase of large datasets availability. The scientific researches related to applied artificial intelligence for every aspect of daily life and a growing number of sampling and measuring IoT systems generate the possibility of creating many innovative solutions [1]. Those can improve the experience, quality of life, and development progress in many industry verticals, e.g. medical and manufacturing [2].

For the sovereignty-constrained solution, it is essential that the machine learning models can be created and managed regardless of the underlying platform (public cloud, data centre). Fig. 1 is presented a cloud-native solution, based on the containerisation of machine learning pipelines, that brings in an abstraction layer which allows running the machine learning model on hybrid and distributed platforms. To coordinate the execution of machine learning pipelines, the cloud-native platforms require so-called orchestrators that introduce overhead in resource consumption, including memory and processor usage [3, 4].

An increase in the effectiveness of running machine learning pipelines without extending the hardware capabilities of the edge device is critical for serving the machine learning models in real time. Therefore, it is essential to research and design the software elements of the solution that will minimise the overheads and reduce the delays caused by the orchestrator for the cases where edge devices are used to provide sovereignty for data processing and analysis. The state-of-the-art technologies enabling machine learning (ML) on edge systems are K3s, MicroK8s and Kubeflow [5].

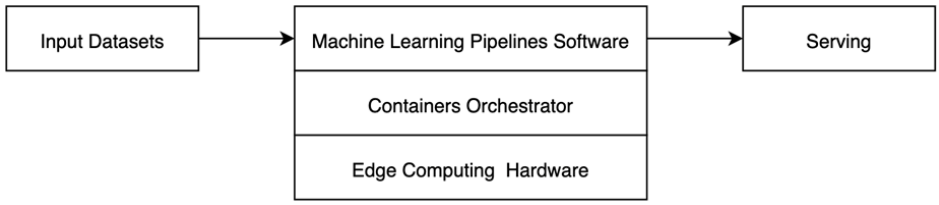


Figure 1: Machine Learning Pipelines on Edge Devices Architecture. Source: own work.

2. Scope of research

In the first phase of the research, the effectiveness of the existing software solutions for managing the lifecycle of the machine learning pipelines is being conducted. The long-term plan consists of several steps. The next phase of the research of the critical elements related to the system's effectiveness will be identified for the case of the platform that cannot use public cloud resources due to data protection restrictions. Finally, we will pinpoint the bottlenecks and perform research to improve the effectiveness of the elements that introduce overheads and delays. The result of the study will include (i) identification of methods of improving the effectiveness of running machine learning pipelines on edge devices allowing protection of confidential data, (ii) a model of architecture based on cloud-native technologies, (iii) a prototype for a particular industry use case (iv) documented proof for improvement of the effectiveness of running machine

learning pipelines on edge devices.

3. Laboratory environment setup

The experimental laboratory setup consisted of two layers – hardware and software. Regarding hardware choices, a decision has been taken to build a lab environment that will characterise itself with maximised stability for testing the system. We virtualised all the Kubernetes cluster nodes and got the static configuration to the nodes based on the Ubuntu Linux distribution, the environment setup was defined in a code, and a rebuild of the virtualised environment could be executed significantly faster. In the next step, based on testing of stability (failure like suspension, unexpected switch off, etc.) K3s was chosen as a Kubernetes distribution.

The last stage of the laboratory environment study resulted in continuing experiments with Kubeflow Pipelines components of Kubeflow. This element is crucial to perform ML operations. Installing additional features of the Kubeflow package can only put overhead on the cluster and will not contribute to the experiment's candidate will perform. On each of the machines, there were K3s nodes installed. All nodes hosted control planes and actual workloads and constituted a fully operational Kubernetes cluster. Finally, the Kubeflow Pipelines were deployed on that cluster. The components of the Kubeflow pipelines were distributed to the node by the Kubernetes Scheduler, and no affinity or anti-affinity rules have been applied.

4. Experiment description

The main research was focused on identifying how the number of nodes in an edge Kubernetes cluster can affect the efficiency of executing the pipelines. We decided to run simple arithmetical computations within containers and measure the pipeline's time to execute. A varying number of pipelines was run parallel.

The experiment started with a single execution, ending with k=30 pipelines executed in parallel. These executions have been conducted on clusters with 1-3 nodes. This allowed analysing both how execution time increases when we increase the number of parallel executions and what is the effect of increasing the nodes number. Each execution was run at least ten times. For the experiment, a benchmarking script was developed. The script allowed us to run multiple requests to the Kubeflow API to create and execute computational pipelines. The time of the execution of a container was measured directly by the Kubeflow software, so any latency related to network connection or data generation could be avoided. The scheme of the experiment environment setup is presented in Figure 2. The hardware used for the experiment has not been used for any other computation

at the time of the experiment to avoid interruptions. The number of nodes was controlled by powering up and down virtual machines on a Qemu hypervisor.

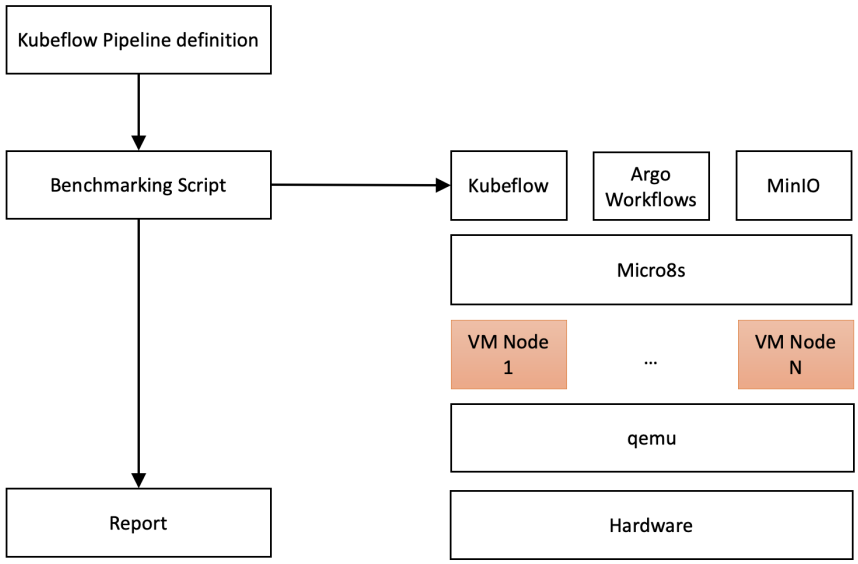


Figure 2: Experiment environment setup. Source: own work.

5. Experiment results

The experiment results of the exaction for N=1-3 nodes are presented in 3. The collected data are visualised on the plots, with the minimum, maximum and average of the executions values. The plotted results indicate that the execution times display almost linear growth with an increase in the number of parallel executions. The behaviour was expected as the load on the system increased. Additionally, we observed that increasing the number of nodes in the system had a very low impact on the execution time, with execution times being almost the same for up to 10 parallel runs. Additionally, one can observe minimal performance increase from 11 parallel runs when we compare 1 node system with 2 and 3 nodes. But it isn't essential. We also see that the difference between a 2 and 3 nodes system is virtually indistinguishable.

6. Conclusions

Based on conducted experiments, it can be concluded that the simple computational tasks in the pipeline are not generating enough load on the nodes to justify the increased effectiveness of the system with an increase in the number of nodes.

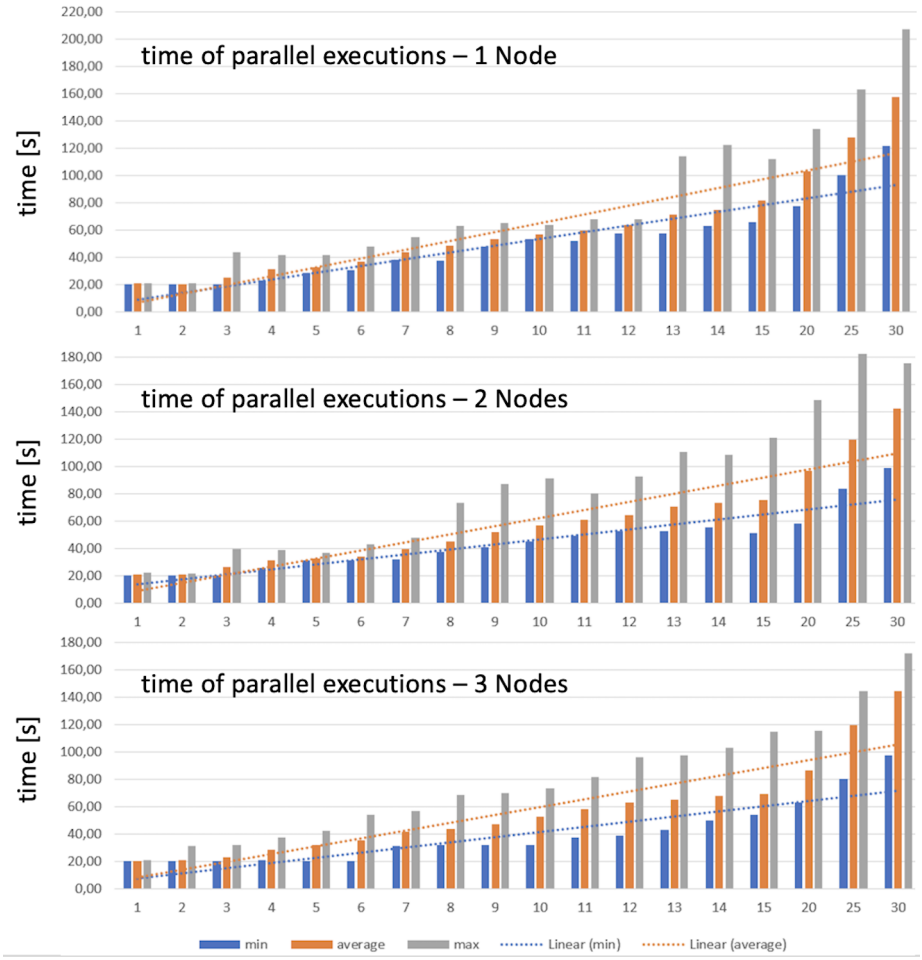


Figure 3: Experiment results for N node systems. Source: own work.

The analyses of results show that increasing the number of nodes in the system had a shallow impact on the execution time. However, such an effect was visible. It is necessary to expand the laboratory environment regarding increased nodes and prepare more demanding calculations. The future study will be focused on testing the system with more computational power-demanding workloads and measuring additional parameters like the pipeline creation time to reflect the system's performance indicators.

Acknowledgment

This research is conducted as a part of the 5th edition of the Implementation Doctorate program financed by the Polish Ministry of Education and Science.

References

[1] Xiong J., Chen H., *Challenges for building a cloud native scalable and trustable multi-tenant aiot platform*, [In:] *Proceedings of the 39th International Conference on Computer-Aided Design*, ICCAD '20, Association for Computing Machinery, New York, NY, USA, 2020, doi: 10.1145/3400302.3415756.

[2] Lv Z., Chen D., Lou R., Wang Q., *Intelligent edge computing based on machine learning for smart city*, *Future Generation Computer Systems*, 2021, vol. 115, pp. 90–99, doi: 10.1016/j.future.2020.08.037.

[3] Rausch T., Rashed A., Dustdar S., *Optimized container scheduling for data-intensive serverless edge computing*, *Future Generation Computer Systems*, 2021, vol. 114, pp. 259–271, doi: 10.1016/j.future.2020.07.017.

[4] Toka L., Dobreff G., Fodor B., Sonkoly B., *Machine learning-based scaling management for kubernetes edge clusters*, *IEEE Trans. on Netw. and Serv. Manag.*, 2021, vol. 18, no 1, p. 958–972, doi: 10.1109/TNSM.2021.3052837.

[5] Fathoni H., Yang C.T., Chang C.H., Huang C.Y., *Performance comparison of lightweight kubernetes in edge devices*, [In:] *Pervasive Systems, Algorithms and Networks*, Springer International Publishing, Cham, 2019, pp. 304–309.

RNN-based Phase Unwrapping for Enabling Vital Parameter Monitoring with FMCW Radars

Piotr Łuczak[1[0000−0002−2530−0283]], **Sławomir Hausman**[2[0000−0000−0000−0000]], **Krzysztof Ślot**[1[0000−0003−1228−0970]]

Lodz University of Technology
[1]*Institute of Applied Computer Science* / [2]*Institute of Electronics*
[1]*Stefanowskiego 18* / [2]*Aleje Politechniki 10, Łódź, Poland*
pluczak@iis.p.lodz.pl

DOI:10.34658/9788366741928.32

Abstract. *Application of radar technology enables remote breathing and heart rate monitoring by analyzing motion waveforms, which are reconstructed from phase signals extracted from radar-delivered data. However, nonlinear deformations introduced by phase recovery procedure make accurate motion reconstruction highly challenging, especially for millimeter-long waves that are commonly generated by state-of-the-art radar devices. In the presented paper we show that a GRU-based neural predictor is capable of correct phase unwrapping under presence of noise (originating e.g. from random subject's movements), enabling vital parameter monitoring in realistic scenarios, which cannot be accomplished using standard approaches.*
Keywords: *regression, GRU, vital parameter estimation, FMCW radar*

1. Introduction

Unobtrusive, privacy-preserving remote monitoring of basic vital signs: breathing and heart rates, becomes an attractive choice for intelligent assistive technologies. This functionality is offered by Frequency-Modulated Continuous-Wave (FMCW) radar devices, which are capable of sensing millimeter and sub-millimeter body displacements caused by respiration and skin surface micro-motion caused by blood pressure variations [1]. The latter phenomenon to be detectable, requires a use of short, millimeter-range radar wavelengths. However, reducing the wavelength poses severe problems for reconstructing micro-displacements accompanying the considered vital processes. As micro-displacements are represented through a phase of radar-produced signals, which gets wrapped if displacement magnitudes exceed a wavelength, information on motion activity becomes distorted, especially in additional presence of inevitable random, tiny body movements. This issue is rarely raised in the literature, where research typically assumes

immobilizing a subject to avoid aggregating of different motion sources. However, practical utility of data analyses, performed in such a case becomes limited.

The following paper introduces a novel method for alleviating the problem of phase reconstruction, which makes reasonable estimation of basic vital parameters feasible. We propose to ground the displacement reconstruction procedure on multiple data streams that are produced by radar devices and employ a recurrent neural network as a means for nonlinear transformation that provides recovery of the actual waveform.

2. Problem formulation

Principles of FMCW radar operation and its application to vital sign monitoring can be found in several publications, e.g. [2]. The radar-monitored space is split into a set of adjacent spatial bins, and objects located in this space are represented by intermediate-frequency (IF) components of frequencies characteristic to these bins. Reconstruction of small object's displacements from FMCW radar data is based on phase information, extracted from the corresponding IF component. As phase is determined via arc-tangent operation on real and imaginary spectral components, it gets wrapped if object displacements exceed radar's signal wavelength. Another level of complexity is introduced by subject's movements and spatial extent of a body surface, which can be seen as a continuous grid of reflection points. Due to movements, these points perpetually change their locations, producing a time-dependent mixture of arbitrary-phase signals that arrive at radar receiving antennas. As tracking of vital activity-related displacements is commonly based on a simple model of a single reflecting point, the real case scenario, involving presence of a reflecting surface combined with movements that are unrelated to vital processes significantly impedes accurate recovery of location-related information from spectra. Therefore, to expect reasonable reconstruction of body surface displacements, on needs to resort to complex nonlinear data processing methods.

The problem with correct phase unwrapping using the standard approach (detection of phase jumps that exceed some adopted threshold level), when respiration activity interferes with subject's movements, has been depicted in Fig.1. The presented scenario assumes that signals are registered using 77GHz FMCW radar (approximately 3.9 mm wavelength) and shows actual and reconstructed phase waveforms for idealized respiratory activity of magnitude 1.5 cm, without (red plots) and with (blue plots) Gaussian motion noise with zero mean and 0.5 mm standard deviation (originating e.g. from chest orientation fluctuations). As it can be seen, the reconstructed phase does not provide the correct basis neither for breathing or heart rate estimation. Although the adopted noise model is highly simplified, it is not far from reality, as similar structures can be observed in unwrapped phase waveforms for actual recordings.

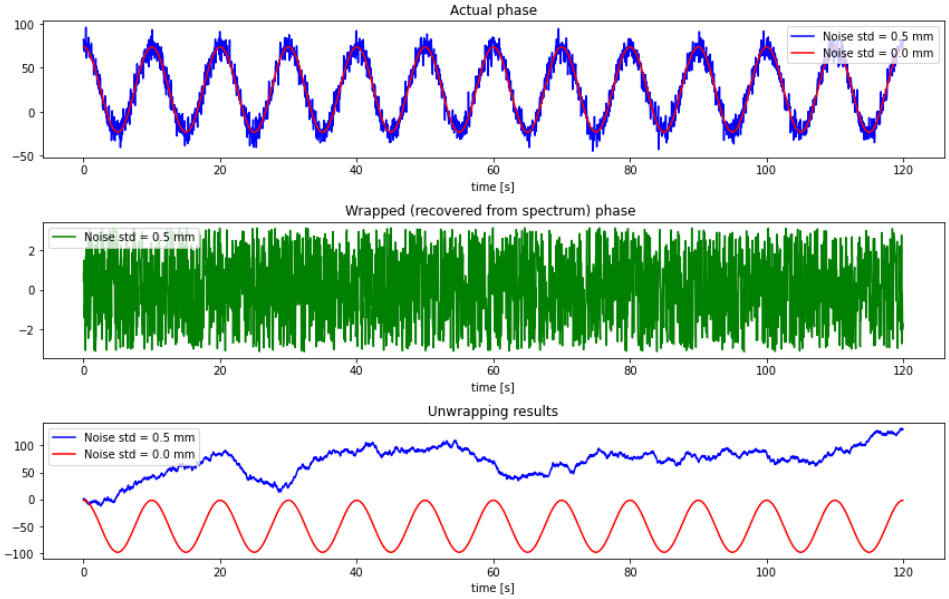


Figure 1. Phase unwrapping problem for respiratory activity monitoring, resulting from random subject's motion. Source: own work.

3. The proposed method

To alleviate the presented difficulties with correct reconstruction of small body surface displacements, we propose to fully exploit capabilities offered by majority of current FMCW radars, which typically contain several receiving (RX) and transmitting (TX) antennas. Since antenna locations are different, one can expect that phase shifts that exist between different RX/TX antenna pairs, might facilitate reaching valid consensus in reconstruction of actual object location. Therefore, the considered problem can be viewed as a prediction of a scalar sequence (phases reconstructed at subsequent time instants) from a sequence of multidimensional observations, where the dimension of input vectors is determined by the product of the number of receiving and transmitting antennas.

Solution to the considered problem requires inversion of highly nonlinear data transformations, so one needs to resort to machine learning methods as the only reasonable approach. We propose to apply Recurrent Neural Network (RNN), made up of Gated Recurrent Units (GRU) [3] to solve the posed prediction problem. Block diagram of the adopted computational architecture for vital-sign analysis, with the proposed GRU-based phase unwrapping module, has been presented in Fig.2.

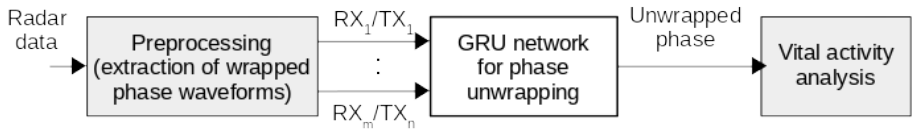


Figure 2. Diagram of the proposed vital activity monitoring algorithm. Source: own work.

4. Experiments

For data collection, the Texas Instruments IWR1443BOOST FMCW radar operating at 76-81 GHz range, has been used. To train the phase-unwrapping neural module, a dataset comprising more than ten hours of recordings of still subjects, positioned at different locations and orientations with respect to the radar, has been created [4]. Radar recordings are accompanied with reference information (targets) that has been collected using a wearable Zephyr BioHarness belt [5] that provides estimates of actual breathing and heart rates.

Of several architectures of the GRU-based neural networks tested during the experiments, the optimum balance between performance and complexity has been offered by the four-layer structure, involving one GRU layer with two units and three dense layers. The total number of parameters of the network is only 682, which makes it easy for hardware implementation. Training of the network was driven by MSE loss with additional L2 weight regularization, for 10 epochs using Adam optimizer. Prediction of output is made based on ten element-long input sequence segments in a sequence-to-sequence conversion scheme.

The network was tested on a disjoint set of recordings and it produces location estimates that serve as a good basis for accurate evaluation of respiratory-related activity. The result for unwrapping of the noisy signal presented in Fig.1 by the proposed algorithm has been shown in Fig.3. As it can be seen, it provides a robust basis for estimation of a fundamental frequency of the target waveform. In addition, average MSE results produced for the test data analyzed using the reference scheme and the proposed method, are shown in Table.1

Table 1. Phase unwrapping accuracy for the proposed and reference methods

	Reference scheme	Proposed scheme
MSE	663.8512	1.0859

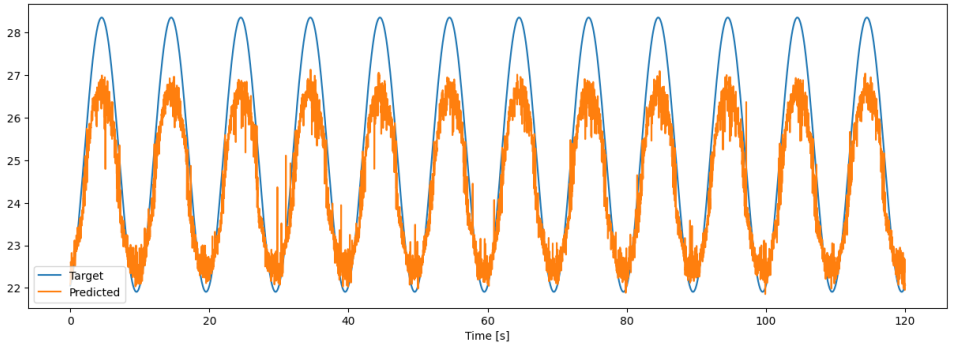


Figure 3. Result of unwrapping the simulated phase with the proposed method. Source: own work.

5. Conclusions

A novel phase-unwrapping approach that enables tracking small displacements of objects located in a space monitored by an FMCW radar, has been presented in the paper. The proposed method utilizes GRU-based recurrent neural network and offers monitoring of respiration-related motor activity in a realistic scenario, where arbitrary sources of subject's micro-movements are allowed. The only limitation of the proposed approach is a requirement that a person subject to monitoring is not 'ideally' (within 1 degree interval) perpendicular to the radar axis. As the proposed algorithm involves tiny network, it can be easily implemented in hardware, enabling delegation of intelligent computing to edge devices.

Acknowledgements

This work has been done as a part of the AIR project, supported by the National Science Centre, Poland under CHIST-ERA III, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no 76897. This work has been completed while the 1st author was the Doctoral Candidate in the Interdisciplinary Doctoral School at the Lodz University of Technology, Poland.

References

[1] Anitori L., de Jong A., Nennie F., *Fmcw radar for life-sign detection*, [In:] *2009 IEEE Radar Conference*, pp. 1–6, doi: 10.1109/RADAR.2009.4976934.

[2] Jardak S., Alouini M.S., Kiuru T., Metso M., Ahmed S., *Compact mmWave FMCW radar: Implementation and performance analysis*, *IEEE Aerospace and Electronic Systems Magazine*, 2019, vol. 34, no 2, pp. 36–44, ISSN 0885-8985, doi: 10.1109/maes.2019.180130.

[3] Cho K., van Merrienboer B., Bahdanau D., Bengio Y., *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches*, *arXiv:1409.1259 [cs, stat]*, 2014.

[4] Ślot K., Łuczak P., Lewandowski P., Jaśkiewicz M., Bielińska-Ślot A., *CHIST-ERA AIR - FMCW radar recordings*, (access: 12-07-2023).
https://chist-era-air.iis.p.lodz.pl/

[5] Hailstone J., Kilding A.E., *Reliability and Validity of the Zephyr™ BioHarness™ to Measure Respiratory Responses to Exercise*, *Measurement in Physical Education and Exercise Science*, 2011, vol. 15, no 4, pp. 293–300, ISSN 1091-367X, doi: 10.1080/1091367X.2011.615671.

Statistical Method for Photovoltaic Power Forecasting Basing on Signal Components Decomposition

Paweł Parczyk[1][0009−0004−3287−9520], **Robert Burduk**[1][0000−0002−3506−6611]

[1]*Wrocław University of Science and Technology*
Department of Systems and Computer Networks
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
pawel.parczyk@pwr.edu.pl, robert.burduk@pwr.edu.pl

DOI:10.34658/9788366741928.33

Abstract. *Since climate and environmental protection have become an important point for society, the industry and business have focused on increasing the share of renewable energy sources in the energy mix. This brought us new challenges. In this paper, we propose a method for photovoltaic power production forecasting. We compared our model with a state-of-the-art Auto Regressive model. We used Mean Absolute Error and Mean Absolute Percentage Error as metrics. Finally, our model turned out to be statistically better than reference model in generating one-hour and two-and-a-half-hour forecasts.*
Keywords: *RES, time series forecasting, PV production forecasting, AR*

1. Introduction

Nowadays, society and businesses aim to decrease pollution and make our lives more environmentally friendly. To achieve this, there is high pressure put on increasing the share of Renewable Energy Sources (RES) such as wind or photovoltaic (PV) energy in the energy mix. This presents us with new challenges. RES, for example, are unstable and difficult to control. The effects of aforementioned instability affect the electricity distribution system due to high variance in localized weather conditions. This results in PV energy generation becoming detached from actual needs of the system at a given time. As such, it is impossible to fully rely on RES energy. Conventional energy sources are still required, to reliably provide energy regardless of weather conditions. Many methods for preventing the negative impact of described phenomena have been proposed. One of the methods is forecasting RES production and charging or discharging batteries. Power production forecasting is a crucial, while best possible battery policy is being established.

Given the above, in this paper, we propose a method for short-term PV production forecasting. We used our model and a reference state-of-the-art Auto Regressive (AR) model for both one-hour and two-and-a-half-hour forecasting on half a year of data. We measured the forecast quality through the usage of Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) metrics.

2. Model description

In this paper, we propose a model which is suited for PV production data. We have observed that this type of data has a few significant features that can be useful in its processing. Photovoltaic production consists of three components.

- The first component is directly connected to the sun and its motion over the sky and the local latitude of the power plant [1]. The solar altitude is changing over time on the day-night cycle and is defined by a sinus shape function.

- The second component depends on the local non-changeable factors of the installation. The azimuth and declination angles of panels, local shadows (e.g. neighbourhood of a forest or other buildings, many chimneys on the roof, etc.).

- The third component is introduced by the weather.

Our time series is also characterized by strong seasonality which can be observed within a one day range. Decomposition of those elements allow us to first focus on the two particular components. We create a model, which maps a learning data into the most anticipated one day power production curve. In other words, it observes data and creates the most adjusted and most expected power path. Using this strategy we included both geographical and local factors (first two components). The third component is also included, but not during the learning process, but when forecasting.

Regarding previously described features of the considered time series, we introduce a division of daytime into k sections defined as $S \in \{s_i \in \{B_i, B_{i+1}\}, i \in \{1, \ldots, k\}\}$ where B is an ordered sequence of boundaries $B = (0, b_1, \ldots, b_{k-1}, 1440)$, each of $k+1$ elements b is an integer from the range (0, 1440) and b stands for minutes of a day, so maximum k is 1440, but is has also to be adjusted to sampling rate in dataset. For example, maximum value for five minutes sampling interval dataset is $k = 288$. Then, the model parameters $\Phi = \{\phi_{S_1}, \phi_{S_2}, \ldots, \phi_{S_k}\}$ are calculated according to the equation 1.

$$\phi_{S_i} = \phi_{\{B_i, B_{i+1}\}} = \overline{Y_i}, Y_i \in \{y_j \in Y \wedge B_i > MoD(t_j) > B_{i+1}, j \in \{0, \ldots, n\}\} \quad (1)$$

Where n is a number of samples in a dataset and $n >> k$. MoD is a function that calculates minutes of a day of the given timestamp. $Y = \{y_0, y_1, \ldots, y_n\}$ is a set of time series data used to learn the model, and $T = \{t_0, t_1, \ldots, t_n\}$ is a set of corresponding to the learning data timestamps. When the model is constructed, the forecasting for a given timestamp $TS = \{ts_1, ts_2, \ldots, ts_{fh}\}$ (where ts_i is a timestamp of a previous forecast value to the ts_{i+1} and fh is a forecast horizon) and the last observed time series value w is as on equation 2. We also introduced an additional random element N.

$$Y_{pred} = (\phi_{\{B_i, B_{i+1}\}} + N \wedge B_i > MoD(TS_j) > B_{i+1}, i \in \{0, \ldots, k\} | j \in \{0, \ldots, fh\}) \quad (2)$$

In that point, the third signal component is included. The temporary circumstances like the weather are taken into account by adjusting our most expected path to the current observation w, by subtracting difference between first forecast Y_{pred_0} and w. The formula is show on equation 3.

$$Y_{pred} = (Y_{pred_i} - (Y_{pred_0} - w), i \in \{0, \ldots, fk\}) \quad (3)$$

3. Experimental protocol

The tests have been conducted on our own dataset, which consists of nearly half a year of observation of a 6 kWp photovoltaic power plant located in Lower Silesian Voivodeship, sampled with a five minutes interval. The dataset contains more than 42 000 samples. We used models to create forecasts for one-hour and two-and-a-half-hour time frames. Both the research environment and models have been implemented in Python programming language. We used an AR model developed in *Statmodels* library.

During the tests we used a direct forecasting approach [2] to forecast a coming period for each sample within dataset. We introduce some hyperparameters which define the learning process. *Learning_size* is count of samples taken to the process of learning in model formal description it is defined as n. *Forecast_horizon* is a number of samples to be forecasted, defined as fh in a model. The last parameter is *window*. It is a number of previous observations given to the model as a set of features. Our model uses only the last observation, which is called w.

During forecasting new data is given to the model and obtained forecasts are compared with the ground true values *y_true* with usage of MAE and MAPE metrics. The process of forecasting is shown in figure 1.

From time to time, the learning process needs to be repeated, to ensure that model is up to current trends in the dataset. This process occurs when *samples_w/o_learning* samples passed from last learning process. A diagram presenting repeated learning is shown in figure 2. Due to the informativeness of all samples gathered during the day being equal we decided to make all segment of equal

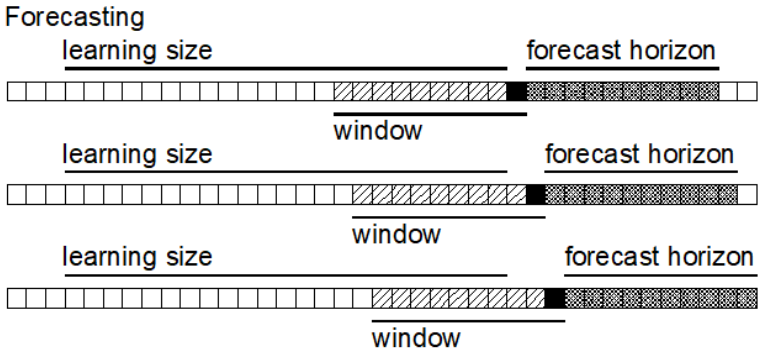


Figure 1. Forecasting process. Source: own work.

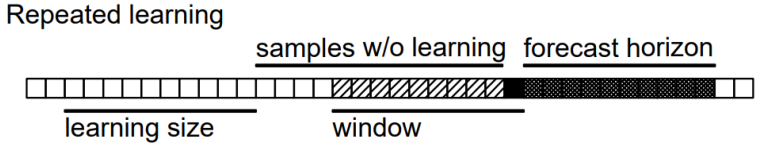


Figure 2. Repeated learning. Source: own work.

length. Segments S and boundaries sequence B have been reduced to one variable, the count of segments. With the usage of a grid search method we found the best combinations of hyperparameters for each model and $forecast\_horizon$ separately. Descriptions and mean scores of these models are shown in Table 1. Afterward, we compared the best-tuned models for both values of a $forecast\_horizon$ parameter using T-Test and P-value. In the Fig. 3 and 4 there are two sample forecasts created from both models for each $forecast\_horizon$. The blue line represents the ground truth value. Green and purple represent forecasts made by Ar and our own model, respectively. The orange and red are the current MAE.

Table 1. Model description with the obtained MAE and MAPE metrics

Fh [hours]	Model	param.	*window*	*learning size*	*samp_w/o learning*	MAE	MAPE
1	Own	50	1	6000	288	0.26	15%
1	Ar	15	2500	-	-	0.31	63%
2.5	Own	50	1	6000	288	0.35	48%
2.5	Ar	20	2250	-	-	0.47	149%

There can be observed that our model makes stairs-like forecast. Depending on the date time the forecasts rising or falling. The second observation regarding

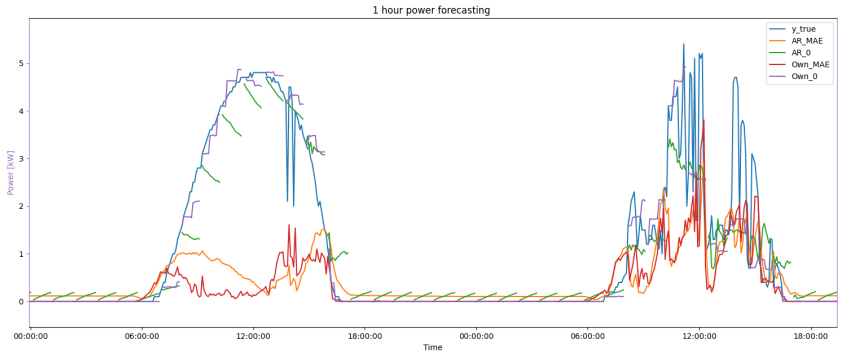


Figure 3. One hour power forecasting. Source: own work.

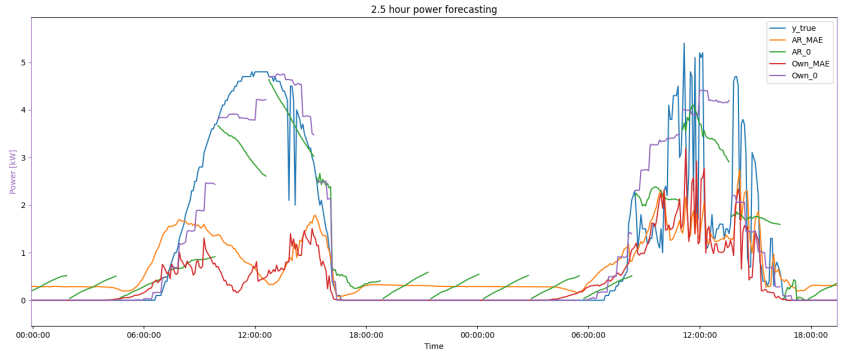


Figure 4. Two and half hour power forecasting. Source: own work.

our model is that the forecast is always adjacent to the *y_true*. In comparison Ar for both forecasts horizons creates line-like curves. Those curves are rising when the beginning is over 1-2 kW and falling in other case. In the middle the prediction are more flat.

4. Conclusions

In this paper, we proposed a model for photovoltaic power plant power production forecasting. We proved that for our dataset spanning almost half a year, our model generates better forecasts measured by both metrics than the reference model. During the tests, we made forecasts for one-hour and two-and-a-half-hour horizons. We obtained MAE 0.26 and MAPE 15%for our model and MAE 0.31 and MAPE 63% for reference Ar model for one-hour forecasting. Further, for two-and-a-half-hour forecasting we obtained MAE 0.35 MAPE 48% for our model and MAE 0.47 and MAPE 149% for Ar model. Height MAPE for Ar model was

caused by the occurrence of null values in the dataset during the night, where Ar model made a significant errors in forecasts. The statistical analysis proved that our model is statistically significantly better than the Ar model. However we are aware that there is still a place for improvements. In the further work we plan to compare our model with more complex state-of-the-art models such as ARMA, SARIMA or ANN-based as other has been done in other research [3, 4].

References

[1] Kalogirou S.A., *3.01 – solar thermal systems: Components and applications—introduction*, [In:] T.M. Letcher (ed.), *Comprehensive Renewable Energy (Second Edition)*, Elsevier, Oxford, second edition edn., 2022, pp. 1–25, doi: https://doi.org/10.1016/B978-0-12-819727-1.00001-7.

[2] Ben Taieb S., Bontempi G., Atiya A.F., Sorjamaa A., *A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition*, *Expert Systems with Applications*, 2012, vol. 39, no 8, pp. 7067–7083, doi: https://doi.org/10.1016/j.eswa.2012.01.039.

[3] Vagropoulos S.I., Chouliaras G.I., Kardakos E.G., Simoglou C.K., Bakirtzis A.G., *Comparison of sarimax, sarima, modified sarima and ann-based models for short-term pv generation forecasting*, [In:] *2016 IEEE International Energy Conference (ENERGYCON)*, pp. 1–6, doi: 10.1109/ENERGYCON.2016.7514029.

[4] Hassanzadeh M., Etezadi-Amoli M., Fadali M., *Practical approach for sub-hourly and hourly prediction of pv power output*, [In:] *North American Power Symposium 2010*, pp. 1–5, doi: 10.1109/NAPS.2010.5618944.

Text-to-music Models and Their Evaluation Methods

Mateusz Modrzejewski[0000−0002−6363−8584],
Przemysław Rokita[0000−0002−4433−2133]

Warsaw University of Technology
Institute of Computer Science
Nowowiejska 15/19, 00-665 Warszawa, Poland
mateusz.modrzejewski@pw.edu.pl

DOI:10.34658/9788366741928.34

Abstract. *Text-to-music models are a very recent approach to generative music, allowing to generate music based on an abstract, rich description input in natural language. In this paper, we propose guidelines for evaluation in text-to-music models, highlighting the need for musical insight and clear descriptions of perceptual quality upon investigating the metrics of currently developed approaches. We also present a critical analysis of the capabilities and evaluation methods of the pioneering text-to-music models.*
Keywords: *generative music, music information retrieval, text-to-music, deep learning, diffusion models*

1. Introduction

Deep learning based generative music enables new tools and means of expression for musicians and is a step towards the democratization of music creation. The immense success of NLP systems like ChatGPT and text-to-image systems like DallE-2 is now being reflected in generative music systems, with the emergence of text-to-music models.

Text-to-music (TTM) models differ significantly from previous approaches. A number of models have had no input control whatsoever, apart from the qualities of their training data, while other allowed for simple conditioning e.g. on a few starting notes. In contrast, TTM model input is a rich text description of the desired music, e.g. *"The main soundtrack of an arcade game. It is fast-paced and upbeat, with a catchy electric guitar riff. The music is repetitive and easy to remember, but with unexpected sounds, like cymbal crashes or drum rolls"*. This enables a new, natural and abstract way of interacting with the model. These approaches are often based on diffusion models and in many cases also able to perform additional tasks, like audio inpainting.

Our contribution is twofold: we propose guidelines for the missing musically informed evaluation, allowing for clear descriptions of the perceptual musical qualities of the generated content. We also provide a critical analysis of 7 recent pioneering generative TTM neural network models and their evaluation methods: 6 of the considered models have been presented in 2023 and one in late 2022. This makes our contribution one of the first, if not the first, comparisons in the discussed area.

2. Evaluation of generative music

The issue of clear evaluation of generative audio systems has long been an unsolved area of music information retrieval. In [1], the authors generate single-track folk music in a symbolic ABC format using character-level RNNs. The evaluation and analysis are very rich and include both a statistical analysis of the output tokens, as well as a detailed, musically informed expert-level analysis of the rhythmic, harmonic and compositional aspects of the outputs. Sadly, musical analysis like this is very rarely seen in the literature. The authors of [2] develop generative adversarial networks (GAN) for generating multi-track MIDI and propose a number of automated analytic metrics, e.g. the ratio of empty bars, number of pitch classes used, tonal dissonance, scale consistenty and pitch entropy. In terms of audio quality, a recent study [3] on neural audio synthesis systems has shown that objective metrics such as Fréchet Audio Distance (FAD), Kernel Inception Distance (KID), and reconstruction errors are insufficient to measure the audio quality and to provide meaningful estimates thereof.

With the emerging text-to-music models, except for musical analysis and audio quality analysis, there is also a need to verify how well the text input matches the produced output. We therefore find the need of TTM evaluation within three categories:

1. musically-informed metrics (*compositional qualities*)
2. audio quality (*audio qualities*)
3. text-to-music relevance (*description and output correspondence*)

3. Critical analysis

We investigate the following recent models: MusicLM [4], Noise2Music [5], Make-an-Audio [6], Moûsai [7], AudioLDM [8], ERNIE [9] and Riffusion [10]. [4] and [5] are works by Google, [6] is a joint work by academics and ByteDance (TikTok), while the others are academic contributions. MusicLM encompasses three pre-trained models, with a fully convolutional encoder-decoder architecture

serving as the neural audio codec, MuLAN as the audio embedding model and w2v-Bert for the semantic tokens. Noise2Music, Moûsai, Make-an-Audio, AudioLDM and ERNIE utilize diffusion models conditioned on text. Riffusion [10] employs an unique approach, adapting image-based diffusion to pairs of text and spectrograms. While it is commonly known that spectrograms should not be treated as ordinary images, the *Riffusion* model seemingly ignores this fact and generates a stream of spectrograms which is subsequently converted into audio. Since its publishing, the model has gained significant attention in the community.

3.1. Capabilities

An overview of the capabilities of the TTM models can be seen in Table 1. We distinguish generating music from short captions, which span over a few keywords, and rich captions, which can span over multiple sentences. Inpainting is a procedure of filling in an existing audio file with generated data. Timbral style transfer is a process of applying a different timbre to a given audio file. Output samples which preserve coherence over several minutes are considered long.

Table 1. Capabilities of TTM models

Model	**short captions**	**rich captions**	**music inpainting**	**timbral style transfer**	**long samples**
MusicLM	✓	✓	-	-	✓
Noise2Music	✓	✓	-	-	✓
Moûsai	✓	-	-	-	-
AudioLDM	✓	✓	✓	✓	✓
ERNIE	✓	-	-	-	-
Make-An-Audio	✓	-	✓	✓	-
Riffusion	✓	-	-	-	✓

3.2. Evaluation methods

As presented in Table 2, there is no agreement on the evaluation methods of TTM models. Four of the considered models are evaluated using variants of the Fréchet Audio Distance. This metric, however, is dependent on the used dataset, and makes objective model comparison difficult. MusicLM and Noise2Music, both works by Google, additionally include MuLAN based metrics. MuLAN is a joint text and audio embedding model, trained also by Google using 44 million music recordings. The usage of MuLAN embeddings allows to investigate how closely do the pairs of music and text correspond to each other. The CLAP metric

used by [6] has a similar meaning, but it uses a different embedding model, making it difficult to compare the two related metrics.

Table 2. Evaluation methods of TTM models

Model	FAD	MuLAN based	IS	KLD	CLAP	manual evaluation
MusicLM	✓	✓	-	✓	-	-
Noise2Music	✓	✓	-	-	-	-
Moûsai	-	-	-	-	-	✓
AudioLDM	✓(FD)	-	✓	✓	-	-
ERNIE	-	-	-	-	-	✓
Make-An-Audio	✓	-	-	✓	✓	✓
Riffusion	-	-	-	-	-	-

[7], [9] and [6] employ ways of manual evaluation with human participants. [6] reports Mechanical Turk results, while [7] and [9] employ 3 and 10 expert participants, respectively, in order to quantify selected aspects of the generated audio, such as genre compliance and text-audio relevance. Finally, IS (inception score) is measured in [6] and KLD (Kullback-Leibler divergence) scores are measured in [4], [8] and [6].

Finally, none of the models make an attempt to analyze the musical qualities of the generated music, e.g. harmony, rhythmic structure or instrumentation.

3.3. Guidelines for clear musical evaluation

As mentioned in Section 2, clear evaluation of TTM models has to consider three categories. While audio quality and text-to-music relevance seem to be tackled in some ways, the musical qualities are either completely ignored, or left to questionable manual evaluation. We propose the utilization of music information retrieval techniques in order to enable a wider, musically-informed analysis of the results. Beat tracking solutions [11] facilitate rhythm analysis on audio data. Existing source separation techniques allow to obtain audio stems (single-instrument tracks) [12], which can further be processed using neural transcription models [13], obtaining symbolic music with high fidelity to the original audio track. Symbolic formats, like MIDI, are well suited for automated musical analysis, enabling quantitative evaluation of attributes such as key, rhythm, tempo, chord progressions and other compositional qualities.

Furthermore, the same pipeline could be adapted to the training dataset, providing an additional, musically-informed way of evaluating the generated samples against the original dataset, alongside variants of FAD and KLD. Finally, manual evaluation would also benefit significantly from such a pipeline, facilitating expert musical analysis similar to the one seen in [1].

4. Conclusions

We propose guidelines for automating selected aspects of musical analysis for the needs of TTM systems in order to facilitate comparison between the outputs of various systems. We also present a critical analysis of emerging text-to-music generative models (published in 2023) and their evaluation methods. We find that the models have similar functionality and provide promising results, but are almost impossible to objectively compare, making further development extremely difficult. The metrics differ in all of the investigated approaches and do not give clear indications of perceptual qualities of the generated music.

In future work, we would like to perform a comparative listening study of the outputs (e.g. in the MUSHRA methodology), propose a set of new musically informed metrics and experiment with models of our own.

References

[1] Sturm B.L., Santos J.F., Ben-Tal O., Korshunova I., *Music transcription modelling and composition using deep learning*, *arXiv preprint arXiv:1604.08723*, 2016.

[2] Dong H.W., Hsiao W.Y., Yang L.C., Yang Y.H., *MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment*, [In:] *AAAI*, vol. 32.

[3] Vinay A., Lerch A., *Evaluating generative audio systems and their metrics*, *arXiv preprint arXiv:2209.00130*, 2022.

[4] Agostinelli A., Denk T.I., Borsos Z., Engel J., Verzetti M., Caillon A., Huang Q., Jansen A., Roberts A., Tagliasacchi M., et al., *MusicLM: Generating music from text*, *arXiv preprint arXiv:2301.11325*, 2023.

[5] Huang Q., Park D.S., Wang T., Denk T.I., Ly A., Chen N., Zhang Z., Zhang Z., Yu J., Frank C., et al., *Noise2music: Text-conditioned music generation with diffusion models*, *arXiv preprint arXiv:2302.03917*, 2023.

[6] Huang R., Huang J., Yang D., Ren Y., Liu L., Li M., Ye Z., Liu J., Yin X., Zhao Z., *Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models*, *arXiv preprint arXiv:2301.12661*, 2023.

[7] Schneider F., Jin Z., Schölkopf B., *Mo\ˆusai: Text-to-music generation with long-context latent diffusion*, *arXiv preprint arXiv:2301.11757*, 2023.

[8] Liu H., Chen Z., Yuan Y., Mei X., Liu X., Mandic D., Wang W., Plumbley M.D., *AudioLDM: Text-to-audio generation with latent diffusion models*, *arXiv preprint arXiv:2301.12503*, 2023.

[9] Zhu P., Pang C., Wang S., Chai Y., Sun Y., Tian H., Wu H., *ERNIE-music: Text-to-waveform music generation with diffusion models*, *arXiv preprint arXiv:2302.04456*, 2023.

[10] Forsgren S., Martiros H., *Riffusion – Stable diffusion for real-time music generation*, 2022, (access: 12-07-2023).
`https://riffusion.com/about`

[11] Böck S., Krebs F., Widmer G., *Joint beat and downbeat tracking with recurrent neural networks.*, [In:] *ISMIR*, New York City, pp. 255–261.

[12] Défossez A., *Hybrid spectrogram and waveform source separation*, [In:] *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*.

[13] Bittner R.M., Bosch J.J., Rubinstein D., Meseguer-Brocal G., Ewert S., *A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation*, [In:] *ICASSP*, Singapore.

Towards Ontology-Driven Verification of Car Claims Settlement

Krzysztof Pancerz[1,3][0000−0002−5452−6310],
Jacek Wolski[2,3][0000−0001−6308−8693]

[1]*The John Paul II Catholic University of Lublin*
Institute of Philosophy
Al. Racławickie 14, 20-950 Lublin, Poland
kpancerz@kul.pl
[2]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
jacek.wolski@dokt.p.lodz.pl
[3]*MakoLab S.A.*
Ogrodowa 8, 91-062 Łódź, Poland

DOI:10.34658/9788366741928.35

Abstract. *In the paper, we outline an intelligent tool enabling the users to automatize the process of verification of the car claims settlement. Two data sources power the tool. The first one is the source of car images in which damaged elements are recognized. The second one is the source of PDF files in which cost estimates are extracted. The designed ontology of car repair, described in the paper, is used both in the pre-processing step and in recognition of atypical situations.*
Keywords: *OWL, ontology of car repairing, deep learning, image recognition, claims settlement*

1. Introduction

The car claims settlement is vulnerable to abuse. Therefore, there is a need to implement an intelligent tool enabling insurance companies to detect insurance frauds. In the paper, we propose a tool aided by the semantic knowledge included in the designed ontology that is the main contribution of our research. The general scheme of the procedure implemented in our tool is shown in Figure 1. Data collected and processed by the tool come from two sources. The first one is the source of car images in which damaged elements are recognized. The second one is the source of PDF files in which cost estimates are extracted. The process of damaged element recognition is described in Section 3.1. The process of cost estimates extraction is described in Section 3.2.

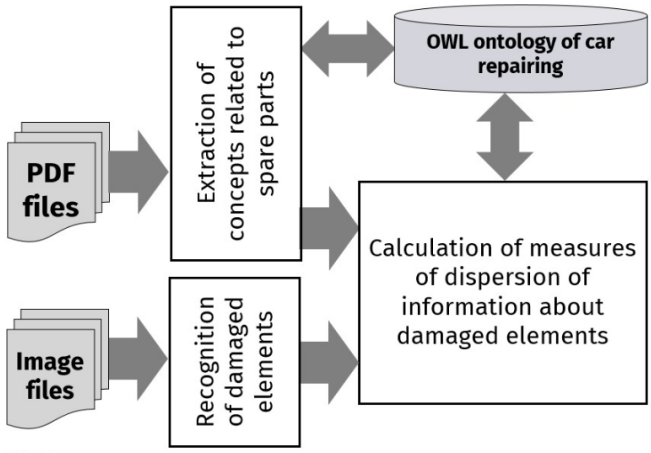


Figure 1. The general scheme of the procedure. Source: own work.

The verification process of car claims settlement is supported by the designed ontology of car repairing presented in Section 2. An increasing attention has been recently focused on ontologies since modern computer tools require semantic and well-structured knowledge bases covering different aspects of information that is processed. The use of ontology in our application allows us: (i) to unify terminology concerning car repairs (data come from variety of formats to record cost estimates, moreover, abbreviations or acronyms are commonly used in tools that generate automatically cost estimates); (ii) to fuse information coming from cost estimates extraction and damaged elements recognition; (iii) to define measures of dispersion about damaged elements for recognition of atypical situations.

2. OWL Ontology of Car Repairing

In this section, we give the outline of the OWL ontology of car repairing, created by us as a central unit of our application. There are many definitions and interpretations of the term *ontology* in the literature. Some of them are recalled in [1]. In general, an ontology describes: concepts in a given domain of interest, instances of concepts, properties representing semantic relations expressing various types of associations between instances as well as various features of instances. Our ontology is built in accordance with the OWL 2 Web Ontology Language (shortly OWL 2) [2]. An OWL ontology consists of three components: classes representing concepts, individuals being instances of classes, properties being binary relations on individuals. OWL 2 distinguishes two types of properties: object properties linking individuals and data properties linking individuals to data values. The OWL ontology of car repairing (identified later by acronym CARRONT) has been im-

plemented using Protégé [3] that is a free, open source, platform-independent environment for creating and editing ontologies and knowledge bases. CARRONT represents the semantic knowledge about damaged (replaced) spare parts and performed activities included in cost estimates. In the hierarchical structure of the OWL classes we have distinguished three basic classes: (1) car element (the basic concept for the spare parts hierarchy); (2) operation (the basic concept for the hierarchy of operations performed during car repair, e. g. *assembly*, *disassembly*, *painting*, etc.); (3) car side (the basic concept for the car side hierarchy, i.e., at the lowest level: *left side*, *right side*, *front side*, *back side*).

At the lowest level of abstraction, concepts are represented by individuals, while at the higher levels of abstraction – by classes). CARRONT includes information about equivalence of individuals, for example *front cover* is the same individual as *engine mask*.

In the OWL ontology we have also defined properties allowing to represent relations between concepts: (i) the relation of performing a specific operation on a specific part of the car, e. g. *dismantling the door*; (ii) the relation of a specific part of the car to a specific side of the car, e. g. *the left rear door is on the left side*; (iii) the co-occurrence relationship of parts of the car, e. g. *the left rear door window occurs with the left rear door*. Currently, CARRONT includes: about 120 classes, about 150 individuals, 4 object properties.

3. Data Acquisition

3.1. Identification of Damaged Car Body Elements

First, photos (JPG images) included in a claim file are undergoing selection. i.e., vehicle type (car/motorcycle/bus/truck), zoom in/out, etc. Sample images have been annotated using the Label Studio tool and split into train, test and validation sets. Subsequently the annotated data are fed into models to train them. Next, the selected images are processed by models to detect bounding boxes for parts matching the ontology (e.g. left back door, trunk etc.) and for damages (e.g. dent, scratch etc.). Overlapping parts and damage areas are used to compute probability of a part being damaged.

The damaged part list is constructed on the basis of all images in a given claim file where damaged parts have been identified. We selected the Convolutional Neural Network YOLO v.7 [4] mainly due to the fact that it is open source, fast (need to process a large number of images) and supported by a considerable developer community.

In order to facilitate car part annotations we use model over-compression, which means that parts are only annotated according to their shape, exclusive of their direction. This allows us to mark only 9 classes instead of 22, which reduces

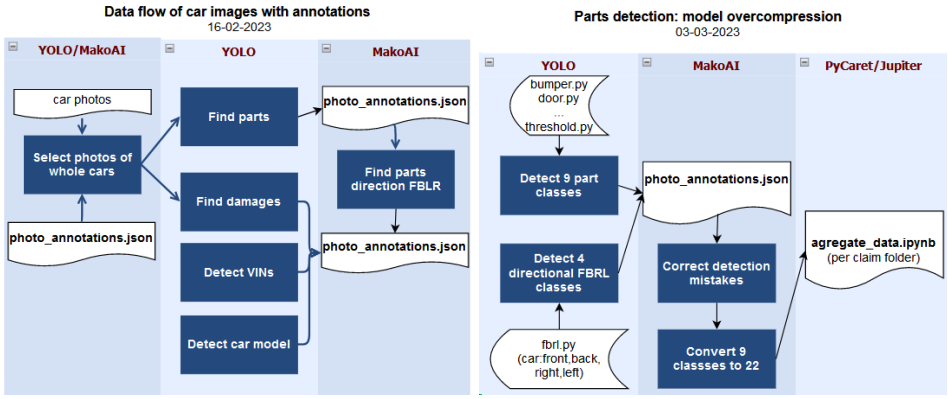


Figure 2. Image processing for car body elements. Source: own work.

the number and complexity of models to train. Hence, the energy use is reduced during the training process.

We obtained the following mean average precision (mAP) of detection for the selected parts: rear bumper – 0.92, left rear door – 0.96, right rear door 0.87, front bumper – 0.91, left front door – 0.96, right front door – 0.96, left mirror – 0.94, right mirror – 0.95, right front fender – 0.92, right rear fender – 0.92, left front fender – 0.89, left rear fender – 0.89.

3.2. Parsing Cost Estimates for Car Repairs

Pre-processing of data relevant for the verification of damaged elements includes the following operations: data cleaning and recognition of abbreviations or acronyms. Assignment of extracted concepts related to spare parts to concepts defined (as individuals) in the OWL ontology of car repairing (CARRONT), described in Section 2, is based on text analysis as well as a fuzzy approach to matching names of spare parts to concepts in CARRONT. This is related to specific names of spare parts listed in cost estimates (e.g., abbreviated parts names). Currently, the mapping procedure identifies about 200 unique pairs (name of the spare part in the cost estimate – concept in the ontology).

4. Measures of Information Dispersion

In order to recognize atypical situations, we propose to define measures of dispersion of information about damaged elements on the basis of the similarity measures between two concepts defined in the literature for semantic networks organized hierarchically (see e.g. [5], [6]). These measures utilizes information about the level of abstraction of two compared concepts in the hierarchy. Concepts

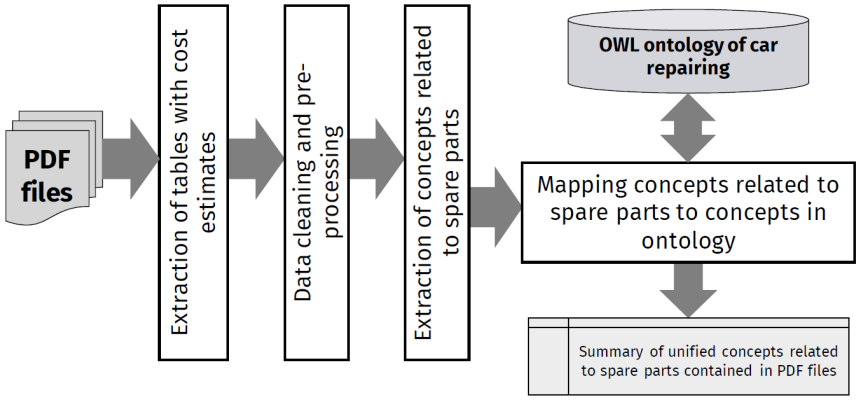


Figure 3. The procedure of parsing the PDF files. Source: own work.

at upper levels of the hierarchy have more general semantics and less similarity between them, while concepts at lower levels have more concrete semantics and stronger similarity. The similarity between two concepts v and v' is considered to be governed by the length l of the shortest path in the semantic network, as well as the depth h of the subsumer (the depth is measured as a distance between the closest concept generalizing v and v' to the top of hierarchy – the root concept). Moreover, in measures of information dispersion, we take into account other relations between the concepts. Particularly, the relation attributing the side of the car to the part of the car as well as the relation determining the part co-occurrence provide enriching information to these measures.

5. Conclusions

In the paper, we have outlined the architecture of an intelligent tool supporting automated verification of car claims settlement aided by the designed ontology of car repairing (CARRONT). One of the further directions of development of the proposed approach will be the extension of ontology with a part representing knowledge derived from surveys completed during damage settlement.

References

[1] Gomez-Perez A., Fernandez-Lopez M., Corcho O., *Ontological Engineering*, Springer-Verlag, London, 2004.

[2] Hitzler P., Krötzsch M., Parsia B., Patel-Schneider P.F., Rudolph S., *OWL 2 Web Ontology Language Primer*, (access: 12-07-2023). https://www.w3.org/TR/owl2-primer/

[3] Musen M.A., *The protégé project: A look back and a look forward*, *AI Matters*, 2015, vol. 1, no 4, p. 4–12, doi: 10.1145/2757001.2757003.

[4] Chien-Yao Wang A.B., Liao H.Y.M., *Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*, *arXiv.org*, 2022, doi: arXiv: 2207.02696.

[5] Li Y., Bandar Z., Mclean D., *An approach for measuring semantic similarity between words using multiple information sources*, *IEEE Transactions on Knowledge and Data Engineering*, 2003, vol. 15, no 4, pp. 871–882, doi: 10.1109/TKDE.2003.1209005.

[6] Rada R., Mili H., Bicknell E., Blettner M., *Development and application of a metric on semantic nets*, *IEEE Transactions on Systems, Man, and Cybernetics*, 1989, vol. 19, no 1, pp. 17–30, doi: 10.1109/21.24528.

Using Security Games against Wild Dumping Sites

Marek Adrian[0000−0002−0435−0994], **Jerzy Markiewicz**

AGH University of Krakow,
al. A. Mickiewicza 30, 30-059 Kraków, Poland
madrian@agh.edu.pl; jerzykmarkiewicz@gmail.com

DOI:10.34658/9788366741928.36

Abstract. *Several types of criminal activity can by reduced, or prevented, by randomly patrolling areas where it could take place. However, man-made schedules tend to have hidden patterns, which reduces their effectiveness. Thus in several cases game theory has been used to achieve true randomness, while minimizing the potential gain of any hostile entity. Building on this approach we propose a model for creating random schedules for placing cameras to prevent creation of wild dumping sites. While the model is constructed specifically to deal with this particular problem, it can be easily used to plan other schedules involving roads and traffic.*
Keywords: *game theory, Stackelberg games, security games*

1. Introduction

Game theory for years has been successfully used to model situations in which agents with their own agendas make rational decisions. As the entities do not know what decisions are made by others, this uncertainty has to be addressed in a scientific manner. This is done by using the language of probability, and while there are arguments against such an approach to uncertainty, it has been proven to be the only "rational" choice with respect to maximizing a sort of utility function[1].

The relation between the goals of the entities can be positive, which will lead to modeling a form of coordination, but they can also be adversarial in their nature, with goals staying in contradiction with each other. In such cases it is not uncommon for one of the entities to observe the other for an extended period of time, to try to predict their moves and act accordingly. Such cases have been modeled by so called security games.

These kind of games have been successfully implemented in different kinds of systems for over a decade now[2]. Starting with a system to randomize checkpoints at the LAX airport[3], a system to schedule the flights of US Marshals, or to help combat poaching[4], just to name a few examples.

Seeing that this approach can be introduced to combat a variety of problematic behaviours, we purpose a model that can be used to prevent, or at least increase the chance of catching in the act, the creation of illegal dumping sites.

The structure of this article is the following: in section 2 we introduce the basic preliminaries needed for our work and a brief overview of the situations where security games have successfully been implemented. After that we describe fully our proposed model. We conclude with our final thoughts and ideas for further improvement of the model.

2. Preliminaries

The basic ideas behind the general model in game theory are very straightforward. We have a set of agents, each of them has a set of possible actions, of which every game round each of the will pick exactly one, and a valuation function that describes their gain depending all the actions chosen by the agents. The idea how should the players choose their actions is widely discussed, but one of the acceptable opinions is for them to maximize their utility function. The decision which action to pick is assumed to happen at the same time, but obviously that is not always the case. For example if one of the agents utility always decreases when another agent picks the same action (which can represent the idea of doing something the other player is trying to prevent), he could wait and observe what pattern of moves is chosen by the other player, and only then pick a strategy to minimize the chance of coordination. This kind of games have been captured by the idea of the Stackelberg security games[5]. In these games we usually consider two players: one is called the attacker and one the defender. It is assumed that the attacker has an arbitrary long time to observe the decisions made by a defender and thus knows what strategy is chosen by the defender. With such approach it is obvious that for the defender to have a realistic chance against the attacker no pattern in his behaviour is allowed. Of course if the defender had resources to cover all of the places of interest there would be no problem, so we need to assume that his resources are finite. At this point the problem comes down to deciding how to apply the given resources in a random manner. Because of the necessity of using randomness in the assignment it becomes clear that to measure the success of the defender it is necessary to minimize the expected utility of the attacker. While there are several ways to approach the utility function for the attacker, the model proposed in [6] seems to be sufficient for the purposes of this article.

3. The purposed model

We decided to try and implement security game solutions to the problem of wild dumping sites as it is a pressing problem[7]. To that end we need to make

some assumptions about the situation. Firstly, wild dumping sites will tend to be created alongside roads, as a vehicle is needed to transport the waste. Preferably it will be along smaller roads, even dirt roads if possible, to minimize the chance of being caught by a random witness. Also we assume that they will be created in forests rather than in a plains area, due to the natural camouflage from the surrounding trees. Thus when assigning the vulnerability of the roads for the model, smaller roads in a forested area are given a higher priority for the potential attacker. Obviously the area being observed should be relatively large, as with small areas there should be little difference with the coverage proposed by the model, which would lead to uninteresting observations.

While the implementation of the model could easily be used on any area for which a map is provided, we decided to focus on the forest around Niepołomice as our test example. The are is large enough so that observing all vulnerable areas at the same time is economically unfeasible, and it has different types of roads, thus allowing to naturally assign different classes of interest from the potential attacker to them.

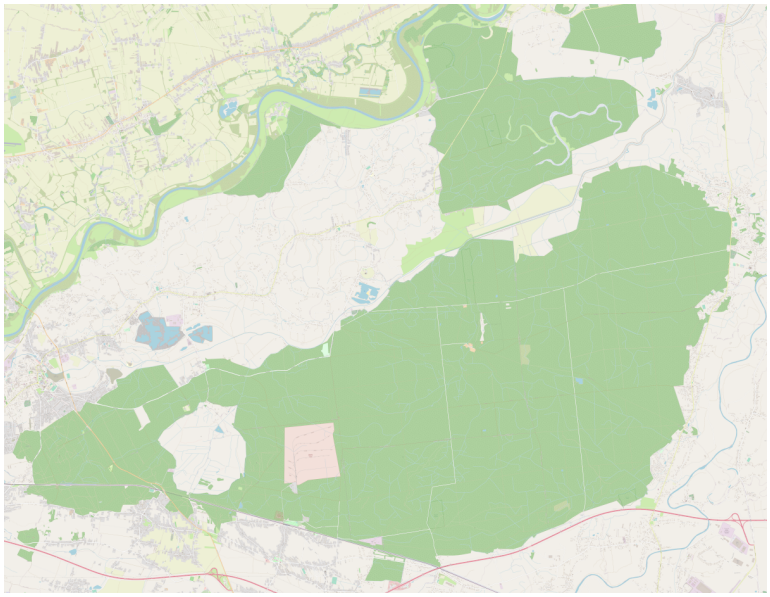


Figure 1. The map used as taken from Bigmap.

To implement the ideas behind the security game model of our choice we need to convert the map of the area in question into a matrix. In the matrix each cell represents a small area of the map, and has a value attached to it witch represents the attractiveness of that spot for a potential dumping site. For simplicity sake, we attached the value of zero to any cell that did not include a part of the road in its representation of the map. Thus the matrix that we created visually shows the

layout of roads from the original map. The map of Niepołomice forest and roads attached has been taken from the service Bigmap as seen in Figure 1. To create the matrix we assigned divided the map into areas of size 15px × 15px and assigned each such area to a cell. Then we detected which cells included any pixel form the roads visible on the map, assigning a non zero value to that cell, the higher the less important road it was attached to. The visualization of this matrix can be seen in Figure 2 where the red color represents non zero cells in the matrix, the more intense the color, the higher the value.

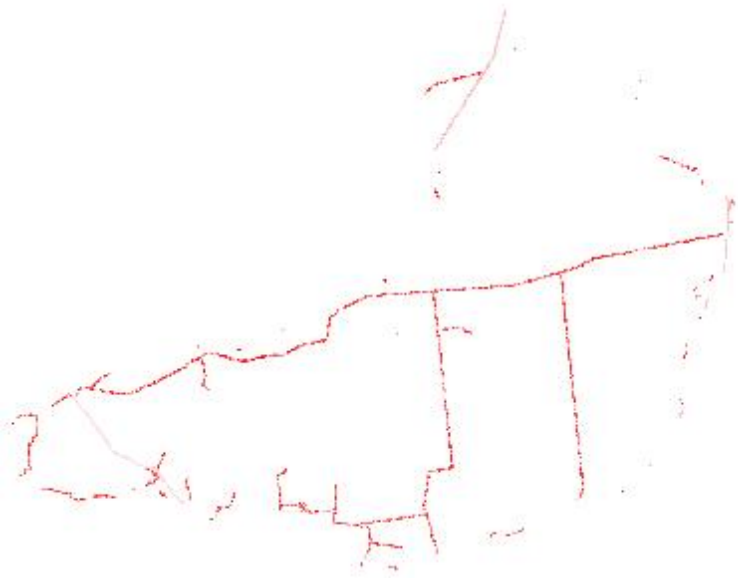


Figure 2. Visualisation of the matrix. Source: own work.

Now we will shortly describe how we define the strategies for each player in such a game. The defender is given a set number of cameras to distribute through out the cells of the matrix (the number of cameras considered can go from one to as many as needed to cover the whole matrix). His action is to give a distribution of cameras among the cells and the strategy is given by assigning a probability value of placing a camera in that spot to each cell of the matrix. The sum of all these probabilities cannot exceed the number of available cameras. The attacker chooses a cell in the matrix depending on his reward function which is directly correlated to the values we attached to each cell. His strategy is also given by assigning probabilities of picking each cell for the dumping site. In our work we assume there is only one attacker for simplicity, but the model used can easily deal with more. The game can be treated as a multistep game, but we focus just on one step in this paper. In one step first the defender decides his distribution of probabilities, then the attacker given that information decides his distribution.

After those assignments are complete the placement of cameras is determined by from the strategy as well as the placement of the dumping site and and the utility for each player is calculated.

At this point what is left is to conduct the proper calculations on the given matrix. For that end we decided to follow the ideas of Nguyen et al.[6] and recognise the subjective expected utility (SEU) of the attacker given by the formula

$$U_t^a = w_1 x_t + w_2 R_t^a + w_3 P_t^a \tag{1}$$

where x_t relates to the marginal coverage on a potential target, R_t^a and P_t^a are the potential rewards and penalties respectively for the attacker, and w_1, w_2, w_3 are weights assigned in our case arbitrarily. We then follow the procedure from the aforementioned article to calculate the probabilities of attack on each cell of the matrix.

In the end the output matrix allows to assign decide what is the optimal placement of cameras to minimize the potential utility of the attacker, and as such increase the chance of prevention. After one placement of cameras is created the information can be included to our matrix by decreasing the value around the spots where the cameras have been place. This allows for the next schedule to have a better chance of not placing the cameras in the same spaces.

4. Conclusion

In this paper we have described briefly how we model roads in the forest around Niepołomice and how we use this information to create random schedules of layouts of cameras that can be used to combat the creation of wild dumping sites.

There are several ways to improve what has already been done. Firstly, the constants in the model have been chosen in an arbitrary fashion and replacing them by constants that arise from experimental data could improve the scheduling. Also the way we assigned weights to the roads was done in an arbitrary way and there may be a better distribution of weights that will lead to better results.

While this model has been created with wild dumping sites in mind, it should be clear that it can be easily adjust for other traffic situations. Obviously, one area were this model could be reused is to combat speeding, by randomly assigning the placement of speeding cameras and patrol cars, but we think it is not limited to static prevention, and we could find more active uses for it.

References

[1] Tadelis S., *Game Theory: An Introduction*, Princeton University Press, 2013, access: 12-07-2023). https://books.google.pl/books?id=eLkOJPwAdu8C

[2] Sinha A., Fang F., An B., Kiekintveld C., Tambe M., *Stackelberg security games: Looking beyond a decade of success*, [In:] *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, International Joint Conferences on Artificial Intelligence Organization, pp. 5494–5501, doi: 10.24963/ijcai.2018/775.

[3] Kiekintveld C., Jain M., Tsai J., Pita J., Ordóñez F., Tambe M., *Computing optimal randomized resource allocations for massive security games*, vol. 2, pp. 689–696, doi: 10.1145/1558013.1558108.

[4] Fang F., Stone P., Tambe M., *When security games go green: Designing defender strategies to prevent poaching and illegal fishing*, [In:] *International Joint Conference on Artificial Intelligence (IJCAI)*.

[5] Wilczyński A., Jakóbik A., Kołodziej J., *Stackelberg security games: Models, applications and computational aspects*, *Journal of Telecommunications and Information Technology*, 2016, vol. 3, pp. 70–79.

[6] Nguyen T., Yang R., Azaria A., Kraus S., Tambe M., *Analyzing the effectiveness of adversary modeling in security games*, *Proceedings of the 27th AAAI Conference on Artificial Intelligence, AAAI 2013*, 2013, vol. 27, pp. 718–724, doi: 10.1609/aaai.v27i1.8599.

[7] Bielinis E., Janeczko E., Janeczko K., Bielinis L., *Effect of an illegal open dump in an urban forest on landscape appreciation*, *PLoS One*, 2020, vol. 17, no 11, doi: 10.20944/preprints202011.0326.v1.

VideoAI – System for Synchronization of Electronic Program Guides

Jan Wasilewski[1,2][0000−0002−1990−9279], **Bartosz Sochaj**[1], **Adam Gaca**[1]

[1]*OKE Software*
[2]*University of Gdansk*
Institute of Informatics, Faculty of Mathematics, Physics and Informatics,
Wita Stwosza 57, 80-308 Gdańsk, Poland

DOI:10.34658/9788366741928.37

Abstract. *Electronic Program Guides (EPG) do not contain commercials and are prepared before the real-time broadcast; therefore are not precise and sometimes shifted compared to real-time TV streams due to unexpected events such as football overtime. These inaccuracies in EPG may harm features often offered by TV stream providers, such as planned recordings. Here we present a system for mitigating this problem called VideoAI. First, the broadcast is analyzed, and commercials are detected by the first module using YOLO neural network. Next, the EPG is synchronized with real broadcast by minimizing the partition dissimilarity score. We evaluate VideoAI on different scenarios and show that it can adjust EPG to the real broadcast.*
Keywords: *Electronic Program Guides synchronization, detecting TV commercials, broadcast monitoring*

1. Introduction

Many features, such as planned recording, offered by TV stream providers rely on precise information about when a particular program begins. The approximate times of starting and ending points of programs are present in Electronic program guides (EPG). Differences in EPG and real broadcasts are caused by the absence of commercials in the former. Moreover, unexpected events, such as unplanned program extensions and the fact that EPGs are prepared in advance, not based on real-time TV streams, are the source of further inaccuracies. In this project, we develop VideoAI – an end-to-end pipeline to synchronize EPG and real-time broadcasts. Previous research [1, 2] relies on the time difference between the television and the EPG server. In contrast, our approach analyses real-time streams without needing any additional information from the EPG server. The system detects all the commercials in the broadcast by recognizing specific visual and audio

patterns by using machine learning methods [3]. The synchronization is done by the analysis of advertising breaks in broadcast. Apart from the accuracy and effectiveness of the analysis, a key element of this project is to achieve the shortest possible delay in EPG updating (e.g., no more than 30 minutes).

2. The developed process of TV stream analysis

VideoAI, the system for EPG synchronization, consists of two parts: an advertisement detector and a synchronization system. The stream is pre-analyzed to establish the dynamics of frames and the presence of black margins on the sides. Information from pre-analysis is processed in a preprocessing step, and the parts with a low rate of change, i.e., long static fragments and frames containing symmetric black margins on sites, are detected. These stream parts are highly likely to be parts of the movies and are excluded from further analysis. Preprocessed stream is passed to the commercial detection block consisting of three algorithms. Two heuristic detectors search for particular sequences of beginnings and ends of the programs and typical commercials, such as an analyzed TV channel commercial, in video and audio provided before. Using YOLO neural network, the third algorithm detects specific graphic forms, such as channel logos or credits. The results obtained from the aforementioned detectors are further post-processed to erase erroneous predictions. The synchronization part is based on the advertisement detector's results. Its goal is to synchronize the electronic program guide, i.e., to adjust EPG to real broadcast. The algorithm operates on the defined time horizon, synchronizing EPG based on the commercials detected by the first part within it. The time horizon is set to 10 hours. The synchronization turns on automatically every half an hour.

3. Commercials detection part

Let us present the pipeline of the commercial detection part of the VideoAI system. This part outputs the location of detected commercials in the real broadcast.

Film preprocessing. The recorded video stream is pre-analyzed to detect characteristic features in the recorded films to find fragments that the commercial detection block should analyze further. In particular, the significant changes between frames and the presence of black margins are detected. Long, static shots and parts of the broadcast with constant, symmetric black margins on the sides are assumed to be parts of the movie rather than a commercial, therefore, excluded from further processing.

Recognising commercials. This block employs machine learning models to recognize commercials. First, a set of characteristic elements in the stream, such as the channel logo and credentials, is defined, and the decision rules for commercial detection are established. The characteristic elements and decision rules vary by channel and are chosen in such a way, that makes it possible to mark the advertisement breaks reliably. For example, the channel logo is in commercials and is not usually found in programs. Therefore, the logo belongs to the characteristic elements' set, and one of the decision rules is: detected logo ⇒ commercial. This module requires a trained, custom model. VideoAI system provides the *training module* for manual annotation and model training. The YOLOv5 [4] neural network is trained using the gradient-based method on the annotated dataset. Apart from the trained model and the decision rules, the input data of this method comprise the broadcast, the time series specifying which frames are to be processed, and optionally the locations (i.e., the coordinates of the rectangle) in which elements belonging to different classes and types (program, commercial) may occur in the film frame. It outputs a time series with information about the recognized element class and frame classification.

Postprocessing. In post-processing, data obtained from recognizing commercials block is converted into a list of points in time at which commercials started or ended. For this purpose, all commercial/program transitions are selected, and anomalies such as excessively short breaks between commercials and excessively short commercials are removed.

4. Synchronisation part

Combinatorial EPG synchronization is an algorithm that adjusts EPG by analyzing real-time TV stream. It is based on the idea that despite not having precise timestamps separating programs and temporal shifts in EPG, the lengths of the programs are approximately correct. The algorithm uses information about commercials from the first part of the system. The commercials that are candidates for separating points are determined for every timestamp separating programs. The commercial is marked as a candidate if its center is closer to the timestamp than constant R, where R is the parameter denoting the maximum value of the broadcast shift. Next, the set of every possible combination of candidates for every timestamp is created, resulting in the set of divisions Π of the considered timeline on different programs. Then, the algorithm searches for the division π_0, which minimizes the partition dissimilarity score

$$\pi_0 = \arg\min_{\pi \in \Pi} \sum_{i}^{N} |\frac{l_i}{\hat{l}_i} - 1|, \tag{1}$$

where l_i and $\hat{l}_i$ are lengths of $i - th$ program obtained from EPG and scenario respectively, and N is the number of programs. This results in choosing such commercials to separate points that produce gaps between them with the most similar lengths to the lengths of movies in the electronic program guide. This approach mitigates problems with the shift between real broadcast and EPG.

5. Results

The experiments aim to evaluate the performance of the proposed synchronization algorithm on 240h of a stream from RTL 4 channel recorded between 4th and 14th September 2022. The first part detects commercials with 94% accuracy. Almost all the errors (93%) are caused by False Positive predictions. Program separating commercials are usually easier to detect due to higher length and frequent channel advertisements and were properly detected in 99.3% cases. During the synchronization part, the morning news panel turned out to be very problematic. In particular, programs RTL Nieuws and RTL Weer tend to have systematically overestimated lengths in EPG. We multiply the partition dissimilarity score of those programs by flexibility parameter α reducing the influence of differences between real length and EPG length. We achieved 83% synchronization accuracy with $\alpha = 1$ and 96% with $\alpha = 0.3$.

We now aim to evaluate the performance of the proposed synchronization algorithm on more diverse data. We generate three scenarios: no-shift, shift, and post-shift scenarios. In the first case, the broadcast approximately matches with

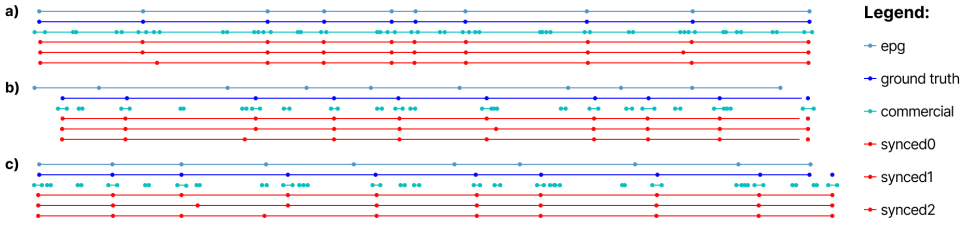


Figure 1. Results of synchronization algorithm on three scenarios: no-shift (a), post-shift (b), and shift (c). Every panel comprises six timelines: EPG, real broadcast, commercials, and three most probable synchronizations. Source: own work.

EPG. The second scenario relates to the situation when some programs were extended in the past, and the real broadcast was shifted compared to EPG. In the third case, the shift happens in the processing time window. Such a scenario simulates the situation when one of the programs is unexpectedly extended; for example, there will be extra time in a football match, or the length of commercials in the middle of a program is underestimated. The lengths of programs and commercials were generated from the uniform distributions $\mathcal{U}(20, 120)$ $\mathcal{U}(4, 10)$ respectively.

Table 1. Results of synchronization algorithm

Scenario	Accuracy
no-shift	0.99
post-shift	0.97
shift	0.91

The commercial block separates every two programs, and the number of commercials during the program was derived as $\lfloor length/30 \rfloor$. A thousand timelines with nine programs, EPGs, and commercials were generated for every scenario. The expected length of one timeline is 630 minutes, which is approximately equal to the synchronization window for this algorithm. Nine programs produce ten separating points which the algorithm aims to find. For every scenario, we run the synchronization on generated timelines and calculate the average accuracy, i.e., the number of correctly established separating points divided by the number of all separating points. The results of the synchronization are presented in Fig.1. Each panel consists of EPG, real broadcasts, commercials, and the three most probable results of the synchronization algorithm. The first panel shows the scenario when the broadcast shift did not occur. Panel b shows the post-shift scenario when the real broadcast is shifted by 20 minutes. The last panel presents the most challenging third scenario in which the third program is extended in the real-time broadcast. The results of simulations on every scenario are presented in table Tab.1.

6. Conclusion

We have introduced VideoAI, the system for EPG synchronization, consisting of the advertisement detector and synchronization module. We evaluated it in three scenarios: no-shift, post-shift, and shift, and demonstrated the synchronization capacity of our system. The further work includes but is not limited to the automation of the decision rules-building process in the commercial recognizing block by employing tree-based classifiers.

Acknowledgments

Publication financed by the National Centre for Research and Development under the Smart Growth Operational program, contract number POIR.01.01.01-00-0032/19. We would like to thank the referees for their comments, which helped improve this paper considerably.

References

[1] Huiyun Z., *Time synchronization method, device, intelligent television and computer readable storage medium*, 2018, CN109040820A, China, pub. 18-12-2018.

[2] Wu H., Sun Z., Zhou X., *Deep learning-based frame and timing synchronization for end-to-end communications*, *Journal of Physics: Conference Series*, 2019, vol. 1169, no 1, p. 012060, doi: 10.1088/1742-6596/1169/1/012060.

[3] Li M.e.a., *Cnn-based commercial detection in tv broadcasting*, ISBN 978-1-4503-5366-3, pp. 48–53, doi: 10.1145/3171592.3171619.

[4] Jocher G., Chaurasia A., Stoken A., Borovec J., NanoCode012, Kwon Y., Michael K., TaoXie, Fang J., imyhxy, Lorna, Yifu Z., Wong C., V A., Montes D., Wang Z., Fati C., Nadar J., Laughing, UnglvKitDe, Sonck V., tkianai, yxNONG, Skalski P., Hogan A., Nair D., Strobel M., Jain M., *ultralytics/yolov5: v7.0 – YOLOv5 SOTA Realtime Instance Segmentation*, 2022, doi: 10.5281/zenodo.7347926.

Chapter 4

Medical Applications of Artificial Intelligence

Domain Editors:

1. Włodzisław Duch, Nicolaus Copernicus University, Toruń.
2. Julian Szymański, Gdańsk University of Technology, Gdańsk.
3. Marian Bubak, Sano Centre for Computational Medicine, AGH and ACC Cyfronet, Kraków.

Identification of Melanocytic Skin Lesions Using Deep Learning Methods

Wiesław Paja[1[0000−0002−6446−036X]],
Jarosław Szkoła[1[0000−0002−6043−3313]],
Krzysztof Pancerz[3[0000−0002−5452−6310]],
Jaromir Sarzyński[1[0000−0002−2133−4187]],
Małgorzata Żychowska[2[0000−0001−8268−0529]]

[1]*University of Rzeszów, Institute of Computer Science*
Pigonia 1, 35-310 Rzeszów, Poland
{wpaja,jszkola,jsarzynski}@ur.edu.pl
[2]*University of Rzeszów, Institute of Medical Sciences*
Kopisto 2a, 35-959 Rzeszów, Poland
mzychowska@ur.edu.pl
[3]*The John Paul II Catholic University of Lublin, Institute of Philosophy*
Aleje Racławickie 14, 20-950 Lublin, Poland
kpancerz@kul.pl

DOI:10.34658/9788366741928.38

Abstract. *Detection of skin cancer at an early stage is a priority in the fight to reduce mortality. The aim of the paper is to develop a method of computer aided diagnosis of melanocytic skin lesions through analysis of dermatoscopic images using deep NN methods. In particular, the goal is to use the multiple binary CNN model approach. The results obtained are much better in distinguishing between categories of lesions compared to the model built on the entire 7-class image database.*
Keywords: *melanocytic skin lesions, computer aided diagnosis, convolutional neural network*

1. Introduction

Skin cancer is a type of cancer that develops in the skin cells. There are three main types of skin cancer: basal cell carcinoma, squamous cell carcinoma, and melanoma. The first one is the most common type of skin cancer and is usually caused by exposure to the sun. It is a slow-growing cancer that is rarely fatal, but can cause disfigurement if left untreated. The second type is a more serious type of cancer that can spread to other parts of the body if not treated early. It is

usually caused by long-term sun exposure or exposure to other sources of ultraviolet light. Melanoma, caused by the abnormal growth of pigment-producing cells (melanocytes) in the skin, is the most dangerous type of skin cancer and can spread quickly to other parts of the body.

Diagnosing skin cancer usually involves a physical examination of the skin and a biopsy of the suspicious area. However, AI algorithms have shown promising results in the field of dermatology and have the potential to improve the accuracy and speed of skin cancer diagnosis [1]. For example, deep neural networks can be trained to analyze dermatoscopic images of lesions and identify features that are indicative of skin cancer [2, 3, 4]. It can help to improve the accuracy of diagnosis, particularly in difficult cases where it may be challenging for a dermatologist to make a definitive diagnosis. It's important to note that AI algorithms for skin cancer diagnosis are still in the early stages of development and are not yet widely used in clinical practice. Additionally, AI algorithms should not be used as a substitute for a thorough physical examination by a dermatologist or other healthcare professional.

The sources of data for training deep learning algorithms for skin cancer diagnosis include large, publicly available datasets of dermatoscopic images, as well as private datasets collected by healthcare organizations and academic institutions. Some examples of publicly available datasets include the ISIC Archive (International Skin Imaging Collaboration) and the PH2 Dataset (Dermo-pediatrics Image Analysis Group). During research HAM10000 dataset from ISIC Archive was used.

2. The HAM10000 dataset

The HAM 10000 image collection is the largest publicly available collection of images of skin pigment lesions. It contains 10015 dermatoscopic images of lesions pertaining to 7 diagnostically relevant categories (see Table 1). This collection comes from different sources, acquired by different methods which provide a variety of images. More than half of them are confirmed by histopathological examination while the rest are confirmed by other means. This database is also heavily unbalanced, with 67% of cases from one *nv* category, typical of benign skin lesions.

3. Methods

3.1. Data augmentation

In order to solve the problem of an unbalanced dataset, which does not allow to significantly improve the quality of the model, it was decided to appropriately

modify the original input set in order to obtain a representative set. Since it was not possible to acquire images for the deficient classes, the technique of expansion by modification, called the augmentation technique, was applied. It means augmenting the set with new samples by appropriately transforming already available data. Various transformations are applied to images, i.e., reflections, rotations, cropping, changing the color model, and others. Rotation operations were applied to the current base by an appropriate angle with respect to the input image (Fig. 1). The

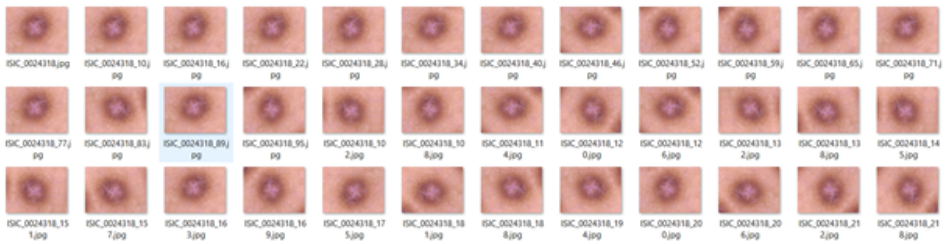


Figure 1. The sample set of rotated images. Source: own work.

number of generated images was set so as to obtain a fully balanced set, in which each decision class has a similar number of images. In total, the database contains 46543 files, that is 4.6 times more than samples (Table 1).

Table 1. Number of cases in categories before and after augmentation.

Id	Type	Acronym	Before	After
0	Actinic keratosis / Bowen's disease	akiec	327	6540
1	Basal cell carcinoma	bcc	514	6682
2	Benign keratosis	bkl	1099	6594
3	Dermatofibroma	df	115	6670
4	Melanocytic nevus	nv	6705	6705
5	Vascular lesion	vasc	142	6674
6	Melanoma	mel	1113	6678

3.2. Model architecture

The architecture of the entire neural network was designed as two convolutional subnetworks with two independent data outputs and a decision-making system in the form of a voting mechanism [5]. Data from the individual outputs of each network are fed to the inputs of the voting algorithm. The general architecture of our approach is shown in Fig. 2. Each model returns the probability for each of the possible decisions, and then the voting algorithm calculate the mean value for every class and approves the class with the highest mean value (see Fig. 3). The

presented model is based on observation that the network classifies well-known patterns, other data are not classified well. By linking data into unique disjoint sets, with two pairs of classes, the data sub-model, that will recognize a maximum of one class from the input dataset, could be created. Due to the fact that each of the sub-models contains one class shared with another sub-model (see Fig. 2), the recognition of data matching the pattern is unambiguous.

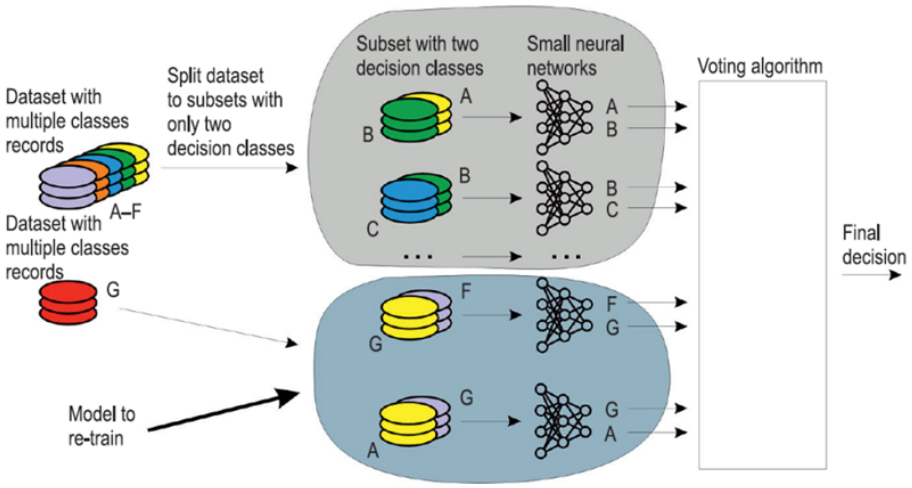


Figure 2. The architecture of the multiple binary CNN model. Source: own work.

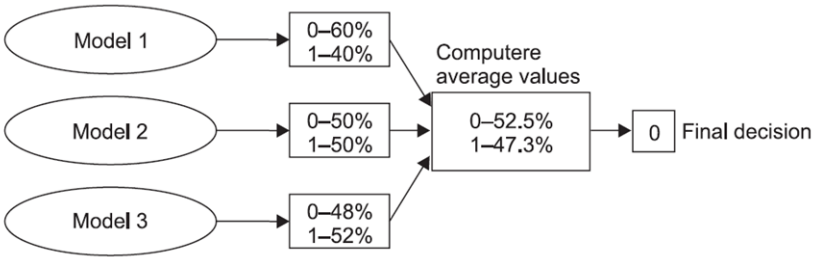


Figure 3. The voting procedure. Source: own work.

4. Results

For the HAM10000 dataset, the following division of decision classes was made, for each neural subnetwork, see Table 2. The learning process was implemented on input data, divided into the training and validation part, in a ratio of 80/20%.

Table 2. The division of decision-making classes and the accuracy of the model for the training and validation sets.

Subnetwork	Class pair	Training	Validation
SN1	akiec & bcc	0.97	0.89
SN2	bcc & bkl	0.97	0.89
SN3	bkl & df	0.98	0.93
SN4	df & nv	0.98	0.97
SN5	nv & vasc	0.98	0.97
SN6	vasc & mel	0.98	0.97
SN7	mel & akiec	0.96	0.91

5. Conclusions

The presented network architecture has many features that can be useful for analyzing multi-class data, such as images. Based on the course of the learning process for individual subnetworks, it can be seen that the learning efficiency for a pair of classes is very high, with the vast majority above 90%.

- Models that have had the same class as the class of the test image within a pair of learning classes show very high activity on one of the outputs, while the other output returns a very small value of it.
- Models for which data does not fit model at all, return conflicting data.
- Models that return inconsistent classifications for a given input image, but other than the correct label assigned to the image, indicate that a given image is difficult to classify unambiguously, and fits into several different classes.

Acknowledgment

This work was supported by the Subcarpatian Center for Innovation in Rzeszów, grant no. N3-471 titled: “Intelligent identification of melanocytic skin lesions using machine learning algorithms”.

References

[1] Cudek P., Paja W., Wrzesień M., *Automatic system for classification of melanocytic skin lesions based on images recognition*, *Man-Machine Interactions 2*, 2011, vol. 103, pp. 189–196, doi: https://doi.org/10.1007/978-3-642-23169-8_21.

[2] Esteva A., Kuprel B., Novoa R.A., Ko J., Swetter S.M., Blau H.M., Thrun S., *Dermatologist-level classification of skin cancer with deep neural networks*, *Nature*, 2017, vol. 7639, no 542, p. 115–118, doi: https://doi.org/10.1038/nature21056.

[3] Xia M., Kheterpal M.K., Wong S.C., Park C., Ratliff W., Carin L., Henao R., *Lesion identification and malignancy prediction from clinical dermatological images*, *Scientific Reports*, 2022, , no 12, p. 15836, doi: https://doi.org/10.1038/s41598-022-20168-w.

[4] Haenssle H.A., Fink C., Schneiderbauer R., Toberer F., Buhl T., Blum A., Kalloo A., Hassen A.B.H., Thomas L., Enk A., Uhlmann L., *Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists*, *Annals of Oncology*, 2018, vol. 29, no 8, p. 1836–1842, doi: https://doi.org/10.1093/annonc/mdy166.

[5] Szkoła J., *Multiclass voice commands classification with multiple binary convolution neural networks*, *Technical Sciences*, 2022, , no 25, p. 149–170, doi: https://doi.org/10.31648/ts.8098.

Loss Function Influence on Uncertainty Estimation for White Matter Lesions 3D Segmentation in a Shifted Domain Setting

Marta Kaczmarska[0000−0000−0000−0000],
Karol Majek[0000−0002−1351−8496]

Cufix, 05-825 Grodzisk Mazowiecki, Poland

DOI:10.34658/9788366741928.39

Abstract. *The aim of this study is to address the problem of distributional shift for white matter Multiple Sclerosis lesion segmentation models. The impact of loss function on models performance and uncertainty estimation is evaluated. The evaluation is performed on two in-domain and one out-of-domain dataset consisting of 3D FLAIR Magnetic Resonance images. Our experiments show that application of segmentation losses (eg. Dice) translate into reduced models robustness and poorer uncertainty estimation compared with classification losses (eg. CE). The source code is publicly available*[1].

Keywords: *White Matter Multiple Sclerosis Lesions, Multiple Sclerosis, 3D Segmentation, Magnetic Resonance Imaging*

1. Introduction and related work

Multiple Sclerosis (MS) is a disease of the central nervous system that manifests by the presence of White Matter Lesions (WML). Magnetic Resonance Imaging (MRI) is an important tool in MS diagnosis as well as prognosis and therapy monitoring [1]. Analyzing WMLs, ie. their number and size, plays a crucial role in these procedures [2]. The development of automated WML segmentation models is limited by the low availability of medical images. Furthermore, variability within images collected from different: medical centers, MRI scanners, population (including different MS stages) represents distributional shift. The shift between training and real-world data causes decrease in models performance and increase in segmentation uncertainty [3].

In this study we experimentally compare the influence of loss function on uncertainty estimation in a domain shift scenario. The application of nested UNet with efficient attention for the task of WML segmentation is investigated as well.

[1] https://github.com/deepdrivepl/shifts

2. Methodology

We use data prepared by neurologists participating in the framework of Observatoire Français de la Sclérose en Plaques (OFSEP), the French MS registry [4]. They collect clinical data prospectively in the European Database for Multiple Sclerosis (EDMUS) software [5]. MRI of patients were provided as part of a care protocol. Nominative data are deleted from MRI before transfer and storage on the Shanoir platform [2].

Table 1. Mean nDSC for evaluation subsets for models trained with different loss functions.

		nDSC ↑		
Loss	**threshold**	**dev_in**	**eval_in**	**dev_out**
Dice	0.20	**0.68** (±0.10)	0.70 (±0.10)	0.59 (±0.17)
Focal	0.40	0.48 (±0.15)	0.57 (±0.18)	0.47 (±0.17)
Dice + Focal	0.35	**0.68** (±0.11)	0.72 (±0.11)	0.60 (±0.13)
CE	0.30	0.66 (±0.13)	**0.73** (±0.12)	0.61 (±0.13)
BCE	0.35	**0.68** (±0.11)	**0.73** (±0.10)	0.61 (±0.13)
Gen. Dice + Focal	0.05	0.66 (±0.12)	0.71 (±0.11)	0.59 (±0.17)
nDSC	0.35	**0.68** (±0.10)	0.72 (±0.13)	0.61 (±0.14)
nDSC + Focal	0.40	0.65 (±0.11)	0.72 (±0.11)	0.57 (±0.14)
nDSC + Focal + CE	0.40	0.66 (±0.11)	**0.73** (±0.11)	0.60 (±0.11)
Log-Cosh Dice	0.05	**0.68** (±0.10)	0.70 (±0.11)	**0.62** (±0.13)
Tversky	0.05	0.66 (±0.09)	0.68 (±0.14)	0.55 (±0.17)

The dataset is provided within Shifts Challenge 2022[3] and includes Fluid Attenuated Inversion Recovery (FLAIR) and T1-weighted MRI images divided into four subsets: *train*, *dev_in*, *eval_in* and *dev_out*. *dev_in* and *eval_in* serve as in-domain validation sets, while *dev_out* constitutes a test set that contains data with a distributional shift compared to the rest of the subsets. The shared data have already undergone the following preprocessing: denoising, skull stripping (brain mask is calculated from the T1 images registered to FLAIR space), bias field correction and interpolation to the 1 mm isovoxel space. A more extensive dataset description can be found in [3].

The main segmentation metric in this study is normalized Dice Similarity Coefficient (nDSC) [3]. The uncertainty estimation is performed on the voxel-scale by constructing nDSC error Retention Curves (RC). This approach captures the relation between the uncertainty measure and the model errors in segmentation. The following uncertainty measures were evaluated: mutual information, expected

[2]Sharing NeurOImagingResources, https://shanoir.org
[3]https://shifts.grand-challenge.org/

pair-wise KL divergence, reverse mutual information, expected entropy, entropy of expected and negated confidence [6]. The area under the error RC (R-AUC) is a metric that assesses both model's robustness to distributional shift and uncertainty quality. As the main measure for comparing R-AUC between the subsets we chose entropy of expected, because it produced one of the lowest R-AUCs and it captures both data and knowledge uncertainty [6].

Table 2. Mean R-AUC for evaluation subsets for models trained with different loss functions. The values were calculated considering entropy of expected as an uncertainty measure.

	$100 \cdot$ R-AUC ↑		
Loss	**dev_in**	**eval_in**	**dev_out**
Dice	63.10 (±16.79)	66.33 (±13.31)	61.72 (±12.86)
Focal	98.11 (± 1.18)	98.55 (± 1.10)	97.43 (± 1.34)
Dice + Focal	97.36 (± 2.31)	98.74 (± 1.36)	98.28 (± 1.21)
CE	**99.10** (± 0.60)	**99.48** (± 0.49)	98.86 (± 0.52)
BCE	99.05 (± 0.70)	99.33 (±0.68)	**98.88** (±0.68)
Gen. Dice + Focal	63.94 (±16.64)	71.18 (±14.42)	65.00 (±13.65)
nDSC	98.83 (± 1.00)	99.33 (± 0.82)	98.75 (± 1.06)
nDSC + Focal	99.05 (± 0.62)	99.39 (± 0.64)	98.33 (± 1.99)
nDSC + Focal + CE	99.09 (± 0.62)	99.36 (± 0.63)	98.64 (± 1.20)
Log-Cosh Dice	64.34 (±16.21)	68.24 (±13.56)	63.97 (±12.65)
Tversky	64.69 (±17.20)	69.48 (±14.74)	62.22 (±13.89)

3. Experiments and results

In all experiments, we used a nested UNet with attention XUnet[4]. We evaluated following loss functions: Dice, Focal, weighted sum of Dice and Focal with weights 0.5 and 2.0 respectively (Dice + Focal), Cross Entropy (CE), Binary Cross Entropy (BCE), average of Generalized Diceand Focal (Gen. Dice + Focal), nDSC, weighted sum of nDSC and Focal with weights 0.01 and 1.0 respectively (nDSC + Focal), weighted sum of nDSC, CE and Focal with weights 0.01, 0.3 and 1.0 respectively (nDSC + Focal + CE), Log-Cosh Diceand Tversky loss.

The model's input is 3D FLAIR image divided into smaller patches of size 64×64×64. For training random patches were extracted from the volume, while for inference sliding window method was used and the patches were aggregated using Gaussian-weighted averaging. Binary WML segmentation masks are obtained by thresholding the model's output at a value that maximizes nDSC on *eval_in* subset.

[4]https://github.com/lucidrains/x-unet

Each model was trained for 100 epochs with RAdam optimizer and batch size of 6. One Cycle learning rate policywith initial 10^{-5}, maximum 10^{-4} at second epoch and final learning rate 3.3×10^{-8} was chosen for Focal, CE, nDSC, nDSC CE, nDSC CE Focal. For the rest of losses initial 10^{-4}, maximum 10^{-3} at second epoch and final learning rate 3.3×10^{-6} were used.

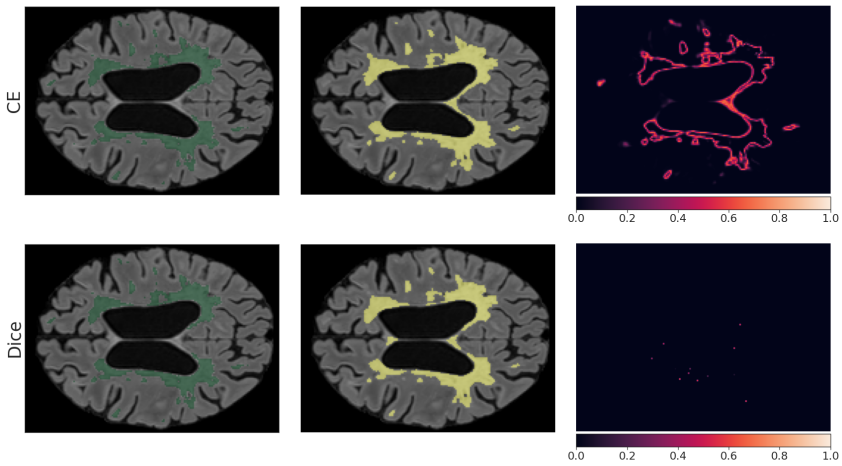


Figure 1. An example from *dev_out* subset; True (green), Predicted (yellow) WML segmentation, and entropy of expected uncertainty maps for models trained with Dice and CE loss. Source: own work.

Data augmentation was applied during training, including random intensity shifting and scaling, random crop, flip, rotation, zoom. The voxels intensity range was normalized before feeding the patch to the model. We calculated nDSC for each study and took the average across all studies for each evaluation subset. The mean nDSC, R-AUC values are shown in Tables 1, 2. Then, we compared area under the RCs constructed with different uncertainty measures for each subset. Exemplary true and predicted WML segmentations with entropy of expected uncertainty maps for models from both loss categories are shown in Fig. 1.

4. Conclusions

We evaluated the models' segmentation performance with nDSC metric and estimated uncertainty with error retention curves. All models achieved comparable segmentation results on evaluation sets with noticeable nDSC drop on out-of-domain *dev_out* set. Models trained with typical segmentation losses, like Dice or Tversky, showed higher uncertainty compared to the ones trained with classi-

fication losses, like CE or Focal. However, with nDSC used as loss function, the uncertainty is similar to classification losses. We also observe a rise in uncertainty, ie. drop in R-AUC values, on *dev_out* set with Focal, Dice + Focal, CE, BCE, nDSC, nDSC + Focal and nDSC + Focal + CE being the most robust ones.

Since WML are characteristic not only to MS [7], the future promising direction is to investigate if WML are predictable at the comparable level of uncertainty in other conditions like stroke or cerebrovascular disease. The proposed experiments could also study if similar correlation between loss function and uncertainty occurs in other diseases or it is specific to MS, in particular the challenge dataset.

Acknowledgment

This work was carried out in collaboration with The Observatoire Français de la Sclérose en Plaques (OFSEP), who is supported by a grant provided by the French State and handled by the"Agence Nationale de la Recherche," within the framework of the "Investments for the Future" program, under the reference ANR-10-COHO-002, by the Eugène Devic EDMUS Foundation against multiple sclerosis and by the ARSEP Foundation.

References

[1] Polman C.H., Reingold S.C., Banwell B., Clanet M., Cohen J.A., Filippi M., Fujihara K., Havrdova E., Hutchinson M., Kappos L., et al., *Diagnostic criteria for multiple sclerosis: 2010 revisions to the McDonald criteria*, *Annals of neurology*, 2011, vol. 69, no 2, pp. 292–302.

[2] Ghosh S., Huo M., Shawkat M.S.A., McCalla S., *Using convolutional encoder networks to determine the optimal magnetic resonance image for the automatic segmentation of multiple sclerosis*, *Applied Sciences*, 2021, vol. 11, no 18, p. 8335.

[3] Malinin A., Athanasopoulos A., Barakovic M., Cuadra M.B., Gales M.J., Granziera C., Graziani M., Kartashev N., Kyriakopoulos K., Lu P.J., et al., *Shifts 2.0: Extending the dataset of real distributional shifts*, *arXiv preprint arXiv:2206.15407*, 2022.

[4] Vukusic S., Casey R., Rollot F., Brochet B., Pelletier J., Laplaud D.A., De Seze J., Cotton F., Moreau T., Stankoff B., et al., *Observatoire français de la sclérose en plaques (ofsep): A unique multimodal nationwide ms registry in france*, *Multiple Sclerosis Journal*, 2020, vol. 26, no 1, pp. 118–122.

[5] Confavreux C., Compston D.A., Hommes O.R., McDonald W.I., Thompson A.J., *Edmus, a european database for multiple sclerosis.*, *Journal of Neurology, Neurosurgery & Psychiatry*, 1992, vol. 55, no 8, pp. 671–676, ISSN 0022-3050, doi: 10.1136/jnnp.55.8.671.

[6] Molchanova N., Raina V., Malinin A., La Rosa F., Muller H., Gales M., Granziera C., Graziani M., Cuadra M.B., *Novel structural-scale uncertainty measures and error retention curves: application to multiple sclerosis*, *arXiv preprint arXiv:2211.04825*, 2022.

[7] Manjón J.V., Coupé P., Raniga P., Xia Y., Desmond P., Fripp J., Salvado O., *Mri white matter lesion segmentation using an ensemble of neural networks and overcomplete patch-based voting*, *Computerized Medical Imaging and Graphics*, 2018, vol. 69, pp. 43–51, doi: 10.1016/j.compmedimag.2018.05.001.

Multi-task Learning for Classification, Segmentation, Reconstruction, and Detection on Chest CT Scans

Weronika Hryniewska-Guzik[1[0000−0003−2903−6050]], **Maria Kędzierska**[1], **Przemysław Biecek**[1,2[0000−0001−8423−1823]]

[1]*Warsaw University of Technology*
Faculty of Mathematics and Information Science
Koszykowa 75, 00-662 Warsaw, Poland
{weronika.hryniewska.dokt, przemyslaw.biecek}@pw.edu.pl
[2]*University of Warsaw*
Faculty of Mathematics, Informatics, and Mechanics
Stefana Banacha 2, 02-097 Warsaw, Poland

DOI:10.34658/9788366741928.40

Abstract. *Lung cancer and COVID-19 have one of the highest morbidity and mortality rates in the world. For physicians, the identification of lesions is difficult in the early stages of the disease and time-consuming. Therefore, multi-task learning is an approach to extracting important features, such as lesions, from small amounts of medical data because it learns to generalize better. We propose a novel multi-task framework for classification, segmentation, reconstruction, and detection. To the best of our knowledge, we are the first ones who added detection to the multi-task solution. Additionally, we checked the possibility of using two different backbones and different loss functions in the segmentation task.*
Keywords: *Multi-task learning, Computed tomography, detection.*

1. Introduction

The recent worldwide high contagiousness of the COVID-19 virus has stressed the importance of tools that support physicians' work. However, not only COVID-19 is the reason why such tools are important. Among cancers, lung cancer has one of the highest morbidity and mortality [1]. In the early stages of cancer, due to mild symptoms, it is usually difficult to diagnose [2]. Moreover, physicians are overloaded with work, and the identification of lesions is very time-consuming.

Computer-aided diagnosis (CAD) systems are designed to assist physicians in interpreting medical images and have to provide the highest possible precision and recall in indicating lesions [3]. For this reason, deep learning models seem to be

a good solution that meets these requirements. However, the need for responsible solutions that learn correct image features and do not overfit to the data led to multi-task learning.

Multi-task learning solutions are an approach to extracting important features even from a small amount of training data, which is common in medical cases. It is a type of learning algorithm that combines information from different tasks (auxiliary tasks) in order to improve the ability to generalize the main task better. In the hard parameter sharing approach, multi-task solutions share some layers and parameters between all the tasks [4].

Various solutions that use multi-tasking for lung medical data have already been developed [4, 5, 2]. Amyar et al. [4] created a framework based on the VGG-13 backbone that solved the classification, segmentation, and reconstruction problem. However, after analyzing the publicly available datasets used in that work and the lack of preprocessing, we can say with a high degree of probability that their model fitted too closely to the selected datasets.

This paper proposes a novel multi-task framework for classification, segmentation, reconstruction, and detection. We are the first ones who show that it is possible to add detection. Additionally, we checked the possibility of using a different backbone – ResNet-50 and altered the loss function in the segmentation task. In our solution, we showed that multi-task solution can be extended to new tasks.

2. Multi-task model training on CT chest scans

2.1. Data and preprocessing

For training and evaluation, the following datasets were used:

- 1816 images for classification and reconstruction: non-COVID patients from MedSeg [6], UCSD-AI4H [7]; COVID-19 patients from UCSD-AI4H [7]; cancer patients from Lung-PET-CT-Dx [8],
- 472 images for segmentation and reconstruction: MedSeg [6] (only images with masks for COVID lesions),
- 99 images for detection and reconstruction: MedSeg [6] Image masks have 3 possible COVID lesions: ground-glass opacity, consolidation, and pleural effusion.

Due to the fact that, in selected CT datasets, not all images were in 3D, we decided to use slices from CT scans, that is, 2D images. In order to unify the images, we equalized their histograms and rescaled them with their masks to a size of 256x256. We scaled the pixels to take values in the range $[0, 1]$. Then, images were split into training, validation and testing sets according to Table 1.

Table 1: Data split into training, validation and testing set.

Tasks	Train	Valid	Test
classification & reconstruction (CR)	1331	244	241
segmentation & reconstruction (SR)	377	48	47
detection & reconstruction (DR)	79	10	10

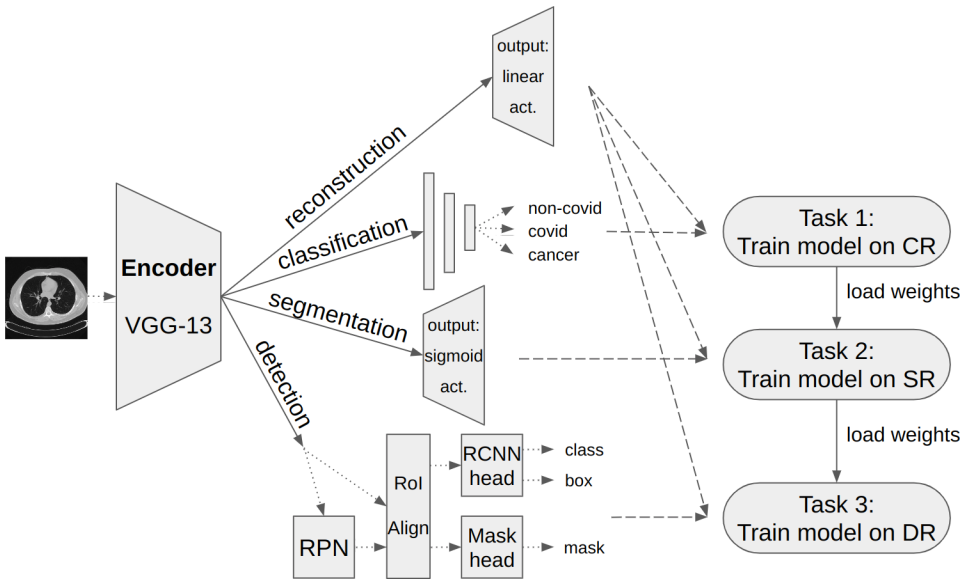


Figure 1: Multi-task architecture and training diagram for tasks: classification (C), segmentation (S), reconstruction (R), and detection (D). Source: own work.

2.2. Multi-task architecture

The proposed multi-task learning architecture is based on 4 tasks: classification, segmentation, reconstruction and detection. As presented in Fig. 1, the architecture is based on U-net [9] architecture, thus, the shared encoder is a VGG-13 neural network.

An encoder takes images in size 246x256x1. The reconstruction decoder is the second half of U-net has changed the activation function on the output layer to linear activation. The segmentation decoder is in the form of the standard second half of U-net, which means that it has a sigmoid activation function on the output layer. The classification decoder consists of 3 fully connected layers with softmax activation on the output layer. Mask-RCNN [10] returns bounding boxes and masks for detected objects.

$$\mathcal{L}_{total} = w_1 \cdot \mathcal{L}_{classif} + w_2 \cdot \mathcal{L}_{segm} + w_3 \cdot \mathcal{L}_{recon} + w_4 \cdot \mathcal{L}_{detect}. \quad (1)$$

The final loss function is a sum of weighted losses for specific tasks (Equation 1): categorical cross-entropy loss, generalized Dice loss, mean squared error, and mask R-CNN losses. Generalized Dice loss [11] in the segmentation task takes into account the unbalanced area of the lesion relative to the area of the entire image, therefore, providing better training results. Weights w_i in our case are $\{0, 1\}$. The mask R-CNN losses is a sum of the following losses:

$$\mathcal{L}_{detect} = \mathcal{L}_{MRCNNclassif} + \mathcal{L}_{MRCNNbbox} + \mathcal{L}_{MRCNNmask}. \tag{2}$$

2.3. Model training and results

The training procedure was divided into 3 steps, shown in Fig. 1. Firstly, the model was trained on image reconstruction and multiclass classification tasks. In order to verify whether the reconstruction task was unnecessary, the model performed only the classification task. The results of training two tasks simultaneously, presented in Table 2, were slightly better than the results of training classification only.

Table 2: Performance metrics of the model trained concurrently on two tasks: classification & reconstruction and on the only classification task.

Task	Accuracy	Macro F1	F1 non-COVID	F1 COVID-19	F1 cancer
Classification & reconstruction	0.89	0.91	0.88	0.90	0.97
Only classification	0.87	0.89	0.83	0.89	0.97

Secondly, the model was given the following tasks: segmentation of COVID-19 lesions and image reconstruction. This was done on a smaller dataset. In the first approach, the model had preloaded weights from the previous task, and in the second approach, the network was trained from scratch. The results of training for 700 epochs with preloaded weights, shown in Table 3 and Figure 2, were better.

Table 3: Performance metrics of the model trained on two tasks: segmentation & reconstruction with and without loading model's weights from the previous task: classification & reconstruction. IoU is an abbreviation for Intersection over Union.

Task	Accuracy	F1	Sensitivity	Specificity	Precision	ROC AUC	IoU
With loaded weights	0.99	0.78	0.76	0.99	0.80	0.88	0.64
Without loaded weights	0.99	0.75	0.75	0.99	0.76	0.87	0.60

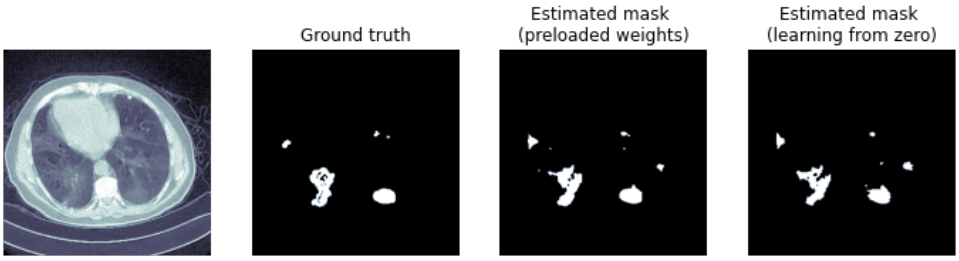


Figure 2: Masks generated by concurrent segmentation & reconstruction task with and without loading weights from classification task. Source: own work.

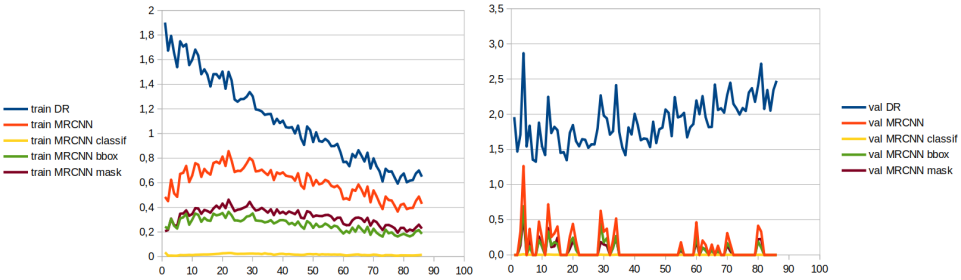
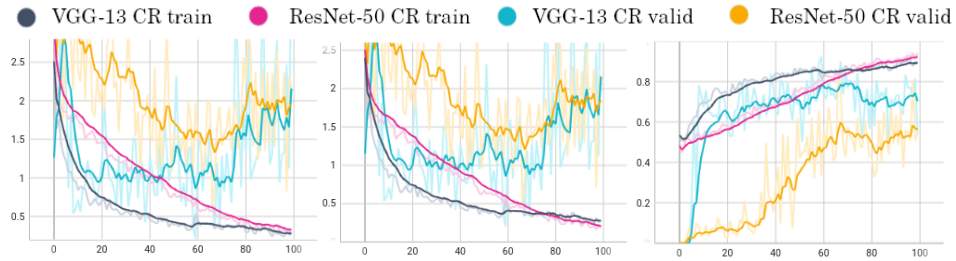


Figure 3: Evolution of the loss during training detection & reconstruction task as a function of epoch. Source: own work.

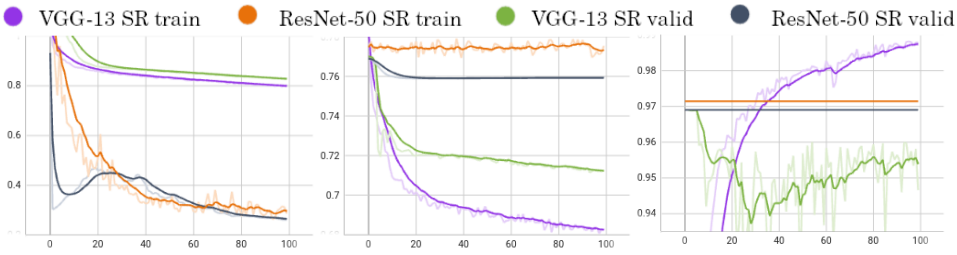
In order to combine previous networks with mask RCNN, the backbone of MaskRCNN was changed to VGG-13. Then, the weights from the best model in the segmentation & reconstruction task were loaded. However, due to the small training set, the network was overfitting from the beginning, presented in Figure 3. To overcome overfitting various data augmentation was applied, such as elastic transformation, rotating by a small angle, and cropping. Nonetheless, it did not help to obtain satisfactory results.

3. Evaluation on different backbone

We decided to evaluate whether the multi-task model obtains similar results on different backbones. Therefore, we change VGG-13 backbone to ResNet-50, which is the default backbone in the detection task. We trained a model for 100 epochs on two different tasks: classification & reconstruction and segmentation & reconstruction. The results in Fig. 4 show that there is no strong advantage of one backbone over another. Multi-task model loss is lower in classification & reconstruction when the backbone is VGG-13, while in segmentation & reconstruction, the multi-task model loss is lower for backbone ResNet-50.



(a) Loss for CR task (b) Classification loss in CR task (c) Classification accuracy in CR task



(d) Loss for SR task (e) Segmentation loss in SR task (f) Segmentation accuracy in SR task.

Figure 4: Using two different backbones: VGG-13 and ResNet-50 for training classification & reconstruction (CR) and segmentation & reconstruction (SR) models. Source: own work.

4. Conclusions

The framework was successfully created and tested. Obtained classification and segmentation results are satisfactory, especially due to the fact that segmentation applies to small lesions, not whole lungs. However, much more data is needed to get desired results in the detection task, even in a multi-task approach.

Acknowledgment

This work was financially supported by the Polish National Center for Research and Development grant number INFOSTRATEG-I/0022/2021-00, and carried out with the support of the Laboratory of Bioinformatics and Computational Genomics and the High Performance Computing Center of the Faculty of Mathematics and Information Science, Warsaw University of Technology.

References

[1] Siegel R.L., Miller K.D., Jemal A., *Cancer statistics, 2019*, *A Cancer Journal for Clinicians*, 2019, vol. 69, no 1, pp. 7–34, doi: 10.3322/caac.21551.

[2] Zhai P., Tao Y., Chen H., Cai T., Li J., *Multi-task learning for Lung Nodule classification on Chest CT*, *IEEE Access*, 2020, vol. 8, pp. 180317–180327, doi: 10.1109/ACCESS.2020.3027812.

[3] Monkam P., Qi S., Ma H., Gao W., Yao Y., Qian W., *Detection and classification of pulmonary nodules using convolutional neural networks: A survey*, *IEEE Access*, 2019, vol. 7, pp. 78075–78091, doi: 10.1109/ACCESS.2019.2920980.

[4] Amyar A., Modzelewski R., Li H., Ruan S., *Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation*, *Computers in Biology and Medicine*, 2020, vol. 126, p. 104037, doi: 10.1016/j.compbiomed.2020.104037.

[5] Li J., Zhao G., Tao Y., Zhai P., Chen H., He H., Cai T., *Multi-task contrastive learning for automatic CT and X-ray diagnosis of COVID-19*, *Pattern Recognition*, 2021, vol. 114, p. 107848, doi: 10.1016/j.patcog.2021.107848.

[6] Jun M., Cheng G., Yixin W., Xingle A., Jiantao G., Ziqi Y., Minqing Z., Xin L., Xueyuan D., Shucheng C., et al., *COVID-19 CT Lung and Infection Segmentation Dataset*, 2020, doi: 10.5281/zenodo.3757476.

[7] Zhao J., Zhang Y., He X., Xie P., *COVID-CT-Dataset: a CT scan dataset about COVID-19*, *arXiv*, 2020.

[8] Li P., Wang S., Li T., Lu J., HuangFu Y., Wang D., *A Large-Scale CT and PET/CT Dataset for Lung Cancer Diagnosis*, *The Cancer Imaging Archive*, 2020, doi: 10.7937/TCIA.2020.NNC2-0461.

[9] Ronneberger O., Fischer P., Brox T., *U-net: Convolutional networks for biomedical image segmentation*, [In:] *MICCAI*, Springer International Publishing, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[10] He K., Gkioxari G., Dollar P., Girshick R., *Mask R-CNN*, [In:] *Proceedings of the IEEE ICCV*, doi: 10.1109/ICCV.2017.322.

[11] Sudre C.H., Li W., Vercauteren T., Ourselin S., Jorge Cardoso M., *Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations*, *Lecture Notes in Computer Science*, 2017, p. 240–248, doi: 10.1007/978-3-319-67558-9_28.

Supporting Surgical Training with the Help of Computer Vision and Machine Learning Methods

Paweł Forczmański[1][0000−0002−3618−9146], **C. Yoonhee Ryder**[2], **Nicole M. Mott**[2], **Christopher L. Gross**[3], **B. Joon Yu**[4], **Deborah M. Rooney**[5], **David R. Jeffcoach**[6], **Serena Bidwell**[2], **Chioma Anidi**[2], **Lindsay Rosenthal**[2], **Grace J. Kim**[7],

[1]*West Pomeranian University of Technology, Szczecin*
Żołnierska Str. 49, 71-210 Szczecin, Poland
pawel.forczmanski@zut.edu.pl
[2]*University of Michigan Medical School, Ann Arbor, MI, USA*
[3]*University of Florida College of Medicine, Gainesville, FL, USA*
[4]*Abstract Partners, New York, NY, USA*
[5]*Dept. of Learning Sciences, University of Michigan, Ann Arbor, MI, USA*
[6]*Dept. of Surgery, Soddo Christian Hospital, Soddo, Ethiopia*
[7]*Dept. of Surgery, University of Michigan, Ann Arbor, MI, USA*

DOI:10.34658/9788366741928.41

Abstract. *The paper presents a novel concept of laparoscopic skills evaluation based on the automated analysis of videos recorded during simulation-based training exercises via an artificial intelligence algorithm. It has been tested on data collected during the training of actual surgeons. Its performance is promising, providing an opportunity to build an automatic system used mainly in developing countries.*
Keywords: *surgical training, skills evaluation, laparoscopic intervention, computer vision, machine learning, object detection, classification*

1. Introduction

Surgical skills training is founded on principles of deliberate and frequent practice and ongoing assessment of specified psychomotor skills. Such a process requires substantial human involvement to perform an assessment. The possibility of using an automated expert system to assess operative skills has the potential to reduce such overhead, mitigating the limiting effect of poor access to high-level surgical specialists, especially in low- and middle-income countries (LMICs). Machine learning has been used for surgical education for several years, applied

mainly to surgical decision-making and assessment of technical skills [1, 2]. Typically, such expert systems work on sequences of images taken during actual interventions or kinematic data gathered from simulations [1, 3] to estimate the general skill level of the operator, identify atomic movement patterns [4], detect critical errors [5], and even locate blood within the surgical field [6]. Many systems' measures are founded on global surgical performance assessment tools, such as Objective Structured Assessment of Technical Skills (OSATS) scores [2, 3]. Automated calculated motion metrics have been used for surgical assessment [2, 7], including discrete cosine transform, discrete Fourier transform, motion texture (e.g. frame kernel matrices) [8], sequential motion texture, augmented bag-of-words, and entropy-based features [7, 9]. However, to our knowledge, only two studies have used estimated motion metrics on laparoscopic surgical videos. These studies provided indirect assessment using variations of convolutional neural networks [10, 11]. Two studies have used kinematic data on intracorporeal suturing [12] and knot tying [13] within a laparoscopic box trainer to predict OSATS scores with an accuracy ranging from 59-70%. Hence, this is the first study to predict OSATS scores from a laparoscopic model or actual surgery purely from video data.

2. Method description

2.1. Motivation

The proposed method is a part of a larger project realized by ALL-SAFE, a global collaboration of surgeons and education researchers [14]. In this study, a low-cost laparoscopic training system was developed, intended to be used in LMICs to support the learning and assessment of cognitive and psychomotor skills of surgeons in training. The first series of experiments were conducted on the laparoscopic treatment of the ectopic pregnancy module, given its morbidity and mortality implications [15]. Learners completed the web-based scenario, reviewed expert demonstration videos, practised the associated laparoscopic skills in the trainer box, then recorded and uploaded their procedure using their personal phones. After the automated analysis of recorded learner videos, the system reports feedback in the format of predicted OSATS scores.

2.2. General overview

The algorithm assumes using several classifiers trained on manually tagged data. Exemplary frames extracted from all videos were used for laparoscopic instrument localization, while all available videos were tagged with standard global OSATS domains, namely Overall Performance, Flow of Operation, Economy of Time and Motion, Instrument Handling, and Respect for Tissue, each scored from 1.0 to 5.0 (Novice to Expert). Surgical instruments detection is performed using a

custom-trained YOLOv5 detector, while instrument tip detection uses a dedicated classifier (built upon pre-trained ImageNet layers). Movement patterns calculated for the tracked tools are normalized and filtered to fill gaps or values where the detector/classifier failed. Twenty-five mathematical correlates associated with the time and movement of the tools (both left and right) were chosen, including features calculated for Carthesian and polar representations of the tip's position, e.g., mean speed, jerk index, standard deviation of the area covered by the tools, path length, number of time segments with no movement, arc length etc.

2.3. Processing pipeline

Building the reference data (training the algorithm) consists of the following steps for all the reference videos.

1. Track instruments in an input video – for each frame from the video:
   (a) Perform YOLOv5 to detect instruments' bounding boxes;
   (b) Classify detected instruments and estimate tool's tip position;
   (c) Store the tool's tip position with information about tool's orientation.
2. Build movement sequence and extract movement characteristics
   (a) Collect each tool's tip position in a time series;
   (b) Calculate a feature vector for the left and right tools.
3. For each OSATS domain:
   (a) Select the most informative features using a simple exhaustive search for a subset giving the highest accuracy;
   (b) Store the most effective features combination.

The testing part contains steps (1)-(2) from the above algorithm, executed for a single test video, followed by the classification of the feature vector – evaluation of OSATS scores. In our work, we investigated several classical classifiers, namely k-Nearest Neighbours, Random Forest Classifier, Linear Discriminant Analysis, and simple Multi-Layer Perceptron. Considering the limited training data, k-NN was chosen for the final implementation.

3. Experiments

3.1. Dataset

Forty-seven surgical novices and experts contributed 74 unique laparoscopic ectopic pregnancy simulation videos. They were recruited from the residents

and attending surgeons of Pan-African Academy of Christian Surgeons (Soddo, Ethiopia), the resident and student body at Michigan Medicine (Ann Arbor, MI, USA) and the University of Florida (Gainesville, FL, USA) through the research recruiting platform UserInterviews (USA). The videos were graded by trained research coordinators using the OSATS to create ground truth values. All videos have the exact resolution (640x480 pixels) and framerate (25 frames per second).

3.2. Results

The experiments were performed using 5-fold cross-validation, ensuring all available files were taken for testing. As a result, the accuracy was calculated, as the frequency of videos for which OSATS scores were correctly predicted. The average accuracy of the five domains is 74.8%, which is quite good for such a small dataset. Confusion matrices show typical misclassification problems, i.e. they confirm that the less represented classes are often misclassified (see Tab. 1).

Table 1. Confusion matrices for all OSATS measures

	Eco. Ti. & Mot. Acc. 74 %					Instr. Handling Acc. 70 %					Resp. for Tissue Acc. 77 %					Flow of Operat. Acc. 76 %					Overall Perfor. Acc. 77 %				
Predicted / Actual	**1**	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**	**1**	**2**	**3**	**4**	**5**
1	0	2	1	0	0	3	4	1	0	0	4	3	0	0	0	0	1	0	0	0	3	2	0	0	0
2	0	39	3	0	0	2	41	0	0	0	2	22	3	2	0	0	39	1	1	0	1	47	1	0	0
3	0	5	11	1	0	2	6	5	0	0	1	7	14	2	0	0	5	10	2	0	0	7	7	0	0
4	0	5	1	5	0	1	5	0	3	0	0	4	1	7	0	0	4	3	7	0	1	2	2	0	0
5	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	0	0

4. Summary

This study demonstrated that computer-generated metrics from purely video input could predict OSATS scores for an entire procedure within a laparoscopic box trainer with an average complete accuracy of almost 75%. This novel work was the first application of AI for laparoscopic salpingostomy simulation-based training. It demonstrates the value of automated computer measures, which minimizes potential bias originating from manually tagged or annotated videos. This technology also has the potential to be transferrable to other procedures.

References

[1] Ward T.M., Mascagni P., Madani A., Padoy N., Perretta S., Hashimoto D.A., *Surgical data science and artificial intelligence for surgical education*, *Journal of Surgical Oncology*, 2021, vol. 124, no 2, pp. 221–230.

[2] Bilgic E., Gorgy A., Yang A., Cwintal M., Ranjbar H., Kahla K., Reddy D., Li K., Ozturk H., Zimmermann E., Quaiattini A., Abbasgholizadeh-Rahimi S., Poenaru D., Harley J.M., *Exploring the roles of artificial intelligence in surgical education: A scoping review*, *The American Journal of Surgery*, 2022, vol. 224, no 1, Part A, pp. 205–216.

[3] Lam K., Chen J., Wang Z., Iqbal F.M., Darzi A., Lo B., Purkayastha S., Kinross J., *Machine learning for technical skill assessment in surgery: a systematic review*, *npj Digital Medicine*, 2022, vol. 5, p. 24.

[4] Perumalla C., Kearse L., Peven M., Laufer S., Goll C., Wise B., Yang S., Pugh C., *Ai-based video segmentation: Procedural steps or basic maneuvers?*, *Journal of Surgical Research*, 2023, vol. 283, pp. 500–506.

[5] Nagaraj M., Namazi B., Sankaranarayanan G., Scott D., *Developing artificial intelligence models for medical student suturing and knot-tying video-based assessment and coaching*, *Surgical Endoscopy*, 2022, vol. 37.

[6] Sasaki S., Kitaguchi D., Takenaka S., Nakajima K., Sasaki K., Ogane T., Takeshita N., Gotohda N., Ito M., *Machine learning-based automatic evaluation of tissue handling skills in laparoscopic colorectal surgery: A retrospective experimental study*, *Annals of surgery*, 2022, vol. 278, no 2, pp. e250–e255.

[7] Zia A., Sharma Y., Bettadapura V., Sarin E., Ploetz T., Clements M., Essa I., *Automated video-based assessment of surgical skills for training and evaluation in medical schools*, *Int. Journal of Computer Assisted Radiology and Surgery*, 2016, vol. 11.

[8] Sharma Y., Ploetz T., Hammerla N., Mellor S., McNaney R., Olivier P., Deshmukh S., McCaskie A., Essa I., *Automated surgical osats prediction from videos*, [In:] *2014 IEEE 11th Int. Symposium on Biomedical Imaging, ISBI 2014*, pp. 461–464.

[9] Zia A., Sharma Y., Bettadapura V., Sarin E., Essa I., *Video and accelerometer-based motion analysis for automated surgical skills assessment*, *Int. Journal of Computer Assisted Radiology and Surgery*, 2017, vol. 13.

[10] Lavanchy J., Zindel J., Kirtac K., Twick I., Hosgor E., Candinas D., Beldi G., *Author correction: Automation of surgical skill assessment using a three-stage machine learning algorithm*, *Scientific Reports*, 2021, vol. 11, no 8933.

[11] Jin A., Yeung S., Jopling J., Krause J., Azagury D., Milstein A., Fei-Fei L., *Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks*, [In:] *2018 IEEE Winter Conf. on Applications of Computer Vision (WACV)*, pp. 691–699.

[12] Oquendo Y., Riddle E., Hiller D., Blinman T., Kuchenbecker K., *Automatically rating trainee skill at a pediatric laparoscopic suturing task*, *Surgical Endoscopy*, 2018, vol. 32, pp. 1–18.

[13] Kowalewski K.F., Garrow C., Schmidt M., Benner L., Müller B., Nickel F., *Sensor-based machine learning for workflow detection and as key to detect expert level in laparoscopic suturing and knot-tying*, *Surgical Endoscopy*, 2019, vol. 33.

[14] Rooney D.M., Mott N.M., Ryder C.Y., Snell M.J., Ngam B.N., Barnard M.L., Jeffcoach D.R., Kim G.J., *Evidence supporting performance measures of laparoscopic salpingostomy using novel low-cost ectopic pregnancy simulator*, *Global Surgical Education - Journal of the Association for Surgical Education*, 2022, vol. 1, no 1, p. 41.

[15] Goyaux N., Leke R., Keita N., Thonneau P., *Ectopic pregnancy in african developing countries*, *Acta obstetricia et gynecologica Scandinavica*, 2003, vol. 82, pp. 305–12.

Chapter 5

Natural Language Processing, Automatic Speech Recognition, Conversational AI, Uncertainty in Artificial Intelligence, Knowledge Engineering

Natural Language Processing, Automatic Speech Recognition, Conversational AI domain Editors:

1. Maciej Piasecki, Wrocław University of Science and Technology
2. Agnieszka Mykowiecka, Institute of Computer Science, Polish Academy of Sciences, Warsaw
3. Piotr Pęzik, University of Lodz

Uncertainty in Artificial Intelligence domain Editors:

1. Dominik Ślęzak, QED Software & University of Warsaw
2. Beata Zielosko, University of Silesia in Katowice
3. Piotr Wasilewski, Systems Research Institute, Polish Academy of Sciences

Knowledge Engineering domain Editors:

1. Agnieszka Ławrynowicz, Poznan University of Technology
2. Dariusz Krol, Wrocław University of Science and Technology
3. Grzegorz J. Nalepa, Jagiellonian University

A Convolutional and Recurrent Neural Network-based Approach for Speech Emotion Recognition

Piotr Duch[0000−0003−0656−1215], **Izabela Wiatrowska, Paweł Kapusta**[0000−0002−3527−7208]

[1]*Lodz University of Technology*
Institute of Applied Computer Science
Stefanowskiego 18, 90-537 Łódź, Poland
piotr.duch@p.lodz.pl
pawel.kapusta@p.lodz.pl

DOI:10.34658/9788366741928.42

Abstract. *Speech emotion recognition (SER) is a crucial aspect of human-computer interaction. In this article, we propose a deep learning approach, using CNN and RNN architectures, for SER using both convolutional and recurrent neural networks. We evaluated the approach on four audio datasets, including CREMA-D, RAVDESS, TESS, and EMOVO. Our experiments tested various feature sets and extraction settings to determine optimal features for SER. Our results demonstrate that the proposed approach achieves high accuracy rates and outperforms state-of-the-art algorithms.*
Keywords: *artificial intelligence, speech emotion recognition*

1. Introduction

Speech emotion recognition (SER) is a critical aspect of human-computer interaction, particularly as more interactions are based on spoken communication. Emotions are conveyed not only through posture, facial expressions and gestures but also through the tone, pitch, and other acoustic features of spoken language. However, recognizing emotions from speech patterns can be challenging due to the subjective nature of emotions, the difficulty of distinguishing between multiple emotions expressed in a single conversation, and the time-consuming process of collecting and classifying data. In this article, we investigate the application of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) in SER, which have potential applications in various fields, such as healthcare, psychology, criminal investigations, and customer service. The use of trained classifiers can help computers better understand human needs and respond appropriately,

particularly when the conversation context is essential. Overall, SER has the potential to improve the quality and effectiveness of human-computer interactions and contribute to our understanding of emotional expression and communication.

2. Datasets and methodology

In our research on speech emotion recognition, we have selected four databases: CREMA-D [1], RAVDESS [2], TESS [3], and EMOVO [4] to evaluate the proposed algorithm. The combination of these databases enables a comprehensive evaluation of speech emotion recognition with the ability to consider various emotions, cultures, genders, and languages.

In this study, we selected three features for sound transformation from the time domain to the frequency domain to extract features from audio files. The Mel spectrogram was chosen as the first feature due to its frequent use in deep learning and ability to transform frequencies comparable to how humans perceive sound differences expressed in Hertz. The second feature are the Mel frequency cepstral coefficients (MFCCs), which consist of 13 coefficients that capture the shape of the human vocal system and tone color. The last feature is the Chromagram, a pitch-based profile of 12 pitch classes that captures harmonic and melodic sound features, resistant to changes in tone color. For RNN, the extracted features were stacked, forming a single matrix that becomes the input to the network, while for CNN, each feature was sent separately to the network to enable the convolutional layers to learn specific weights for each feature. These three features enable a more comprehensive analysis of the emotional state of the speaker, which is particularly important in the context of the proposed deep learning models (see Fig. 1).

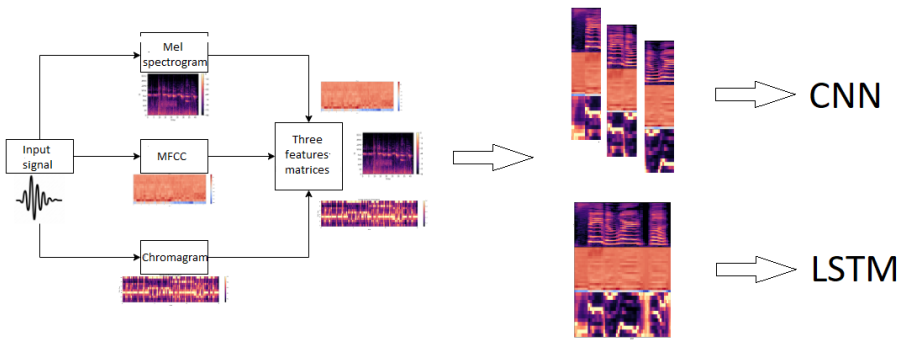


Figure 1. Architecture of the proposed algorithm. Source: own work.

During the training process, the input size had to be standardized. This was particularly important for the CNN where each input signal had to have the same

dimension. In contrast, the length of the signals could differ in the RNN. To ensure compatibility with the CNN, each sample was divided into fragments that overlapped by 25%. Furthermore, the datasets were augmented using three approaches: adding noise with an amplitude of 0.035 to the sample, slowing down the speed of speech, and lowering the pitch. The CNN architecture consisted of two convolutional layers with 128 and 64 filters for the Mel spectrogram and 64 and 32 filters for the other features, respectively. The results for all three features were then combined and flattened. Two fully connected layers with 64 and 6 neurons were used to complete the network. For RAVDESS and CREMA-D, the optimal neural network consisted of one additional convolutional layer for each feature with 32 filters. The RNN architecture consisted of two Long Short-Term Memory (LSTM) layers with 128 and 64 cells for RAVDESS, TESS, and EMOVO and three LSTM layers with 128, 64, and 64 cells for CREMA-D. Similar to CNN, the RNN also used two fully connected layers with 64 and 6 neurons to complete the network.

3. Results

In our study, we evaluated the performance of our proposed deep learning models for speech emotion recognition using four datasets: TESS, RAVDESS, CREMA-D, and EMOVO. During our experiments, we tested several different feature sets and feature extraction settings to determine the optimal features for speech emotion recognition. Our final results were very promising, with the model achieving satisfactory results on all tested datasets (Table 1).

Furthermore, we have compared the accuracy of our approach with several state-of-the-art methods using three different datasets: RAVDESS, EMOVO, and CREMA-D (Table 2). Our results indicate that our proposed approach has a higher accuracy rate than other algorithms in RAVDESS and EMOVO datasets and comparable accuracy in CREMA-D.

Table 1. The classification performance on chosen datasets using CNN and RNN.

Dataset	CNN	LSTM
TESS	100%	99%
RAVDESS	84%	77%
CREMA-D	64%	62%
EMOVO	87%	89%

Table 2. Accuracy comparison with existing SER algorithms

Method	Accuracy	Year
RAVDESS dataset		
DCNN [5]	71.6%	2020
Multimodal fine-grained learning [6]	74.7%	2020
Head Fusion [7]	77.4%	2020
BiLSTM [8]	82%	2020
Our approach – LSTM	77%	2023
Our approach – CNN	84%	2023
EMOVO dataset		
Multi-Level Local Binary and Ternary [9]	73.87%	2020
Mel frequency magnitude coefficient [10]	73.81%	2021
Twine shuffle pattern [11]	79.08%	2021
Statistical Feature Extraction for Deep SER [12]	83.9%	2022
Our approach – LSTM	89%	2023
Our approach – CNN	87%	2023
CREMA-D dataset		
SE-ResNet + GhostVLAD layer + emotion constrain [13]	64.92%	2021
ANN+ReLU (MFCC) [14]	71.96%	2021
2D CNN [15]	70.1%	2022
BYOL-S, 2048 [16]	76.9%	2022
Our approach – LSTM	66.2%	2023
Our approach – CNN	64%	2023

4. Conclusions

The research presented in this article provides a comprehensive evaluation of the proposed deep-learning algorithms for speech-emotion recognition. The high accuracy of our method suggests that it is a promising technique for accurate speech emotion recognition, which can be applied in various fields, such as healthcare, psychology, and customer service. Furthermore, our approach can facilitate the development of more sophisticated human-computer interaction systems that can better understand the emotional state of the speaker and respond appropriately.

References

[1] Cao H., Cooper D.G., Keutmann M.K., Gur R.C., Nenkova A., Verma R., *Crema-d: Crowd-sourced emotional multimodal actors dataset*, *IEEE transactions on affective computing*, 2014, vol. 5, no 4, pp. 377–390.

[2] Livingstone S.R., Russo F.A., *The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english*, *PloS one*, 2018, vol. 13, no 5, p. e0196391.

[3] Dupuis K., Pichora-Fuller M.K., *Toronto emotional speech set (tess) collection*, 2010, (access: 04-05-2022).
https://tspace.library.utoronto.ca/handle/1807/24487

[4] Costantini G., Iaderola I., Paoloni A., Todisco M., *Emovo corpus: an italian emotional speech database*, [In:] *International Conference on Language Resources and Evaluation (LREC 2014)*, ELRA, pp. 3501–3504.

[5] Issa D., Fatih Demirci M., Yazici A., *Speech emotion recognition with deep convolutional neural networks*, *Biomedical Signal Processing and Control*, 2020, vol. 59, no 101894, doi: https://doi.org/10.1016/j.bspc.2020.101894.

[6] Li H., Ding W., Wu Z., Liu Z., *Learning fine-grained multimodal alignment for speech emotion recognition*, *arXiv preprint arXiv:2010.12733*, 2020.

[7] Xu M., Zhang F., Zhang W., *Head fusion: Improving the accuracy and robustness of speech emotion recognition on the iemocap and ravdess dataset*, *IEEE Access*, 2021, vol. 9, pp. 74539–74549.

[8] Sajjad M., Kwon S., et al., *Clustering-based speech emotion recognition by incorporating learned features and deep bilstm*, *IEEE access*, 2020, vol. 8, pp. 79861–79875.

[9] Sönmez Y.Ü., Varol A., *A speech emotion recognition model based on multilevel local binary and local ternary patterns*, *IEEE Access*, 2020, vol. 8, pp. 190784–190796.

[10] Ancilin J., Milton A., *Improved speech emotion recognition with mel frequency magnitude coefficient*, *Applied Acoustics*, 2021, vol. 179, p. 108046.

[11] Tuncer T., Dogan S., Acharya U.R., *Automated accurate speech emotion recognition system using twine shuffle pattern and iterative neighborhood component analysis techniques*, *Knowledge-Based Systems*, 2021, vol. 211, p. 106547.

[12] Sekkate S., Khalil M., Adib A., *A statistical feature extraction for deep speech emotion recognition in a bilingual scenario*, *Multimedia Tools and Applications*, 2023, vol. 82, p. 11443–11460.

[13] Mocanu B., Tapu R., Zaharia T., *Utterance level feature aggregation with deep metric learning for speech emotion recognition*, *Sensors*, 2021, vol. 21, no 12, p. 4233.

[14] Dolka H., VM A.X., Juliet S., *Speech emotion recognition using ann on mfcc features*, [In:] *2021 3rd international conference on signal processing and communication (ICPSC)*, IEEE, pp. 431–435.

[15] Mittal R., Vart S., Shokeen P., Kumar M., *Speech emotion recognition*, [In:] *2022 2nd International Conference on Intelligent Technologies (CONIT)*, IEEE, pp. 1–6.

[16] Scheidwasser-Clow N., Kegler M., Beckmann P., Cernak M., *Serab: A multilingual benchmark for speech emotion recognition*, [In:] *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, pp. 7697–7701.

Aaron Earned an Iron Urn: Speech-to-IPA Models Improve Diagnostic of Pronunciation

Franciszek Olejnik[1[0009−0003−6263−0200]],
Rafał Stachowiak[1,2[0009−0002−7978−900X]],
Izabela Krysińska[1,2[0000−0003−3345−9291]],
Mikołaj Morzy[1[0000−0002−2905−9538]]

[1]*Poznan University of Technology*
Piotrowo 2, 60-965 Poznań, Poland
[2]*Pearson AI Learning Capabilities*
Dąbrowskiego 77A, 60-529 Poznań Poland

DOI:10.34658/9788366741928.43

Abstract. *Learning the proper pronunciation is one of the key aspects of foreign language acquisition. Assessment of the correctness of pronunciation requires the involvement of expert phoneticians and linguists, severely limiting the scalability of learning solutions. However, the recent adaptation of the Transformer architecture to the audio domain opens the way for automatic model-based assessment of pronunciation. In this paper, we present the pronunciation diagnostic tool developed at PUT and we experimentally evaluate the correlation between expert human assessment and automatic model assessment. By combining the Wav2Vec model and the IPA representation, we prove that pronunciation assessment can be performed automatically with high precision.*
Keywords: *Wav2Vec, IPA, pronunciation diagnostic, ASR*

1. Introduction

Language learning is a challenging task that requires learners to acquire new sounds and language structures. Among the most challenging aspects of learning a new language is developing correct pronunciation. Proper pronunciation is crucial for effective communication and can significantly affect a learner's success in language acquisition. Identifying phonemes that learners struggle with can be a challenging task, often requiring the assistance of a professional linguist, phonetician, or native speaker.

Recent advances in artificial intelligence and machine learning open new ways in which pronunciation learning can be accelerated and improved. In particular, the development of the Transformer model architecture and its adaptation to the

domain of audio and speech data allows for the design and implementation of efficient methods for automatic pronunciation assessment and improvement. In this paper, we describe a system for pronunciation assessment developed at Poznan University of Technology. The system uses a Wav2Vec model to transcribe speech to IPA (International Phonemic Alphabet) automatically and evaluates the correctness of pronunciation in the IPA space. Our main hypothesis is that pronunciation assessment can be performed automatically by a machine learning model without the involvement of expert linguists and phoneticians.

2. Speech-to-IPA

2.1. International Phonemic Alphabet

A *phonetic transcription* represents a sequence of sounds and other speech qualities enclosed with square brackets. For instance, the proper pronunciation of the word *tie* could be transcribed as [tʰaɪ] *tie*, where the [ʰ] diacritic means that after the consonant [t], there is a release of air. These transcriptions are used only in specific cases when arbitrary precision is needed. In practice, a more useful tool to learn the pronunciation is the *phonemic transcription*, i.e., the sequence of *phonemes*. A phoneme is defined as a set of sounds called *allophones* which, used interchangeably, do not change the meaning of the word. The phonemic transcription is represented in forward slashes. The International Phonetic Alphabet (IPA) is the set of all characters used to graphically encode the sounds (*phones*) produced by human beings. IPA characters are used to create both phonetic and phonemic transcriptions [1]. Phonemes depend on the language they describe. One phoneme in two different languages, represented with the same IPA character, may describe different allophones (e.g., /a/ in the Arabic language has three allophones [ɑ], [ɐ] and [a], but in Russian /a/ may be mapped to [æ] or [ɑ]) [2].

2.2. Wav2Vec model

Speech processing technology has seen remarkable advancements in recent years, with numerous applications in various fields, such as virtual assistants, speech-to-text applications, and automated call centers. The introduction of the Transformer architecture[3] marked a significant breakthrough in natural language processing (NLP). This innovative approach to NLP made it possible to achieve higher levels of accuracy and robustness in language processing solutions by overcoming the limitations of previous architectures based on recurrent or convolutional neural networks. The Transformer model utilizes a self-attention mechanism that shifts focus to different parts of the input sequence, enabling it to capture long-range dependencies more effectively. This ability to incorporate longer context made the Transformer architecture the go-to choice in speech processing.

By adopting the transformer architecture to the unique characteristics of speech data, researchers have developed a new Wav2Vec family of models [4, 5, 6]. Compared to previous speech recognition models such as CNNs, LSTMs, and GRUs [7], Wav2Vec has shown remarkable gains in phoneme recognition accuracy, particularly in low-resource settings. This is due to its ability to capture the contextual information of speech signals effectively and to learn phonetic representations that are more discriminative and robust. One of the most significant advantages of Wav2Vec in terms of phoneme recognition is the potential to reduce the amount of labeled data required for training. It is particularly important, as phoneme annotations are tedious, slow, and must be performed by qualified phoneticians. Given the impressive performance of the Wav2Vec model in speech processing tasks, we are motivated to investigate its potential in the phoneme production assessment of English language learners. While traditional methods for assessing phoneme production typically require expert phoneticians or human evaluators, the Wav2Vec model's ability to learn representations of speech signals through self-supervised learning and the transformer architecture may offer a promising alternative.

3. Experiment

3.1. Experimental system for pronunciation learning

(a) Configuration (b) Pronunciation assessment

Figure 1: System for pronunciation learning developed at PUT. Source: own work.

We have developed a system that automates the process of conducting pronunciation diagnostic tests. The application allows users to select phonemes to be tested (Fig. 1a), then randomly select words from an English dictionary that contain the chosen phonemes. Users record their pronunciation using the Web browser interface. The IPA transcription of the user's pronunciation is displayed along with the golden IPA transcription (i.e., the prescriptive pronunciation of the word as defined by the English dictionary). The application clearly indicates the phonemes which differ between user pronunciation and the gold pronunciation and calculates

the pronunciation score, which reflects the distance between the two pronunciations in the IPA space (Fig. 1b). We use the Wav2Vec XLSR model [6] fine-tuned on multilingual Common Voice dataset [8] to transcribe input WAV files into IPA.

3.2. Diagnostic test

Diagnostic pronunciation assessment aims to determine the phonemes that language learners struggle with so that subsequent learning activities can be tailored to meet the learners' specific needs. There are several methods for performing phoneme assessment, one of which involves computing formant frequencies on a segment of audio containing a particular phoneme and comparing them to the formant frequencies of native speakers. While this method has been widely used in phonetic research, it does require access to specialized software and expertise in speech analysis. Additionally, it may not be well-suited for large-scale assessments of phoneme production, as it is time-consuming and labor-intensive to collect and analyze formant frequency data for large numbers of speakers. Another approach to phoneme assessment is acoustic analysis which involves orthographic transcription, phonetic transcription of speech, and phoneme quality assessment.

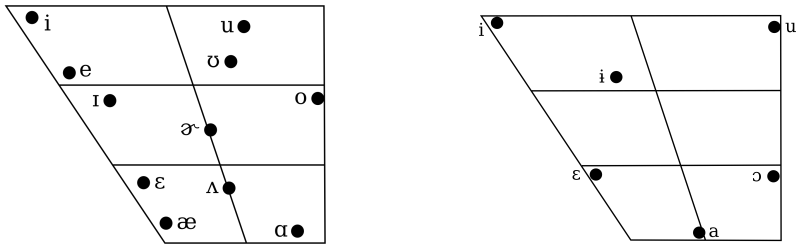
The main goal of this paper is to evaluate the usefulness of our experimental system for pronunciation assessment. We want to see if the pronunciation assessment computed from the IPA representations of user and gold pronunciations correlates with the pronunciation assessment by expert phoneticians.

Table 1: Survey of participants in the experiment

feature	**value**
gender	female: 4 (30,8%), male: 9 (69,2%)
age	20-30: 6 (46,2%), 31-40: 5 (38,5%), 41-50: 2 (15,4%)
CEFR level	B1: 1, B2: 7, C1: 3, C2: 1
environment	school education: 11 (84,6%), self-learning: 1 (15,4%)

We enroll 13 English learners to conduct the study. Table 1 presents information about the participants obtained using Language History Questionnaire [9]. As we can see, the cohort is sufficiently varied with respect to gender and self-reported CEFR language level (Common European Framework of Reference for Languages).

For our study, we record participants reading the passage from "The Boy Who Cried Wolf", a well-known fable by Aesop that "[...] has been substantially rewritten in order to provide suitable material for the description of English pronunciation" [10]. The fable contains a variety of sounds and phonetic features that are relevant to the study of language production and perception. The recordings are assessed by expert phoneticians in terms of the production of three vowels:



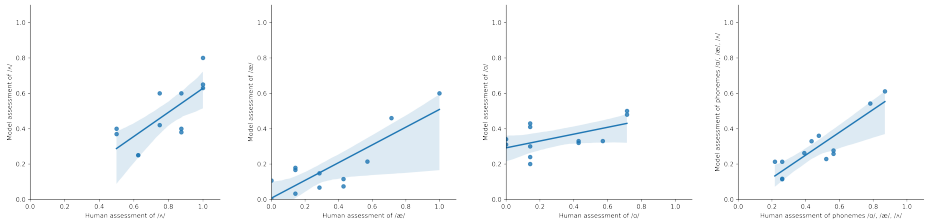
(a) English (b) Polish

Figure 2: IPA charts of the phonetic distance between vowels. Source: own work.

/ʌ/, /ɑ/, and /ae/. The choice of vowels is not arbitrary. As all participants are native Polish speakers, we have chosen sounds that are hard to differentiate for Poles. As can be seen in Fig. 2, English phonemes /ʌ/, /ɑ/, and /ae/ all map to a single Polish phoneme /a/.

After reading the passage, the participants conducted the automated diagnostic test in our application. The algorithm selected 30 words containing the chosen phonemes to record. All instances of phonemes were assessed as either correct or incorrect both by the speech-to-IPA model and by a phonetician. Thus, for each phoneme, we have obtained the model and the human assessment of the correctness of pronunciation.

4. Results



(a) phoneme /ʌ/ (b) phoneme /ae/ (c) phoneme /ɑ/ (d) all phonemes

Figure 3: Correlation of the assessment of pronunciation of selected phonemes by expert phoneticians and our model. Source: own work.

The results of the experiments are presented in Fig. 3. We find a strong Pearson correlation between human and model assessments, 0.73, 0.85, and 0.56 for phonemes /ʌ/, /ae/, and /ɑ/ respectively. We note that the model assessments are consistently lower than human assessments. The reason is probably the choice of words selected for human and model assessment. The words selected by our

diagnostic test are less frequently used than the words from "The Boy Who Cried Wolf". The lower scores are likely more related to the lack of knowledge of how to pronounce the word than the lack of ability to produce a given phoneme. When we combine the results for all phonemes, the overall Pearson correlation is 0.89. Thus, we conclude that the model assessment of pronunciation is very similar to the expert assessment. As a consequence, our model can be used without human supervision to assist with language acquisition and pronunciation assessment.

5. Conclusions

Our experiments show that there is a strong positive correlation between human and model assessment of pronunciation for all three tested phonemes. This indicates, at least partially, that our model can accurately assess the correctness of pronunciation. Of course, the data is only based on a small sample size of native Polish learners of English and may not be representative of other learners or other phonemes. More studies are required to evaluate the generalizability of the results of this preliminary study. Nevertheless, we find these results very encouraging, and we plan to evaluate further the usefulness of the speech-to-IPA approach for the automatic assessment of English pronunciation acquisition.

References

[1] Daniels P.T., Bright W., *The World's Writing Systems*, Oxford University Press on Demand, 1996, ISBN 9780195079937.

[2] Jones D., Ward D., *The phonetics of Russian*, Cambridge University Press, 2011.

[3] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I., *Attention is all you need*, *Advances in neural information processing systems*, 2017, vol. 30.

[4] Schneider S., Baevski A., Collobert R., Auli M., *wav2vec: Unsupervised pre-training for speech recognition*, *arXiv preprint arXiv:1904.05862*, 2019.

[5] Baevski A., Zhou Y., Mohamed A., Auli M., *Wav2vec 2.0: A framework for self-supervised learning of speech representations*, *Advances in Neural Information Processing Systems*, 2020, vol. 33, pp. 12449–12460.

[6] Conneau A., Baevski A., Collobert R., Mohamed A., Auli M., *Unsupervised cross-lingual representation learning for speech recognition*, *arXiv preprint arXiv:2006.13979*, 2020.

[7] Ravanelli M., Parcollet T., Bengio Y., *The pytorch-kaldi speech recognition toolkit*, [In:] *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE 2019, pp. 6465–6469.

[8] Ardila R., Branson M., Davis K., Henretty M., Kohler M., Meyer J., Morais R., Saunders L., Tyers F.M., Weber G., *Common voice: A massively-multilingual speech corpus*, [In:] *Proc. of the 12th Conference on Language Resources and Evaluation LREC 2020*, pp. 4211–4215.

[9] Li P., Sepanski S., Zhao X., *Language history questionnaire: A web-based interface for bilingual research*, *Behavior research methods*, 2006, vol. 38, no 2, pp. 202–210.

[10] Deterding D., *The north wind versus a wolf: short texts for the description and measurement of english pronunciation*, *Journal of the International Phonetic Association*, 2006, vol. 36, no 2, pp. 187–196.

Anonymizer for Polish Language

Tomasz Walkowiak[0000−0002−7749−4251],
Mateusz Gniewkowski[0000−0002−5011−5573],
Michał Pogoda[0000−0002−5011−5573],
Norbert Ropiak[0000−0003−3616−1298]

Wroclaw University of Science and Technology
Faculty of Information and Communication Technology
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
tomasz.walkowiak@pwr.edu.pl

DOI:10.34658/9788366741928.44

Abstract. *Researchers and enterprises require anonymization of unstructured text. This is not only due to the GDPR regulation, but also due to the increasing use of large language models (LLMs) such as GPT-3, where there is growing concern about the privacy and security risks associated with these models. The texts to be processed by such models need to be anonymized beforehand, and very often they need to be anonymized at the data providers' premises rather than at the machine learning teams. In this paper, we present an effective anonymization pipeline for Polish. It provides a modular and configurable solution that employs different modes, including the challenging pseudo-anonymization mode in languages with complex inflectional systems. The system can be easily integrated with existing systems and deployed in different environments using a microservices architecture solution with a REST interface.*

Keywords: *natural language processing, anonymization, Polish language, Kubernetes*

1. Introduction

GDPR compliance and data protection in sensitive applications require effective anonymization of unstructured text. Our study proposes Anonymizer, a modular and customizable end-to-end anonymization pipeline for Polish that can run completely offline.

It is an extended version of the system presented in [1]. Our solution uses a combination of deep neural name-entity recognition models, morphosyntactic dictionaries, and expert rules to effectively anonymize unstructured text and protect sensitive personal data in various applications. We emphasize the importance of effective anonymization pipelines, especially with the increasing use of large

language models (LLMs) such as GPT-3, there is growing concern about the privacy and security risks associated with these models. LLMs have the potential to process and store large amounts of personal data, which may include sensitive information such as names, addresses, and other personally identifiable information. This poses significant privacy and security risks.

2. Anonymizer architecture

Anonymization can take many forms. We have implemented three: **deleting** the strings containing sensitive data (this approach is simple and effective, but may result in the loss of valuable information from the text); **tagging** to convert phrases containing personal information into different tag types, we use tags corresponding to different categories, e.g., PLACE, PERSON; and **pseudo-anonymization** – to replace sensitive data with a false name or identifier, e.g., to replace the names of persons with fictitious names.

The last technique is the most interesting from the point of view of machine learning applications, because the lexical and syntactic structure of the text is not disturbed by pseudo-anonymization. However, it is the most challenging approach because the sensitive term must be replaced by a word in the correct grammatical form. Since Polish has a very complex inflection, a part-of-speech tagger is needed. It indicates the grammatical form of the words. In addition, we need a morphological synthesizer, a tool that creates an inflected form based on the lemma (from the dictionary) and the desired inflectional features (obtained by the part-of-speech tagger).

The reported system consists of five elements:

1. document format conversion – the Apache Tika library was used, it allows to convert text document formats (like doc, docx, pdf) into pure text;
2. part-of-speech tagger – Morphodita[2] trained for Polish;
3. name entity recognizer – the XLM-RoBerta[3] model fine-tuned for Polish to detect name entity boundary and fine-grained categorization (82 types were used) [4].
4. expert rules module – to detect data with a certain structure such as phone number, username, email address, URL or date.;
5. pseudo-anonymization module – with Morfeusz2[5] as a morphological synthesizer.

The anonymization process must take place within the data owner's infrastructure. Therefore, it is important that the application can be easily integrated with existing systems and deployed in different environments. To achieve this,

we have developed a microservices architecture solution with a REST interface with the possibility of synchronous and batch operation (for large sets of documents). Orchestration with microservices can be done using the Docker-Compose tool, as well as in production systems based on Kubernetes with the possibility of autoscaling individual system components to achieve high performance.

3. Conclusion

In conclusion, effective anonymization pipelines are crucial for complying with privacy regulations and protecting personal data in sensitive applications. The Anonymizer system proposed in this study provides a modular and configurable solution that employs different modes, including the challenging pseudo-anonymization mode in languages with complex inflectional systems. The system can be easily integrated with existing systems and deployed in different environments using a microservices architecture solution with a REST interface. The demo on-line version is avaliable at https://services.clarin-pl.eu/services/Anonymizer/interactive.

Acknowledgment

Financed by the European Regional Development Fund as a part of the 2014-2020 Smart Growth Operational Programme, CLARIN – Common Language Resources and Technology Infrastructure, project no. POIR.04.02.00-00C002/19.

References

[1] Oleksy M., Ropiak N., Walkowiak T., *Automated anonymization of text documents in polish*, *Procedia Computer Science*, 2021, vol. 192, pp. 1323–1333, doi: https://doi.org/10.1016/j.procs.2021.08.136.

[2] Straková J., Straka M., Hajič J., *Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition*, [In:] *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics 2020*, Association for Computational Linguistics, pp. 13–18.

[3] Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L., Stoyanov V., *Unsupervised cross-lingual representation learning at scale*, [In:] *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, pp. 8440–8451, doi: 10.18653/v1/2020.acl-main.747.

[4] Marcińczuk M., Kocoń J., Oleksy M., *Liner2 – a generic framework for named entity recognition*, [In:] *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, Association for Computational Linguistics, Valencia, Spain, pp. 86–91, doi: 10.18653/v1/W17-1413.

[5] Kieraś W., Woliński M., *Morfeusz 2 – analizator i generator fleksyjny dla języka polskiego*, *Język Polski*, 2017, vol. XCVII, no 1, pp. 75–83.

A Hybrid Fuzzy-Rough Approach to Handling Missing Data in a Fall Detection System

Teresa Mroczek[1[0000−0002−6064−9528]], **Dorota Gil**[1[0000−0001−6663−7571]], **Barbara Pękala**[1,2[0000−0002−5501−5467]]

[1] *University of Information Technology and Management*
Department of Artificial Intelligence
Sucharskiego 2, 35-225 Rzeszow, Poland, {tmroczek, dgil}@wsiz.edu.pl
[2] *University of Rzeszów, Institute of Computer Science*
Pigonia 1, 35-310 Rzeszow, Poland, bpekala@ur.edu.pl

DOI:10.34658/9788366741928.45

As an improvement of camera-based systems for depth maps analyzing [1], a new hybrid system, named FRSystem, based on fuzzy [2] and rough sets [3], has been developed [4]. It should be taken into account that in systems based on sensors, various types of disturbances may occur caused, for example, by power failure, battery depletion, or temporary damage. Due to the fact that FRSystem, created in cooperation with the Elderly Care Home in Rzeszow, is dedicated to elderly monitoring, it is desirable to extend it by the possibility to mine incomplete data. Therefore, a new method featuring two new concepts for mining incomplete data was proposed, based on:

- usage of a new K measure of knowledge to reduce the uncertainty due to incompleteness and imprecision of data;
- usage of a new method for computing maximal consistent blocks from incomplete data.

We evaluate the proposed methodology in stages.

In the first knowledge measure-only approach, from all objects with the same decision class as the object with a missing value, the most similar objects are selected from among the objects closest in terms of distance.

Whereas in the second approach, we include the idea of computing maximal consistent blocks. Thus, from the maximal collection of indiscernible objects i.e. maximal consistent blocks, the blocks with the highest probabilistic approximation are selected. Next, from the most similar objects the objects with the biggest Knowledge measure are chosen. Such an assessment of the most similar objects reduces the degree of uncertainty and improves the quality of the data.

Deployment of a hybrid approach, based on interval-valued fuzzy set theory [5, 6] and rough set theory [7], to mining incomplete data in a real decision-making problem, i.e. in a posture detection system was tested on data with 5%, 25%, and 50% of missing values. The best results for the interval-valued fuzzy model with knowledge measure were obtained using the geometric together with the arithmetic means as aggregations and for the rough-fuzzy model using the maximum as aggregation in the inference process. In addition, the usage of the knowledge measure allowed for the reduction of the uncertainty due to incompleteness and imprecision of data through an additional selection of objects, for which the degree of the information measure is the highest.

References

[1] Kwolek B., Kepski M., *Fuzzy inference-based fall detection using kinect and body-worn accelerometer*, *Applied Soft Computing*, 2016, vol. 40, pp. 305–318.

[2] Zadeh L.A., *Fuzzy sets*, *Information and control*, 1965, vol. 8, no 3, pp. 338–353.

[3] Pawlak Z., *Rough sets*, *International Journal of Computer and Information Sciences*, 1982, vol. 11, p. 341–356.

[4] Pękala B., Mroczek T., Gil D., Kepski M., *Application of fuzzy and rough logic to posture recognition in fall detection system*, *Sensors*, 2022, vol. 22, no 4, p. 1602.

[5] Sambuc R., *Fonctions ϕ-floues: Application á l'aide au diagnostic en pathologie thyroidienne*, Ph.D. thesis, Faculté de Médecine de Marseille, 1975, (unpublished).

[6] Zadeh L., *The concept of a linguistic variable and its application to approximate reasoning–i*, *Information Sciences*, 1975, vol. 8, no 3, pp. 199–249.

[7] Leung Y., Li D., *Maximal consistent block technique for rule acquisition in incomplete information systems*, *Information Sciences*, 2003, vol. 153, pp. 85–106.

Customer Churn Analytics Using Monotonic Rules

Marcin Szeląg[1][0000−0001−5884−7958], **Roman Słowiński**[1,2][0000−0002−5200−7795]

[1]*Institute of Computing Science, Poznań University of Technology*
[2]*Systems Research Institute, Polish Academy of Sciences*

DOI:10.34658/9788366741928.46

Abstract. *Using bank customer churn data, we demonstrate the explanatory and predictive capacity of monotonic decision rules. Since the data are partially ordinal, they are structured by a new version of the Variable Consistency Dominance-based Rough Set Approach before the induction of monotonic decision rules. The induced rules characterize loyal customers and the ones who left the bank. Such an approach is in line with explainable AI, aiming to obtain a transparent and understandable decision model. In the course of a computational experiment, we compare the predictive performance of monotonic rules with several well-known machine learning models.*
Keywords: *Dominance-based Rough Set Approach, Ordinal classification with monotonicity constraints, Monotonic decision rules, Customer churn*

1. Introduction

We perform data analytics on bank customer churn data publicly available at kaggle.com[1]. Our aim is to demonstrate the explanatory and predictive capacity of monotonic decision rules. Customer churn prediction is a frequent subject of data analytics [1]. This kind of data falls into the category of ordinal classification problems with monotonicity constraints. In ordinal classification, a finite set of objects constituting a universe U is described by a finite set of attributes A, among which there are condition attributes C and decision attributes D, such that $A = C \cup D$ and $C \cap D = \emptyset$. Particular condition attributes may have nominal or ordinal value sets (scales). The set of decision attributes is usually reduced to a singleton $D = \{d\}$ – its numerical value set indicates p ordered decision classes – class Cl_1 is the worst, and class Cl_p is the best. In case of bank customers, there are two classes: Cl_1 containing churning customers, and Cl_2 with loyal ones. Some ordinal condition attributes are monotonically related to class code, i.e., the higher (or lower) the attribute value the less (or more) probable the churn.

In supervised learning, the customers described by condition and decision attributes are training examples constituting a data set from which summaries in the

[1]https://www.kaggle.com/datasets/mathchi/churn-for-bank-customers

form of *"if . . . , then . . . "* decision rules are induced. However, the training examples can be partially inconsistent, making it difficult to draw clear patterns from the data. Due to inconsistency of bank customer churn data, we use the Dominance-based Rough Set Approach (DRSA) [2] to structure the data before the induction of monotonic decision rules. We will compare this approach experimentally with other available methods of supervised learning. The paper is a follow-up of computational experiments reported in [3, 4].

Section 2 presents the setup of a computational experiment. Section 3 presents the analysis of results, and Section 4 groups conclusions.

2. Experiment setup – bank customer churn data

The analyzed data set has 10 condition attributes (CreditScore, Age, Tenure, Balance, EstimatedSalary,Geography, Gender, NumberOfProducts, HasCrCard, IsActiveMember). Decision attribute *Exited* is binary. Contrary to previous work [3, 4] (concerning a balanced subset of 4000 customers), we considered all 10000 customers – 2037 disloyal customers, labeled by $Exited = 1$ (minority class), and 7963 loyal ones, labeled by $Exited = 0$ (preferred class from bank's viewpoint).

We performed calculations not only for the original data set, without missing attribute values (mv), but also for its several variants involving 5%, 10%, . . . , 25% of mv (all obtained in WEKA using *ReplaceWithMissingValue* filter with seed 1).

The experiment setup is adapted from [4], with several changes. First, we advocate ϵ-$\mathbb{VC}$-$\mathbb{DRSA}^{mv}_{1.5}$ only, which is the version of Variable Consistency DRSA (VC-DRSA) with ϵ object consistency measure, handling mv using dominance relations $D^{mv}_{1.5}$ and $d^{mv}_{1.5}$ [5], and enhanced by three extensions: (i) generalization of elementary conditions during rule induction, (ii) pruning of the set of induced rules, and (iii) using co-trained Naive Bayes classifier as a fallback when no rule matches a classified object. Second, we exclude from the comparison OLM and MoNGEL – for their lowest accuracy [4]. Third, we assume the same monotonicity constraints as in [3, 4], and the same parameters for all methods as in [4].

The methods compared with ϵ-$\mathbb{VC}$-$\mathbb{DRSA}^{mv}_{1.5}$ are those considered in [4], except OLM and MoNGEL. They are all implemented and referenced in WEKA[2]: C4.5, Naive Bayes (NB), Support Vector Machine (SVM), Random Forest (RF), Multilayer Perceptron (MP), RIPPER (RIPP), and Ordinal Stochastic Dominance Learner (OSDL). The methods offering an explanation of recommended decision are: ϵ-$\mathbb{VC}$-$\mathbb{DRSA}^{mv}_{1.5}$ (matched decision rules), C4.5 (matched path in the decision tree), RF (matched paths in the decision trees forming an ensemble), NB (conditional probability distribution), and RIPP (matched decision rule). However, apart from our method, only OSDL explicitly considers monotonicity constraints.

[2]https://www.cs.waikato.ac.nz/~ml/weka; used version: 3.8.6

Due to imbalanced distribution of classes, we compared the methods using true positive rates (TPR) of both classes and their geometric mean (Gmean). We also took into account the explainability of induced models and the monotonic relationship between ordinal attributes and assignment to decision classes.

To make the comparison reliable, we run 10 times a 10-fold cross-validation with a different seed. For each variant of the data set (0%, 5%, ..., 25% of mv), we got 100 splits, each having 9000 training objects and 1000 test objects.

The code implementing calculations is available on Github[3].

Table 1 shows the quality of classification [2] for each data set, assuming threshold θ_X for $\epsilon_X(y)$ equal to 0, $X \in \{Cl_1, Cl_2\}$, and $y \in U$. The quality of classification measures the % of objects consistent with the dominance principle.

Table 1. Quality of classification for $\theta_X = 0$ (classical DRSA) for a given % of missing values (first row), when applying dominance relations $D^{mv}_{1.5}$ and $d^{mv}_{1.5}$.

%mv	**0**	**5**	**10**	**15**	**20**	**25**
$\gamma^{mv}_{1.5}$	0.6478	0.6412	0.6319	0.6092	0.5988	0.5835

3. Analysis of results

Tables 2 and 3 present a comparison of avg. true positive rates (TPR)[4] – for minority class $Exited = 1$ in the superscript, and for majority class $Exited = 0$ in the subscript, and their geometric mean (main numbers) from 10 independent runs of 10-fold cross-validation (CV). Undersampling the majority class in each fold's training set gave much better results than using the original training set. We also tested oversampling the minority class, and both simultaneously, but the results for undersampling were the best. Thus, in the following, we present only them.

In Table 3, one can note relatively good Gmean of ϵ-$\mathbb{VC}$-$\mathbb{DRSA}^{mv}_{1.5}$, and its consistently best TPR for the most interesting class $Exited = 1$.

Table 4 shows average consistencies (calculated over 100 folds) of test data, in terms of $\gamma^{mv}_{1.5}$, when the customers are assigned new classification decisions by the trained models. For each % of mv, we considered two variants of the calculation of $\gamma^{mv}_{1.5}$. The first one concerns all test objects and verifies their *post-consistency* (i.e., consistency after new decisions) [6]. The second variant verifies post-consistency of test objects on a subset of originally consistent (*pre-consistent*) test objects [4]. Note that our method has all the best results, except OSDL for 0% mv.

[3] https://github.com/ruleLearn/rulelearn-experiments

[4] TPR(Cl_i) = (number of objects from class Cl_i assigned to Cl_i) / (number of objects from Cl_i)

Table 2. Gmean and avg. TPR (%) in 10×10-fold CV – imbalanced training data

%mv	ϵ-VC-DRSA$^{mv}_{1.5}$	C4.5	NB	SVM	RF	MP	RIPP	OSDL
0	$\mathbf{72.1}^{\mathbf{58.1}}_{89.6}$	$66.7^{46.8}_{95.1}$	$65.3^{45.0}_{94.7}$	0^{0}_{100}	$67.8^{47.9}_{95.9}$	$67.2^{47.5}_{95.2}$	$65.1^{44.2}_{95.9}$	$58.3^{35.1}_{96.9}$
5	$\mathbf{68.7}^{\mathbf{51.8}}_{90.9}$	$63.8^{42.2}_{96.6}$	$64.2^{43.4}_{94.9}$	0^{0}_{100}	$59.4^{36.3}_{97.2}$	$61.7^{40.3}_{94.3}$	$63.1^{41.4}_{96.1}$	$57.8^{34.5}_{97.0}$
10	$\mathbf{65.9}^{\mathbf{47.4}}_{91.6}$	$59.3^{36.2}_{96.9}$	$62.9^{41.6}_{95.0}$	0^{0}_{100}	$53.0^{28.6}_{98.1}$	$58.6^{36.4}_{94.4}$	$59.7^{37.1}_{96.3}$	$54.7^{30.8}_{97.1}$
15	$\mathbf{62.2}^{\mathbf{41.8}}_{92.6}$	$54.8^{30.8}_{97.5}$	$61.3^{39.4}_{95.3}$	0^{0}_{100}	$49.5^{24.8}_{98.4}$	$55.9^{33.0}_{94.9}$	$57.7^{34.6}_{96.2}$	$53.0^{29.0}_{97.0}$
20	$48.6^{24.3}_{97.3}$	$48.0^{23.4}_{98.4}$	$\mathbf{59.6}^{\mathbf{37.2}}_{95.5}$	0^{0}_{100}	$44.8^{20.4}_{98.8}$	$54.3^{31.1}_{94.6}$	$52.6^{28.6}_{96.8}$	$50.9^{26.6}_{97.2}$
25	$39.8^{16.1}_{98.4}$	$44.4^{19.9}_{98.9}$	$\mathbf{58.2}^{\mathbf{35.4}}_{95.8}$	0^{0}_{100}	$41.9^{17.7}_{99.1}$	$53.0^{29.6}_{94.8}$	$51.1^{27.0}_{96.8}$	$48.1^{23.7}_{97.7}$

Table 3. Gmean and avg. TPR (%) in 10×10-fold CV – undersampling train. data

%mv	ϵ-VC-DRSA$^{mv}_{1.5}$	C4.5	NB	SVM	RF	MP	RIPP	OSDL
0	$75.7^{\mathbf{77.0}}_{74.5}$	$75.0^{73.6}_{76.4}$	$76.1^{75.9}_{76.3}$	$71.0^{69.8}_{72.3}$	$\mathbf{77.0}^{74.6}_{79.4}$	$75.9^{74.2}_{77.6}$	$75.8^{74.3}_{77.4}$	$71.3^{60.1}_{84.5}$
5	$74.9^{\mathbf{75.8}}_{74.0}$	$75.5^{73.0}_{78.1}$	$75.5^{74.9}_{76.1}$	$70.3^{68.5}_{72.2}$	$\mathbf{75.9}^{73.7}_{78.2}$	$74.8^{73.9}_{75.7}$	$74.0^{72.8}_{75.2}$	$70.4^{59.2}_{83.8}$
10	$73.9^{\mathbf{75.1}}_{72.7}$	$74.8^{71.6}_{78.1}$	$75.0^{74.5}_{75.4}$	$69.6^{67.9}_{71.3}$	$\mathbf{75.1}^{71.6}_{78.7}$	$73.2^{72.9}_{73.5}$	$72.5^{71.5}_{73.6}$	$68.7^{56.2}_{84.0}$
15	$72.8^{\mathbf{74.1}}_{71.5}$	$73.8^{70.0}_{77.7}$	$73.9^{73.0}_{74.9}$	$68.8^{67.1}_{70.5}$	$\mathbf{74.5}^{71.3}_{77.9}$	$71.9^{71.1}_{72.7}$	$71.2^{69.2}_{73.2}$	$67.0^{53.6}_{83.6}$
20	$71.4^{\mathbf{73.3}}_{69.5}$	$72.4^{68.7}_{76.4}$	$73.0^{72.2}_{73.8}$	$67.9^{65.7}_{70.1}$	$\mathbf{73.6}^{71.2}_{76.2}$	$70.1^{71.1}_{69.2}$	$70.3^{68.5}_{72.1}$	$66.0^{52.4}_{83.2}$
25	$70.5^{\mathbf{71.8}}_{69.3}$	$71.7^{67.8}_{75.7}$	$72.1^{71.2}_{73.1}$	$66.9^{64.6}_{69.3}$	$\mathbf{72.9}^{70.7}_{75.2}$	$68.9^{67.2}_{70.7}$	$69.1^{67.1}_{71.1}$	$65.3^{51.3}_{83.2}$

Next, we checked the classification models trained on the entire set of 10000 customers without mv, using undersampling. ϵ-VC-DRSA$^{mv}_{1.5}$ induced 162 rules of avg. length 6, avg. support 82, and avg. confidence factor 0.9. C4.5 generated 196 rules (a path in C4.5 tree = a rule). RIPPER induced an ordered list of 10 rules, 9 for class *Exited* = 1, and only one for class *Exited* = 0 (default rule without elementary conditions). We observed several problems. First, the C4.5 tree corresponds to many rules ignoring monotonicity constraints, e.g., the rule suggesting class *Exited* = 1: "if (NumOfProducts > 2) and (Age ≤ 42) and (Balance > 55853.33), then (Exited = 1)" involves two conditions violating monotonicity constraints: Age ≤ 42 (Age is a cost-type criterion, so the relation should be > or ≥), and Balance > 55853.33 (Balance is a gain-type criterion, so the relation should be < or ≤). Second, the RIPPER's rule set also exhibits the same deficiency, e.g., the rule: "if (Balance ≥ 3768.69) and (Age ≥ 39) and (Gender = Female), then (Exited=1)", involves condition Balance ≥ 3768.69. Third, the RIPPER's rules are not useful in explaining decisions for class *Exited* = 0, as only the default rule does it. Moreover, the default rule covers as much as 527 customers with decision *Exited* = 1, which implies rule's ϵ consistency around 0.26, while ϵ-VC-DRSA$^{mv}_{1.5}$ makes rules having in the worst case ϵ consistency equal to 0.005 (52 times better).

Table 4. Average consistency (in terms of $\gamma_{1.5}^{mv}$, in %) of test data after new classification decisions concerning all test objects (*post* consistency) or only originally consistent test objects (*pre-post* consistency); ϵ-$\mathbb{D}_{1.5}^{mv}$ stands for ϵ-$\mathbb{VC}$-$\mathbb{DRSA}_{1.5}^{mv}$

%mv	ϵ-$\mathbb{D}_{1.5}^{mv}$	C4.5	NB	SVM	RF	MP	RIPP	OSDL
0	99.8|99.9	93.3|94.4	97.5|98.1	98.9|99.1	95.1|96.2	96.7|97.4	97.7|98.2	**100|100**
5	**98.6|99.1**	94.6|96.2	95.8|97.1	96.8|97.7	94.2|95.7	94.3|95.8	95.8|96.9	98.0|98.7
10	**97.3|98.4**	94.1|96.3	93.9|96.2	94.7|96.5	93.1|95.6	92.4|94.5	94.3|96.1	96.3|97.8
15	**96.3|98.1**	93.0|96.0	92.7|95.7	93.2|95.7	92.0|95.3	91.1|94.1	92.0|94.7	95.0|97.3
20	**94.4|97.1**	91.2|95.2	90.7|94.8	90.7|94.3	90.1|94.5	88.5|92.5	90.5|94.2	93.1|96.5
25	**92.6|96.6**	89.6|94.7	88.7|94.2	88.7|93.5	88.8|94.4	87.7|92.7	89.0|93.6	91.2|95.9

When analyzing the classification performed by ϵ-$\mathbb{VC}$-$\mathbb{DRSA}_{1.5}^{mv}$ on the entire set of 10000 objects without mv, we noticed that classification of a particular customer is usually based on a small fraction of 162 rules matching this customer, making it explainable [3, 4]. The avg. number of rules matching a customer is 3.61. E.g., customer no. 182 is matched by one rule only: "if (Age $\geq$ 51) and (CreditScore $\leq$ 567) and (EstimatedSalary $\leq$ 158325.87) and (HasCrCard = 1), then (Exited = 1)" (supp.=65, ϵ=0.0049). Thus, (s)he is assigned to class *Exited* = 1.

In Table 5, we present top 4 rules for class *Exited* = 1. Remark that IsActive-

Table 5. Top rules induced by ϵ-$\mathbb{VC}$-$\mathbb{DRSA}_{1.5}^{mv}$ using undersampling for all 10000 customers without missing values

ID	**Conditions**	**Decision**	ϵ	**Support**
102	Age $\geq$ 51, IsActiveMember = 0, NumOfProducts $\leq$ 1, CreditScore $\leq$ 825	*Exited* = 1	0.004	242
86	Age $\geq$ 51, IsActiveMember = 0, CreditScore $\leq$ 674	*Exited* = 1	0.005	217
98	Age $\geq$ 49, IsActiveMember = 0, Geography = Germany, CreditScore $\leq$ 849	*Exited* = 1	0.003	178
76	NumOfProducts $\geq$ 3, Age $\geq$ 43	*Exited* = 1	0.000	161

Member = 0 is present in 3 of the 4 top rules, and Age is used in all these rules. The rules explain the patterns observed in data and respect monotonicity constraints.

Comparing the results given in Tables 3 and 4, one can see that whenever our method achieves a slightly worse Gmean, it has a better average post-consistency. This suggests misclassification of some test objects for a good reason – making classification decisions more consistent with the dominance principle.

4. Conclusions

The comparison of the enhanced version of Variable Consistency Dominance-based Rough Set Approach, ϵ-$\mathbb{VC}$-$\mathbb{DRSA}_{1.5}^{mv}$, with 7 other ML methods on bank

customer churn data show that our method outperforms its competitors on predictive accuracy and respect of monotonicity constraints, while giving an interpretable insight into the problem at hand in terms of monotonic decision rules.

The authors wish to acknowledge the TAILOR project funded by EU Horizon 2020 programme under GA No 952215, and the SBAD funding.

References

[1] De Caigny A., Coussement K., De Bock K.W., *A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees*, *Eur. J. Oper. Res.*, 2018, vol. 269, pp. 760–772.

[2] Greco S., Matarazzo B., Słowiński R., *Rough sets theory for multicriteria decision analysis*, *Eur. J. Oper. Res.*, 2001, vol. 129, no 1, pp. 1–47.

[3] Szeląg M., Słowiński R., *Dominance-based rough set approach to bank customer satisfaction analysis*, [In:] P. Jędrzejowicz, et al. (eds.), *PP-RAI'2022, Proceedings*, Gdynia Maritime University, pp. 147–150.

[4] Szeląg M., Słowiński R., *Explaining and predicting customer churn by monotonic rules induced from ordinal data*, *Eur. J. Oper. Res.*, 2023, vol. (subm.).

[5] Szeląg M., Błaszczyński J., Słowiński R., *Rough set analysis of classification data with missing values*, [In:] L. Polkowski, et al. (eds.), *IJCRS 2017, Proceedings*, *LNAI*, vol. 10313, Springer, pp. 552–565.

[6] Cano J.R., Gutiérrez P.A., Krawczyk B., Woźniak M., García S., *Monotonic classification: An overview on algorithms, performance measures and data sets*, *Neurocomputing*, 2019, vol. 341, pp. 168–182.

Application of Pawlak's Conflict Model to Generate Coalitions of Local Tables with Similar Values on Conditional Attributes

Małgorzata Przybyła-Kasperek[1][0000−0003−0616−9694]
Katarzyna Kusztal[1][0000−0002−9970−5339]

[1]*University of Silesia in Katowice*
Institute of Computer Science,
Będzińska 39, 41-200 Sosnowiec, Poland
malgorzata.przybyla-kasperek@us.edu.pl, kkusztal@us.edu.pl

DOI:10.34658/9788366741928.47

Abstract. *Generating a model or pattern based on dispersed data available in many different tables is difficult because there can be numerous inconsistencies in the data. One way to deal with such a problem is to analyze conflicts and generate coalitions of consistent local tables. This paper proposes a model in which coalitions of tables with consistent data are created using Pawlak's conflict analysis approach. A model, decision tree, is created based on the aggregated data within the coalition. This way, we get rules that better describe the concepts found in consistent local tables.*
Keywords: *Pawlak conflict analysis model, independent data sources, coalitions, decision trees, dispersed data*

1. Introduction

Data on various issues can be collected in a dispersed and decentralized manner. In such a situation, we cannot expect that the data collected by independent entities are consistent. That is why simple aggregation of dispersed data is not possible. More sophisticated methods must be used to agree on a common model based on all local data. An example of such dispersed data can be tables collected by medical departments located in different cities but performing diagnostics in one area. Another example is mobile applications that many users use, and dispersed local data is collected. Alternatively, other institutions such as banks, energy stations, and atmospheric sensors can collect data in dispersed forms. Using dispersed data should guarantee a higher classification quality than operating based on only one local data set. The main approach to building model based on dispersed data is federated learning [1, 2], in which we care for the privacy and protection of local data. This approach responds to GDPR arrangements [3] and

general concerns about too much access by artificial intelligence to personal data. The method is to build models in the local space and then share only the models' parameters with a central server. The local models are then aggregated into a global model and returned to the local spaces. The local units can modify their local model or leave it unchanged. Such an exchange is iterated until convergence is achieved. Federated learning has found many practical applications [4, 5]. However, this approach does not allow any collaboration of local units, even if such units are very similar and closely related. The approach proposed in this paper is different because we assume the cooperation of local tables and the exchange of data within the coalition.

The second approach to dealing with dispersed data is classifier ensembles [6, 7, 8]. Here we have free access to local data. Base classifiers can be sensitized to complex cases with which previous base classifiers in the series had trouble, as in AdaBoost [9]. In this approach, originally, the data are collected in a single and consistent decision table. The algorithm controls the process of data dispersion. The proposed in this paper approach is different from classifier ensembles approach as the considered local tables are collected independent and can be inconsistent.

We can also find dispersed data classification approaches in the literature that use the cooperation of local tables. In [10, 11], a dynamic approach for creating the system structure is considered, in which coalitions of classifiers making similar decisions for objects are defined using Pawlak's conflict analysis model. The effect of using the strength of these coalitions on classification quality is analyzed in paper [12]. However, this approach is different from the one proposed in this paper. In the proposed approach local tables need to have the same sets of conditional attributes. In [10, 11, 12] there was no such assumption. System with a static structure is considered in the proposed approach – the same structure for all classified objects is used. In [10, 11, 12] the dynamic system structure was considered – different structure for each classified object. Coalitions of local tables are generated based on characteristics of conditional attributes' values stored in local tables. In [10, 11, 12] coalitions are generated in relation to prediction vectors' generated for classified objects. In the proposed approach, the focus is on the common concept described in the data, rather than the compatible prediction. So the two approaches are completely different. Establishing a coalition of tables describing a common concept allows later to generate better models of hidden patterns in the data.

There are many different approaches to conflict analysis in the literature [13, 14, 15]. Some approaches consider constraints and look for an optimal solution that meets resource limitations [16]. Others are drawn from game theory and focus on optimizing payoff functions [17]. This paper uses an approach based on Pawlak's conflict analysis, as it is relatively simple and very effective in modelling relations and calculating the intensity of conflicts between the parties.

The structure of the paper is as follows. In section 2 we describe the proposed model and object's classification process. The article ends with a conclusion.

2. Proposed model

The idea behind the model is to designate coalitions of local tables that store similar data. The similarity is calculated in terms of the values of conditional attributes stored in tables. We assume that all tables have the same conditional attributes. Formally, we assume that a set of local decision tables are given $D_i = (U_i, A, d), i \in \{1, \ldots, n\}$, where U_i is the universe, a set of objects; A is a set of conditional attributes; d is a decision attribute. We define certain characteristics for all attributes. However, we proceed in a different way in the case of qualitative and quantitative attributes. For each quantitative attribute $a_{quan} \in A$, we determine the average of all attribute's values occurring in local table D_i, we denote this as $\overline{Val}^i_{a_{quan}}$. We also calculate the global average and the global standard deviation of values stored in all local tables: $\overline{Val}_{a_{quan}}$ and $SD_{a_{quan}}$. For each qualitative attribute $a_{qual} \in A$, we determine a vector over the values of that attribute. If attribute a_{qual} has c values $val_1, \ldots, val_c$. The vector $Val^i_{a_{qual}} = (n^i_1, \ldots, n^i_c)$ represents the number of occurrences of each of these values in the decision table D_i.

These attributes' characteristics are used to define the information system $S = (U, A)$ that is used in Pawlak's conflict analysis [18]. In the system S, U is a set of local decision tables and A is a set of conditional attributes. For the quantitative attribute $a_{quan} \in A$ a function $a_{quan} : U \to \{-1, 0, 1\}$ is defined

$$a_{quan}(D_i) = \begin{cases} 1 & \text{if } \overline{Val}_{a_{quan}} + SD_{a_{quan}} < \overline{Val}^i_{a_{quan}} \\ 0 & \text{if } \overline{Val}_{a_{quan}} - SD_{a_{quan}} \leq \overline{Val}^i_{a_{quan}} \leq \overline{Val}_{a_{quan}} + SD_{a_{quan}} \\ -1 & \text{if } \overline{Val}^i_{a_{quan}} < \overline{Val}_{a_{quan}} - SD_{a_{quan}} \end{cases} \quad (1)$$

In this way, the values -1, 0, 1 distinguish local tables in which the stored values, are below the minimum value of the global range of typical variability, belong to this range or are above the maximum value of the global range of typical variability. For the qualitative attribute a_{qual} we use the 3−means clustering algorithm for vectors $Val^i_{a_{qual}}, i \in \{1, \ldots, n\}$. In this way three groups of local tables with similar distribution of the attribute's a_{qual} values are defined. Then for the attribute a_{qual} and the tables in the first group are assigned 1, in the second group 0, in the third group -1. In the next step, Pawlak's conflict analysis model is used to define coalitions of local tables that store similar attributes' values. The conflict intensity between pairs of tables are calculated using the function $\rho(D_i, D_j) = \frac{card\{a \in A : a(D_i) \neq a(D_j)\}}{card\{A\}}$. A coalition is a set of local tables that for every two tables D_i, D_j, $\rho(D_i, D_j) < 0.5$ is satisfied. In other words, a coalition is a set of local tables that are in a friendship relation. We assume that the tables within a

coalition cooperate. This collaboration involves the exchange of data. An aggregated table is generated in which the universe is the sum of the objects present in the local tables from the coalition. A model is built based on each aggregate table in the next step. The models determined for the coalition is used to classify the object. The final decision is made by voting. Any type of model can be used in the proposed approach.

In the paper [19], decision trees with a Gini index were used. It was shown that this approach gives a much better classification quality than the one in which we do not allow local tables cooperation. The baseline approach from the paper [19] built a decision tree based on each local table separately. The improvement obtained was over 0.1 for some dispersed data sets. The obtained differences in average classification accuracy were shown to be statistically significant.

In the next research stage, we look closely at the knowledge generated by the proposed model. The paper [20] showed that the confidence of rules generated using the proposed approach is significantly higher than those generated using the baseline approach. This gives promise and good perspectives on the use of the proposed model both with other decision rule-based models and the application of the proposed approach to real-life cases.

3. Conclusions

The paper presents a new model for classification based on dispersed data. The novelty is to combine local tables with similar values on conditional attributes into coalitions. For this purpose, Pawlak's conflict analysis model was used. This approach improves the quality of classification and the confidence of the generated decision rules. In the next stage of the research, it is planned to consider the variability of the conditional attributes' values with respect to decision classes. Only local tables that have similar values within decision classes will be combined into coalitions.

References

[1] Połap D., Woźniak M., *A hybridization of distributed policy and heuristic augmentation for improving federated learning approach*, *Neural Networks*, 2022, vol. 146, pp. 130–140.

[2] Dyczkowski K., Pękala B., Szkoła J., Wilbik A., *Federated learning with uncertainty on the example of a medical data*, [In:] *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, IEEE, pp. 1–8.

[3] Truong N., Sun K., Wang S., Guitton F., Guo Y., *Privacy preservation in federated learning: An insightful survey from the gdpr perspective*, *Computers & Security*, 2021, vol. 110, p. 102402.

[4] Ślazyk F., Jabłecki P., Lisowska A., Malawski M., Płotka S., *Cxr-fl: Deep learning-based chest x-ray image analysis using federated learning*, [In:] *ICCS 2022, London, UK, June 21–23, 2022*, Springer, pp. 433–440.

[5] Shubyn B., Kostrzewa D., Grzesik P., Benecki P., Maksymyuk T., Sunderam V., Syu J.H., Lin J.C.W., Mrozek D., *Federated learning for improved prediction of failures in autonomous guided vehicles*, *Journal of Computational Science*, 2023, p. 101956.

[6] Czarnowski I., *Weighted ensemble with one-class classification and oversampling and instance selection (wecoi): An approach for learning from imbalanced data streams*, *Journal of Computational Science*, 2022, vol. 61, p. 101614.

[7] Jedrzejowicz J., Jedrzejowicz P., *Gep-based classifier for mining imbalanced data*, *Expert Systems with Applications*, 2021, vol. 164, p. 114058.

[8] Ślęzak D., Stawicki S., *The problem of finding the simplest classifier ensemble is np-hard–a rough-set-inspired formulation based on decision bireducts*, [In:] *Rough Sets: International Joint Conference, Havana, Cuba, June 29–July 3, 2020*, Springer, pp. 204–212.

[9] Freund Y., Schapire R., Abe N., *A short introduction to boosting*, *Journal-Japanese Society For Artif. Intell.*, 1999, vol. 14, no 771-780, p. 1612.

[10] Przybyla-Kasperek M., Wakulicz-Deja A., *Global decision-making system with dynamically generated clusters*, *Inf. Sci.*, 2014, vol. 270, pp. 172–191.

[11] Przybyła-Kasperek M., *Coalitions' weights in a dispersed system with pawlak conflict model*, *Group Decision and Negotiation*, 2020, vol. 29, pp. 549–591.

[12] Przybyla-Kasperek M., Wakulicz-Deja A., *The strength of coalition in a dispersed decision support system with negotiations*, *Eur. J. Oper. Res.*, 2016, vol. 252, no 3, pp. 947–968.

[13] Skowron A., Deja R., *On some conflict models and conflict resolutions*, *Romanian J. Inf. Sci. Technol.*, 2002, vol. 3, no 1-2, pp. 69–82.

[14] Ramanna S., Peters J.F., Skowron A., *Approaches to conflict dynamics based on rough sets*, *Fundamenta Informaticae*, 2007, vol. 75, no 1-4, pp. 453–468.

[15] Skowron A., Dutta S., *Rough sets: past, present, and future*, *Natural computing*, 2018, vol. 17, pp. 855–876.

[16] Deja R., Ślęzak D., *Rough set theory in conflict analysis*, [In:] *New Frontiers in Artif. Intell.: Joint JSAI 2001 Workshop Post-Proceedings*, Springer, pp. 349–353.

[17] Rzepecki Ł., Jaśkowski P., *Application of game theory against nature in supporting bid pricing in construction*, *Symmetry*, 2021, vol. 13, no 1, p. 132.

[18] Pawlak Z., *On conflicts*, *International Journal of Man-Machine Studies*, 1984, vol. 21, no 2, pp. 127–134.

[19] Przybyla-Kasperek M., Kusztal K., *New classification method for independent data sources using pawlak conflict model and decision trees*, *Entropy*, 2022, vol. 24, no 11, p. 1604.

[20] Przybyla-Kasperek M., Kusztal K., *Rules' quality generated by the classification method for independent data sources using pawlak conflict analysis model*, *PrePrint*.

Chapter 6

Neural Network and Deep Learning Systems

Domain Editors:

1. Aleksander Byrski, AGH University of Science and Technology
2. Marcin Kurdziel, AGH University of Science and Technology

A Novel DNN-based Image Watermarking Algorithm

Slavko Kovačević[1], **Kosta Pavlović**[2], **Igor Djurović**[1,3]

[1]*Faculty of Electrical Engineering*
The University of Montenegro
skovacevic@ucg.ac.me
[2]*Faculty of Natural Sciences and Mathematics*
The University of Montenegro
[3]*Montenegrin Academy of Sciences and Arts*

DOI:10.34658/9788366741928.48

Abstract. *DNN architecture for image watermarking that balances the tasks of embedding and detecting a watermark is presented. The system consists of two networks: the embedder and the detector. A loss function based on a structural similarity index measure minimizes the difference between the original and watermarked signal. The average SSIM is* 0.98 *while the accuracy is* 99.99%.
Keywords: *deep neural networks, image watermarking, autoencoder, ssim*

1. Introduction

With recent advancements in deep neural networks, it is now possible to generate artificial data, also known as "deep fakes", which can be used for malicious purposes. To secure the copyright and authenticity of digital data, a watermark is added to the signal carrier. The process of watermark detection consists in extracting the watermark from the signal carrier. In the context of images, adding a watermark is essential not only for safeguarding intellectual property but also for preserving the information contained within the image.

The field of digital watermarking has been the subject of active research since the 1990s [1], and many techniques have been developed for watermarking various types of digital media. The primary concepts for watermarking were developed for images, and numerous methods have been proposed, including those by [2, 3, 4]. The resurgence of deep learning has led to its incorporation into various signal processing fields [5, 6, 7]. Several studies have utilized deep neural networks (DNNs) for both watermark embedding and detection, as demonstrated in papers by [8, 9, 10, 11].

We propose a DNN architecture that balances the conflicting tasks of watermark embedding and detection. The system consists of two networks: the embedder, a U-Net-like autoencoder, which creates its transform domain and inserts the watermark, and the detector, which extracts the watermark from the signal carrier. In theory, the watermarked image produced by the embedder should be an exact replica of the original image. However, it must also be possible for the detector to identify the presence of the watermark.

2. Architecture and training procedure

The specifications for the design of the embedder and the detector are shown in Tables 1 and 2, respectively. The encoding component of the embedder is tasked with creating a latent representation of the image in which the watermark is embedded, and the decoding component reconstructs the image while preserving the watermark. Convolutional layers of the encoder are batch normalized with the ReLU activation function, while transposed convolution of the decoder, in the addition to batch normalization and ReLU, uses dropout with the rate .5. Sigmoid is used as the output activation function of the embedder as input images are normalized to range $[0, 1]$. The detector is a simple image classifier whose output layer size depends on the watermark length. The watermark message is a randomly sampled binary vector of length 3. Watermark is embedded by concatenating it at the end of the channel dimension of the latent representation of the image whose dimensions are $4 \times 4 \times 256$. Therefore, the watermark message is repeated 4×4 times to match the dimensions of the latent representation of the image, but also to increase its robustness.

The proposed network is trained on the CIFAR-10 [12] dataset for 20 epochs. As the input of the embedder are unwatermarked images and the desired output are watermarked images that are identical to the input, a novel loss function based on SSIM is used. Standard SSIM implementation from [13] is utilized in defining the loss function. The final SSIM index value ranges from -1 to 1, with 1 signifying complete similarity, 0 indicating no correlation, and -1 representing perfect dissimilarity. Therefore, embedder loss can be calculated as $L_e = 1 - SSIM$. The detector uses binary cross entropy as a loss function. We used Nadam as an optimizer and the learning rate parameter is set to $1e{-}5$. Training two neural networks with opposing tasks at once is a challenging process. The losses for these networks are balanced in a way that gives priority to the detector network at the beginning of training, but as training progresses, the focus shifts to the embedder network until both networks reach their optimal performance. Embedder and the detector loss trajectories during the training are shown in Figure 1a.

Table 1: The specifications for the design of the embedder.

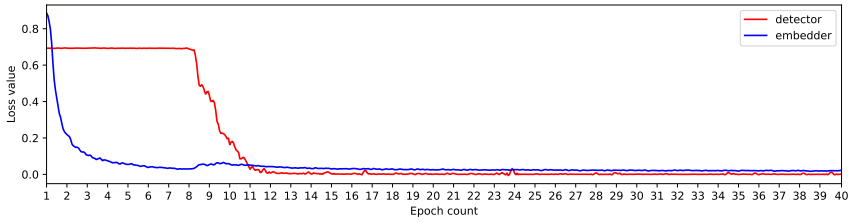
Type	Filters	Size/Stride	Output
Convolutional	8	5×5 / 1	32×32
Convolutional	16	5×5 / 1	32×32
Convolutional	32	5×5 / 1	32×32
Convolutional	64	3×3 / 2	16×16
Convolutional	128	3×3 / 2	8×8
Convolutional	256	1×1 / 2	4×4
-watermark embedding-			
Convolutional	256	1×1 / 1	4×4
Transposed Convolutional	128	1×1 / 2	8×8
Transposed Convolutional	64	3×3 / 2	16×16
Transposed Convolutional	32	3×3 / 2	32×32
Transposed Convolutional	16	5×5 / 2	256×32
Transposed Convolutional	8	5×5 / 1	32×32
Convolutional	3	5×5 / 1	32×32

Table 2: The specifications for the design of the detector. The symbol L_w stands for the size of the watermark, which is indicated by the number of bits in the watermark.

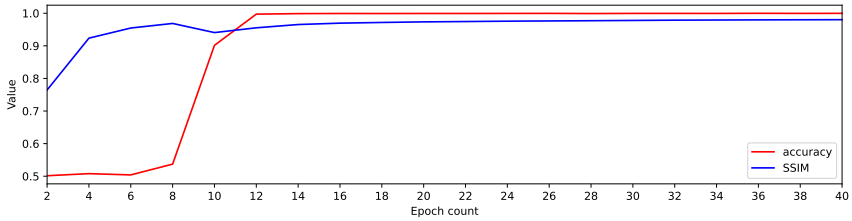
Type	Filters	Size/Stride	Output
Convolutional	32	5×5 / 1	32×32
Convolutional	32	5×5 / 1	32×32
Convolutional	64	3×3 / 2	16×16
Convolutional	64	3×3 / 2	16×16
Convolutional	64	3×3 / 2	8×8
Convolutional	128	3×3 / 2	4×4
Convolutional	128	3×3 / 2	2×2
Fully Connected		$512 \times L_w$	L_w

3. Results

The proposed solution achieves satisfactory results in the terms of image reconstruction and watermark extraction accuracy. The average SSIM is 0.98 while the accuracy is 99.99%. Accuracy and SSIM vary from epoch to epoch. These metrics are shown in Figure 1b. At the start of the training, the embedder and the detector are unbalanced and the detector is not able to detect watermark bits which drastically changes during the 9th epoch after which the quality of the watermarked



(a)



(b)

Figure 1: (a) Embedder and the detector losses as a function of the current epoch, (b) accuracy and SSIM values as a function of the current epoch. Source: own work.

image becomes a priority and is slowly improved. Following this starting period the metrics are converging together – an increase in accuracy does not result in a decrease in SSIM and vice versa. It can be concluded that the system reached optimum. Randomly sampled watermarked and original images can be seen in Figure 2.

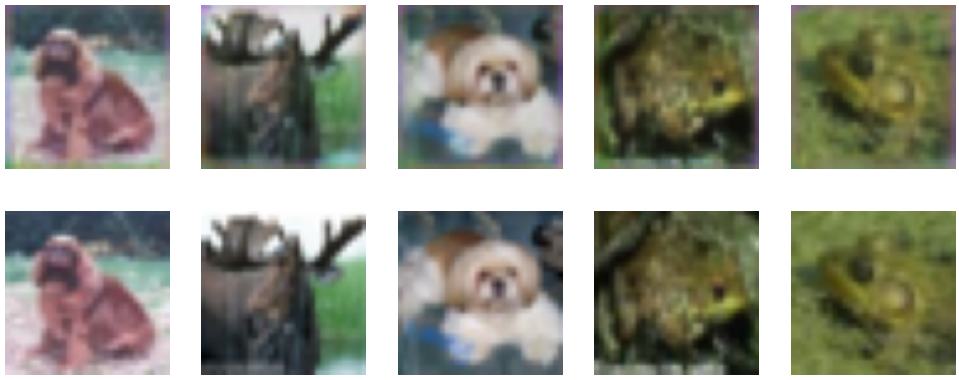


Figure 2: Five randomly sampled watermarked images (first row) and original images (second row). Source: own work.

4. Conclusion

We proposed a DNN architecture capable of hiding the watermark within an image. The embedder creates a latent space representation of the raw input image and embeds the watermark in it. A watermarked image is passed to the detector which can detect watermark bits. The system represents a proof of concept that demonstrates that an autoencoder could be used in image watermarking. Our future work will concentrate on lengthening the watermark message, introducing image manipulation attacks, and testing the system on a more diverse dataset.

References

[1] Cox I.J., Kilian J., Leighton F.T., Shamoon T., *Secure spread spectrum watermarking for multimedia*, *IEEE Transactions on Image Processing*, 1997, vol. 6, no 12, pp. 1673–1687.

[2] Djurović I., Stanković S., Pitas I., *Digital watermarking in the fractional Fourier transformation domain*, *Journal of Network and Computer Applications*, 2001, vol. 24, no 2, pp. 167–173, ISSN 1084-8045.

[3] Huang Y., Niu B., Guan H., Zhang S., *Enhancing image watermarking with adaptive embedding parameter and PSNR guarantee*, *IEEE Transactions on Multimedia*, 2019, vol. 21, no 10, pp. 2447–2460.

[4] Xiao D., Zhao A., Li F., *Robust watermarking scheme for encrypted images based on scrambling and Kronecker compressed sensing*, *IEEE Signal Processing Letters*, 2022, vol. 29, pp. 484–488.

[5] Liu Y., Xia C., Zhu X., Xu S., *Two-stage copy-move forgery detection with self deep matching and proposal superglue*, *IEEE Transactions on Image Processing*, 2022, vol. 31, pp. 541–555.

[6] Cui Z., Bao C., *Power exponent based weighting criterion for DNN-based mask approximation in speech enhancement*, *IEEE Signal Processing Letters*, 2021, vol. 28, pp. 618–622.

[7] Lopac N., Hržić F., Vuksanović I.P., Lerga J., *Detection of non-stationary gw signals in high noise from cohen's class of time–frequency representations using deep learning*, *IEEE Access*, 2022, vol. 10, pp. 2408–2428.

[8] Mun S.M., Nam S.H., Jang H., Kim D., Lee H.K., *Finding robust domain from attacks: A learning framework for blind watermarking*, *Neurocomputing*, 2019, vol. 337, pp. 191–202, ISSN 0925-2312.

[9] Kandi H., Mishra D., Gorthi S.R.S., *Exploring the learning capabilities of convolutional neural networks for robust image watermarking*, *Computers & Security*, 2017, vol. 65, pp. 247–268, ISSN 0167-4048.

[10] Zhu J., Kaplan R., Johnson J., Fei-Fei L., *Hidden: Hiding data with deep networks*, [In:] *15th European Conference*, Munich, Germany, ISBN 978-3-030-01266-3, pp. 682–697.

[11] Pavlović K., Kovačević S., Djurović I., Wojciechowski A., *Robust speech watermarking by a jointly trained embedder and detector using a DNN*, *Digital Signal Processing*, 2021, vol. 122, p. 103381, ISSN 1051-2004.

[12] Krizhevsky A., *Learning multiple layers of features from tiny images*, Tech. rep., University of Toronto, 2009, (access: 12-07-2023).
https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[13] Wang Z., Bovik A., Sheikh H., Simoncelli E., *Image quality assessment: from error visibility to structural similarity*, *IEEE Transactions on Image Processing*, 2004, vol. 13, no 4, pp. 600–612.

Autoregressive Label-Conditioned Autoencoder for Controllable Image-To-Video Generation

Kacper Kubicki[1][0000−0002−1811−8395], **Krzysztof Ślot**[1][0000−0003−1228−0970]

[1]*Lodz University of Technology*
Institute of Applied Informatics
Stefanowskiego 18/22, 90-537 Łódź, Poland
kacper.kubicki@p.lodz.pl

DOI:10.34658/9788366741928.49

Abstract. *Generating videos from a single image with user-controlled attributes is a complex challenge in the field of computer vision, despite the significant advancements recently made in the field. This paper presents a novel approach to tackle this issue, leveraging a convolutional autoencoder with supervised principal component analysis and autoregressive inference step. The efficacy of the proposed method is evaluated on two datasets – MNIST handwritten-digits and time-lapse photos of the sky. Results from both quantitative and qualitative analyses show that the proposed approach produces high-quality videos of variable duration with user-defined attributes, while preserving the integrity of original image contents.*
Keywords: *Video generation, Image-To-Video, Autoencoder, Supervised PCA*

1. Introduction

In recent years, the field of video generation has witnessed remarkable progress, particularly in the area of user-controlled video generation. Early research in video generation, much like in image generation, focused on unconditioned approaches [1, 2, 3]. However recently, tremendous advances in the field of controllable video generation have been made, especially by leveraging the text-guided approach [4, 5]. Other approaches include e.g. landmark-based guidance [6], trajectory-based manipulation [7] or pixel-level manipulation [8]. Although text-based descriptions offer a natural interface for users to describe appearance and motion in the video to be generated [9], such descriptions inherently rely on large language models to provide the language embeddings necessary to guide generation process. Large-scale nature of the best-performing language models, however, renders training or using these models challenging and computationally-demanding. As a result, there is a need for novel video generation approaches that can provide users with greater control over the generated videos while minimizing reliance on complex external architectures.

The presented paper extends our previous work [10] and introduces a novel approach for autoregressive generation of arbitrary-length user-controlled video sequences based on a single input image, using a convolutional autoencoder (CAE) and latent space transformation based on supervised principal component analysis (SPCA) that enables attribute manipulation.

2. Method

The proposed method comprises a few data processing steps. The first one is concerned with CAE training, where the objective is to provide correct reconstruction of videos, containing some selected semantic object category (we use MNIST digits and sky textures):

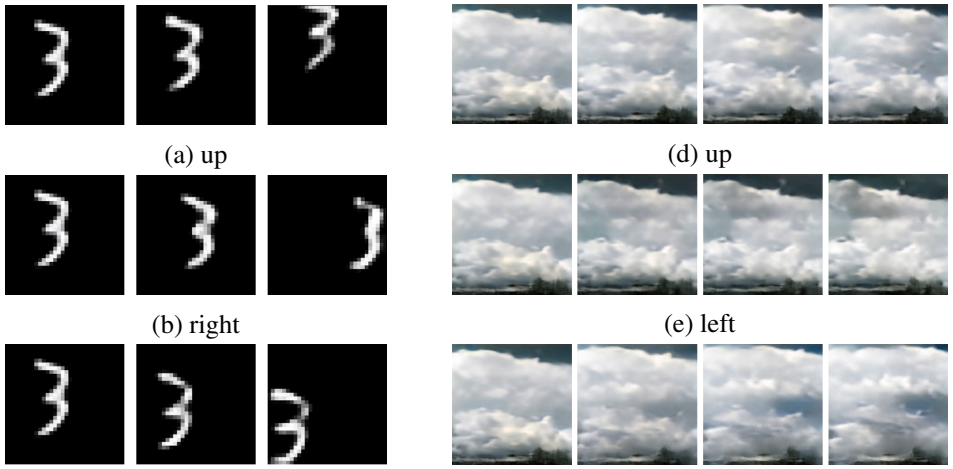
$$\theta^* = \arg\min_{\theta} \sum_{v} \sum \|\mathbf{X}^v - \hat{\mathbf{X}}^v\| \tag{1}$$

where summation is done over all v input videos and all pixels, '||' denote L2-norm, $\mathbf{X}, \hat{\mathbf{X}} \in \mathbb{R}^{L\times H\times W\times C}$ denote input and reconstructed video sequences composed of L frames each, and θ denote CAE parameters. Given the pretrained CAE, Supervised Principal Component Analysis (SPCA) is performed on its latent space to relate selected video attributes (for example, direction of optical flow) and the corresponding input sequence. As leading SPCA eigenvectors provide (possibly disentangled) information on the considered attribute intensity, purposeful modifications of their components implement the assumed video contents control. To enable generation of videos of arbitrary length, we additionally introduce autoregressive scheme for setting-up CAE training sequences.

Motivated by recent work [11] we decided to train the autoencoder in an adversarial manner by introducing a discriminator network $\mathcal{D}$, as well as to use the perceptual loss [12] in order to increase the perceptual quality of the reconstructed video sequences. We use a ResNet-18 architecture in both encoder $\mathcal{E}$ and discriminator $\mathcal{D}$, and a stack of convolutional and sub-pixel convolutional [13] layers with GELU activations in decoder $\mathcal{G}$. The perceptual loss is calculated between the feature maps extracted from the second convolutional layer of an ImageNet-finetuned VGG19 network. We train the autoencoder with batch sizes of 32, using an Adam optimizer with a learning rate of 3×10^{-4} and discriminator $\mathcal{D}$ using the same optimizer, with a learning rate of 3×10^{-6}.

3. Results

We trained the CAE on two distinct datasets: **(a)** an MNIST-based dataset with animated images of handwritten digits (bouncing off canvas edges), and **(b)** a large-scale time-lapse video dataset featuring diverse motion patterns and complex



(a) up (d) up

(b) right (e) left

(c) down-left (f) up-right

Figure 1: Selected frames from generated video sequences for (left) MNIST-based and (right) sky time-lapse dataset, with modification of video contents: motion in various directions (a-f) induced by SPCA transformation. Source: own work.

Table 1: Experimental evaluation of MoCoGAN [3], DTVNet [17] and our method on Sky Time-lapse dataset in terms of SSIM. The upward-pointing arrow signifies that as the value increases, the model performance improves.

Method	SSIM ↑
MoCoGAN [3]	0.849
DTVNet [17]	0.916
Ours	0.879

contents of sky scenes [14]. The videos belonging to both datasets were divided into 8-frame sequences, and labeled using either **(a)** a mean translation in both vertical and horizontal directons or **(b)** horizontal and vertical components of a mean vector of dense optical flow calculated between the consecutive sequence frames using Farnebäck's method [15]. The selected frames from the generated video sequences are presented in Fig. 1.

Apart from the qualitative results, we assessed performance of the trained CAE by calculating a structural similarity index measure (SSIM) [16] between the source frame and generated video frames from the Sky Time-lapse test dataset and compared them with other works using the same dataset. The results, presented in Table 1, show that the generated videos exhibit high consistency – indicated by high SSIM value – that is comparable with the state-of-the-art methods.

4. Conclusions

The presented paper describes a novel approach for generating user-controllable videos from a single source image, which does not require computationally expensive language models for guiding video-generation process. The proposed method has proved successful, both in terms of objective and subjective assessment, in generating clips displaying motion, both for artificial as well as natural contents.

Looking forward, we propose that future research should focus on extending the method's capabilities to enable manipulation of other content attributes, such as for example, scene appearance that can be quantified by a variety of descriptors including mean illumination, shadowing or style.

References

[1] Vondrick C., Pirsiavash H., Torralba A., *Generating Videos with Scene Dynamics*, [In:] D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett (eds.), *Advances in Neural Information Processing Systems*, vol. 29, Curran Associates, Inc., Barcelona 2016.

[2] Saito M., Matsumoto E., Saito S., *Temporal generative adversarial nets with singular value clipping*, [In:] *Proceedings of the IEEE international conference on computer vision*, pp. 2830–2839.

[3] Tulyakov S., Liu M.Y., Yang X., Kautz J., *MoCoGAN: Decomposing Motion and Content for Video Generation*, [In:] *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1526–1535.

[4] Molad E., Horwitz E., Valevski D., Acha A.R., Matias Y., Pritch Y., Leviathan Y., Hoshen Y., *Dreamix: Video Diffusion Models are General Video Editors*, *arXiv preprint arXiv:2302.01329*, 2023.

[5] Singer U., Polyak A., Hayes T., Yin X., An J., Zhang S., Hu Q., Yang H., Ashual O., Gafni O., Parikh D., Gupta S., Taigman Y., *Make-A-Video: Text-to-Video Generation without Text-Video Data*, [In:] *International Conference on Learning Representations*.

[6] Yang C., Wang Z., Zhu X., Huang C., Shi J., Lin D., *Pose guided human video generation*, [In:] *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 201–216.

[7] Hao Z., Huang X., Belongie S., *Controllable Video Generation with Sparse Trajectories*, [In:] *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7854–7863.

[8] Blattmann A., Milbich T., Dorkenwald M., Ommer B., *Understanding object dynamics for interactive image-to-video synthesis*, [In:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5171–5181.

[9] Wu C., Huang L., Zhang Q., Li B., Ji L., Yang F., Sapiro G., Duan N., *Godiva: Generating open-domain videos from natural descriptions*, *arXiv preprint arXiv:2104.14806*, 2021.

[10] Ślot K., Kapusta P., Kucharski J., *Autoencoder-based image processing framework for object appearance modifications*, *Neural Computing and Applications*, 2021, vol. 33, pp. 1079–1090.

[11] Esser P., Rombach R., Ommer B., *Taming transformers for high-resolution image synthesis*, [In:] *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883.

[12] Larsen A.B.L., Sønderby S.K., Larochelle H., Winther O., *Autoencoding beyond pixels using a learned similarity metric*, [In:] *International conference on machine learning*, PMLR, pp. 1558–1566.

[13] Shi W., Caballero J., Huszár F., Totz J., Aitken A.P., Bishop R., Rueckert D., Wang Z., *Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network*, [In:] *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1874–1883, 2016.

[14] Xiong W., Luo W., Ma L., Liu W., Luo J., *Learning to Generate Time-Lapse Videos Using Multi-Stage Dynamic Generative Adversarial Networks*, [In:] *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] Farnebäck G., *Polynomial expansion for orientation and motion estimation*, Ph.D. thesis, Linköping University Electronic Press, 2002.

[16] Wang Z., Bovik A.C., Sheikh H.R., Simoncelli E.P., *Image quality assessment: from error visibility to structural similarity*, *IEEE transactions on image processing*, 2004, vol. 13, no 4, pp. 600–612.

[17] Zhang J., Xu C., Liu L., Wang M., Wu X., Liu Y., Jiang Y., *Dtvnet: Dynamic time-lapse video generation via single still image*, [In:] *European Conference on Computer Vision*, Springer, pp. 300–315.

Building Energy Use Intensity Prediction with Artificial Neural Networks

Kamil Stokfiszewski[1][0000−0002−2707−7353], **Przemysław Sztoch**[2], **Ryszard Sztoch**[2], **Agnieszka Wosiak**[3][0000−0001−6124−1236]

[1,3]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
[1]*kamil.stokfiszewski@p.lodz.pl*,[3]*agnieszka.wosiak@p.lodz.pl*
[2]*FINN Sp. z o.o., Wrońsko 1A, 98-313 Konopnica, Poland*
[2]*sekretariat@finn.pl*

DOI:10.34658/9788366741928.50

Abstract. *In this paper the authors propose the construction and examine the performance of the artificial neural network for energy use intensity prediction for residential buildings. The network's type is the standard multilayer perceptron and its training dataset contains the data of 768 residential buildings where the training pattern for an individual building consists of 8 parameters describing the building's geometry along with its lighting and glazing conditions while the only output value is the building's actual energy use intensity characteristics. Experimental study shows that the mean absolute percentage error of prediction of the energy use intensity evaluated for buildings data present in the network's test set does not exceed 1.8%, what might be considered a highly satisfactory result.*
Keywords: *energy use intensity, neural networks, statistical prediction*

1. Introduction

In recent years buildings energy performance has become an increasingly vital issue in the context of energy saving policy undertaken by the European countries, what is legally reflected in numerous European Commission directives, such as e.g. [1]. This has led to an considerable growth of interest in the research aiming at optimization of building design in terms of energy use efficiency, especially for the case of the most energy consuming internal building installations such as heating, ventilation and air conditioning (HVAC) systems, see e.g. [2, 3]. In this research area statistical analysis methods, such as those presented in [3, 4], and especially artificial neural networks (ANNs), turned out to be particularly useful tool for the analysis and prediction of buildings energy use efficiency, see [2, 5, 6, 7].

Encouraged by the results regarding the use of the ANNs for prediction of buildings energy use efficiency, presented in the aforementioned works, the authors decided to verify the effectiveness of neural network approach in the task of predicting the specific type of the indicator, i.e. the building's *Energy Use Intensity*[1] (EUI), in whose prediction the authors are particularly interested, in the context of the ongoing research project [8], which the authors are engaged in.

In this article we present the ANN architecture for EUI prognostics along with detailed descriptions of the dataset used and experimental prediction results.

2. Dataset description

In this section we present the detailed dataset description which was used in our experiments. Brief presentation of the network's input and output parameters for a single training pattern is presented below in Tab. 1.

Table 1. Description of the dataset used for neural network training and validation.

Variable type	Building parameter	Attribute name	Data type	Unit
Input	Relative compactness	x_1	`float`	none
	Surface area	x_2	`float`	m^2
	Wall area	x_3	`float`	m^2
	Roof area	x_4	`float`	m^2
	Overall height	x_5	`float`	m
	Orientation	x_6	`int`	none
	Glazing area	x_7	`float`	m^2
	Glazing area distribution	x_8	`int`	none
Output	Energy use intensity	y	`float`	$kWh/(m^2 \cdot year)$

The used dataset was created and made publicly available by T. Athanasios and A. Xifara as the part of their work published in [3]. The dataset consists of characteristics of 768 different residential buildings, modeled in *Autodesk Ecotec* design software, each described by 8 different input parameters, gathered in Tab. 1 and denoted by symbols x_i, and 2 output values, namely the required minimum modeled heating and cooling loads of the HVAC systems installed inside the considered buildings.[2] In order to adopt the original data [3] to our particular needs we only had to recalculate the dataset output values of heating and cooling loads to form a

[1] In Polish nomenclature the EUI coefficient is a exactly equivalent to so called *areal WWE* indicator (*pl.* powierzchniowy Wskaźnik Wyniku Energetycznego)

[2] The detailed description of the dataset parameters can be found in the mentioned work [3].

single EUI indicator (denoted in Tab. 1 by symbol y) using the following formula:

$$EUI_k = 100 \cdot E_f^{-1} \cdot \ 0.2931 \cdot 10^{-3} \cdot 24 \cdot 365 \cdot (HL_k + CL_k) , \qquad (1)$$

where for $k = 1, \ldots, 768$, the heating and cooling loads, HL_k and CL_k respectively, for each building were originally given in BTUs (British thermal units), E_f is the HVAC system effectiveness coefficient (which was set to a constant value of 85%) and the resulting EUI factor is given in $kWh \ / \ (m^2 \cdot year)$ units. The 8 input parameters, present in Tab. 1, of the considered dataset were left unchanged. After such preparation the dataset was randomly divided in to the training set, consisting of 70% of the input-output data records (i.e. 537 training patterns) and the test set covering 30% of the rest of the data records (i.e. 231 test patterns). Prepared in such way the set was directly used in the process of training and validation of the proposed neural network.

3. Neural network architecture

For our investigation we have chosen the standard multilayer perceptron neural network whose simplified schematic is depicted in Fig. 1. The proposed ANN consisted of an input layer whose copying neurons propagated 8 building input parameters to the neurons of the first hidden layer, 2 hidden layers, each consisting of 240 biased neurons with ReLU activation functions, and a single biased output neuron with identity activation function meant to code the output EUI indicator for a given building present in the dataset. The network was implemented using `Keras` high-level neural network programming library for `Python` programming language [9]. The network undergone the training process which took 5000 epochs and used the standard `Adam` optimization method with learning rate $\mu = 10^{-3}$. During the training process the mean squared error (MSE) has been chosen as the loss function and mean absolute percentage error (MAPE) served as the accuracy metrics. For the sake of clarity the respective formulas for a single training (or test) epoch of both of the mentioned measures are given below in Equ. (2):

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2, \quad MAPE = \frac{1}{N} \sum_{i=1}^{N} \frac{| y_i - \hat{y}_i |}{\hat{y}_i} \cdot 100 \ [\%] , \qquad (2)$$

where $\hat{y}_i$ and y_i, $i = 1, \ldots N$, are the desired and the obtained network output values, respectively. In the next section we present the results of the conducted experimental study.

4. Experimental results

In order to validate the effectiveness of the proposed network in the task of the buildings' EUI indicators prediction on the basis of their geometry, construction

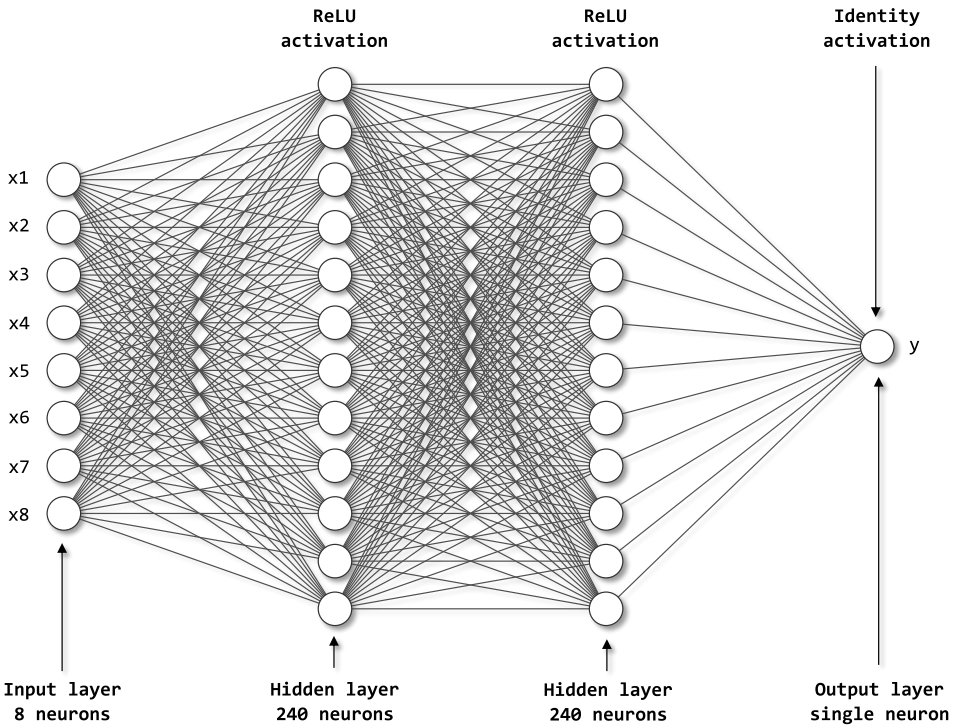


Figure 1. Architecture of the neural network for prediction of energy use intensity. Source: own work.

characteristics and lighting conditions data, we have performed experimental study involving the training and test datasets, described earlier in detail in Section 2. After the training process, whose conditions were comprehensively characterized in Section 3, we arrived at the final MAPE measure for the predicted buildings' EUI not exceeding the value of 1.8% calculated for all of the 231 patterns present in the test set. A brief summary of the training and test processes computational loads as well as their time effectiveness[3] is gathered below in Tab. 2

Table 2. Network's training and validation computational loads and the results.

Num. of training epochs	Num. of training parametres	Num. of training patterns	Num. of test patterns	EUI MAPE for the test set
5000	60241	537	231	1.8%
Training time	94s	Test Time	2s	

[3]All of the computations were performed using `Keras` neural network library configured for the standard CPU operation running on top of `Python 3.9.13` programming language on the system equipped with 2.20 GHz Intel(R) Core(TM) i7-8750H CPU.

MAPE training curve with the scatter plot of the predicted vs. actual values of the test set buildigs' EUIs are shown below in Fig. 2.

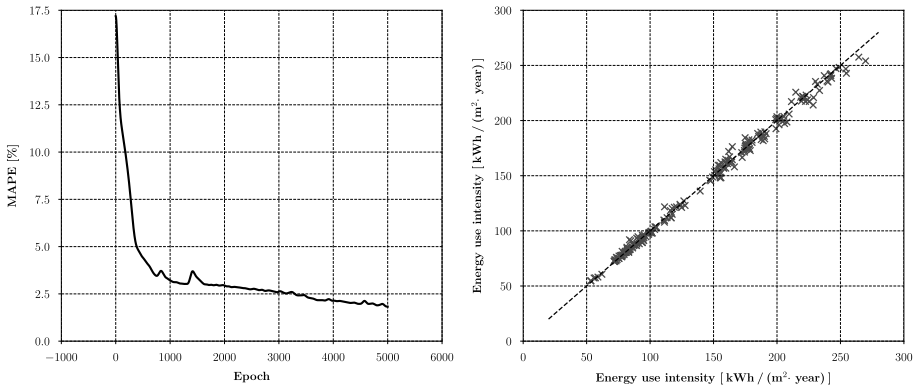


Figure 2. On the left: training curve for the considered neural network with mean absolute percentage error (MAPE) as the training error metrics. On the right: neural network's validation scatter plot of the predicted (horizontal axis) versus actual (vertical axis) values of the test buildings' Energy Use Intensities (EUIs). Source: own work.

5. Conclusions

In their present work the authors examined the construction and performance of the artificial neural network for energy use intensity prediction in residential buildings. The results show that the proposed neural network not only gives highly satisfactory results regarding the EUI prediction, with the MAPE not exceeding the value of 1.8% for the validation dataset, but is also effective in terms of computational workloads and time characteristics of both training and operational stages.

Acknowledgment

The authors would like to thank FINN Sp. z o.o., Wrońsko 1A, 98-313 Konopnica, for their help in conducting this research, which was financed by the National Center for Research and Development project no. POIR.01.01.01-00-0281/20-00, entitled: "Predictive energy management system EnMS (PE)".

References

[1] European Commission, *Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings*, 2010.

[2] Zhang H., Feng H., Hewage K., Arashpour M., *Artificial neural net- work for predicting building energy performance: A surrogate energy retrofits decision support framework*, *Buildings*, 2022, vol. 12, no 6, p. 829, doi: https://doi.org/10.3390/buildings12060829.

[3] Tsanas A., Xifara A., *Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools*, *Energy and Buildings*, 2012, vol. 49, pp. 560–567.

[4] Sajjad M., Khan S., Khan N., Haq I., Ullah A., Lee M., Baik S., *Towards efficient building designing: Heating and cooling load prediction via multi-output model*, *Sensors*, 2020, vol. 20, no 22, p. 6419, doi: https://doi.org/10.3390/s20226419.

[5] Chari A., Christodoulou S., *Building energy performance prediction using neural networks*, *Energy Efficiency*, 2017, vol. 10, pp. 1315–1327.

[6] Khalil A., Barhoom A., Abu-Nasser B., Musleh M., Abu-Naser S., *Energy efficiency prediction using artificial neural network*, *International Journal of Academic Pedagogical Research (IJAPR)*, 2019, vol. 3, no 9, pp. 1–7.

[7] Sun M., Han C., Nie Q., Xu J., Zhang F., Zhao Q., *Understanding building energy efficiency with administrative and emerging urban big data by deep learning in glasgow*, *Energy and Buildings*, 2022, vol. 273, p. 112331, doi: https://doi.org/10.1016/j.enbuild.2022.112331.

[8] *National Center for Research and Development project no. POIR.01.01.01-00-0281/20-00, Predictive energy management system EnMS (PE)*, 2022.

[9] *Keras neural network programming library*, (access: 23-02-2023). https://keras.io/

Grounded HyperSymbolic Representations Learned through Gradient-Based Optimization

Piotr Łuczak[0000−0002−2530−0283], **Krzysztof Ślot**[0000−0003−1228−0970], **Jacek Kucharski**[0000−0002−8704−1950]

Lodz University of Technology
Institute of Applied Computer Science
Stefanowskiego 18, 93-537 Łódź, Poland
pluczak@iis.p.lodz.pl

DOI:10.34658/9788366741928.51

Abstract. *Hyperdimensional computing is a novel paradigm, capable of processing complex data structures with simple operations. Its main limitations lie in the conversion of real world data onto hyperdimensional space, which due to lack of a universal translation scheme, oftentimes requires application-specific methods. This work presents a novel method for unsupervised hyperdimensional conversion of arbitrary image data. Additionally, this method is augmented by the ability of creating HyperSymbols, or class prototypes, provided that such class labels are available. The proposed method achieves promising performance on MNIST dataset, both in translating individual samples as well as producing HyperSymbols for downstream classification task.*

Keywords: *artificial intelligence, hyperdimensional computing, representation extraction, neuromorphic architectures*

1. Introduction

The problem of translating latent data representations into meaningful symbols is of paramount importance to developing real world applications of hyperdimensional computing. While capable of processing a diverse range of data types and structures in a unified manner, hyperdimensional computing [1] is currently limited by the need for custom methods of converting real world data into hyperdimensional space. Since autoencoders have attained wide-spread adoption in the domain of representation learning, they provide an ideal starting point for the construction of universal hyperdimensional encoders.

2. Related work

Hyperdimensional computing (HDC, a.k.a. Vector Symbolic Architectures), proposed by Pentti Kanerva [1] is a computational paradigm based on long (around 10^4 bits), random binary or bipolar vectors. By leveraging the properties of high-dimensional spaces (hypervectors can be thought of as vertices of a hypercube) and spreading the encoded information over all bits, this bio-inspired framework can efficiently perform complex data processing using only a few operators. Due to the simplicity of hardware implementation and resilience to noise, hyperdimensional computing has found application in domains such as event-based vision, EEG classification [2] or language classification[3]. HDC is an attractive alternative or complement to deep neural networks [4], particularly in the case of deployment to edge devices due to its ease of hardware implementation [2].

A crucial problem with hyperdimensional representation is in translating real-world data into hyperdimensional space. While a lot of classic data structures can be build with proper application of basic hyperdimensional operations, more complex data, such as images, typically require application of representation learning techniques [5]. Some successful applications have leveraged direct naive coding of values and their positions into latent vectors [6], yet the currently dominant family of methods is based on vector quantization [7]. This can be implemented e.g. by means of Sparse Distributed Memory (SDM), proposed by Pentti Kanerva [8]. As in the case of hyperdimensional vectors, stored data is distributed over the entire contents matrix $\mathbf{C}$, providing similar structural noise resilience. Both writing to ($\mathbf{C} := \mathbf{C} + ((\mathbf{A}\mathbf{x}_{addr}) \geq d)\mathbf{x}_{data}^T$) and reading from ($\mathbf{x}_{data} = sign(\mathbf{C}^T((\mathbf{A}\mathbf{x}_{addr}) \geq d))$) the SDM are based on the similarity between $\mathbf{x}_{addr}$ and the address matrix $\mathbf{A}$.

3. Methods

The main contribution of this work is a novel method for deriving hyperdimensional data representation, based on a modified Autoencoder than can be trained using standard gradient-based methods, as shown in Fig.1. An ancillary contribution of this work is a proposal of the simplified, bidirectional version of SDM.

The mapping from latent space to hypervectors is simply a linear composition of seed hypervectors stored in the value book $\mathbf{V} \in \mathbb{B}^{n \times v}$, with weights based on the similarity of a latent vector $\mathbf{k} \in \mathbb{Z}^{k \times 1}$ to learned values in the key book $\mathbf{K} \in \mathbb{Z}^{n \times k}$:

$$\mathbf{v} = \tanh(\mathbf{V}^T(\mathbf{k} \approxeq \mathbf{K})) \tag{1}$$

The inverse mapping, that is moving from the hyperdimensional space to Autoencoder's latent space, follows the same principle. In this case the output is a linear composition of latent vectors from the key book based on the similarity between

hypervector $\mathbf{v} \in \mathbb{B}^{\nu \times 1}$ and the value book:

$$\mathbf{k} = \mathbf{K}^T softmax(\mathbf{V}\mathbf{v}) \tag{2}$$

The complete structure, showing the data flow through the proposed model is shown in Fig.1.

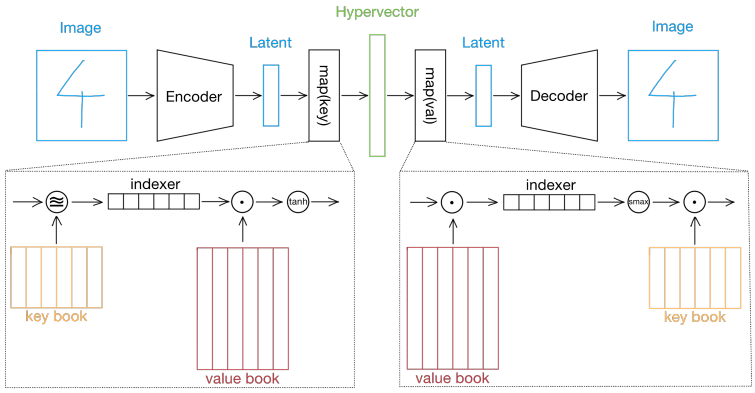


Figure 1: Structure of the hyperdimensional autoencoder. Source: own work.

The final step of deriving the HyperSymbolic representation is grounding it, that is building hyperdimensional prototypes associated with each known label. The construction of a class prototype can be done by simply bundling hypervector representations derived for a set of training images with the same label. The number of components can be relatively low, oftentimes requiring only around 100 examples from each class.

While hypervectors can be compared using the dot product, a gradient-friendly method of measuring floating point vector similarity was needed to enable bidirectional mapping between floating point and hyperdimensional vectors. For this purpose, the relaxed equality operator (equation 3) [9] has been chosen and extended into vector version as the mean of elementwise computations.

$$a \cong b \equiv sech^2\left(\frac{b-a}{2\beta}\right) \tag{3}$$

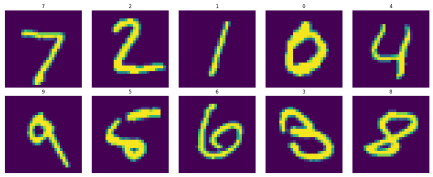
The support of this operator can be controlled by changing the β parameter. Given that the input to the hyperdimensional encoder has been produced by the sigmoid layer of the convolutional encoder, the value of β was tuned so that the furthest 20% of values were not matched.

4. Results

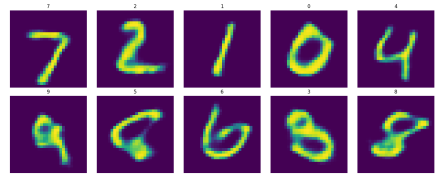
The proposed model has been evaluated on the MNIST dataset after being trained for 100 epochs with the PyTorch's implementation of the Adam optimiser.

Since the model had been designed to behave like a classic autoencoder, it was trained using mean square error loss. Data augmentation, learning rate schedulers and pretraining were omitted deliberately, in order to evaluate the learning ability of the hyperdimensional component without additional confounding factors.

The performance of the proposed model can be evaluated in a twofold manner. Firstly, the model can be assessed simply as an autoencoder, as shown in figure 2. The set of images in Fig. 2a has been randomly sampled from the testing set, while the set of images in Fig 2b is the output of the autoencoder for those images. As it can be observed, the input handwriting exhibits a number of deviations from the "ideal" digit shapes, however the autoencoder is capable of providing their legible reconstructions.



(a) Random sample of input data.



(b) Autoencoder output.

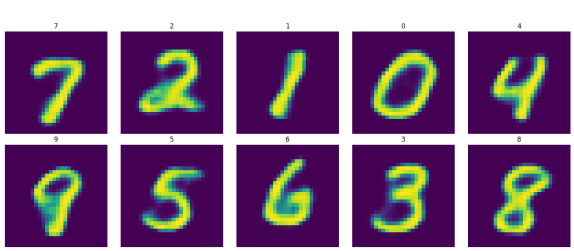
Figure 2: Autoencoder performance on a random sample of dataset. Source: own work.

Secondly, the model can be evaluated based on its ability to assemble well formed HyperSymbols, as shown in Fig.3. While the training dataset contained digits with varying degrees of nonideality, the prototypes constructed from random samples of 150 representatives of each digit, strongly resemble the "textbook" versions of digits.This indicates that the prototypes are indeed well formed, as the model was capable of extracting the "ideal" shapes of digits, from their non-ideal representatives. The result of decoding of these prototypes can be seen in fig. 3a.

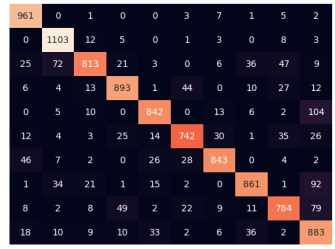
HyperSymbols can be used for a number of downstream applications, such as encoding expert knowledge. For the derived HyperSymbols, we test their generic nature using a task of image classification. As shown in Fig.3b, the hyperdimensional classifier achieves promising performance, with the average accuracy of 0.8725. While the classification performance is not perfect, the hyperdimensional component of the model was trained with only 150 occurrences of each digit, which is significantly less than required by most neural networks.

5. Conclusion

The proposed model enables simple conversion of real data into hyperdimensional space and vice versa, with simple, unsupervised gradient-based training. It enables easy inclusion of HDC paradigm in real world applications, without incur-



(a) Decoded hyperdimensional prototypes for all classes in MNIST dataset.



(b) Confusion matrix for hyperdimensional classification.

Figure 3: Performance of the hyperdimensional classifier. Source: own work.

ring the cost of developing custom, application-specific encoding schemes. Future work on this method could investigate the degree of transferability of trained encoders between similar domains, as well as the impact of state-of-the-art neural network training techniques and models on the quality of obtained HyperSymbols.

Acknowledgements

This work was funded by European Union's Horizon 2020 research and innovation programme under grant agreement no 101016734. This work has been completed while the 1st author was the Doctoral Candidate in the Interdisciplinary Doctoral School at the Lodz University of Technology, Poland.

References

[1] Kanerva P., *Hyperdimensional Computing: An Introduction to Computing in Distributed Representation with High-Dimensional Random Vectors*, *Cognitive Computation*, 2009, vol. 1, no 2, doi: 10.1007/s12559-009-9009-8.

[2] Rahimi A., Kanerva P., Benini L., Rabaey J.M., *Efficient Biosignal Processing Using Hyperdimensional Computing: Network Templates for Combined Learning and Classification of ExG Signals*, *Proceedings of the IEEE*, 2019, vol. 107, no 1, ISSN 1558-2256, doi: 10.1109/JPROC.2018.2871163.

[3] Joshi A., Halseth J.T., Kanerva P., *Language Geometry Using Random Indexing*, [In:] *Quantum Interaction*, Lecture Notes in Computer Science, Springer International Publishing, doi: 10.1007/978-3-319-52289-0_21.

[4] Łuczak P., Ślot K., Kucharski J., *Combining Deep Convolutional Feature Extraction with Hyperdimensional Computing for Visual Object Recognition*,

[In:] *2022 International Joint Conference on Neural Networks (IJCNN)*, ISSN 2161-4407, doi: 10.1109/IJCNN55064.2022.9892281.

[5] van den Oord A., Vinyals O., kavukcuoglu k., *Neural Discrete Representation Learning*, [In:] *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc.

[6] Nazemi M., Fayyazi A., Esmaili A., Pedram M., *SynergicLearning: Neural network-based feature extraction for highly-accurate hyperdimensional learning*, [In:] *Proceedings of the 39th International Conference on Computer-Aided Design*, ICCAD '20, ACM, doi: 10.1145/3400302.3415696.

[7] Mitrokhin A., Sutor P., Summers-Stay D., Fermüller C., Aloimonos Y., *Symbolic Representation and Learning With Hyperdimensional Computing*, *Frontiers in Robotics and AI*, 2020, vol. 7, ISSN 2296-9144.

[8] Kanerva P., *Sparse Distributed Memory and Related Models*, [In:] *Associative Neural Memories: Theory and Implementation*, Oxford University Press, ISBN 978-0-19-507682-0, 1993, pp. 50–76.

[9] Petersen F., *Learning with Differentiable Algorithms*, 2022, doi: 10.48550/arXiv.2209.00616.

Increasing Skin Lesions Classification Rates using Convolutional Neural Networks with Invariant Dataset Augmentation and the Three-Point Checklist of Dermoscopy

Piotr Milczarski[1[0000−0002−0095−6796]],
Norbert Borowski[1[0000−0002−3861−2411]],
Michał Beczkowski[2[0000−0001−5220−138X]]

[1]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
{piotr.milczarski, norbert.borowski}@p.lodz.pl
[2]*University of Lodz*
Faculty of Physics and Applied Informatics
Pomorska str. 149/153, 90-236 Łódź, Poland
michal.beczkowski@uni.lodz.pl

DOI:10.34658/9788366741928.52

Abstract. *In the paper, we show how to tackle the problem of lack of the rotation invariance in CNN networks using the authors' Invariant Dataset Augmentation (IDA) method. The IDA method allows to increase the classification rates taking into account as an example the classification of the skin lesions using a small image set. In order to solve the problem of the lack of rotation invariance, IDA method was used and the dataset was increased in an eightfold and invariant way. In the research, we applied the IDA methods and compared the results of VGG19, XN and Inception-ResNet-v2 CNN networks in three skin lesions features classification defined by well-known dermoscopic criterions e.g. the Three-Point Checklist of Dermoscopy or the Seven-Point Checklist. Due to Invariant Dataset Augmentation, the classification rate parameters like true positive rate by almost 20%, false positive rate as well as the F1 score and Matthews correlation coefficient have been significantly increased opposite to type II error that has significantly decreased. In the paper, the confusion matrix parameters result in: 98-100% accuracy, 98-100% true positive rate, 0.0-2.3% false positive rate, tests F1=0.95 and MCC=0.95. That general approach can provide higher results while using CNN networks in other applications.*

Keywords: *invariant dataset augmentation, dermoscopic images, blue-white veil, lesion symmetry, convolutional neural networks, artificial intelligence*

1. Introduction

The authors' motivation is to enhance screening methods by raising the classification probability of the positive features that point to the possible health problems. That general approach can provide higher results while using CNN networks in other disciplines not only dermatology. The results vary for different CCN networks but they can be used in a support system for the general practitioners.

The problem of melanoma and other skin illnesses is the fourth vital death problem in society according to European Cancer Information System (ECIS) [1] and American Cancer Society (ACS) [2].

In dermoscopy, the Three-Point Checklist of Dermoscopy (3PCLD)[3] and Seven-Point Checklist (7PCL) [4] is defined and it is proved to be a sufficient screening method in the skin lesions assessments during the checking by dermatology expert. In the 3PCLD method, there are three criteria: asymmetry of shape, hue and structure distribution within the lesion with discrete value either 0, 1 or 2; blue-white structures and atypical pigment network. Apart from 3PCLD and 7PCL there is the ABCD rule [5] that has its own methodology of the skin lesion assessment.

In the problem of data acquisition and data analysis, the feature extraction or classification should not depend on the object position within the image. Let us imagine that our sensor e.g. camera matrix can acquire the image with the features depending on the object rotation. That is why, we propose to use while testing by machine learning methods e.g. CNN networks, the original image and its invariant copies.

The rotation-invariance property is the network's capability to predict the class regardless of the object's orientation [6, 7]. Generally the CNN networks have problem with the rotation invariance and the lack of rotation-invariance is regarded as their weakness [7]. There are some solutions proposed to tackle the problem and provide the CNN networks and deep learning applications this property [8, 9, 10].

In computer science, there are feature-based and appearance-based approaches [4, 11, 12, 13, 14, 15, 16, 17, 18] that show around 70-95% accuracy in feature accuracy of the skin lesions feature assessments.

In the paper, we present how to tackle the problem of rotation invariance with the proposed Invariant Dataset Augmentation (IDA) [14, 15, 19] method. The IDA approach is used not only for data augmentation but also as a proliferation of the validation and test datasets and that results in the eightfold classification of the same image as a set of bytes but different due to its topology. The IDA method shows that the classification results and their confusion matrix parameters e.g. accuracy and true positive rate as well as the F1 test and MCC can be much higher than in the case when only original images are used as a test set. To show the advantages of the IDA method we have prepared the images from the PH2 dataset [20]. After preparing the dataset, several neural networks have been built with the

help of the available VGG19, Xception, and Inception-ResNet-v2 pretrained CNN networks. The networks are trained, validated and tested on the original images taken from the PH2 [20] datasets.

In the research, we achieved for the blue-white veil/symmetry classification using 3PCLD approach around 78-87% for the original PH2 dataset. But, the true positive rate (the blue-white veil is present) was between 65-80% in comparison to the true negative rate (the blue-white veil is absent) that varied from 95 to 98%. The reason for that is the fact that there are only around 20% of the images with the blue-white veil feature in the PH2 [20] dataset. However, after using the IDA approach, the achieved in the research average accuracy is around 94 % but the weighted accuracy is around 90% for the blue-white veil, while the sensitivity is even 20% higher than in a standard approach and reached even 98-100% for the images from PH2 dataset.

The paper is organized as follows. In Section 2 the Invariant Dataset Augmentation (IDA) is shown. The screening methods and dermoscopic datasets are discussed in Section 3. The research and its results are presented in Section 4. Finally, Section 5 shows the conclusions.

2. Invariant Dataset Augmentation

The Invariant Dataset Augmentation has been described in [14, 15, 19] and its foundation are based on the idea of how to make the available image dataset bigger as much as possible not changing the data within (i.e. pixels) and its order and relation to other data saved in pixels using the available geometrical invariant transformations. These transformations do not change the asymmetry of shape, hue, and structure distributions, as well as other features that are in the original copy of the image that is taken into account. Fig.1 shows the example of the original image IMD168 from PH2 set.

While data augmentation authors often use a rotation of scaling the image, but while looking for very fragile features in the images, these features can vanish after that procedure e.g. blue-whitish veil. We have chosen seven image transformations: rotation by 90°, 180° and 270°, mirror reflection by a vertical and horizontal axis of the images and their rotations by 90°. In the previous publications [14, 15, 19] we have shown that pretrained CNN networks have not rotation invariant property.

3. IDA and PH2 Dataset

In dermoscopy, the Three-Point Checklist of Dermoscopy (3PCLD) [3] is defined and regarded as a sufficient screening method. The dermatology experts proved it is a sufficient screening method of skin lesions assessments. There are

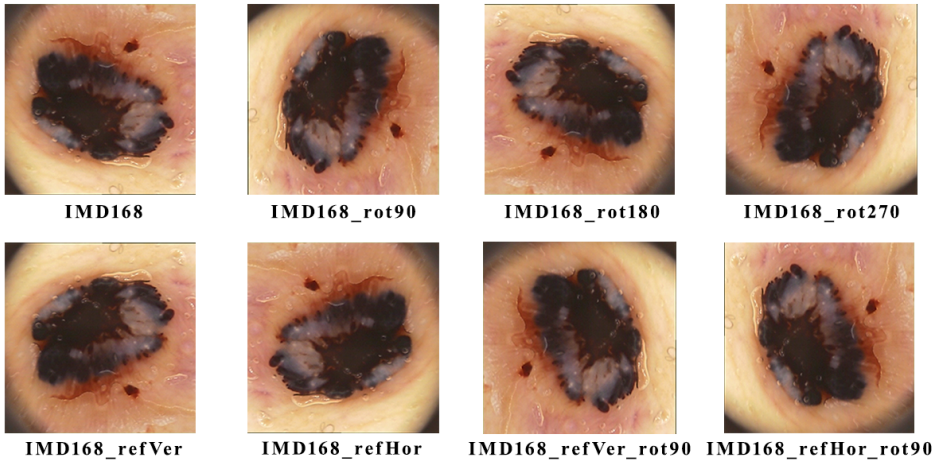


Figure 1. The image IMD168 from PH2 [20] and its invariant copies. Source: own work.

three criteria in the 3PLCD method: asymmetry of shape, hue, and structure distribution within the lesion with discrete value either 0, 1, or 2; blue-white structures and atypical pigment network. The Seven-Point Checklist (7PCL) [4] is another screening method used in skin lesions assessment. In the 7PCL case, dermatology experts examine pigment network, blue-white veil, streaks, pigmentation, regression structures, dots and globules, and vascular structures as the important assessment factors. The ABCD and ABCDE rules [5] are examples of the screening methods that take into account the symmetry/asymmetry, border, color, diameter, and evolving of the lesion.

These transformations used by IDA do not change the pixels, they are pixel invariant, mutually unambiguous, and reversible. Seven new copies for each image are achieved. Altogether, we have 1600 images out of 200 based on the PH2 dataset with:

- 288 images with present blue-white veil and 1312 without a blue-white veil;
- 936 fully symmetric images, 248 symmetric in 1 axis and 416 fully asymmetric;
- 928 images with atypical pigment network and 672 with typical one.

4. Results

In order to show the advantages of IDA method we have compared this methodology with training and testing on the same image sets several times using the IDA

transformed images. The set was split into: 75% of the images for the train and validation set; 25% for the test one which. The train set has IDA copies only. Some of the results were presented i.e. symmetry property in [15] and blue-white veil in [14, 19].

The research has been conducted using Matlab 2022b with up-to-date versions of Deep Learning Toolbox™. To achieve a high validation accuracy i.e. 100% or close and a low validation loss i.e. around 0.5 or less we have trained the networks using 30 epochs. The higher values of the epochs have not changed the validation accuracy and the loss as well as the testing values of the accuracy, true positive rate, etc. To achieve the highest classification rates we have tested the wide variety of the following parameters: 30-100 epochs; learning rate from 5e-5 to 1e-2. The times of training have varied from around 12 minutes for VGG19, 40 minutes for XN, and 60 minutes for IRN2 for PH2 dataset.

The confusion matrix factors are used for the chosen CNN network and they are calculated according to the well-known equations [10, 19]. In the T1 and T8 columns, the results of the tests are shown using each image obtained from the IDA procedure as a separate one. In Tables 1 and 2 the classification results for the blue-white veil training on PH2 dataset using IDA transformations with the test results for a single original image set (T1), augmented dataset with 8 copies of the original file (T8), and using the worst-case scenario (IDA) classification are presented. The chosen confusion matrix factors: true positive rate (TPR), false positive rate (FPR) with their average (AVG), variance (VAR), minimum (MIN) and maximum (MAX) values and F1 score and Matthews correlation coefficient (MCC) for the chosen CNN network. The chosen confusion matrix factors true positive rate for full asymmetry (TPR0), true positive rate for symmetry in one axis (TPR1), true positive rate for full symmetry (TPR2), false positive rate for full asymmetry (FPR0), false positive rate for symmetry in one axis (FPR1), false positive rate for full asymmetry full symmetry (FPR2) with their average (AVG), variance (VAR), minimum (MIN) and maximum (MAX).

The original authors' approach shows that the classification characteristics like accuracy and true positive rate as well as the F1 and MCC tests can be much higher (5-20%) than using only original images. In the paper for the reaching 98-100% accuracy, 98-100% true positive rate, 0.0-2.3% for a minimum of a false positive rate with tests F1=0.95 and MCC=0.95 as well as AUC=1. For the blue-white veil the area under curve (AUC) for the receiver operating characteristic curve (ROC) the average values are 0.966, 0.969 and 0.957, as well as the maximum values 0.996 for VGG19, 0.985 for Xception and 0.971 for Inception-ResNet-v2.

In the case of the symmetry/asymmetry research, the IDA method turned out to be in many cases more effective than training the network only on the original images, even by 20% higher. In the dermatological asymmetry studies, we achieved for VGG19, the best CCN network, a maximum of 68.56% weighted accuracy, 92.3% true positive rate, tests F1 = 0.56 and MCC = 0.62, AUC = 0.965, and a

Table 1. The results for the blue-white veil classification

CM factor		VGG19			Xception			Inception-ResNet-v2		
		T1	T8	IDA	T1	T8	IDA	T1	T8	IDA
w.ACC [%]	AVG	87.2	88.1	92.2	82.2	80.2	88.0	79.8	78.2	83.9
	MAX	98.8	97.5	100	87.7	85.3	94.4	88.9	86.1	93.2
TPR [%]	AVG	77.8	79.6	89.4	66.7	61.8	81.7	62.2	58.4	73.9
	VAR	11.7	10.2	8.2	7.9	7.1	9.5	13.3	16.1	10.1
	Min	55.6	66.7	77.8	55.6	51.4	66.7	33.3	23.6	55.6
	Max	100	97.2	100	77.8	73.6	88.9	77.8	76.4	88.9
FPR [%]	AVG	3.4	3.3	5.0	2.3	1.5	5.6	2.7	2.1	6.1
	VAR	2.2	2.0	2.2	1.6	1.1	3.9	1.7	1.3	2.7
	Min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	2.4
	Max	7.3	7.0	7.3	4.9	3.0	14.6	4.9	5.2	12.2
Test	F1 Max	0.95	0.96	1.0	0.82	0.79	0.94	0.88	0.80	0.89
	MCC	0.94	0.95	1.0	0.79	0.74	0.94	0.86	0.76	0.86

minimum of 2.7% a false positive rate for the asymmetric cases.

The time of training for the BW approach has varied from around 12 minutes (VGG19), 40 minutes (Xception), and 60 minutes (IRN2) for the PH2 dataset. The times of the training of the CNNs in the BW2 case varied from 5 minutes (VGG19 with 60 epochs) to 14 minutes (XN with 100 epochs), and 21 minutes (IRN2 with 60 epochs) for the PH2 dataset.

5. Conclusions

The correct selection of specific features, in particular, the blue-whitish veil has an impact on the analysis of specific disease fragments. In the research, we have used the three pretrained CNN networks: Xception, VGG19 [11], and Inception-ResNet-v2 as well as PH2 dermoscopic image dataset [20]. The results achieved are quite promising. The average accuracy was above 90 %. The networks usually quite well classified the images of the lesions. VGG19 appears to show the best results from three CNNs.

In the previous papers we have shown that the classifications results are higher when the networks have been trained, validated and tested on the image datasets achieved using the Invariant Dataset Augmentation (IDA). That general approach can also provide higher results while using CNN networks in other disciplines not only dermatology, chest X-ray images of the patients with different lung illnesses e.g. viral pneumonia and COVID-19 [21].

Table 2. The results for the asymmetry classification

CM factor		VGG19			Xception			Inception-ResNet-v2		
		T1	T8	IDA	T1	T8	IDA	T1	T8	IDA
TPR_0 [%]	AVG	58.8	60.2	71.9	65.4	67.1	80.4	53.1	55.9	70.4
	VAR	8.4	6.8	10.4	13.1	9.6	12.3	7.9	4.4	5.7
	Min	46.2	49.0	53.8	38.5	52.9	61.5	38.5	49.0	61.5
	Max	69.2	74.0	92.3	84.6	80.8	92.3	69.2	67.3	84.6
TPR_1 [%]	AVG	26.9	26.7	33.1	25.6	24.4	21.9	7.5	10.3	13.1
	VAR	15.3	12.9	9.3	22.0	16.3	12.7	8.5	10.0	9.5
	Min	12.5	12.5	12.5	0.0	6.3	0.0	0.0	0.0	0.0
	Max	62.5	57.8	50.0	62.5	53.1	37.5	25.0	31.3	25.0
TPR_2 [%]	AVG	80.0	80.9	70.7	82.2	83.0	66.4	83.6	82.5	62.8
	VAR	11.8	10.6	13.0	7.1	5.7	7.2	5.7	4.7	9.9
	Min	58.6	61.2	44.8	72.4	75.9	55.2	75.9	73.3	44.8
	Max	96.6	95.3	86.2	93.1	91.4	75.9	89.7	89.2	75.9
FPR_0 [%]	AVG	9.2	8.9	15.9	11.6	11.1	25.0	10.8	10.9	23.4
	VAR	4.7	3.8	6.3	3.2	4.2	5.8	3.3	2.1	5.8
	Min	0.0	2.0	2.7	5.4	6.1	16.2	5.4	8.1	13.5
	Max	16.2	14.5	2.4	18.9	17.2	2.4	16.2	15.5	4.8
FPR_1 [%]	AVG	11.1	11.4	14.9	10.4	8.7	12.6	9.6	8.7	16.5
	VAR	9.8	8.7	11.6	7.8	6.4	9.3	7.3	5.3	9.9
	Min	0.0	2.1	2.4	0.0	3.3	2.4	0.0	2.1	4.8
	Max	33.3	29.5	35.7	23.8	20.2	28.6	23.8	18.8	33.3
FPR_2 [%]	AVG	42.6	40.5	25.5	33.1	35.7	19.0	48.6	49.0	28.6
	VAR	9.6	8.4	9.4	8.2	11.7	11.7	6.7	7.2	9.9
	Min	23.8	25.0	9.5	14.3	16.1	0.0	38.1	37.5	9.5
	Max	61.9	54.8	38.1	42.9	48.2	33.3	66.7	63.7	42.9
w.ACC [%]	AVG	55.2	56.0	58.6	57.8	58.1	56.2	48.1	49.6	48.8
	Max	69.2	68.9	68.3	78.9	73.3	68.6	54.8	56.8	57.0
Test	F1 Max	0.543	0.548	0.555	0.560	0.560	0.511	0.446	0.465	0.439
	MCC	0.578	0.524	0.621	0.732	0.650	0.643	0.429	0.424	0.449

References

[1] ECIS, *European cancer information system*, 2023, (access: 14-02-2023). https://ecis.jrc.ec.europa.eu

[2] ACS, *American cancer society*, 2023, (access: 14-02-2023). https://www.cancer.org/research/cancer-facts-statistics.html

[3] Soyer H.P., Argenziano G., Zalaudek I., Corona R., Sera F., Talamini R., Barbato F., Baroni A., Cicale L., Di Stefani A., Farro P., Rossiello L., Ruocco E., Chimenti S., *Three-Point Checklist of Dermoscopy: A New Screening Method for Early Detection of Melanoma*, *Dermatology*, 2004, vol. 208, no 1, pp. 27–31, doi: 10.1159/000075042.

[4] Kawahara J., Daneshvar S., Argenziano G., Hamarneh G., *Seven-point checklist and skin lesion classification using multitask multimodal neural nets*, *IEEE Journal of Biomedical and Health Informatics*, 2019, vol. 23, no 2, pp. 538–546, doi: 10.1109/JBHI.2018.2824327.

[5] Nachbar F., Stolz W., Merkle T., Cognetta A.B., Vogt T., Landthaler M., Bilek P., Braun-Falco O., Plewig G., *The abcd rule of dermatoscopy: High prospective value in the diagnosis of doubtful melanocytic skin lesions*, *Journal of the American Academy of Dermatology*, 1994, vol. 30, no 4, pp. 551–559, doi: https://doi.org/10.1016/S0190-9622(94)70061-3.

[6] Rodriguez Salas R., Dokladal P., Dokladalova E., *Rotation invariant networks for image classification for hpc and embedded systems*, *Electronics*, 2021, vol. 10, no 2, doi: 10.3390/electronics10020139.

[7] Tarasiuk P., Szczepaniak P.S., *Novel convolutional neural networks for efficient classification of rotated and scaled images*, *Neural Computing and Applications*, 2022, vol. 34, no 13, pp. 10519–10532.

[8] Cheng G., Zhou P., Han J., *Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images*, *IEEE Transactions on Geoscience and Remote Sensing*, 2016, vol. 54, no 12, pp. 7405–7415, doi: 10.1109/TGRS.2016.2601622.

[9] Dieleman S., Willett K.W., Dambre J., *Rotation-invariant convolutional neural networks for galaxy morphology prediction*, *Monthly Notices of the Royal Astronomical Society*, 2015, vol. 450, no 2, pp. 1441–1459, doi: 10.1093/mnras/stv632.

[10] Weiler M., Hamprecht F.A., Storath M., *Learning steerable filters for rotation equivariant cnns*, [In:] *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 849–858, doi: 10.1109/CVPR.2018.00095.

[11] Simonyan K., Zisserman A., *Very deep convolutional networks for large-scale image recognition*, [In:] *International Conference on Learning Repre-sentations*. https://www.robots.ox.ac.uk/~vgg/publications/2015/Simonyan15/

[12] Esteva A., Kuprel B., Novoa R.A., Ko J., Swetter S.M., Blau H.M., Thrun S., *Dermatologist-level classification of skin cancer with deep neural networks*, *Nature*, 2017, vol. 542, no 7639, pp. 115–118.

[13] Milczarski P., Stawska Z., *Classification of skin lesions shape asymmetry using machine learning methods*, [In:] L. Barolli, F. Amato, F. Moscato, T. Enokido, M. Takizawa (eds.), *Web, Artificial Intelligence and Network Applications*, Springer International Publishing, Cham, pp. 1274–1286, doi: 10.1007/978-3-030-44038-1_116.

[14] Milczarski P., Beczkowski M., Borowski N., *Blue-white veil classification of dermoscopy images using convolutional neural networks and invariant dataset augmentation*, [In:] L. Barolli, I. Woungang, T. Enokido (eds.), *Advanced Information Networking and Applications*, Springer International Publishing, Cham, pp. 421–432, doi: 10.1007/978-3-030-75075-6_34.

[15] Beczkowski M., Borowski N., Milczarski P., *Classification of dermatological asymmetry of the skin lesions using pretrained convolutional neural networks*, [In:] *Artificial Intelligence and Soft Computing*, Springer International Publishing, 2021, pp. 3–14, doi: 10.1007/978-3-030-87897-9_1.

[16] Menzies S.W., Menzies S.W., Crotty K., Ingvar C., McCarthy W., *Dermoscopy: An Atlas*, McGraw-Hill Australia, 2009.

[17] He K., Zhang X., Ren S., Sun J., *Deep residual learning for image recognition*, [In:] *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[18] Celebi M.E., Iyatomi H., Stoecker W.V., Moss R.H., Rabinovitz H.S., Argenziano G., Soyer H.P., *Automatic detection of blue-white veil and related structures in dermoscopy images*, *Computerized Medical Imaging and Graphics*, 2008, vol. 32, no 8, pp. 670–677, doi: 10.1016/j.compmedimag.2008.08.003.

[19] Milczarski P., Beczkowski M., Borowski N., *Enhancing dermoscopic features classification in images using invariant dataset augmentation and convolutional neural networks*, [In:] *Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, December 8–12, 2021, Proceedings, Part III 28*, Springer, pp. 403–417, doi: 10.1007/978-3-030-92238-2_34.

[20] Mendonça T., Ferreira P.M., Marques J.S., Marcal A.R., Rozeira J., *Ph 2-a dermoscopic image database for research and benchmarking*, [In:] *2013 35th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, IEEE, pp. 5437–5440, doi: 10.1109/EMBC.2013.6610779.

[21] Milczarski P., Beczkowski M., Borowski N., *Covid-19 lungs assessment in chest x-ray images using convolutional neural networks*, [In:] *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, vol. 2, pp. 1062–1067, doi: 10.1109/IDAACS53288.2021.9661046.

Chapter 7

Problem Solving and Optimisation

Domain Editors:

1. Jarek Arabas, Warsaw University of Technology
2. Karol Opara, Systems Research Institute, Polish Academy of Sciences
3. Robert Nowak, Warsaw University of Technology

A Novel Learning Multi-Swarm Particle Swarm Optimization

Bożena Borowska[1]

[1]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
bozena.borowska@p.lodz.pl

DOI:10.34658/9788366741928.53

Abstract. *Particle swarm optimization (PSO) is one of the metaheuristic optimization methods. Because of its many advantages, it is often used to solve real-world engineering problems. However, in case of complex, multi-dimensional tasks, PSO faces some problems related to premature convergence and stagnation in local optima. To eliminate this inconveniences, in this paper, a new learning multi-swarm particle swarm optimization method (LMPSO) with local search operator has been proposed. The presented approach was tested on a set of nonlinear functions and a CEC2015 test suite. The obtained results were compared with other optimization methods.*
Keywords: *learning particle swarm optimization, learning strategy, multi-swarm, particle swarm optimization, pso, optimization, swarm intelligence*

1. Introduction

Particle swarm optimization (PSO) is one of the nature-inspired metaheuristic methods used in real-world optimization problems. It was proposed by Kennedy and Eberhart [1] as an alternative to the genetic algorithm (GA). It is appreciated by scientists for its simplicity, robustness and search capability and is successfully applied in many different areas such as image segmentation [2, 3], feature selection [4, 5, 6], and many others. However, in case of complex, multi-dimensional tasks, PSO experiences some problems related to premature convergence and stagnation in local optimal solutions. To avoid this difficulties, in this paper, a new learning multi-swarm PSO method (LMPSO) with local Cauchy search operator has been proposed. LMPSO is a two-phase method. In the first phase the sub-swarms of the LMPSO method operate independently. In each sub-swarm, the particles learn from their neighbors. In the second phase the best particles of a sub-swarm can learn from the best particles of other sub-swarms. To more deeply search local area Cauchy operator is used. The presented approach was tested on a set of nonlinear functions and a CEC2015 test suite. The obtained results were compared with other optimization methods.

2. Standard PSO

Particle swarm optimization (PSO) is one of the nature-inspired methods used in optimization. It is based on a population called a swarm [7]. Swarm individuals are called particles. Each particle is described by a position vector $X_i=(x_{i1}, x_{i2}, \ldots, x_{iD})$ and a velocity vector $V_i=(v_{i1}, v_{i2}, \ldots, v_{iD})$. The velocity vector determines the speed and direction of particle motion. Each particle location represents one of the possible potential solutions to the problem under study. The goodness of the particle is evaluated by the fitness function. In the first iteration, the positions of the particles are randomly generated. Then they move in the search space and remember their best position (pbest) and the best position found in the entire swarm (gbest). In each iteration velocity of the particles is changing based on the equation 1:

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (pbest_i - X_i(t)) + c_2 r_2 (gbest - X_i(t)) \tag{1}$$

The position of the particles is updated according to the formula 2:

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{2}$$

where ω is the inertia weight, $pbest_i$ means the best i particle position, $gbest$ is the best position in a swarm, c_1 and c_2 are acceleration coefficients, r_1 and r_2 are numbers generated from the uniform distribution on interval (0, 1).

3. Proposed Learning Strategy

The proposed LMPSO (learning multi-swarm particle swarm optimization) is a two-phase method based on particle swarm optimization, multi-swarm and learning concept. In the first phase the entire population of N particles is randomly divided into several sub-swarms. Each particle has a different position and velocity, and different searching abilities. During the search space all sub-swarms work independently. In each sub-swarm the particles move in the search space according to their velocity vectors and remember their best found position (*pbest*). The best position discovered in the entire sub-swarm is recorded as *sbest*. Half of the randomly selected particles of the sub-swarm update their position by learning from the average of the personal best positions ($\overline{pbest}$) found by all particles of the sub-swarm according to the formula 3 and 4:

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (pbest_i - X_i(t)) + c_2 r_2 (\overline{pbest} - X_i(t)) \tag{3}$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{4}$$

where $pbest_i$ is the best position of the i particle, $\overline{pbest}$ is the average of the best personal positions found so far by the particles in sub-swarm.

The other half of the sub-swarm particles update their position by learning from

the best particle in the entire sub-swarm (*sbest*). The update proceeds according to the equations 5 and 4:

$$V_{\mathrm{i}}(t+1) = \omega V_{\mathrm{i}}(t) + c_1 r_1 (pbest_{\mathrm{i}} - X_{\mathrm{i}}(t)) + c_2 r_2 (sbest - X_{\mathrm{i}}(t)) \tag{5}$$

In the second phase, a temporary set E of the best particles (*sbest*) of each sub-swarm is created. Then the best particle from them is selected, and the other particles learn from it. In this way sub-swarms can share knowledge and learn from other sub-swarms. This means that good information found by one of the sub-swarms is not lost but can be used by other sub-swarms. Moreover, the sub-swarm that is trapped into local optimum can jump out of it by learning from the best particles of other sub-swarm. In addition, to deeply search vicinity area the local search Cauchy operator is used. The proposed strategy helps maintain population diversity and better search the space of possible solution.

4. Results

The tests of the proposed LMPSO method were performed on a set of classical benchmark problems and on the CEC 2015 functions [8]. The results of the tests were compared with performance of FIPS (fully informed particle swarm optimization) [9], MSPSO (pso with multiple subpopulations) [10] and PSO (standard particle swarm optimization).
For all tested algorithms, the population consist of 40 particles. The maximum number of iterations is 8000. The dimension of the search space was D=30. Inertia weight ω=0.9 to 0.4, sub-swarm's number s=4. For each problem, the algorithms were run 32 times independently. The exemplary research results for 5 functions from the CEC2015 set [8] are presented in Table1.

The results of the test indicate that the proposed LMPSO algorithm achieves superior performance over the other methods. Only in case of f3 function all algorithms reached similar average value but the mean value found by LMPSO was a bit better. However, it should be noted that LMPSO was slower than SPSO and FIPS but its results were more accurate. In case of f1, f2 and f6 function, the worst results were achieved by MSPSO. The MSPSO agorithm worked slower and performed worse in local optima. The FIPS algorithm generally performed better than the MSPSO method, but worse than LMPSO.
The proposed sub-swarm topology of LMPSO helps maintain diversity of the particles and improve exploration capacity. The learning strategy prevents the loss of valuable information found by swarms and improves effectiveness of the method. Cauchy operator increases exploitation ability and prevents stagnation in local optimum.

Table 1. The comparison test results

Function	Criteria	SPSO	MSPSO	FIPS	LMPSO
F1	Best	2.25E+04	4.19E+06	2.49E+06	1.56E+03
	Worst	2.11E+07	1.98E+07	1.26E+07	6.17E+05
	Mean	6.53E+06	1.21E+07	5.71E+06	8.98E+04
F2	Best	6.51E-08	3.05E+06	2.26E+01	1.83E-12
	Worst	1.59E-01	3.26E+07	1.20E+04	1.77E-08
	Mean	7.16E-03	1.38E+07	4.13E+03	1.31E-09
F3	Best	20.48E+00	20.67E+00	20.97E+00	20.50E+00
	Worst	21.90E+00	21.03E+00	21.18E+00	21.04E+00
	Mean	20.97E+00	20.98E+00	21.06E+00	20.78E+00
F4	Best	48.37E+00	51.44E+00	73.14E+00	41.20E+00
	Worst	112.03E+00	113.08E+00	151.13E+00	109.63E+00
	Mean	65.47E+00	88.65E+00	114.98E+00	56.24E+00
F6	Best	9.81E+04	1.12E+05	1.10E+05	3.25E+04
	Worst	1.52E+06	3.77E+06	1.14E+06	1.08E+05
	Mean	4.8953E+05	8.79E+05	4.32E+05	6.07E+04

5. Conclusions

In this study, a new two-phase multi-swarm particle swarm optimization (LMPSO) based on learning strategy and Cauchy search operator has been proposed. During the first phase sub-swarms works independently. The particles of the sub-swarm can move according the knowledge about their personal best position (pbest) and the mean best value of all particles in the sub-swarm or towards the best particle in the entire sub-swarm (sbest). In the second phase information is exchanges between sub-swarms. The effectiveness of LMPSO was tested on a set of nonlinear benchmark functions and a CEC2015 test functions.

References

[1] Kennedy J., Eberhart R.C., *Particle swarm optimization*, [In:] *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, 1995, pp. 1942–1948.

[2] Chander A., Chatterjee A., Siarry P., *A new social and momentum component adaptive pso algorithm for image segmentation*, *Expert Systems With Applications*, 2011, vol. 38, no 5, pp. 4998–5004, ISSN 0957-4174, doi: 10.1016/j.eswa.2010.09.151.

[3] Suresh S., Lal S., *Multilevel thresholding based on chaotic darwinian particle swarm optimization for segmentation of satellite images*, *Applied Soft Computing*, 2017, vol. 55, pp. 503–522, ISSN 1568-4946, doi: 10.1016/j.asoc.2017.02.005.

[4] Shafipour M., Rashno A., Fadaei S., *Particle distance rank feature selection by particle swarm optimization*, *Expert Systems with Applications*, 2021, vol. 185, p. 115620, ISSN 0957-4174, doi: 10.1016/j.eswa.2021.115620.

[5] Li A.D., Xue B., Zhang M., *Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies*, *Applied Soft Computing*, 2021, vol. 106, p. 107302, ISSN 1568-4946, doi: 10.1016/j.asoc.2021.107302.

[6] Song X.F., Zhang Y., wei Gong D., yan Sun X., *Feature selection using barebones particle swarm optimization with mutual information*, *Pattern Recognition*, 2021, vol. 112, p. 107804, ISSN 0031-3203, doi: 10.1016/j.patcog.2020.107804.

[7] Kennedy J., Eberhart R.C., Shi Y., *Swarm Intelligence*, Morgan Kaufmann Publishers: San Francisco, CA, USA, 2001.

[8] Liang J., Qu B., Suganthan P., *Problem definitions and evaluation criteria for the cec 2015 competition on learning-based real-parameter single objective optimization*, Tech. rep., Nanyang Technological University (Singapore) and Zhengzhou University (China), 2014.

[9] Chang W.D., *A modified particle swarm optimization with multiple subpopulations for multimodal function optimization problems*, *Applied Soft Computing*, 2015, vol. 33, pp. 170–182.

[10] Mendes R., Kennedy J., Neves J., *The fully informed particle swarm: simpler, maybe better*, *IEEE Trans. Evol. Comput.*, 2004, vol. 8, pp. 204–210, doi: 10.1109/TEVC.2004.826074.

Are Quantified Boolean Formulas Hard for Reason-Able Embeddings?

Jędrzej Potoniec[0000−0002−6115−6485]

Poznan University of Technology
Institute of Computing Science
Piotrowo 2, 60-965, Poznań, Poland
jedrzej.potoniec@cs.put.poznan.pl

DOI:10.34658/9788366741928.54

Abstract. *We aim to establish theoretical boundaries for the applicability of reason-able embeddings, a recently proposed method employing a transferable neural reasoner to shape a latent space of knowledge graph embeddings. Since reason-able embeddings rely on the $\mathcal{ALC}$ description logic, we construct a dataset of the hardest concepts in $\mathcal{ALC}$ by translating quantified boolean formulas (QBF) from QBFLIB, a benchmark for QBF solvers. We experimentally show the dataset is hard for a symbolic reasoner FaCT++, and analyze the results of reasoning with reason-able embeddings, concluding that the dataset is too hard for them.*
Keywords: *artificial intelligence, neural-symbolic reasoning, knowledge representation, description logics*

1. Introduction

Recent years saw renewed research interest in the so-called neural-symbolic artificial intelligence, i.e., approaches combining symbolic and subsymbolic (neural) techniques [1]. Within this trend frameworks, that were traditionally symbolic, are extended with sub-symbolic components. One such framework is knowledge graphs (KGs), extended by embedding concepts and relations in a latent, high-dimensional space. Over the years numerous embeddings for KGs were proposed, mostly exploiting only rather shallow semantics of the labeled graph itself [2, 3, 4, 5, 6]. Recently, approaches for KGs employing more elaborate semantics were proposed. One example, being the central focus of this work, is reason-able embeddings. They enable the computation of embeddings for complex logical expressions (concepts) in the $\mathcal{ALC}$ description logic (DL) by applying neural operators to embeddings of atomic concepts [7]. The embeddings can then be used by a transferable neural reasoner to decide whether one concept is subsumed by another. While reason-able embeddings present remarkable performance in empirical evaluation, there may be a gap between the computational complexity of

neural networks and that of $\mathcal{ALC}$: for the former, the inference is polynomial [8], whereas for the latter concept subsumption is PSPACE-complete [9].

In this paper, we aim to tackle the problem of finding what is hard for Reason-able embeddings. We attack the problem by leveraging the hardest formulas for $\mathcal{ALC}$, i.e., those used to prove PSPACE-hardness of $\mathcal{ALC}$, and arising from transforming the problem of the validity of quantified boolean formulas (QBFs) to the problem of concept satisfiability in $\mathcal{ALC}$ [9]. Under the usual assumption that P is a proper subset of PSPACE, and assuming the employed QBFs are not a simpler subset of QBFs (like 2-SAT for the SAT problem), we expect reason-able embeddings to exhibit poor performance.

2. Materials and methods

2.1. Reason-able embeddings

Reason-able embeddings [7] are rooted in the idea that the latent space of the embeddings should preserve semantic properties of the original concepts in a way that can be shared between multiple separate KGs, similarily how logical operators of $\mathcal{ALC}$ are shared. The latent space is shaped by a neural reasoner, shared between KGs, and the training of embeddings takes place by backpropagation through the reasoner. The framework of reason-able embeddings consists of two main parts: KG-specific embeddings, transforming concepts of the KG to the latent space, and a neural reasoner transferable between KGs. Initially, the reasoner and embeddings for multiple KGs are trained jointly. Once trained, the reasoner is frozen and can be used to train embeddings for a new KG by means of transfer learning. This procedure ensures the latent space and the reasoner remain coupled, and the reasoner can exploit it.

The reasoner consists of two main components: a reasoner head, which given two embeddings, performs binary classification to answer whether the concept represented by one embedding is subsumed by the second embedding's concept; and neural operators, a set of neural networks one-to-one corresponding to logical operators of $\mathcal{ALC}$ and enabling computation of an embedding of a complex concept from the embeddings of its parts. Similarly, the embeddings for the top concept $\top$ and the bottom concept $\bot$ are a part of the reasoner and thus shared between KGs.

Since $\mathcal{ALC}$ contains negation, all concepts of form $\forall r.C$ are transformed to $\neg\exists r.\neg C$, thus transforming two quantifiers into one. Furthermore, in $\mathcal{ALC}$ KGs without individuals the only possible position where a role is employed is under a quantifier, and thus from the point of view of the embeddings, the expressions $\exists r$ can be assumed to carry the same amount of information as r itself. Thus, there are no neural operators corresponding to quantifiers, and instead, relation embeddings are trained and employed in their place.

In [7], alongside the theoretical framework, an implementation was presented and evaluated. The embeddings' dimension was set to 10, and the reasoner head consisted of a single hidden layer of 16 neurons with the ELU activation function. The neural operators were all single layers with no activation function. The reasoner was trained on a set of randomly generated $\mathcal{ALC}$ KGs and tested on a set of 6 real-world, previously not seen $\mathcal{ALC}$ KGs. For each KG, 32,000 random, unique subsumptions were generated and posed to the reasoner to decide whether they follow from the KG. To obtain the ground truth, a symbolic reasoner FaCT++ was employed [10]. Reason-able embeddings exhibited remarkable performance, attaining 0.978 ± 0.016 AUC-ROC.

Reason-able embeddings were also evaluated on a test set of 20 ontologies generated randomly in the same way as the ontologies for training, attaining 0.989 ± 0.009 AUC-ROC. We use the embeddings for these ontologies throughout the paper.

2.2. QBFs and their transformation to $\mathcal{ALC}$

To obtain high-quality non-trivial QBFs, we turned to QBFLIB [11, 12], a repository of over 13,000 QBFs serving as a benchmark for QBF solvers. We concentrated on the QBFs expressed in the QDIMACS format, ensuring that we are dealing only with closed formulas (i.e., all variables are quantified) in the conjunctional normal form. We selected a suite named *Castellini*[1] since it consists of the QBFs with the fewest variables in the whole QBFLIB, and further limited it to QBFs using less than 50 variables, obtaining 17 QBFs using 11–49 variables.

Since we are dealing with closed formulas, the notion of validity and satisfiability are equivalent, and we could employ the procedure described in [9] (Section 6.2) to translate QBFs to DL concepts. For sake of space, we refrain from repeating the details of the procedure here. It is polynomially complex and requires a single distinguished concept and a single distinguished relation. A single QBF is translated into a single complex DL concept such that the concept is satisfiable if, and only if, the QBF is valid. For each of the 20 KGs from the test set, we select uniformly at random a single concept and a single relation. To avoid interference from the KG, we then translate two very simple QBFs: valid $\forall x \exists y\, ((\neg x \vee y) \wedge (x \vee \neg y))$ and invalid $\forall x \exists y\, (\neg x \wedge \neg y)$. We query FaCT++ about the translated concepts and verify the answers. If they are incorrect, the selection procedure is repeated. Otherwise, we use the concept and relation to create the concepts from the selected 17 QBFs. This procedure ensures that the used concept is not overly constrained by the KG, e.g., by being equal to the bottom concept.

By creating 17 concepts for each of the 20 KGs, we have arrived at a dataset of 340 concepts. To obtain the ground truth labels, we have employed DepQBF, a

[1] http://www.qbflib.org/suite_detail.php?suiteId=4

state-of-the-art QBF solver [13]. 10 out of 17 QBFs in the dataset are valid, thus 200 out of 340 (59%) translated concepts are satisfiable.

3. Experimental results

To verify the hardness of the dataset, we used FaCT++ with a time limit of 300 seconds per query on a Linux system equipped with 32 GB RAM and a CPU scoring on average 7122 points in the CPU benchmark[2]. It managed to decide on the satisfiability of only 37 out of 340 concepts (11%), indicating the dataset is indeed hard.

As expected, the results of reason-able embeddings were very poor: averaging over the KGs, they attained accuracy 0.518 ± 0.089 and AUC-ROC 0.515 ± 0.077. Further analysis revealed that even those results must be taken with a grain of salt: for every KG, all 17 concepts were classified the same, as either satisfiable (for 12 KGs), or unsatisfiable (for the remaining 8). This means reason-able embeddings cannot deal with these concepts at all.

We observe there are two sources of the hardness of these concepts. First, the concepts are much more complex than those used to train the reasoner: the syntax tree of the training concepts was at most 3 deep, whereas the concepts corresponding to the selected QBFs are at least 11 deep. It may be the case that the neural operators are too coarse and useful properties are lost after applying them so many times. The other source is the inherent hardness of reasoning with these concepts, as expected by their construction and indicated by the poor performance of FaCT++. Thus an open question arises: are there well-defined boundaries of usefulness for the reason-able embeddings, or must it always be a matter of experimental verification?

4. Conclusions

In this paper, we have presented a method to construct a dataset of $\mathcal{ALC}$ DL concepts which are expected to be computationally expensive for any reasoner, be it symbolic or neural. We experimentally confirmed the hardness of the dataset for a symbolic reasoner. We then established that reason-able embeddings, a recently proposed method of learning KG embeddings employing a transferable neural reasoner, are incapable of dealing with the dataset, attaining results substantially biased and no better than random guessing.

In the future, we envision constructing a comprehensive benchmark for KG embeddings, enabling proper assessment of their capabilities, both in terms of used DL constructors, as well as the complexity of the performed reasoning.

Finally, to answer the question posed in the title: *yes, they are hard*.

[2]https://www.cpubenchmark.net/

Acknowledgment

This research was partially supported by 0311/SBAD/0726 and by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

[1] Hitzler P., Sarker M.K. (eds.), *Neuro-Symbolic Artificial Intelligence: The State of the Art*, *Frontiers in Artificial Intelligence and Applications*, vol. 342, IOS Press, 2021, doi: 10.3233/FAIA342.

[2] Choudhary S., Luthra T., Mittal A., Singh R., *A survey of knowledge graph embedding and their applications*, *CoRR*, 2021, vol. abs/2107.07842.

[3] Wang Q., Mao Z., Wang B., Guo L., *Knowledge graph embedding: A survey of approaches and applications*, *IEEE Trans. Knowl. Data Eng.*, 2017, vol. 29, no 12, pp. 2724–2743, doi: 10.1109/TKDE.2017.2754499.

[4] Dai Y., Wang S., Xiong N.N., Guo W., *A survey on knowledge graph embedding: Approaches, applications and benchmarks*, *Electronics*, 2020, vol. 9, no 5, p. 750, doi: 10.3390/electronics9050750.

[5] Cai H., Zheng V.W., Chang K.C., *A comprehensive survey of graph embedding: Problems, techniques, and applications*, *IEEE Trans. Knowl. Data Eng.*, 2018, vol. 30, no 9, pp. 1616–1637, doi: 10.1109/TKDE.2018.2807452.

[6] Wang M., Qiu L., Wang X., *A survey on knowledge graph embeddings for link prediction*, *Symmetry*, 2021, vol. 13, no 3, doi: 10.3390/sym13030485.

[7] Adamski D.M., Potoniec J., *Reason-able embeddings: Learning concept embeddings with a transferable neural reasoner*, *Semantic Web*, 2023, access: 13-07-2023.
https://www.semantic-web-journal.net/content/reason-able-embeddings-learning-concept-embeddings-transferable-neural-reasoner

[8] Orponen P., *Neural networks and complexity theory*, *Nord. J. Comput.*, 1994, vol. 1, no 1, pp. 94–110.

[9] Schmidt-SchaußM., Smolka G., *Attributive concept descriptions with complements*, *Artificial Intelligence*, 1991, vol. 48, no 1, pp. 1–26, doi: 10.1016/0004-3702(91)90078-X.

[10] Tsarkov D., Horrocks I., *FaCT++ Description Logic Reasoner: System Description*, [In:] U. Furbach, N. Shankar (eds.), *Automated Reasoning*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, ISBN 9783540371885, pp. 292–297, doi: 10.1007/11814771_26.

[11] Narizzano M., Pulina L., Tacchella A., *The qbfeval web portal*, [In:] *Logics in Artificial Intelligence*, Springer Berlin Heidelberg, pp. 494–497.

[12] Giunchiglia E., Narizzano M., Pulina L., Tacchella A., *Quantified Boolean Formulas satisfiability library (QBFLIB)*, 2005, `www.qbflib.org`.

[13] Lonsing F., Egly U., *Depqbf 6.0: A search-based QBF solver beyond traditional QCDCL*, [In:] L. de Moura (ed.), *Automated Deduction - CADE 26 - 26th International Conference on Automated Deduction, Gothenburg, Sweden, August 6-11, 2017, Proceedings*, *LNCS*, vol. 10395, Springer, pp. 371–384, doi: 10.1007/978-3-319-63046-5_23.

Dynamic Mutation Control in Continuous Genetic Algorithms

Łukasz Wieczorek[0000−0002−3438−0174],
Przemysław Ignaciuk[0000−0003−4420−9941]

Lodz University of Technology
Institute of Information Technology
8 Politechniki Av., 93-590 Łódź, Poland
lukasz.wieczorek.1@edu.p.lodz.pl

DOI:10.34658/9788366741928.55

Abstract. *In this paper the adaptability of the mutation operation in continuous genetic algorithms (CGAs) is taken into consideration from an analytical perspective. For this purpose, based on the notation that has previously been used to analyze the classical, binary genetic algorithm, a dynamic system model of CGA has been created. In order to adapt the mutation probability in successive generations, a linear controller has been applied. It allows one to accelerate the evolution process. As a result, faster convergence is obtained, as required in computationally intensive optimization problems.*
Keywords: *genetic algorithms, dynamic system model, linear controller*

1. Introduction

Genetic algorithms (GAs) are one of the most commonly used and best formally studied groups of computational intelligence techniques. Their key mechanism – the mutation operation – allows one to mitigate the risk of convergence to a local optimum and not exploring a search space sufficiently during the optimization process. However, that operation, due to its stochastic nature, poses a significant challenge for analytical treatment. So far, the GAs have been studied using a variety of approaches such as the schema theorem [1], Markov chains [2], or dynamic system models (DSMs) [3], which is also the case in this work.

In the fundamental analysis of GAs using DSMs a simple GA was taken under consideration. Further research involved more sophisticated implementations of GAs, e.g., including rank and tournament selection [3]. Nevertheless, the majority of past works address the classical GAs form in the binary domain. This assumption translates to the mutation operation flipping genes from 0 to 1 (and back). In contrast to those works, however, the DSM analyzed in this paper allows handling continuous-domain optimization problems. In the considered case, the features of

the candidate solution are not converted into a binary form [3]. Consequently, the analysis of the mutation operation is substantially more complex due to numerous potential values into which the gene may mutate, yet rewarding in terms of extended application scope of continuous genetic algorithms (CGAs) in real-life problems with respect to their approximate implementation as binary GAs [4].

In recent years, adaptive GAs have been found to be highly effective in solving continuous-domain optimization problems. The research involves the adaptability of selection and/or crossover operations [5], dynamic mutation probability [6], and fitness function modifications [7]. However, most of the contributions so far have been developed by trial and error and verified empirically, only, i.e., without any formal foundations confirming their effectiveness. For that reason, in this paper, a linear mutation controller has been formulated and studied formally.

The contribution of this manuscript is thus twofold. First, using the notation that has been used to analyze the classical, binary GA, a DSM of CGA has been created. The considered model allows one to study various implementations of CGAs with respect to different genetic operations. Second, a linear controller to adapt the mutation probability is proposed. The designed controller accelerates the evolution process, thus being suitable for problems requiring fast convergence. The controller performance has been verified both analytically and numerically.

2. Dynamic system model

In the considered class of optimization problems, the search space is simplified to natural numbers, only. Let the problem domain be formulated as a vector of all the feasible candidate solutions $\mathbf{x} = [x_1, \ldots, x_n]$, where n denotes its cardinality. Then, the population vector in generation t, $t = 1, 2, \ldots$, may be expressed as $\mathbf{v}(t) = [v_1(t), \ldots, v_n(t)]^T$, where $v_i(t)$ is the number of x_i present in that generation. The elements of $\mathbf{v}(t)$ sum to S, which is the population size.

The proportionality vector, expressing the probability of occurrence of each candidate solutions in generation t as the population size approaches infinity, can be defined by $\mathbf{p}(t) = \mathbf{v}(t)/S = [p_1(t), \ldots, p_n(t)]^T$, where all the elements sum to 1.

According to the law of large numbers, the average of the results obtained from a sufficiently big number of trials must be close to the expected value of a single trial. Also, the result is closer to the expected value as the number of trials becomes large. Thus, if the population size $S \to \infty$, a DSM of the CGA including the fitness-proportionate selection, only, may be formulated as

$$\mathbf{p}(t+1) = \frac{\mathrm{diag}(\mathbf{f})\mathbf{p}(t)}{\mathbf{f}^T\mathbf{p}(t)}, \tag{1}$$

where $\mathbf{f} = [f_1, \ldots, f_n]^T$ is a column vector of fitness values of candidate solutions and $\mathrm{diag}(\mathbf{f})$ is a diagonal matrix containing the elements of $\mathbf{f}$.

If selection is followed by mutation, an additional matrix $\mathbf{M}$ should be defined. It gathers the mutation probabilities among all the candidate solutions such that M_{ij} quantifies the probability that x_i mutates to x_j. That matrix is relatively easy to define in the binary GAs, in which mutation consists in flipping gene's values from 0 to 1. However, in the continuous search domain the gene's value may be replaced by any value within the allowed range (including the original value) [3]. Supposing mutation probability equals m and candidate solutions composed of G natural-valued genes upper-bounded by $\mathbf{g}^{\max}$, where $g_k^{\max}$ is the maximal possible value of gene at index k, the probability that x_i mutates to x_j may be calculated by

$$M_{ij} = \prod_{k=1}^{G} \gamma(x_i(k), x_j(k), g_k^{\max}, m), \tag{2}$$

where:

$$\gamma(x_i(k), x_j(k), g_k^{\max}, m) = \begin{cases} (1-m) + \frac{m}{g_k^{max}} & \text{if } x_i(k) = x_j(k), \\ \frac{m}{g_k^{max}} & \text{if } x_i(k) \neq x_j(k). \end{cases} \tag{3}$$

Then, the DSM of the selection-mutation CGA may be formulated as

$$\mathbf{p}(t+1) = \frac{\mathbf{M}(\mathbf{x}, \mathbf{g}^{\max}, m)^T \operatorname{diag}(\mathbf{f})\mathbf{p}(t)}{\mathbf{f}^T\mathbf{p}(t)}. \tag{4}$$

Let us consider a twenty-four-solution search space upper-bounded by $\mathbf{g}^{\max} = [2, 3, 4]$. Assuming the mutation probability set at 0.1, a column vector of fitness values, $\mathbf{f} = [1, 1, 2, 3, 1, 4, 3, 1, 4, 2, 3, 5, 1, 1, 2, 6, 1, 4, 3, 1, 4, 2, 3, 7]^T$, and an initial proportionality vector $\mathbf{p}(0) = [0.2, 0.2, 0.2, 0.2, 0.2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$, exact proportions of candidate solutions through successive generations may be calculated. Figure 1 depicts the progress of the proportionality vector through a 50-generation evolution process.

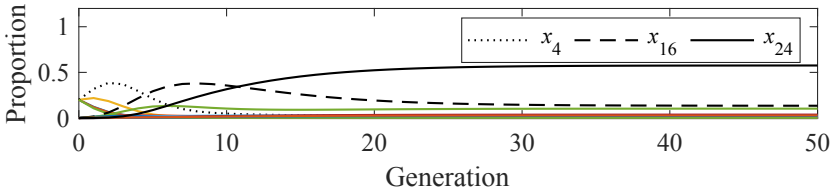


Figure 1. Exact proportions of candidate solutions established using the created DSM with a fixed mutation probability equals 0.1. Source: own work.

3. Mutation controller

In order to accelerate the evolution process and consequently reduce the number of generations required to reach the optimal solution, a linear mutation controller is proposed. The controller enables one to adapt the mutation probability of candidate solutions based on their quality against the solutions already found. Thus, let the mutation probability of x_i in generation t be established according to

$$m_i(t) = Ke_i(t), \tag{5}$$

where K is the controller gain and $e_i(t)$ quantifies process error of x_i in generation t as

$$e_i(t) = \frac{f_{max}(t) - f_i}{f_{max}(t)} = 1 - \frac{f_i}{f_{max}(t)}, \tag{6}$$

in which $f_{max}(t)$ is the maximal fitness value of any solution found until generation t and all the fitness values have to satisfy $f_i > 0$, for any $i = 1, \ldots, n$.

As a result, matrix $\mathbf{M}$ may be expressed as a function of time, and thus the DSM including the proposed mutation controller is given by

$$\mathbf{p}(t+1) = \frac{\mathbf{M}(\mathbf{x}, \mathbf{g}^{\max}, \mathbf{m}(t+1))^T \operatorname{diag}(\mathbf{f})\mathbf{p}(t)}{\mathbf{f}^T\mathbf{p}(t)}, \tag{7}$$

where $\mathbf{m}(t+1) = [m_1(t+1), \ldots, m_n(t+1)]$ is a vector of the mutation probabilities established for all the candidate solutions in the generation.

Figure 2 presents the progress of the proportionality vector in the evolution process performed using the dynamic CGA in which the mutation probabilities were adjusted according to (5) with $K = 1$. In comparison with Fig. 1, the dynamic CGA needed only 13 generations to reach steady state, i.e., the moment in which the proportionality vector remains almost the same in successive generations, whereas the CGA with a fixed mutation probability needed about 30 generations to reach that state. In addition to the faster convergence, the adaptive CGA was also focused on better-suited candidate solutions in the evolution process.

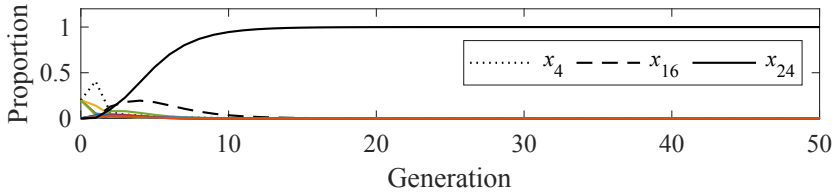


Figure 2. Exact proportions of candidate solutions established using the created DSM with linear mutation controller. Source: own work.

4. Numerical studies

Figure 3 shows the results of the simulation-based optimization of a 200-solution population. The obtained results coincide with the proportionality vectors established analytically shown in Fig. 2. Little differences originate from the finite-size population and a random number generator used in the simulations.

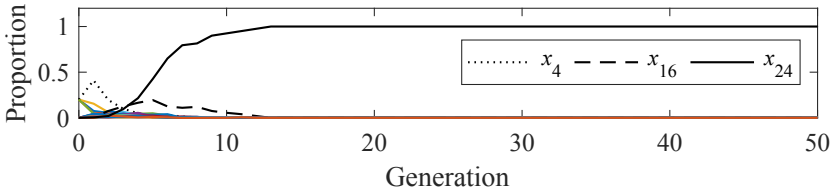


Figure 3. Proportions of candidate solutions through the evolution process performed using the CGA with linear mutation controller. Source: own work.

5. Conclusions

In this paper a DSM of CGA was formulated based on the notation used in the binary GAs so far. Moreover, a dynamic mutation controller was proposed. It accelerates the evolution process by achieving faster convergence than using a fixed mutation probability. The effectiveness of the created controller was illustrated both analytically and numerically. Although DSMs have been restricted to small problems so far, the further formal analysis combined with growing hardware capabilities may allow one to be applied in real-life problems. The future research involves extending the DSM with crossover and other selection operators. Also, the adaptability of CGA can be steered using other classical controllers.

Acknowledgment

The research was supported by the National Science Centre, Poland ['Robust control solutions for multi-channel networked flows', no. 2021/41/B/ST7/00108].

References

[1] Liu D., *Mathematical modeling analysis of genetic algorithms under schema theorem*, *Journal of Computational Methods in Sciences and Engineering*, 2019, vol. 19, no S1, pp. 131–137.

[2] Corus D., Oliveto P.S., *Standard steady state genetic algorithms can hillclimb faster than mutation-only evolutionary algorithms*, *IEEE Transactions on Evolutionary Computation*, 2018, vol. 22, no 5, pp. 720–732.

[3] Simon D., *Evolutionary Optimization Algorithms*, Wiley, 2013.

[4] Wieczorek Ł., Ignaciuk P., *(r, q) inventory management in complex distribution systems of the one belt one road initiative*, *International Journal of Shipping and Transport Logistics*, 2022, vol. 15, no 1-2, pp. 111–143.

[5] Sivanandam S.N., Deepa S.N., *Introduction to Genetic Algorithms*, Springer Publishing Company, 2008.

[6] Li Y.B., Sang H.B., Xiong X., Li Y.R., *An improved adaptive genetic algorithm for two-dimensional rectangular packing problem*, *Applied Sciences*, 2021, vol. 11, no 1.

[7] Iranmanesh A., Naji H.R., *Dchg-ts: a deadline-constrained and cost-effective hybrid genetic algorithm for scientific workflow scheduling in cloud computing*, *Cluster Computing*, 2021, vol. 24, no 2, pp. 667–681.

Local Energy Redistribution Units for Space Dimensionality Reduction in Data Classification

Dariusz Puchała[0000−0001−9070−8042]

Lodz University of Technology,
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland

DOI:10.34658/9788366741928.56

Abstract. *In this paper, we present locally trained* 2*-input to* 2*-output neurons called Local Energy Redistribution Units (LERUs), which enable to transfer most of the input data energy to the selected output, and when organized into properly designed networks, allow for the energy accumulation in lower-indexed elements of output vectors. This property can be used to reduce the dimensionality of the input data space, resulting in a reduction in the number of weights and disk space needed to store neural network models. We test the effectiveness of the proposed approach experimentally in the task of data classification using the well-known MNIST dataset.*
Keywords: *locally trained neurons, compression of neural networks*

1. Introduction

Deep artificial neural networks (DNNs) proved to be highly effective in many tasks of data classification and analysis. It should be noted, however, that DNNs with dense connections operating on high-dimensional data suffer from a huge number of weights that must be trained and stored in a form of the trained model. In the literature, we can find various approaches aimed at reducing the number of connections and network parameters, e.g.,: sparse multilayer neural networks [1, 2], low-rank approximation of dense layers using SVD transform [3], or depthwise separable convolutional neural networks [4].

In this paper, we present a novel structure of neural network based on dedicated LERU neurons, which by energy compaction in the small number of outputs, allows to reduce the number of connections between neural layers. The additional strong point of the proposed approach is local adaptation rule of neurons, which may prevent gradient from vanishing during the training process. As part of the conducted research, we verify experimentally the effectiveness of the proposed approach in the task of data classification.

2. Local Energy Redistribution Units

The method of operation of the proposed Local Energy Redistribution Unit consists in shifting as much input energy as possible to one of the outputs. The LERU can be defined as a neuron with two inputs and two outputs and linear activation function. In the most general case it can be formally described by:

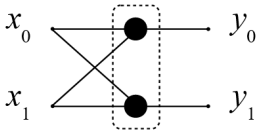
$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}, \tag{1}$$

where $[y_0, y_1]^T$ and $[x_0, x_1]^T$ are output and input vectors, respectively, and a, b, c and d are the weights of the neuron. Since we do not want to change the energy of the input data but only shift it to one of the outputs, we demand that $y_0^2+y_1^2 = x_0^2+x_1^2$. This demand leads directly to the following requirements: $ab+cd = 0$, $a^2+c^2 = 1$, and $b^2+d^2 = 1$. By substituting $c = -b$ and $d = a$, we get the first requirement fulfilled and the remaining two reduced to one condition of the form: $a^2+b^2 = 1$. This requirement can be easily fulfilled by the popular unit length constrain applied to the rows of the weights matrix. In this way, we get:

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \left(\frac{1}{a^2+b^2}\right) \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}. \tag{2}$$

It can be easily verified that the weights of neuron described by (2) constitute an orthogonal matrix.

The next crucial demand is the local rule for weights adaptation, which additionally must be built into the neuron. Here, we assume without the loss of generality that the input energy should be shifted to output y_0. Hence, we want to maximize y_0^2, which, after calculating the partial derivatives for a and b, allows us to determine formulas updating the weights, i.e.: $a_{(i+1)} = a_{(i)}+\beta x_0 y_0$, and similarly $b_{(i+1)} = b_{(i)}+\beta x_1 y_0$, where i is the stage index of the training algorithm and β defines the learning rate. The proposed neuron can be presented graphically as in Fig. (3).



$$\Leftrightarrow \quad \begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \left(\frac{1}{a^2+b^2}\right) \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \tag{3}$$

3. The network topology

In order to achieve energy compaction for vectors longer than 2 elements, the proposed LERU neurons must be organized into a network. We propose the following recursive approach based on the "divide-and-conquer" strategy. Let N be an even number describing the size of input data. The network construction procedure can be defined by the following steps:

- apply LERU neurons to input data vector in a number of $N/2$;
- divide the obtained vector into two possibly equal parts of sizes N_0 and N_1 elements, where both values are even, $N_0 \geq N_1$ and $N_0 + N_1 = N$;
- sort outputs of neurons in the following way: outputs maximizing energy go one by one to the first N_0-element part of the output vector, the remaining outputs go to the second part;
- apply the same steps to the first and the second part of the vector as long as $N_0 > 2$ and $N_1 > 2$.

Fig. 1 shows an example of the network constructed for $N = 8$.

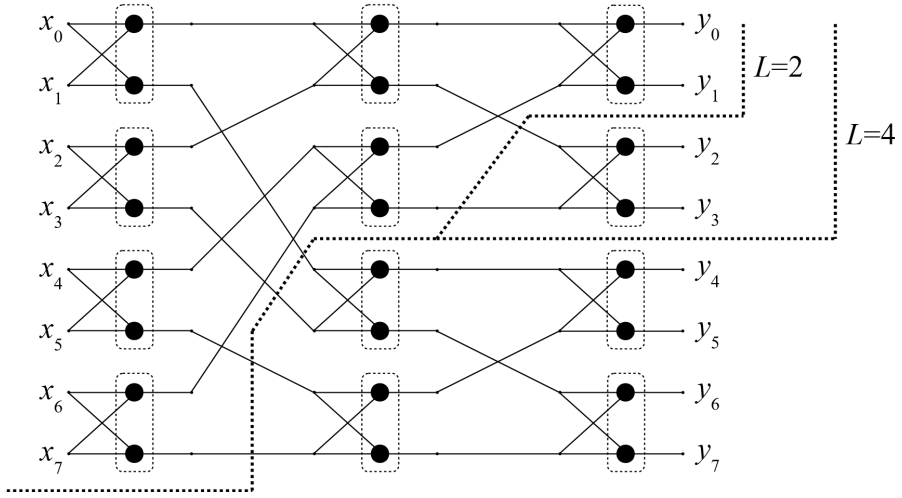


Figure 1. The proposed LRUs organized into the network that enables energy compaction (for $N = 8$). Dashed lines indicate possible pruning (shrinking) of the network and reduction of the size of output data for $L = 2$ and $L = 4$ resulting in a number of 14 and 16 parameters, respectively. Source: own work.

It should be noted that the number of outputs of the structure can be adjusted depending on the value of L parameter by applying the pruning procedure, i.e. the operation of removing redundant neurons (see Fig. 1). In this way, we can only get the number of outputs needed to store the largest amount of input energy. If we heuristically equate the amount of energy with the information carried by the signal, then prunning may allow to achieve the desired results by operating on data in less dimensional space.

In turn, in Fig. 2 we present exemplary results of energy compaction obtained for $N = 8$ and 1-st order Markov signals with autocorrelation coefficients $\rho = 0.7$ and $\rho = 0.95$. Based on the obtained results, we may conclude that the

proposed network constructed with use of LERU neurons allows for good energy compaction in a small number of output coefficients.

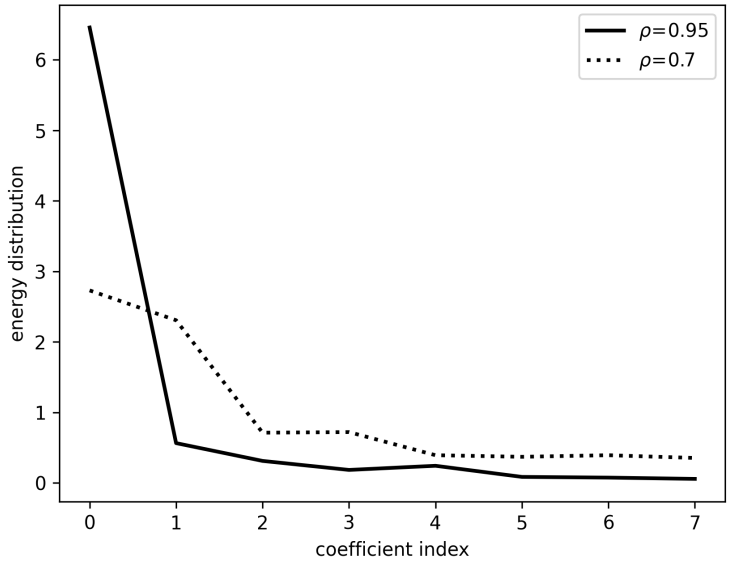


Figure 2. The results in energy compaction by LERU based network for $N = 8$ while operating on 1-st order Markov signals for autocorrelation $\rho = 0.7$ (dotted line) and $\rho = 0.95$ (solid line). Source: own work.

4. Experimental results

In the experimental part of the research work, we focused on practical verification of the effectiveness of the proposed approach. For this purpose, we applied LERU neurons organized in a network similar to the one shown in Fig. 1 (but for input data of size $N = 28$) to the task of handwritten digits classification from scratch using the well-known MNIST (Modified National Institute of Standards and Technology) dataset [5]. The MNIST dataset contains 70000 grayscale images of handwritten digits with resolution of 28 by 28 pixels grouped into training and test datasets in proportions of 60000 to 10000 images. Note that in traditional multi-layer perceptron (MLP) based approach, the input images fed to the classifying neural network are flattened into one-dimensional vectors of size 784 elements, which form the input dataset. At the output of the network, we expect 10-element vectors describing the probability distribution of input data over the set of 10 possible classes. With use of the traditional approach, in this specific experiment, the obtained accuracy of classification was at the level of 97.68%.

Within the framework under consideration adopting LERU neurons, we used a different experimental setup. First, individual rows of input images were fed into LERU-based neural networks, resulting in a number of 28 separate networks. The role of such networks was to reduce space dimensionality of data. The degree of dimensionality reduction depended directly on the number of network outputs and this value was defined by the parameter $L \in \{1, 2, 4, 6, 8\}$. Next, the concatenated outputs of LERU-based networks were fed to the input of the MLP classifier, giving vectors of size $28 * L$ elements. The results obtained in this experiment are summarized in Table 1. It should be noted, as previously mentioned, that the similar classifier but operating on the full image, i.e., $28 * 28$-element input vectors, allowed to obtain the accuracy of classification at the level of 97.68%. The MLP classifier used was a 3-layer network with 128, 64 and 10 biased neurons in each layer, respectively, and with ReLU activation function in two first layers, and softmax function in the output layer. During the training process we used Adam optimizer and categorical cross-entropy as the loss function.

Table 1. The experimental results in classification accuracy obtained for MNIST dataset and different values of parameter $L \in \{1, 2, 4, 6, 8\}$.

Dataset	$L=1$	$L=2$	$L=4$	$L=6$	$L=8$
Training	91.51%	99.88%	100.00%	100.00%	100.00%
Test	88.26%	95.58%	95.87%	97.35%	97.39%

The analysis of results from Table 1 allows to notice that the accuracy of classification for the values of parameter $L=6$ and $L=8$ is close to the accuracy obtained with input data vectors containing whole images. For $L=6$ it is smaller only by 0.33%, and for $L=8$ by 0.29%. At the same time the reduction of the size of input vectors for the MLP classifier results in the essential reduction of the number of weights that must be trained and stored. The number of weights for whole input images equals to 109386. But with $L=8$ it is 37772 weights, and with $L=6$ we have 30602 weights. This gives the following weight reduction percentages, i.e. 65.5% and 72% respectively.

5. Conclusions

Based on the obtained results, we can definitely conclude that the proposed approach based on networks constructed using LERU neurons can be used in the tasks of classification in order to reduce the dimensionality of data representation. The reduction in dimensionality translates directly into significantly smaller numbers of weights that must be trained and stored within the trained models.

References

[1] Robinett R.A., Kepner J., *Neural network topologies for sparse training*, [In:] *IEEE MIT Undergraduate Research Technology Conference (URTC)*, 2018.

[2] Puchala D., Stokfiszewski K., *Sparse neural networks with topologies inspired by butterfly structures*, [In:] *Signal Processing Symp. (SPSympo)*, 2021.

[3] Masana M., van de Weijer J., Herranz L., Bagdanov A.D., Alvarez J.M., *Domain-adaptive deep network compression*, [In:] *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[4] Dang L., Pang P., Lee J., *Depth-wise separable convolution neural network with residual connection for hyperspectral image classification*, [In:] *Remote Sensing*, vol. 12, 2020.

[5] LeCun Y., Bottou L., Bengio Y., Haffner P., *Gradient-based learning applied to document recognition*, [In:] *Proc. of the IEEE*, vol. 86, 1998.

MPTCP Congestion Control Algorithms for Streaming Applications – Performance Evaluation in Public Networks

Łukasz Piotr Łuczak[0000−0003−0892−7276],
Przemysław Ignaciuk[0000−0003−4420−9941],
Michał Morawski[0000−0002−8902−1259]

Lodz University of Technology
Institute of Information Technology
Wólczańska 215, 90-924 Łódź, Poland
lukasz.luczak@dokt.p.lodz.pl

DOI:10.34658/9788366741928.57

Abstract. *Efficient data transfer for high-quality streaming requires fast speed, low latency, and stable transmission parameters. Utilizing multiple communication paths is a promising solution for improving performance. This paper evaluates the most common MPTCP congestion control algorithms in the context of streaming applications in the open Internet. The results show that the BALIA algorithm is the most effective CC algorithm for multi-path streaming. This algorithm achieves the lowest path delay and Head-of-Line blocking degree with consistent throughput. Conversely, the MPTCP CC algorithm wVegas exhibits the weakest performance.*
Keywords: *MPTCP, congestion control, streaming applications*

1. Introduction

IP-based systems are increasingly replacing other network solutions in various industries, such as traditional telecommunications, medicine, entertainment, Internet of Things, and tactile Internet. While IP networks have the advantage of universality and ease of expansion, they lack control over transmission quality. The widespread use of mobile devices with multiple network interfaces presents additional challenges, including rapidly varying link parameters and the need to smoothly switch to another network. To address these challenges, Multipath TCP (MPTCP) was proposed as a TCP protocol tailored for multi-interface traffic [1, 2]. However, MPTCP congestion control (CC) algorithms were primarily designed to boost efficiency without compromising fairness [3, 4], rather than handling delay constraints critical for streaming applications [5]. This study aims to examine the

suitability of popular MPTCP CC algorithms for streaming applications, which require short latency, low error rate, and low jitter, while avoiding extensive buffering [6].

The research on network protocols often relies on simulations or tests conducted in controlled environment, which may not always reflect the real-world conditions. Therefore, this study aims to evaluate the performance of various CC algorithms in the open Internet. The results show that the BALIA algorithm achieves the lowest path delay and Head-of-Line (HoL) blocking degree, along with the highest throughput, making it a suitable candidate for streaming services. Conversely, the wVegas algorithm exhibits the weakest performance among the four tested algorithms in that context.

2. MPTCP Congestion Control Algorithms

The TCP protocol currently underpins the majority of Internet services, with its different versions tailored for specific scenarios through the method used to adjust the transfer speed. While Cubic is the default CC algorithm in Windows and Linux, MPTCP provides additional algorithms [7]: BaLIA, DMCTCP, LIA, OLIA, and wVegas, which are the focus of this study. None of these algorithms are optimized for the streaming traffic, which presents challenges for the transmission of data-intensive multimedia content [8]. It is therefore necessary to evaluate the performance of MPTCP CC algorithms in the context of streaming services and indicate the most suitable algorithm for this type of content.

The LIA, OLIA, BALIA, and wVegas CC algorithms are designed based on the principle of protocol fairness. LIA accelerates the transmission speed faster than the slowest path, while OLIA responds to channel disparities and fluctuations by analyzing the underlying single path control variables. OLIA is better suited for heterogeneous environments. BALIA is a hybrid algorithm that combines the strengths of LIA and OLIA, allowing it to perform well in both homogeneous and heterogeneous environments. Its main advantage is the ability to dynamically adjust the aggressiveness of CC based on network conditions. Finally, wVegas adjusts the congestion window size using the estimated round-trip time and packet loss rate, performing well in high-speed and long-distance connectivity but less effectively in the congested or lossy networks.

3. Quality Measures

Various unpredictable events, such as congestion or buffering, can affect the transmission parameters. Therefore, the following quality metrics were utilized as objective quality measures in the CC algorithm assessment.

3.1. Path Delay

The path delay refers to the time it takes for a packet to traverse a path from the sender to the receiver. In the model developed in this paper, the total delay on path i, denoted by T_i, comprises the SRTT of this path τ_i and the waiting time for processing the data stream θ_i:

$$T_i = \tau_i + \theta_i. \tag{1}$$

3.2. Protocol Delay

The transmitted data is subjected to various types of delays, and the values of these delays vary over time. The protocol delay is defined as the time that a given piece of data, such as a packet, waits in the buffer for the content reassembly. The total delay of the slowest path is taken as the measure of the protocol delay

$$T_{over}(k) = \max_i T_i(k). \tag{2}$$

In the experiments, an average and maximum protocol delay were also calculated as

$$\upsilon^r_{av} = \frac{1}{K}\sum_{k=1}^{K} T_{over}(k), \text{ and } \upsilon^r_{\max} = \max_{k\in[1,K]} T_{over}(k) \tag{3}$$

and the mean protocol delay from all the experiments as

$$\upsilon_{av} = \frac{1}{R}\sum_{r=1}^{R} \upsilon^r_{av}, \text{ and } \upsilon_{\max} = \frac{1}{R}\sum_{r=1}^{R} \upsilon^r_{\max}. \tag{4}$$

3.3. HoL Degree

In a multi-path transmission, the HoL blocking degree is defined as the number of packets that are queued up and waiting to be transmitted in a certain path prior to the HoL packet being delivered.

For each experiment an average and maximum wating time were calculated as

$$\zeta^r_{av} = \frac{1}{K}\sum_{k=1}^{K}\left(\max\left(T_{over}(k) - \max_{i\in[1,m]} \tau_i(k), 0\right)\right), \text{ and} \tag{5}$$

$$\zeta^r_{\max} = \max_{k\in[1,K]}\left(T_{over}(k) - \max_{i\in[1,m]} \tau_i(k)\right) \tag{6}$$

and the mean waiting time from all the experiments as

$$\zeta_{av} = \frac{1}{R}\sum_{r=1}^{R} \zeta^r_{av}, \text{ and } \zeta_{\max} = \frac{1}{R}\sum_{r=1}^{R} \max(\zeta^r_{\max}, 0). \tag{7}$$

4. Experimental Setup

To thoroughly investigate the characteristics of CC algorithms, a test set reflecting a typical data transmission scenario was created. In this scenario, the data are received by a client connected to a high-end server device via a public network. The server is located in a remote location accessible through a public IP address. To obtain an input stream with pre-defined parameters, a specialized program was utilized.

Both the client and server devices are running on the Linux operating system version 4.19, which had been patched to support MPTCP version 0.95. The client device is equipped with two communication interfaces, both of which are connected to an LTE router via an Ethernet cable. One interface transmits packets over 13 hops, while the other over 15 hops. Each scenario was designed to last for 10 seconds and was repeated 30 times to ensure statistical significance.

Table 1. Impact of congestion control algorithms on streaming transmission

		LIA	**OLIA**	**BALIA**	**wVegas**
Protocol delay	υ_{av}	115	116	113	120
[ms]	υ_{max}	860	885	219	1552
HoL Degree	ζ_{av}	29	29	27	32
[ms]	ζ_{max}	769	794	124	1445
SRTT [ms]	$\tau_{1,av}$	84	85	83	86
path 1	$\tau_{1,max}$	115	117	120	156
SRTT [ms]	$\tau_{2,av}$	74	74	73	74
path 2	$\tau_{2,max}$	91	90	92	91
Throughput	*av*	6.30	5.99	6.38	5.32
[Mbps]	*max*	11.55	11.37	11.90	9.54

5. Tests and Results

Four CC algorithms: LIA, OLIA, BALIA, and wVegas, were tested for MPTCP streaming capability assessment. Table 1 summarizes the results. The gathered data demonstrate that the BALIA algorithm leads to the best overall performance. It is particularly visible in terms of HoL blocking degree, implying its superior handling of that multi-path systemic problem. Similar outcomes concern the protocol delay, where the BALIA algorithm also exhibits the lowest average and maximum values, indicating its effectiveness in reducing the protocol delay in the MPTCP transmission. It also achieves the highest throughput.

Overall, the BALIA algorithm reveals to be the most effective CC algorithm for MPTCP streaming transmission, achieving the lowest delay and best handling of HoL blocking, whereas using wVegas is ill-advised.

6. Conclusions

The problem of establishing an efficient data transfer over heterogeneous public networks is the main focus of this study. It was observed that the use of multiple communication paths could potentially be a solution for achieving high-speed, low latency, and stable transmission parameters, which are essential for high-quality streaming. The MPTCP protocol was used to govern the multi-path transfer. It was found that the transmission quality is decisively affected by the choice of CC protocol.

The results indicate that multi-path transmission could increase latency, but it is deemed acceptable and should not negatively affect streaming. Among the tested algorithms, BALIA proved to be the most effective for MPTCP streaming scenario, providing the lowest latency and highest throughput.

Acknowledgment

This work has been performed in the framework of a project "Robust control solutions for multi-channel networked flows" no. 2021/41/B/ST7/00108 financed by the National Science Centre, Poland.

References

[1] Barreiros M., Lundqvist P., *QOS-Enabled Networks: Tools and Foundations*, 2015, doi: 10.1002/9781119109136.

[2] Morawski M., Ignaciuk P., *Network nodes play a game – a routing alternative in multihop ad-hoc environments*, *Computer Networks*, 2017, vol. 122, pp. 96–104, ISSN 1389-1286, doi: https://doi.org/10.1016/j.comnet.2017.04.031.

[3] M. Li et al., *Multipath transmission for the internet: A survey*, *IEEE Commun. Surv. TuT.*, 2016, vol. 18, no 4, pp. 2887–2925, doi: 10.1109/COMST.2016.2586112.

[4] Morawski M., Ignaciuk P., *A green multipath tcp framework for industrial internet of things applications*, *Computer Networks*, 2021, vol. 187, p. 107831, doi: https://doi.org/10.1016/j.comnet.2021.107831.

[5] Morawski M., Ignaciuk P., *Choosing a proper control strategy for multipath transmission in industry 4.0 applications*, *IEEE Transactions on Industrial Informatics*, 2022, vol. 18, no 6, pp. 3609–3619, doi: 10.1109/TII.2021.3105499.

[6] Yedugundla K., Ferlin S., Dreibholz T., Alay O., Kuhn N., Hurtig P., Brunstrom A., *Is multi-path transport suitable for latency sensitive traffic?*, *Comput. Netw.*, 2016, vol. 105, pp. 1–21, doi: https://doi.org/10.1016/j.comnet.2016.05.008.

[7] Xu C., Zhao J., Muntean G., *Congestion control design for multipath transport protocols: A survey*, *IEEE Commun. Surv. Tut.*, 2016, vol. 18, no 4, pp. 2948–2969, doi: 10.1109/COMST.2016.2558818.

[8] Grzyb S., Orłowski P., *Congestion feedback control for computer networks with bandwidth estimation*, [In:] *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 1151–1156, doi: 10.1109/MMAR.2015.7284041.

Optimized Mutation Operator in Evolutionary Approach to Stackelberg Security Games

Adam Żychowski[0000−0003−0026−5183], **Jacek Mańdziuk**[0000−0003−0947−028X]

Warsaw University of Technology
Faculty of Mathematics and Information Science
Koszykowa 75, 00-662 Warsaw, Poland
a.zychowski@mini.pw.edu.pl mandziuk@mini.pw.edu.pl

DOI:10.34658/9788366741928.58

Abstract. *In this paper, we introduce several mutation modifications in Evolutionary Algorithm for finding Strong Stackelberg Equilibrium in sequential Security Games. The mutation operator used in the state-of-the-art evolutionary method is extended with several greedy optimization techniques. Proposed mutation operators are comprehensively tested on three types of games with different characteristics (in total over* 300 *test games). The experimental results show that application of some of the proposed mutations yields Defender's strategies with higher payoffs. A trade-off between the results quality and the computation time is also discussed.*

Keywords: *Security Games, Stackelberg Equilibrium, Evolutionary Computation*

1. Introduction

Evolutionary Algorithms (EAs) are popular and powerful population-based metaheuristic optimization methods. Their effectiveness strongly depends on designed evolutionary operators: mutation, crossover, and selection [1]. One of the potential improvements can be achieved by adding some local optimization techniques – either as a separate algorithm step or as a part of the evolutionary operators [2].

In this study, the above claim is verified with respect to the evolutionary algorithm for sequential Stackelberg Security Games (SSGs) [3], by means of an introduction and experimental evaluation of various greedy optimizations for the mutation operator. Some of the proposed modifications lead to better results and superior Defender's strategies.

2. Problem definition

Sequential SSGs are played by two players: the Defender (D) and the Attacker (A). Each game is composed of m time steps and each player chooses an action to be performed (simultaneously) in each time step. A player's *pure strategy* σ_P ($P \in \{D, A\}$) is a sequence of their actions in consecutive time steps: $\sigma_P = (a_1, a_2, \ldots, a_m)$. The set of all possible pure strategies of player P is denoted by Σ_P. A probability distribution $\pi_P \in \Pi_p$ over Σ_P is the player's *mixed strategy*, where Π_p is the set of all mixed strategies for player P.

For any pair of strategies (π_D, π_A) the expected payoffs for the players are denoted by $U_D(\pi_D, \pi_A)$ and $U_A(\pi_D, \pi_A)$. The goal of the game is to find the *Strong Stackelberg Equilibrium* (SSE), i.e. a pair of strategies (π_D, π_A) satisfying the following conditions:

$$\pi_D = \arg\max_{\bar{\pi}_D \in \Pi_D} U_A(\bar{\pi}_D, BR(\bar{\pi}_D)), \qquad BR(\pi_D) = \arg\max_{\pi_A \in \Pi_A} U_A(\pi_D, \pi_A). \quad (1)$$

The first equation chooses the best Defender's strategy π_D under the assumption that the Attacker always selects the best response strategy ($BR(\pi_D)$) to the Defender's committed strategy. If there exists more than one optimal Attacker's response (with the same highest Attacker's payoff), the Attacker selects the one with the highest corresponding Defender's payoff, i.e. breaks ties in favor of the Defender [4].

Both players choose their strategies at the beginning of a game (first the Defender and then the Attacker) and the strategies cannot be altered during the course of the game. The problem of finding SSE has been proven to be NP-hard [5].

3. Evolutionary Algorithm

The Evolutionary Algorithm for Stackelberg Games (EASG) [6] aims to optimize the Defender's payoff by evolving a population of Defender's mixed strategies. Initially, EASG creates a population of pure Defender's strategies selected at random. The population evolves over successive generations until the stopping criterion is met. Four operations are applied in each generation: crossover, mutation, evaluation, and selection.

Crossover randomly selects two individuals from the population and combines their pure strategies by halving their probabilities and merging them into a single chromosome. The resultant chromosome is simplified by deleting some of its pure strategies, with the probability of deletion being inversely proportional to their probabilities.

The mutation operator randomly selects a pure strategy encoded in the chromosome and modifies it, starting from a randomly selected time step. New actions are drawn from the set of all feasible actions in a given game state.

Each individual in the population is then assigned a fitness value, which represents the expected Defender's payoff. This requires finding the optimal Attacker's response to the mixed Defender's strategy encoded in the chromosome. EASG accomplishes this by iterating over all possible Attacker's pure strategies and selecting the one with the highest Attacker's payoff [6].

In the selection phase, individuals with higher Defender's payoffs have a higher likelihood of being chosen for the next generation. This is achieved through a binary tournament in which two chromosomes are repeatedly selected with return, and the one with the higher fitness value is promoted with a certain probability (the *selection pressure*). Additionally, a set of chromosomes with the highest fitness function value is unconditionally copied to the next generation to preserve the best solutions found so far (the *elite* mechanism).

EASG is a generic framework which can be applied to various SSGs. For instance, it has been successfully applied to games with moving targets [7], signaling games [8], or games that assume bounded rationality of the Attacker [9, 10].

4. Mutation modifications

In EASG the mutation changes random actions from randomly selected pure strategy. We observed that this approach rarely leads to individuals with higher fitness function. For some types of games it is less than 6% of mutation executions, while the recommended literature standard is 20% – *one-fifth rule* [11]. **In order to improve the mutation impact, we propose and test various mutation implementations, other than the baseline [6], which may potentially improve Defender's strategies encoded in chromosomes.**

We test 11 different types of mutation modifications, which are briefly described below. Rather than entirely replacing the original mutation operator in EASG we propose that the new type of mutation be applied (replace the original one) with a probability equal to 0.5. All other EASG operators and parameters remain the same as reported in [6]. Depending on the number of mutations applied, we generally distinguish two cases:
(1) The mutation is applied only once and its result is preserved regardless of the resulting impact on the individual fitness (this approach was used in EASG). All such variants will be denoted with a subscript 1, e.g. MNPS_1,
(2) The mutation is repeated until a better solution is found or a predefined limit of trials n is reached. After each trial, the chromosome is reverted to its previous form if its fitness deteriorates. Such variants will be denoted by subscript n, e.g. MNPS_n. In the experiments, $n = 50$ was used.

In either case, if the encoded mixed strategy probabilities have changed as a result of applied mutation, they are normalized to sum up to 1. The following mutation enhancements have been tested:

- **EASG**$_n$ – EASG algorithm with repeated mutation.
- **MANPS**$_1$, **MANPS**$_n$ – *mutation adds new pure strategy* – a uniformly selected pure strategy is added to a chromosome, with a uniformly sampled probability.
- **MCP**$_1$, **MCP**$_n$ – *mutation changes probability* – a probability of randomly selected pure strategy is uniformly changed.
- **MSP**$_1$, **MSP**$_n$ – *mutation switches probability* – probabilities of two randomly chosen pure strategies are switched.
- **MDPS**$_1$, **MDPS**$_n$ – *mutation deletes pure strategy* – a randomly chosen pure strategy is removed.
- **MCWPS** – *mutation changes the weakest pure strategy* – mutation is applied only to a pure strategy with the lowest payoff.
- **MDWPS** – *mutation deletes the weakest pure strategy* – pure strategy with the lowest payoff is deleted.

5. Results and Conclusions

All mutation variants have been tested on 3 types of Security Games: Warehouse Games (WHG) [12], Search Games (SEG) [13], and FlipIt Games (FIG) [14]. The same game instances were also used for EASG evaluation [6]. Please refer to [6] for a detailed description of the rules and characteristics of the games.

Table 1 shows experimental results averaged over 30 independent runs and all game instances – 150 WHG, 90 SEG, and 60 FIG. The biggest improvement is observed for SEG. This may be attributed to the fact that SEG is the most complex type of game, with the largest search space. Variants with mutation repetitions yield higher results improvements, but also lead to a significant (approximately tenfold) increase in computation time. The reason for that is frequent evaluation (for each chromosome, after each mutation attempt) needed to decide whether the mutation results should be retained or another mutation attempt should be used.

A greedy selection of the worst pure strategies (MCWPS and MDWPS) turned out to be an ineffective approach. This is most likely attributed to the fact that considering the quality of individual pure strategies in isolation from the other elements of a given mixed strategy may not be the right approach. Even if a single pure strategy is weak on its own, it may play a crucial role in the overall mixed strategy by rendering the decision that is unfavorable for the Defender to be also unprofitable for the Attacker.

Overall, the results show that repetition of mutation operation generally leads to improvement of SSGs outcomes, though at the expense of significant increase in

Table 1. The average and standard deviation values of the Defender's payoff and the computation time for various mutation operators. The best results are **bolded**. Results that are better than the baseline version of the algorithm (EASG) are underlined. In cases where the difference between the baseline version (EASG) and a given variation is statistically significant (according to the Wilcoxon test with p-value < 0.05), the result is highlighted with a gray background.

	Defender's payoff			Computation time [s]		
	WHG	SEG	FIG	WHG	SEG	FIG
EASG	**0.017** ±0.001	0.108 ±0.006	0.031 ±0.002	152±6	2534±150	328±20
EASG$_n$	**0.017** ±0.001	0.135 ±0.008	**0.037** ±0.003	1206±84	21913±1264	3051±224
MANPS$_1$	0.014 ±0.001	0.059 ±0.004	0.031 ±0.002	156±8	2548±119	313±11
MANPS$_n$	0.016 ±0.001	**0.139** ±0.013	0.036 ±0.002	1366±62	21892±1463	2988±112
MCP$_1$	0.015 ±0.001	0.074 ±0.007	0.030 ±0.002	148±6	2422±82	336±19
MCP$_n$	0.016 ±0.001	0.131 ±0.012	**0.037** ±0.002	1285±91	22651±751	3008±145
MSP$_1$	0.013 ±0.001	0.099 ±0.007	0.024 ±0.001	156±7	2583±124	316±15
MSP$_n$	0.016 ±0.001	0.108 ±0.006	**0.037** ±0.004	1332±84	21447±1594	2931±203
MDPS$_1$	0.013 ±0.001	0.052 ±0.005	0.029 ±0.002	147±8	2620±79	313±15
MDPS$_n$	0.013 ±0.001	0.053 ±0.005	0.026 ±0.002	1283±81	22026±1599	2900±111
MCWPS	0.013 ±0.001	0.046 ±0.004	0.030 ±0.003	148±6	2612±151	321±20
MDWPS	0.008 ±0.002	0.058 ±0.004	0.018 ±0.002	139±4	2361±141	299±11

computation time. Hence, in situations when computational cost is less important and obtaining the best possible result is critical, the proposed modifications offer a viable alternative to the base EASG formulation.

References

[1] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer Berlin Heidelberg, 1996.

[2] Neri F., Cotta C., *Memetic algorithms and memetic computing optimization: A literature review*, *Swarm and Evolutionary Computation*, 2012, vol. 2, pp. 1–14.

[3] Sinha A., Fang F., An B., Kiekintveld C., Tambe M., *Stackelberg Security Games: Looking Beyond a Decade of Success*, [In:] *Proceedings of the 27th IJCAI conference*, pp. 5494–5501.

[4] Breton M., Alj A., Haurie A., *Sequential stackelberg equilibria in two-person games*, *Journal of Optimization Theory and Applications*, 1988, vol. 59, no 1, p. 71–97.

[5] Conitzer V., Sandholm T., *Computing the optimal strategy to commit to*, [In:] *Proceedings of the 7th ACM conference on Electronic commerce, 2006*, pp. 82–90.

[6] Żychowski A., Mańdziuk J., *Evolution of Strategies in Sequential Security Games*, [In:] *Proceedings of the 20th AAMAS conference, 2021*, pp. 1434–1442.

[7] Karwowski J., Mańdziuk J., Żychowski A., Grajek F., An B., *A memetic approach for sequential security games on a plane with moving targets*, [In:] *Proceedings of the 33rd AAAI conference, 2019*, vol. 33, pp. 970–977.

[8] Żychowski A., Mańdziuk J., Bondi E., Venugopal A., Tambe M., Ravindran B., *Evolutionary approach to Security Games with signaling*, *Proceedings of the 31st IJCAI conference, 2022*, 2022, pp. 620–627.

[9] Żychowski A., Mańdziuk J., *Learning attacker's bounded rationality model in security games*, [In:] *Proceedings of the 28th ICONIP, 2021*, pp. 530–539.

[10] Karwowski J., Mańdziuk J., Żychowski A., *Sequential stackelberg games with bounded rationality*, *Applied Soft Computing 2023*, 2023, vol. 132, p. 109846.

[11] Eiben A.E., Michalewicz Z., Schoenauer M., Smith J.E., *Parameter control in evolutionary algorithms*, *Parameter setting in evolutionary algorithms*, 2007, pp. 19–46.

[12] Karwowski J., Mańdziuk J., *A Monte Carlo Tree Search approach to finding efficient patrolling schemes on graphs*, *European Journal of Operational Research 2019*, 2019, vol. 277, pp. 255–268.

[13] Bošanský B., Čermak J., *Sequence-form algorithm for computing stackelberg equilibria in extensive-form games*, [In:] *Proceedings of the 29th AAAI conference, 2015*, pp. 805–811.

[14] Van Dijk M., Juels A., Oprea A., Rivest R.L., *Flipit: The game of "stealthy takeover"*, *Journal of Cryptology*, 2013, vol. 26, no 4, pp. 655–713.

Simulation of the Quantum Heat Engine in the Quantum Register

Marcin Ostrowski[0000−0001−8985−8123]

Lodz University of Technology
Institute of Information Technology
Wólczańska 215, 90-924 Łódź, Poland
marcin.ostrowski@p.lodz.pl

DOI:10.34658/9788366741928.59

Abstract. *This paper investigates whether a quantum computer can efficiently simulate the transfer of excitation between a pair of quantum systems with energy loss caused by photon or phonon emission. The main contribution of our work is an algorithm that enables the simulation of time evolution of such a system, implemented on a standard two-input gates. The paper examines the properties of the proposed algorithm and then compares the obtained results with theoretical predictions.*
Keywords: *quantum simulations, excitation transfer, quantum heat engine*

1. Introduction

In the near future, quantum calculations are likely to make a major contribution to the development of informatics [1]. Nowadays, some institutions claim to have a quantum computer and offer its computing power. Therefore, it is worth examining the properties of such a machine.

For many years, we have known Shor [2] and Grover [3] algorithms which are faster than their best classical counterparts. Another promising application of a quantum computer is quantum simulation [4], i.e. the computer modeling of behavior of physical quantum systems. It gives the possibility of effective modeling quantum processes, which is not possible using classical computers. Quantum computers can simulate a wide variety of quantum systems, including fermionic lattice models [5], quantum chemistry and quantum field theories [6].

In the present study, we consider a quantum system (the system A from Fig. 1), which returns to the ground state with partial energy transfer to another system (the system B). The rest of the energy is emitted as a photon or phonon (the system C). The process described above is equivalent to operation of the quantum heat engine. It occurs during optical pumping of the laser or during photosynthesis [7], where the energy of an absorbed photon is transferred at a loss in many steps between successive carriers.

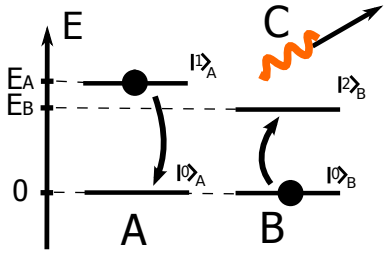


Figure 1. Energy transfer between subsystems A (hot reservoir) and B (quantum heat engine) with emission of photon C (cold reservoir). Source: own work.

The main purpose of our work is to investigate whether this phenomenon could be simulated in the quantum register. In the future, the ability to simulate such phenomena in a quantum computer may prove useful in the process of designing new quantum heat engines. Due to its low complexity level, the algorithm presented here may be used as a part of a more complex simulation.

The results presented here are preliminary. The issue we are considering is related to the optimization of the energy transport process in biological and artificial systems. It is claimed that the process of photosynthesis gains light-harvesting efficiency by exploiting the phenomenon of quantum coherence [8]. This involves the superpositions of electronic quantum states, which seem able to explore many energy-transmitting pathways at once.

2. Description of the simulated system

Let us consider a complex quantum system that is composed of three parts: A, B and C. The subsystem B (quantum heat engine) has three nondegenerate energy levels, which are denoted as follows: $|0\rangle_B$, $|1\rangle_B$ and $|2\rangle_B$. Energies of these states are equal to $E_0 = 0$, $E_1 > 0$ and $E_2 = E_B > 0$, respectively. Subsystems A (hot reservoir) and C (cold reservoir) have two nondegenerate energy levels. Stationary states of the system A we denote by $|0\rangle_A$ and $|1\rangle_A$. We assign them energies equal to $E_0 = 0$ and $E_A > 0$, respectively. Analogously, stationary states of the system C we denote by $|0\rangle_C$, $|1\rangle_C$, and we assign them energies equal to $E_0 = 0$ and $E_C > 0$, respectively. Full structure of the system is shown in Fig. 2.

The free Hamiltonian of the system we can write in the following form:

$$\hat{H}_0 = E_A \hat{a}^\dagger \hat{a} + E_1 \hat{b}_1^\dagger \hat{b}_1 + E_2 \hat{b}_2 \hat{b}^\dagger + E_C \hat{c}^\dagger \hat{c}, \tag{1}$$

where increasing and decreasing energy operators are defined as follows:

$$\hat{a}^\dagger |0\rangle_A = |1\rangle_A, \quad \hat{b}_1^\dagger |0\rangle_B = |1\rangle_B, \quad \hat{b}_2^\dagger |2\rangle_B = |1\rangle_B, \quad \hat{c}^\dagger |0\rangle_C = |1\rangle_C, \tag{2}$$

$$\hat{a}|1\rangle_A = |0\rangle_A, \quad \hat{b}_1|1\rangle_B = |0\rangle_B, \quad \hat{b}_2|1\rangle_B = |2\rangle_B, \quad \hat{c}|1\rangle_C = |0\rangle_C.$$

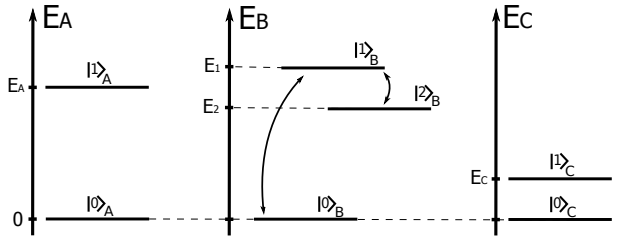


Figure 2. Scheme of the simulated systems. Source: own work.

The Hamiltonian of interaction we choose in the following form:

$$\hat{H}_{int} = g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger + g_2\hat{c}^\dagger\hat{b}_2 + g_2^*\hat{c}\hat{b}_2^\dagger, \tag{3}$$

where g_1 and g_2 are coupling constants.

For the Hamiltonian (3) interaction between subsystems A and B generate the transition $|1\rangle_A|0\rangle_B \leftrightarrow |0\rangle_A|1\rangle_B$. Analogously, interaction between subsystems B and C generate the transition $|1\rangle_B|0\rangle_C \leftrightarrow |2\rangle_B|1\rangle_C$.

In the presented system resonance occurs when:

$$E_A = E_2 + E_C. \tag{4}$$

If we additionally assume that

$$E_A = E_1, \tag{5}$$

resonances occur in AB and BC subsystems.

In this work we are also considering modification of the system presented above. We replace the $|1\rangle_C$ state with a band consisting $n_L - 1$ excited levels with energies given by:

$$E_{C_i} = \Delta E \cdot i \qquad \text{for } i = 0, \ldots, n_L - 1, \tag{6}$$

where $\Delta E = 2E_C/n_L$ is the distance between adjacent levels. Now the E_C is the middle energy level of the band. For this modification the Hamiltonian of interaction takes the form:

$$\hat{H}'_{int} = g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger + \sum_{i=1}^{n_L-1}(g_2\hat{c}_i^\dagger\hat{b}_2 + g_2^*\hat{c}_i\hat{b}_2^\dagger), \tag{7}$$

where $\hat{c}_i^\dagger|0\rangle_C = |i\rangle_C$ and $\hat{c}_i|i\rangle_C = |0\rangle_C$. The extended system C simulate quantum field, which receives energy from the system B irreversibly.

3. Algorithm simulating time evolution of the system

In order to solve the Schrödinger equation for the Hamiltonian $\hat{H} = \hat{H}_0 + \hat{H}_{int}$ we use the time evolution operator in the form:

$$\hat{U}(\mathrm{d}t) = \exp(-\mathrm{i}\hat{H}\mathrm{d}t/\hbar), \tag{8}$$

where $\mathrm{d}t$ is time step. For $\mathrm{d}t \to 0$ operator given by Eq. (8) can be approximated as follows:

$$\begin{aligned}\hat{U}(dt) = \exp(-\mathrm{i}E_A\hat{a}^\dagger\hat{a}\mathrm{d}t/\hbar)\exp\Big(-\mathrm{i}(E_1\hat{b}_1^\dagger\hat{b}_1 + E_2\hat{b}_2\hat{b}_2^\dagger)\mathrm{d}t/\hbar\Big)\exp(-\mathrm{i}E_C\hat{c}^\dagger\hat{c}\mathrm{d}t/\hbar)\\ \times\exp\Big(-\mathrm{i}(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)\mathrm{d}t/\hbar\Big)\exp\Big(-\mathrm{i}(g_2\hat{c}^\dagger\hat{b}_2 + g_2^*\hat{c}\hat{b}_2^\dagger)\mathrm{d}t/\hbar\Big).\end{aligned} \tag{9}$$

The above equation is equivalent to using the first-order Lie-Trotter formula with Trotter error $O(t^2)$. The simulation for a large time t is obtained by dividing the evolution into n Trotter steps ($t = n \cdot \mathrm{d}t$).

The basic version of the algorithm (simulating the system from Fig. 2) is implemented in a four-qubit register, as shown in Fig 3. Stationary states of the subsystem B are encoded in B_1 and B_2 qubits in the following way: $|0\rangle_B \to |0\rangle_{B_2}|1\rangle_{B_1}$, $|1\rangle_B \to |1\rangle_{B_2}|1\rangle_{B_1}$ and $|2\rangle_B \to |1\rangle_{B_2}|0\rangle_{B_1}$. Base state $|0\rangle_{B_2}|0\rangle_{B_1}$ is not used.

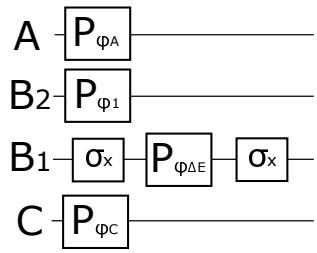


Figure 3. Scheme of the free evolution algorithm. Source: own work.

The free evolution of the system described by Eq. (1) is implemented by the algorithm showed in Fig. 3. Gates σ_x are standard NOT gates. Gates P_ϕ are standard phase-shift gates that operate according to the scheme:

$$|0\rangle \to |0\rangle, \qquad |1\rangle \to \mathrm{e}^{-i\phi}|1\rangle, \tag{10}$$

where: $\phi_A = E_A\hbar^{-1}\mathrm{d}t$, $\phi_1 = E_1\hbar^{-1}\mathrm{d}t$, $\phi_{\Delta E} = (E_2 - E_1)\hbar^{-1}\mathrm{d}t$, $\phi_C = E_C\hbar^{-1}\mathrm{d}t$ and $\mathrm{d}t$ is time step.

Implementation of the algorithm simulating interaction described by the Hamiltonian (3) is shown in the left drawing in Fig. 4. Three-input gates R_ϕ operate as follows:

$$|1\rangle|1\rangle|0\rangle \to \cos\phi\,|1\rangle|1\rangle|0\rangle + \sin\phi\,|1\rangle|1\rangle|1\rangle, \tag{11}$$

$$|1\rangle|1\rangle|1\rangle \to \cos\phi\,|1\rangle|1\rangle|1\rangle - \sin\phi\,|1\rangle|1\rangle|0\rangle, \tag{12}$$

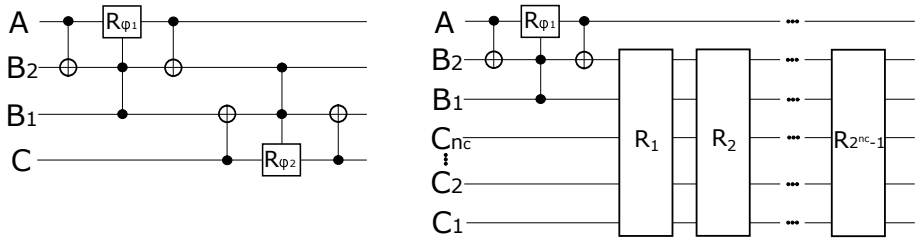


Figure 4. Scheme of the algorithm simulating interaction between subsystems. The left drawing shows the case for the Hamiltonian (3), while the right one shows the case for the Hamiltonian (7). Source: own work.

where $\phi_1 = |g_1|\mathrm{d}t/\hbar$ and $\phi_2 = |g_2|\mathrm{d}t/\hbar$.

Implementation of the interaction algorithm can be obtained by expressing the last two components from Eq. (9) in the following way:

$$\begin{aligned}\exp\left(-\mathrm{i}(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)\mathrm{d}t/\hbar\right) &= \sum_{j=0}^{\infty}\frac{1}{j!}\left(-\frac{i\,dt}{\hbar}\right)^j(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^j = \\ &= \sum_{j=0}^{\infty}\frac{(-1)^j}{(2j)!}\left(\frac{dt}{\hbar}\right)^{2j}(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j} + \\ &\quad + i\sum_{j=0}^{\infty}\frac{(-1)^j}{(2j+1)!}\left(\frac{dt}{\hbar}\right)^{2j+1}(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j+1} \qquad (13)\end{aligned}$$

and using the following formulas:

$$(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j}|0\rangle_A|1\rangle_B = |g_1|^{2j}|0\rangle_A|1\rangle_B, \qquad (14)$$

$$(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j}|1\rangle_A|0\rangle_B = |g_1|^{2j}|1\rangle_A|0\rangle_B, \qquad (15)$$

$$(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j+1}|0\rangle_A|1\rangle_B = |g_1|^{2j}g_1|1\rangle_A|0\rangle_B, \qquad (16)$$

$$(g_1\hat{a}^\dagger\hat{b}_1 + g_1^*\hat{a}\hat{b}_1^\dagger)^{2j+1}|1\rangle_A|0\rangle_B = |g_1|^{2j}g_1^*|0\rangle_A|1\rangle_F. \qquad (17)$$

The implementation of the algorithm simulating interaction of the extended system (described by the Hamiltonian (7)) is shown in the right drawing in Fig. 4. In this case, we simulate the subsystem C in n_c qubit subregister ($n_c = n_q - 3$). In each base state of the subregister, a single energy level is encoded. Therefore, the total number of energy levels of the subsystem C is equal to $n_L = 2^{n_c}$. The state $|0\rangle_C$ we identify with the vacuum state (lack of a photon). The transition $|1\rangle_B|0\rangle_C \leftrightarrow |0\rangle_B|i\rangle_C$ (the i-th component of the sum from the Hamiltonian (7)) is implemented by R_i block. The implementation of R_i blocks can be found in [9].

4. Simulation results

In the first part of our consideration, we examine our algorithm for the Hamiltonian (3) with conditions (4) and (5). As an initial state of the simulated system we choose $|1\rangle_A|0\rangle_B|0\rangle_C$. The simulation parameters are: $dt = 10^{-16}$s, $E_A = E_1 = 2$eV, $E_C = 0.2$eV, and $E_2 = 1.8$eV. The results are shown in Fig. 5. We only present the probabilities for the subsystem B, because $p_{A^*} = p_{B0}$ and $p_{C^*} = p_{B2}$.

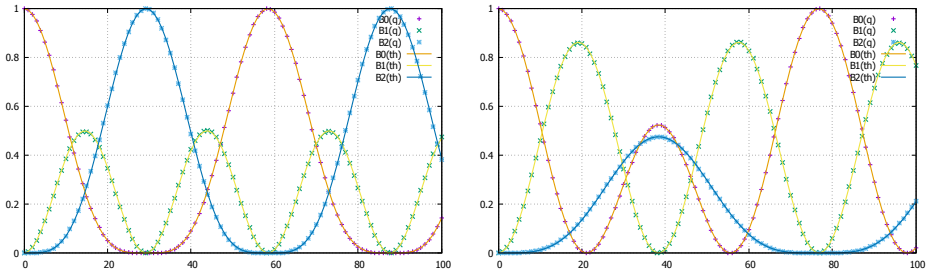


Figure 5. Probability of finding the system B in $|0\rangle_B$, $|1\rangle_B$ and $|2\rangle_B$ states as functions of time (in 10^{-15}s units). The left plot is made for $g_1 = g_2 = 0.05$eV, the right one is made for $g_1 = 0.05$eV and $g_2 = 0.02$eV. The dotted lines shows the results of the simulation. Solids lines represent comparative theoretical results. Source: own work.

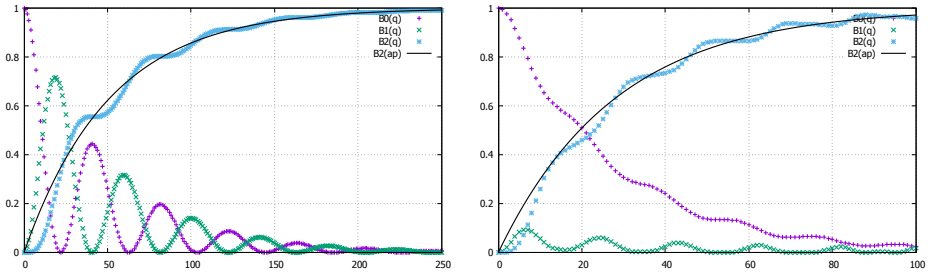


Figure 6. Probability of finding the system B in $|0\rangle_B$, $|1\rangle_B$ and $|2\rangle_B$ states as functions of time (in 10^{-15}s units). The left plot is made for $g_2 = 0.005$eV, the right one is made for $g_2 = 0.02$eV. The dotted lines shows the results of the simulation. Solids line represents result of exponential approximation. Source: own work.

In the next part of our consideration we examine the algorithm for the Hamiltonian (7) for $n_q = 9$ ($n_c = 6$). Other parameters take the following values: $dt = 10^{-16}$s, $E_A = 2$eV, $E_C = 0.2$eV (center of the band), $E_1 = 2$eV, $E_2 = 1.8$eV

and $g_1 = 0.05$eV. The results of the simulation are shown in Fig. 6.

5. Conclusions

- In Fig. 5 we can see very good consistency of results obtained by the simulation and by the comparative method.
- In the case of the extended system C we can observe exponential growth of the state $|2\rangle_B$ occupation probability.

References

[1] Feynman R., *Simulating physics with computers*, *Internat. J. Theor. Phys.*, 1982, vol. 21, pp. 467–488.

[2] Shor P.W., *Algorithms for quantum computation: Discrete logarithm and factoring*, *Proc 35th Ann. Symp. Found. Comp. Sci., IEEE Comp.Soc. Pr.*, 1994, pp. 124–134.

[3] Grover L.K., *From schrodinger equation to the quantum search algorithm*, *Am. J. Phys.*, 2001, vol. 69, pp. 769–777.

[4] Schaetz T., Monroe C., Esslinger T., *Focus on quantum simulation*, *New Journal of Phys.*, 2013, vol. 15.

[5] Kokail C., Maier C., van Bijnen R., *Self-verifying variational quantum simulation of lattice models*, *Nature*, 2019, vol. 569.

[6] Jordan S., Lee K., Preskill J., *Quantum algorithms for quantum field theories*, *Science*, 2012, vol. 336, p. 1130–1133.

[7] Dorfman K.E., Voronine D.V., *Photosynthetic reaction center as a quantum heat engine*, *PNAS*, 2013, vol. 110, no 8, p. 2746–2751.

[8] G. S. Engel T.R.C., *Evidence for wavelike energy transfer through quantum coherence in photosynthetic systems*, *Nature*, 2007, vol. 446, no 12/04.

[9] Ostrowski M., *Simulation of photon emission by an excited atom in the quantum register*, [In:] A. Wojciechowski, P. Napieralski, P. Lipiński (eds.), *TEWI 2021*, TUL Press, Lodz 2021, pp. 140–154.

Socio-cognitive Flock-based Optimization

Aleksandra Urbańczyk[0000−0002−6040−554X], **Krzysztof Czech, Aleksander Byrski**[0000−0001−6317−7012]

Institute of Computer Science
AGH University of Science and Technology
Al. Adama Mickiewicza 30, 30-059 Kraków, Poland
{aurbanczyk, olekb}@agh.edu.pl, kczech@student.agh.edu.pl

DOI:10.34658/9788366741928.60

Abstract. *A novel optimization algorithm inspired by socio-cognitive phenomena and based on flock architecture is presented along with promising preliminary experimental results.*
Keywords: *metaheuristics, global optimization, socio-cognitive computing*

1. Introduction

Tackling difficult optimization problems requires the use of metaheuristics [1]. Based on the famous No Free Lunch Theorem [2] there is always the possibility that a new algorithm will be better suited to find an optimal solution to hard computational problems. Talbi contends that hybridization and modification of current algorithms can be helpful in this regard [3]. Because metaheuristics are frequently inspired by nature, their hybridization frequently brings together different phenomena observed in the real world. Many metaheuristics that process a large number of individuals, particularly when individuals are perceived to be somewhat autonomous, use socio-cognitive inspirations, e.g. EMAS [4]. Among them there is a group of algorithms with dedicated mechanisms rooted in Social-Cognitive Theory by Albert Bandura [5], e.g. s-c PSO [6], s-c ACO [7] and s-c evolution strategies. We came to realize that by harnessing the power of metaphorical thinking[8], we can create novel, inventive mechanisms and operators that improve the functionality of traditional metaheuristics, not just for the sake of creation, but also to advance the field of computational intelligence. In our current work we decided to explore possibilities of using metaphor based on the theory of different, prominent social psychologist – Elliot Aronson [9]. His reward theory of attraction states that attraction is a form of social learning. According to Aronson, we can generally understand why people are attracted to each other by looking at the social costs and benefits. In summary, reward theory states that we prefer those who provide maximum rewards at the lowest possible cost. Social psychologists

have discovered four particularly powerful predictors of interpersonal attraction: proximity, similarity, self-disclosure, and physical attractiveness [10]. We use this inspiration to design a novel socio-cognitive algorithm described below and to perform pilot experiments in order to preliminarily verify its usefulness.

2. Socio-cognitive Flock-based algorithm

The algorithm is based on the concept of the Evolutionary Multi-Agent System with addition of socio-cognitive elements. A flock-based architecture extends the traditional sequential model into the parallel EA, providing an additional level of system organization [11]. The population of individuals is divided into flocks that are managed by agents. Several agents are created at the start of the algorithm. Each of the agents starts with a unique set of individuals in its initial population. Every cycle of the algorithm includes the evolutionary part and the socio-cognitive part. During the evolutionary part, every agent performs an evolutionary algorithm on his flock. The socio-cognitive part consists of a series of communication between two agents. During every communication, one of the agents is gaining information about part of the flock belonging to the other agent, and after a quality check of the acquired data, the agent assimilates part of its own flock to the individuals included in acquired information. The amount of information transferred between agents is determined by the trust between them. The concept of trust is implemented as a global token market where each of the agents starts with a certain amount of trust tokens, which can be passed by the agents during every event of single communication based on the outcome of this event. The more trust agents have, the more information they will acquire from other agents, and the better it will be. The assimilation of flocks is based on the use of simple and fast operators to reduce the distance between two individuals. The algorithm continues until it performs a given number of cycles, the best solution found by the agents is assumed as the solution found by the algorithm.

2.1. Experimental results

The preliminary results of the algorithm running on three standard 100- dimentional benchmark functions: Rastrigin, Ackley and Griewank are shown in Figures 1, 2 and 3, respectively. Each experiment was repeated 10 times and the results were averaged. Each benchmark was tested in 5-agent and 10-agent versions, with single-agent run as a reference. In both experimental settings, in each cycle of the algorithm, every agent does 50 iterations of the evolutionary algorithm, and then every agent attempts to communicate with others 2 times. The evaluation of the fitness function is performed after each such cycle. In the referential system, one agent is making the same amount of evolutionary algorithm iterations as in other experiments.

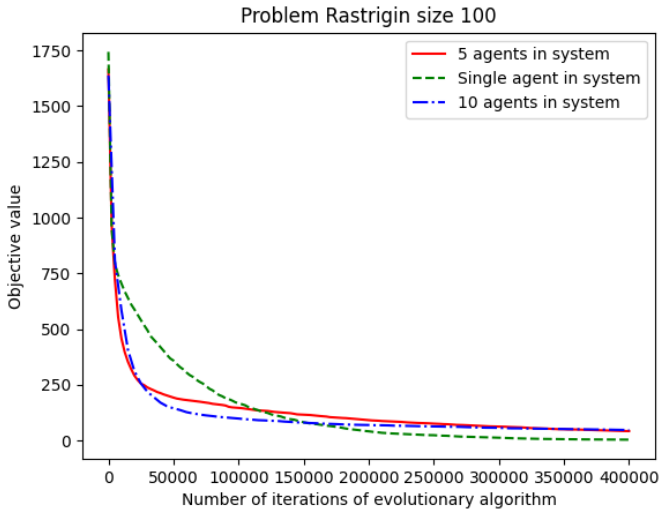


Figure 1. Preliminary results for Rastrigin benchmark. Source: own work.

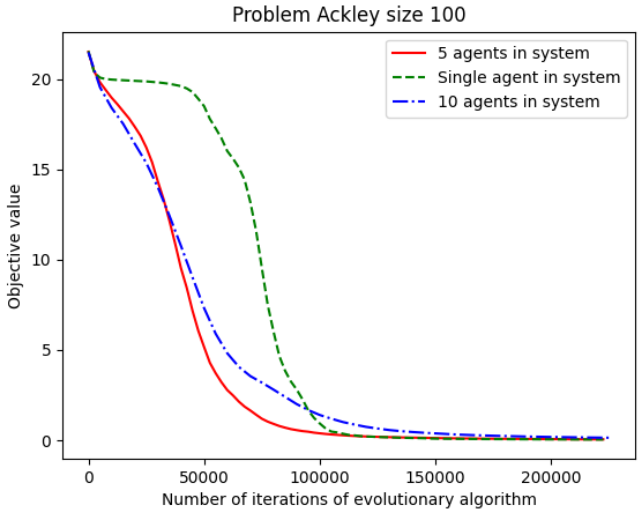


Figure 2. Preliminary results for Ackley benchmark. Source: own work.

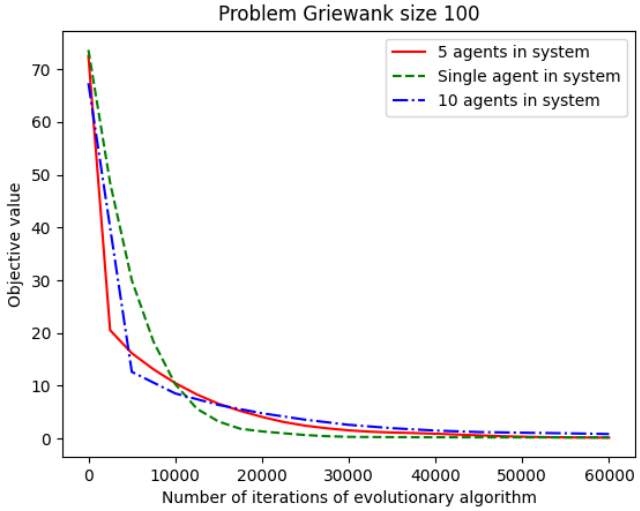


Figure 3. Preliminary results for Griewank benchmark. Source: own work.

3. Conclusions

In the pilot experimental run of the algorithm, promising results were obtained. Both 5-agent and 10-agent versions have found better solutions faster than a single-agent version; however, they were slightly outperformed in the final phase of the Restrigin benchmark. To conclude, communication between agents, based on socio-cognitive mechanisms, facilitates faster convergence in tested benchmark algorithms, but further experiments are needed. In addition to testing the algorithm against various, modern benchmarks, we intend to modify the flock architecture more extensively, by differentiating variation operators' settings among the agents, and compare our idea with an island version of evolutionary algorithm, which seems to be more appropriate.

Acknowledgment

The research presented in this paper has been financially supported by: Polish National Science Center Grant no. 2019/35/O/ST6/00570 "Socio-cognitive inspirations in classic metaheuristics."; Polish Ministry of Science and Higher Education funds assigned to AGH University of Science and Technology.

References

[1] Michalewicz Z., Fogel D., *How to Solve It: Modern Heuristics*, Springer Berlin Heidelberg, 2004, ISBN 9783540224945.

[2] Wolpert D., Macready W., *No free lunch theorems for optimization*, *IEEE Transactions on Evolutionary Computation*, 1997, vol. 1, no 1, pp. 67–82, doi: 10.1109/4235.585893.

[3] Talbi E.G., *Metaheuristics: from design to implementation*, John Wiley & Sons, 2009.

[4] Byrski A., Dreżewski R., Siwik L., Kisiel-Dorohinicki M., *Evolutionary multi-agent systems*, *The Knowledge Engineering Review*, 2015, vol. 30, no 2, pp. 171–186.

[5] Bandura A., *Social Foundations of Thought and Action: A Social Cognitive Theory*, Prentice-Hall series in social learning theory, Prentice-Hall, 1986, ISBN 9780138156145.

[6] Bugajski I., Listkiewicz P., Byrski A., Kisiel-Dorohinicki M., Korczynski W., Lenaerts T., Samson D., Indurkhya B., Nowé A., *Enhancing particle swarm optimization with socio-cognitive inspirations*, [In:] M. Connolly (ed.), *International Conference on Computational Science 2016, ICCS 2016*, *Procedia Computer Science*, vol. 80, Elsevier, pp. 804–813.

[7] Byrski A., Świderska E., Łasisz J., Kisiel-Dorohinicki M., Lenaerts T., Samson D., Indurkhya B., Nowé A., *Socio-cognitively inspired ant colony optimization*, *Journal of Computational Science*, 2017, vol. 21, pp. 397–406, ISSN 1877-7503, doi: https://doi.org/10.1016/j.jocs.2016.10.010.

[8] Lakoff G., Johnson M., *Metaphors we live by*, University of Chicago press, 2008.

[9] Aronson E., Aronson J., *The Social Animal*, Macmillan Learning, 2018, ISBN 9781464144189.

[10] Zimbardo P., Johnson R., McCann V., *Psychology: Core Concepts*, Always learning, Pearson, 2012, ISBN 9780205183463.

[11] Kisiel-Dorohinicki M., *Flock-based architecture for distributed evolutionary algorithms*, [In:] L. Rutkowski, J.H. Siekmann, R. Tadeusiewicz, L.A. Zadeh (eds.), *Artificial Intelligence and Soft Computing - ICAISC 2004*, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-540-24844-6, pp. 841–846.

Chapter 8

Robotics and Autonomous Systems

Domain Editors:

1. Piotr Skrzypczyński, Poznan University of Technology
2. Piotr Lipiński, Lodz University of Technology

A New Approach to Learning of 3D Characteristic Points for Vehicle Pose Estimation

Tomasz Nowak[0000−0002−2635−7732], **Piotr Skrzypczyński**[0000−0002−9843−2404]

Poznan University of Technology
Institute of Robotics and Machine Intelligence
ul. Piotrowo 3A, 60-965 Poznań, Poland
tomasz.nowak@doctorate.put.poznan.pl

DOI:10.34658/9788366741928.61

Abstract. *This article discusses the challenges of estimating the pose of a vehicle from monocular images in an uncontrolled environment. We propose a new neural network architecture that learns 3D characteristic points of vehicles from image crops and coordinates of 2D keypoints on images. To facilitate supervised training of this network, we pre-process the ApolloCar3D dataset to obtain labelled 3D characteristic points of different car models. We evaluate our approach on the ApolloCar3D benchmark and demonstrate results competitive to state-of-the-art methods.*
Keywords: *vehicle pose estimation, 3D scene understanding, deep learning*

1. Introduction

Estimating the pose of other objects is a fundamental task for an autonomous car, as it allows the car to determine its own pose relative to other objects in the environment, such as other vehicles and road infrastructure. Obtaining pose estimates from single camera images addresses a practical problem essential for making decisions about how to navigate the environment safely, without a need for expensive laser scanners. Unfortunately, the problem of recognizing 3D car instances and estimating their poses from monocular images in an uncontrolled environment is ill-posed and challenging [1]. In our recent paper [2], we demonstrated that a single camera can be positioned accurately with respect to a known infrastructure object using a deep neural network derived from the HRNet [3], previously developed for human pose estimation. In this paper, we demonstrate that a similar deep network architecture can estimate the pose of a vehicle in a traffic scenario. This scenario is challenging, because the model of the vehicle needs to be recognized among many prototypes, and the system has to deal with significant viewpoint changes and occlusions. To accomplish this task, a new neural network

architecture is proposed, that estimates 3D coordinates of characteristic points of 3D car models based on features learned from image crops and coordinates of 2D keypoints on images.

The pose estimation problem was decided to be solved using the Perspective-*n*-Point (PnP) algorithm, as in [2]. It requires known coordinates of keypoints in the image, 3D coordinates of corresponding points on the object model, and internal parameters of the camera. Estimating the position of 2D points in the image, solved in [2], is beyond the scope of this paper. In the experiments presented in this paper coordinates of 2D keypoints provided in [1] were used, to facilitate fair comparison. The task that we focused on in this paper is the estimation of 3D coordinates of points of various car models that differ in shape, unlike the precisely known model of electric charger in [2].

To facilitate supervised training of our network we pre-process automatically ApolloCar3D, a large dataset of 3D car instances, in order to obtain associations between the 3D coordinates of characteristic points of different car models and the annotated 2D keypoints in images, as these associations are not provided in the original dataset. Finally, we test our approach on the ApolloCar3D benchmark, demonstrating results competitive to state-of-the-art.

2. Dataset preparation

The ApolloCar3D dataset presented at [1] was used for the experiments. It consists of 5,277 traffic-derived images containing more than 60K vehicle instances. A set of 66 feature points was defined for each instance, and visible points were marked in images. In addition, a set of 34 CAD models of cars appearing in the dataset was provided and assigned to instances visible in the images. For each visible car, pose data relative to the camera coordinates was published. Unfortunately, the dataset creators did not provide a mapping between the 2D points in the image and the corresponding 3D points from the CAD model. To obtain such a mapping, which is required for supervised learning of 3D points detection, we use the procedure shown in Fig. 1. The first step is to transform the CAD model using the known translation and rotation. Next, the parameters of the ray containing all 3D points whose projections onto the image coincide with the point marked as the considered keypoint of the given vehicle are determined. For each CAD model face, we check whether it is intersected by this ray using the Möller-Trumbore algorithm [4]. Then, the coordinates of the intersection point are determined. If more than a single intersection point is found, the one closest to the camera is selected, since only not occluded points were marked in images by the annotators. Finally, the inverse of the ground truth rotation and translation is applied to the given intersection point to obtain the coordinates in the canonical pose of the car. The resulting point coordinates are then corrected to get a more accurate match.

To this end, optimization using the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm is performed on all instances of a given car type in the training set, which modifies the 3D coordinates to minimize the translation error. This error is defined as the square of the distance between the ground truth translation and the translation estimated by the EPnP [5] method using given camera parameters and 2D point coordinates in the image.

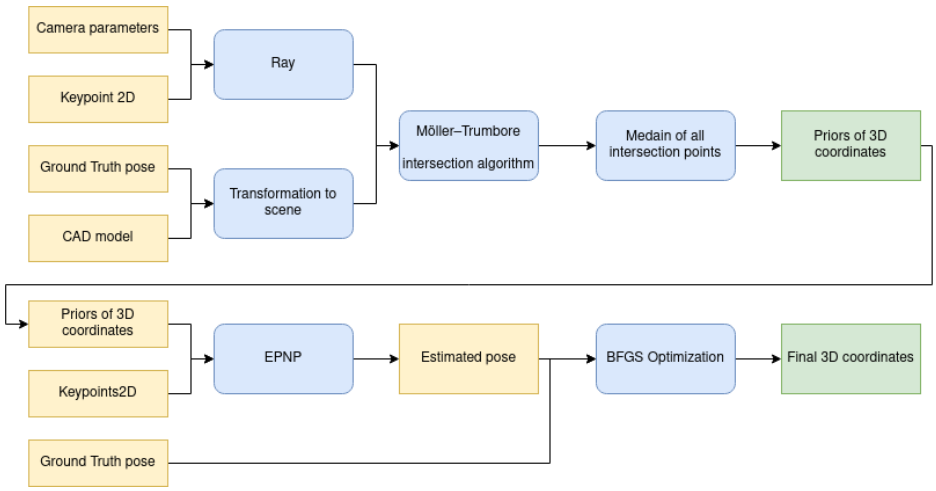


Figure 1. Pipeline for dataset pre-processing in order to obtain labelled 3D points for supervised learning. Source: own work.

3. Pose estimation system architecture

Estimation of the coordinates of 3D keypoints is carried out using a deep neural network, the diagram of which is shown in Fig. 2. It accepts as input an image crop containing the considered vehicle and the coordinates of the 2D keypoints visible in the image. Note that our method does not need a mask from pre-trained Mask R-CNN, which is used by the baselines from [1]. The image slice is processed using the HRNet backbone [3], which generates 48 feature maps of size $w \times h$ on output. The coordinates of 2D points in the image are normalized relative to the bounding box and processed by a Multilayer Perceptron (MLP) consisting of 7 layers. The output of this module is a vector of length $w \cdot h$, which is then transformed to a matrix of size $w \times h$ and concatenated with the feature maps from HRNet. In the model used during experiments w=64 and h=48. This set of feature maps is passed to the 3D keypoints estimation head that predicts two heatmaps per single point. The first heatmap corresponds to the point position on the X-Y plane and provides x and y coordinates. The second one predicts the point position on X-Z

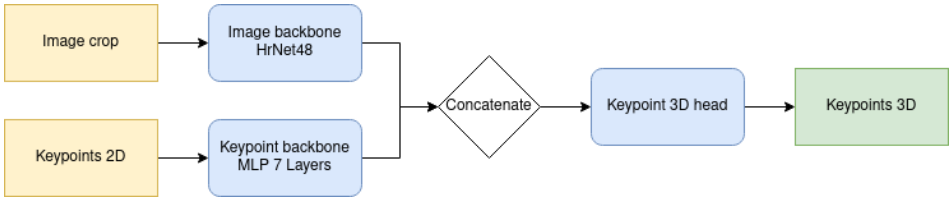


Figure 2. Architecture of keypoint 3D estimation network. Source: own work.

plane and provides z coordinate. The final pose is acquired by minimization of the reprojection error defined as in (1) using the BFGS algorithm.

$$\text{loss}_{\text{repr}} = \sum_{i=1}^{n} \left(\left\| \pi\left(\mathbf{T}, \mathbf{p}_i^{3d}, \mathbf{K}\right) - \mathbf{p}_i^{2d} \right\|_2 \right)^2, \tag{1}$$

where π is the projection function, $\mathbf{T}$ is a transformation to the given pose, $\mathbf{p}_i^{3d}$ are the 3D coordinates of the i-th characteristic point, $\mathbf{K}$ is the camera intrinsics matrix, and $\mathbf{p}_i^{2d}$ are the 2D coordinates of the i-th keypoint on image. A block scheme of the inference pipeline is presented on Fig. 3

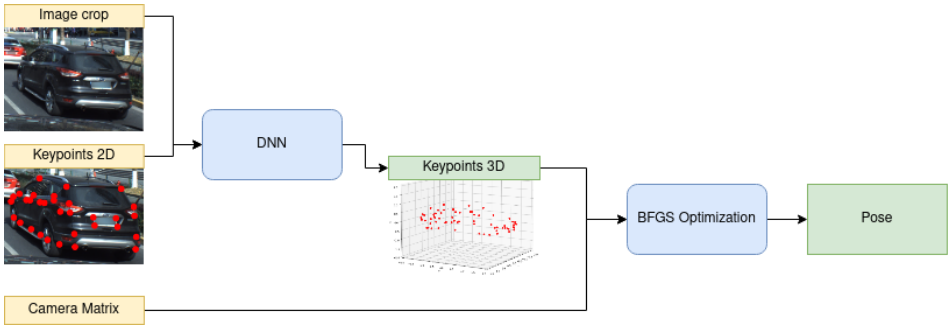


Figure 3. Inference pipeline for pose estimation. Source: own work.

4. Evaluation results

Model evaluation was carried out on the validation set of ApoloCar3D dataset containing 200 images. The A3DP metric presented in [1] was used, in its A3DP-Abs variant considering absolute distances to objects. It evaluates three elements: estimated car shape, position, and rotation. We do not consider for evaluation the car shape error also defined in [1], because we are not estimating the full grid of the car model here, being interested solely in pose estimation. The translation error metric is defined as:

$$c_{\text{trans}} = \left| \mathbf{t}_{gt} - \hat{\mathbf{t}} \right|_2 \leq \delta_t, \tag{2}$$

where $\mathbf{t}_{gt}$ denotes ground truth translation, $\hat{\mathbf{t}}$ denotes estimated translation and δ_t is an acceptance threshold. The rotation error metric is defined as:

$$c_{\text{rot}} = \arccos\left(\left|\mathbf{q}_{gt} \cdot \hat{\mathbf{q}}\right|\right) \leq \delta_{rot}, \tag{3}$$

where $\mathbf{q}_{gt}$ denotes ground truth rotation quaternion, $\hat{\mathbf{q}}$ denotes estimated rotation quaternion and δ_{rot} is an acceptance threshold. Inspired by metrics used in the COCO dataset, the authors of [1] proposed a set of metric thresholds from strict to loose. Thresholds for translation are set from 2.8 m to 0.1 m with a step of 0.3 m. Thresholds for rotation are set from $\pi/6$ to $\pi/60$ with the step of $\pi/60$. In addition to the "mean" metric that averages results for all thresholds, two metrics that use a single threshold were defined. The loose criterion $c-l$ uses $[2.8, \pi/6]$ and the strict criterion $c-s$ uses $[1.4, \pi/12]$ thresholds for translation and rotation respectively.

The results are presented in Tab. 1 and compared against the representative keypoint-less/direct (3D-RCNN [6]) and keypoint-based/indirect (DeepMANTA [7]) methods. Note that for the algorithms proposed in [6] and [7] the implementations provided in [1] as baselines were used for fair comparison.

Table 1. Comparison of results with baseline methods [1] on A3DP-Abs metrics

algorithm	mean	$c-l$	$c-s$
3D-RCNN [6]	16.4	29.7	19.8
DeepMANTA [7]	20.1	30.7	23.8
Ours	16.9	35.0	23.7

Our new method clearly outperforms the direct, keypoint-less approach, while it is on par with the DeepMANTA algorithm, having better results for the loose criterion ($c-l$), and slightly worse for the strict one ($c-s$). These results suggest that methods using 3D points perform better avoiding to regress the global depth or scale, as these tasks are problematic for a neural network. Our algorithm is applied to individual car instances, thus it does not apply the context-aware constraints enforcing the neighboring cars poses to be co-planar. These constraints provide around 1.5% improvement, as stated in [1], that surmounts the difference in the A3DP-Abs strict criterion between our method and the DeepMANTA baseline. On the other hand, our method detects correctly a higher number o vehicle instances while more relaxed thresholds are applied, which seems to be a more practical result for an autonomous car.

5. Conclusions

In conclusion, our study explored the application of a novel pose estimation method in the context of autonomous cars. We proposed a pipeline for automatic

annotation of 3D point positions in a large dataset. This annotation task done manually is time-consuming and requires advanced 3D labeling tools. Despite the lack of manual annotations of 3D points and with fewer assumptions (like pre-trained masks), we demonstrated competitive results compared to existing techniques.

References

[1] Song X., Wang P., Zhou D., Zhu R., Chenye G., Dai Y., Su H., li H., Yang R., *ApolloCar3D: A large 3D car instance understanding benchmark for autonomous driving*, [In:] *IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019*, pp. 5447–5457, doi: 10.1109/CVPR.2019.00560.

[2] Nowak T., Skrzypczyński P., *Geometry-aware keypoint network: Accurate prediction of point features in challenging scenario*, [In:] *17th Conference on Computer Science and Intelligence Systems, 2022*, pp. 191–200.

[3] Wang J., Sun K., Cheng T., Jiang B., Deng C., Zhao Y., Liu D., Mu Y., Tan M., Wang X., Liu W., Xiao B., *Deep high-resolution representation learning for visual recognition*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, vol. 43, pp. 3349–3364.

[4] Möller T., Trumbore B., *Fast, minimum storage ray-triangle intersection*, *Journal of Graphics Tools*, 1997, vol. 2, no 1, p. 21–28.

[5] Lepetit V., Moreno-Noguer F., Fua P., *EPnP: An accurate O(n) solution to the PnP problem*, *International Journal of Computer Vision*, 2009, vol. 81, doi: 10.1007/s11263-008-0152-6.

[6] Kundu A., Li Y., Rehg J.M., *3D-RCNN: Instance-level 3D object reconstruction via render-and-compare*, [In:] *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3559–3568.

[7] Chabot F., Chaouch M., Rabarisoa J., Teuliere C., Chateau T., *Deep MANTA: A coarse-to-fine many-task network for joint 2D and 3D vehicle analysis from monocular image*, [In:] *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2040–2049.

A Reinforcement Learning Framework for Motion Planning of Autonomous Vehicles

Mateusz Orłowski[0000−0002−5583−0197], **Paweł Skruch**[0000−0002−8290−8375]

AGH University of Science and Technology
Department of Automatic Control and Robotics
Adam Mickiewicz Avenue 30/B1, 30-059 Kraków, Poland
morl@agh.edu.pl, pawel.skruch@agh.edu.pl

DOI:10.34658/9788366741928.62

Abstract. *The paper introduces a framework that has been developed for the design and verification of motion planning algorithms for autonomous driving. The framework allows for the use of reinforcement learning for autonomous driving that requires complex and computationally intensive simulations. The key element in the presented approach plays a multi-agent closed-loop simulation of the traffic environment. Using the framework, the training process can be performed in parallel on high-performance computing clusters. Therefore, the framework provides an easy way to explore the potential of reinforcement learning for autonomous driving applications.*
Keywords: *reinforcement learning, autonomous vehicles, motion planning, framework*

1. Introduction

Reinforcement learning (RL) problems concern the derivation of an agent's policy, which tells what to do in a given situation to maximize a defined reward signal. Almost from the definition, those problems are closed-loop, as the agent's actions impact the environment state and, through that, the next input. Moreover, in contrast to many other machine learning methods, the training process is not directly told which actions to take, but instead has to discover on its own what actions will result in the highest rewards by trying them out. A key characteristic of RL is to treat the whole problem as a goal-oriented agent's interaction with an uncertain environment. This case appears in motion planning tasks for vehicles with a high level of automation, where agents have to operate in a closed loop despite serious uncertainty about the environment [1].

Because of the closed-loop nature of the problem, to successfully train a reinforcement learning agent the most common strategy is to use the simulation. When working on the decision-making part of the autonomous driving stack, the requirements for the tool shift from photo-realistic scene presentation and precise vehicle

dynamic simulation to realistic traffic motion and efficiency. As some simulations available on the market could be treated as a good base [2, 3, 4], at the time of conducting research, they lack some features including traffic intelligence, scalability, and efficiency which was required for our reinforcement learning use case. Because of that, we have decided to work on our own simulation environment.

The paper is organized as follows. After the introduction, goal-based behavior planning for autonomous vehicles is discussed. Next, a framework is presented that allows us to employ RL for the defined problem. The last chapter includes final conclusions.

2. Autonomous Driving Behavior Planning

Some of the autonomous driving functions and features for which RL might be used are motion planning, automatic parking, and context-aware decision-making processes. For motion planning, one of the challenges is the successful navigation from point A to point B. The system is most often provided with a navigation module, which based on localization information provides a list of instructions to follow the selected route. These instructions can be represented as lane-based goals and, in most cases, are associated with decision points such as splits or intersections. Those goals define on which lanes we should position ourselves in a given distance with additional information about the desired kind of maneuvers to be executed, in cases where multiple maneuvers can be performed from a given lane (like driving straight or taking a right turn). In the end, to follow the route, the car should perform these maneuvers to end up in the correct lanes at specific locations. A straightforward policy could be to follow such a route by performing accordingly left- and right-lane change maneuvers until arriving at the correct lane; however, it might not be sufficient in real-life scenarios. As an example, traffic rules in some countries indicate that cars should stay in the right lane if possible. Furthermore, lane navigation decisions must be made in the context of current traffic, which may not allow changing lanes at a given moment (Fig. 1) or may cause a situation in which leaving the desired lane from a route perspective is a reasonable option (Fig. 2). Deciding on such high-level actions (for example, in the form of lane change maneuvers) is called behavior planning.

As the driving setting is inherently closed-loop and involves many unobserved states, like the level of aggressiveness of a given driver or their own goals, the correct decision in a lot of cases is not straightforward. The optimal policy could change drastically even with small perturbations of the same scenario and is defined on the basis of the current objective, which should represent a balance between comfort, safety, and effectiveness. Due to the plurality of possible scenarios, the obvious tendency is to use data-driven methods, such as machine learning. The closed-loop property of the system suggest that the RL methodology as a good

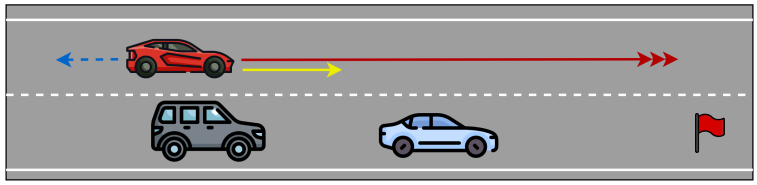


Figure 1. The ego (red car) goal is on the right lane and there are multiple strategies for how to get to that lane. In the most aggressive one (red), ego speeds up and tries squeezing in front of the blue car. The yellow strategy involves keeping speed and negotiating space with the grey vehicle. The most conservative option is blue, where the ego tries to drop behind the last car at the target lane. Source: own work.

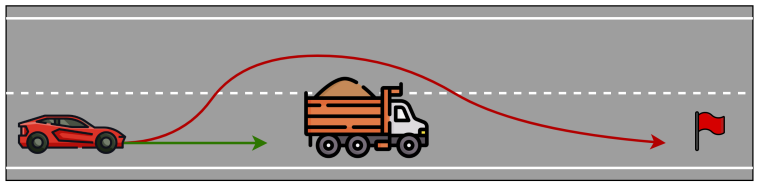


Figure 2. The ego car (red) is traveling in the correct lane but it is stuck behind a slow-moving truck. The safe option is to stay in the correct lane (green trajectory), while the more aggressive one is to overtake the truck (red trajectory). Source: own work.

framework to tackle this problem.

3. Framework Description

AiPilot is a scalable framework written in Python that allows us to employ RL to solve motion planning tasks for autonomous vehicles. The main element of this framework used for road traffic simulation and, therefore, the basic component of the definition of the environment, is TrafficAI [5]. It has been implemented by the Simteract company and adopted based on the requirements regarding the efficient RL training process, as well as execution in high-performance computing clusters.

TrafficAI is a multi-agent, closed-loop simulation of the traffic environment, both for the highway as well as urban scenarios. Regarding the static environment, the simulation includes a multiple-lane road simulation with features such as junctions, merges, or splits, along with traffic signs, traffic lights, or pedestrian crossings (Fig. 3). Such road structures may be created artificially or may be based on real-world map sources, such as OpenDrive or Open Street Map. Talking about the dynamic part, multiple road users of different types, such as cars or trucks,

may be simulated. Controlling of an agent may be realized by direct state setting, using kinematic model simulation, or with the use of a dynamic model of a car. Agent control may also be delegated to the simulation engine. In such a case, a parametrizable behavior model is used to decide on agents' actions. By such parameterization, different types of drivers, such as aggressive or rookie, can be defined. The tool also allows for defining a basic portfolio of simulated sensors that handle both field-of-view limitations and occlusions. By placing the sensors in specific locations in the car, we may represent the target car setup that we are trying to recreate, which later might be queried about objects they detected. The tool also allows for basic visualization, representing cars as bounding boxes, and drawing the road in the form of lines. The main advantage of TrafficAI is focusing on the motion planning aspect of autonomous driving systems, simulating perception at a high level, and presenting the idea of not demanding execution time and resource requirements.

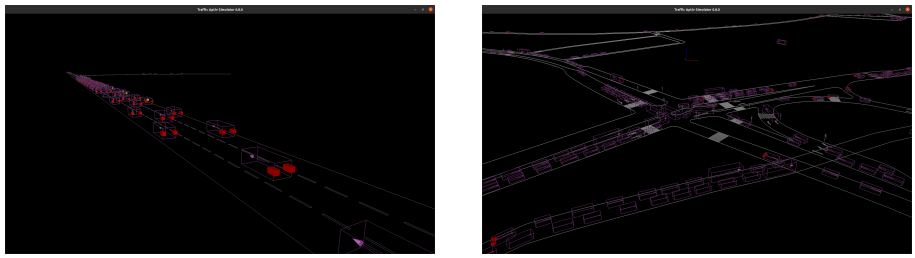


Figure 3. Visualized TrafficAI simulation. Road structures are presented in white; cars in the form of bounding boxes are presented in magenta. The left image presents a highway scenario, while the right one urban use case with high-density traffic, junction, and overpass. Source: own work.

The simulation engine works as a foundation for the definition of the reinforcement learning environment. Fig. 4 illustrates the building blocks and the scheme of the agent (policy) interaction with the designed environment. In each timestamp, based on the ego car's perception systems forming agents' observation, the policy decides on an action to execute, which is derived by running inference through a neural network. Action is defined as an acceleration command, a maneuver to execute, or an analogous control signal. This action is further interpreted and parsed by trajectory generation and control blocks. Next, with low-level control defined for the ego, behavior is executed within the traffic, which is simulated or the real one. Then, the ego car's perception systems are queried again, resulting in a new state (observation) for the next time instance. Along with this new observation, the policy decision, and in general environment state, is evaluated and summarized in the form of a reward signal, whose value depends on such qualities as achieved speed, smoothness, and safety.

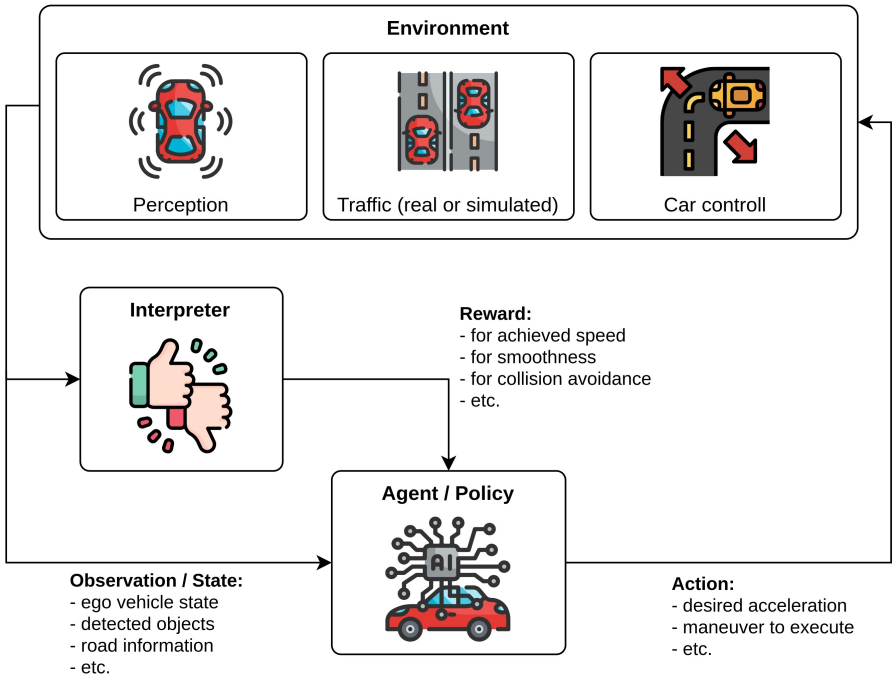


Figure 4. Model of the RL framework and agent interaction for motion planning of autonomous vehicles. Source: own work.

4. Conclusions

In the paper, an efficient and effective RL-based framework for motion planning tasks is presented. The framework is implemented to cope with computationally expensive environments and can retain the parallel efficiency of the external high-performance computing solver. Our main motivation for working on reinforcement learning applications for motion planning of highly automated vehicles is to close the gap between research and industry application. Developing RL planning algorithms with safety guarantees which improve efficiency and present human-like behaviors is our desired end goal, and some preliminary work in that direction might be found in [6].

References

[1] Aradi S., *Survey of deep reinforcement learning for motion planning of autonomous vehicles*, *CoRR*, 2020, doi: 10.48550/arXiv.2001.11231.

[2] Dosovitskiy A., Ros G., Codevilla F., López A.M., Koltun V., *Carla: An open urban driving simulator*, [In:] *Conference on Robot Learning*.

[3] Lopez P.A., Behrisch M., Bieker-Walz L., Erdmann J., Flötteröd Y.P., Hilbrich R., Lücken L., Rummel J., Wagner P., Wiessner E., *Microscopic traffic simulation using sumo*, [In:] *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2575–2582, doi: 10.1109/ITSC.2018.8569938.

[4] Shah S., Dey D., Lovett C., Kapoor A., *Airsim: High-fidelity visual and physical simulation for autonomous vehicles*, *CoRR*, 2017, doi: 10.48550/arXiv.1705.05065.

[5] Simteract, *Traffic AI™ – Simteract*, (access: 14-07-2023). https://simteract.com/projects/traffic-ai/

[6] Orłowski M., Wrona T., Pankiewicz N., Turlej W., *Safe and goal-based highway maneuver planning with reinforcement learning*, [In:] A. Bartoszewicz, J. Kabziński, J. Kacprzyk (eds.), *Advanced, Contemporary Control*, Springer International Publishing, Cham, ISBN 978-3-030-50936-1, pp. 1261–1274.

BDOT10k-seg: A Dataset for Semantic Segmentation

Aleksandra Kos[1,2][0000−0001−9726−4472], **Karol Majek**[2][0000−0002−1351−8496]

[1]*Poznan University of Technology*
60-965 Poznań, Poland
aleksandra.kos@doctorate.put.poznan.pl
[2]*Cufix*
05-825 Grodzisk Mazowiecki, Poland
karolmajek@cufix.pl

DOI:10.34658/9788366741928.63

Abstract. *In this work, we describe BDOT10k-seg, a novel aerial dataset for semantic and instance segmentation. Our data covers almost the entire territory of Poland (314,000 km^2) and provides precise pixel-level annotations for 286 classes of topographical objects, including buildings, roads, rivers, lakes, airports, agricultural areas, and forests. BDOT10-seg consists of 60,718 images with a resolution of 3 to 75 centimeters per pixel, and more than 40 million object instances. The average image size is 12,367 px because, unlike other publicly available datasets, we do not modify the source orthoimages. The code for generating the BDOT10k-seg dataset is publicly available*[1].

Keywords: *BDOT10k, remote sensing, aerial images, semantic segmentation, instance segmentation*

1. Introduction

Remote sensing images consist mainly of data collected by satellites and unmanned aerial vehicles (UAVs) and are significantly different from standard datasets such as COCO [1]. The low signal-to-noise ratio caused by massive and complex backgrounds, the large variations in object densities and sizes, and non-trivial orientations make segmentation and detection in aerial images particularly challenging. For these reasons, despite the significant progress that has been made in generic computer vision in recent years, aerial-based computer vision is still an unsolved problem. In this paper, we introduce BDOT10k-seg – a new remote sensing dataset dedicated to semantic and instance segmentation. The images in BDOT10k-seg cover almost 314,000 km^2, and the dataset was created based on

[1]https://github.com/deepdrivepl/bdot10kseg

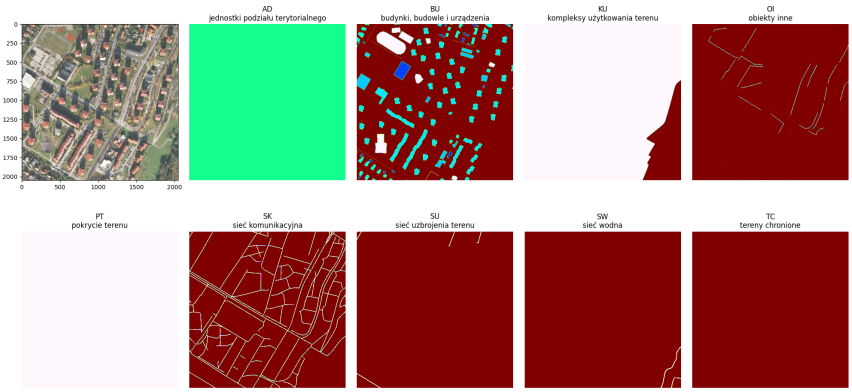


Figure 1. Sample data in the BDOT10k dataset. We show semantic segmentation labels for 9 base classes. Source: own work.

the data published by the Polish Head Office of Geodesy and Cartography – orthoimages and the Database of Topographic Objects (BDOT10k). The main contributions of this work are: BDOT10k-seg, a new challenging aerial dataset with precise pixel-level annotations for 9 base classes, 57 intermediate classes, and 286 3rd-level classes, as well as an in-depth analysis of our dataset and comparison with other publicly available benchmarks. A data sample is shown in Fig. 1.

2. Related Work

Satellite imagery is widely used in tasks such as agricultural and urban planning, and environmental monitoring. Over the last years, many datasets related to object detection [2, 3, 4, 5] and segmentation [5, 6, 7] in aerial images have been released. Most of them focus on general multi-class detection, but [3, 7] also publish geospatial data in addition to the standard labels. The images in most remote sensing datasets have substantial dimensions, and there are often large variations in object sizes. Therefore, methods such as AF-SSD [8] and SCRDet [9] process these images by dividing them into uniform tiles, and use the attention mechanism as well as multi-level feature fusion.

3. BDOT10k Dataset

The BDOT10k-seg dataset covers an area of 313,976 km^2 and contains 60,718 high-resolution orthoimages. Resolutions range from 3 to 75 cm per pixel, with the largest number of images at 25 cm per pixel, as shown in Fig. 2. Our dataset is characterized by a large variability of image size – the largest photo in the dataset

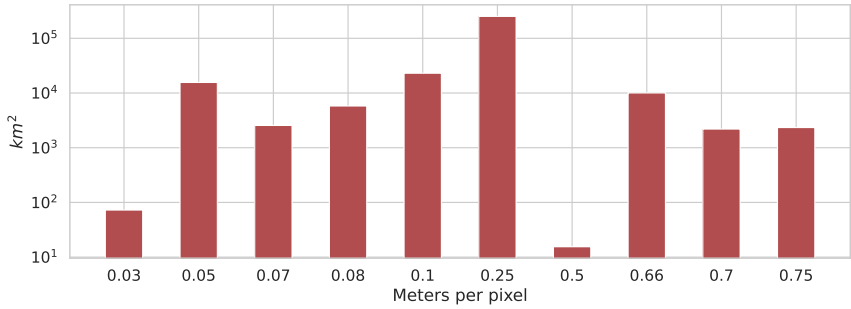


Figure 2. The number of square kilometers at different image resolutions. Source: own work.

has almost 50,000 px, while the smallest has less than 5,000 px. We provide pixel-level annotations for two tasks, semantic and instance segmentation. We created ground-truth labels based on the BDOT10k database published by the Head Office of Geodesy and Cartography and, similarly to BDOT10k, we distinguish 9 base classes, 57 intermediate classes, and 286 precise classes. A complete list of available categories can be found in the source code.

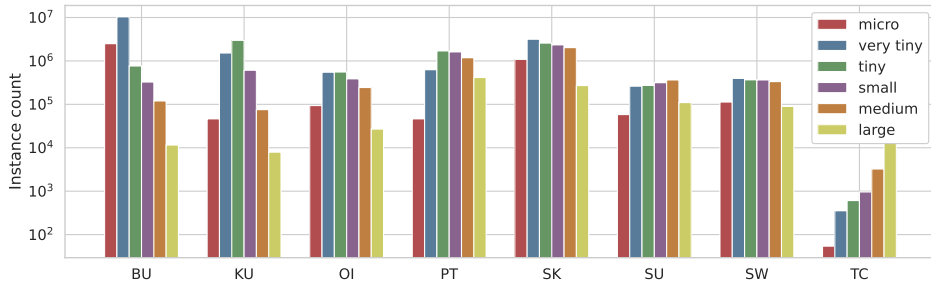


Figure 3. Distribution of object sizes based on relative size thresholds. Size: geometric mean of height and width. BU – buildings, KU – land use, OI – other objects, PT – land cover, SK – communication network, SU – lines and pipes, SW – river systems, TC – protected areas; AD – territorial division is ommited. Source: own work.

Objects in our dataset come in a wide range of sizes, which makes BDOT10k-seg challenging and suitable for small and multi-scale instance segmentation. Fig. 3 shows the number of instances in each of 6 predefined sizes (*micro*, *very tiny*, *tiny*, *small*, *medium*, and *large*) for 9 base classes. The images in our dataset have very high resolution, so we use relative threshold values as in [10]. Since we also compare BDOT10k-seg with object detection datasets, bounding boxes, not segmentation masks, were used to calculate object sizes for a fair comparison.

Table 1. Comparison of the main features of selected remote sensing datasets. S_{image} and S_{object} are the average image and object sizes, respectively. The size is defined as the geometric mean of width and height of a box. AABB: Axis-Aligned Bounding Box, OBB: Oriented Bounding Box, IS: Instance Segmentation, SS: Semantic Segmentation.

dataset	labels	images	objects	classes	S_{image}	S_{object}
DOTAv2 [2]	AABB, OBB	2,422	349,675	18	3756±3536	33.0±49.0
xView [3]	AABB	846	601,806	62	3148±317	34.9±39.9
SODA-A [4]	AABB	2,512	872,613	10	3627±162	15.6±7.7
iSAID [6]	SS,IS	1,869	471,438	15	3165±1606	22.7±35.2
DIOR [11]	AABB	23,463	192,518	20	800±0	65.7±91.7
VHR-10 [5]	AABB, IS	650	3,896	10	813±137	74.5±46.8
SpaceNet2 [7]	IS	8,519	217,360	1	650±0	53.0±42.2
BDOT10k	**SS, IS**	**60,718**	**41,159,701**	**286**	**12367±8939**	**442.1±1022.7**

As shown in Tab. 1, our dataset contains significantly more images and objects than other datasets, with 60,718 images and over 40 million instances. The average image size (S_{image}), as well as the average object size (S_{object}), show the greatest variability, which makes BDOT10k particularly challenging. However, by filtering out some of the classes, it is possible to obtain a subset that is well-suited for small and tiny instance segmentation.

Table 2. Validation Jaccard Index values for 7 base classes (without AD and PT)

Model name	BU IoU	KU IoU	OI IoU	SK IoU	SU IoU	SW IoU	TC IoU
UNet R50	43.26	45.23	2.36	99.04	9.16	5.38	6.10
UNet EffNet B3	41.21	45.86	2.11	79.15	0.07	12.59	38.4

We report training results using UNet architecture with backbones ResNet50 (*batchsize* = 24) and EfficientNet-B3 (*batchsize* = 16) as shown in Tab. 2. Training set: 19438 images (1024px) sampled from 490 original images from the dataset. Validation set: 2985 images sampled from 55 images – without overlap with the training set. We used Adam optimizer, One Cycle Learning Rate schedule (startLR $3.33e^{-6}$, maxLR $1e^{-4}$@10%, finalLR $1.11e^{-6}$), Dice Loss, Jaccard index as a metric, 1024px images, training time augmentation only (horizontal flip, scale, shift, Gauss noise, CLAHE, brightness, contrast, gamma, sharpen, blur, motion blur, HSV). The model was trained on a single RTX 8000 GPU using FP16 half precision. Experiments are reproducible using the published code.

4. Conclusions and Future Work

In this article, we introduced a new remote sensing dataset – BDOT10k-seg. We provide labels for two computer vision tasks (semantic segmentation and instance segmentation), which we developed on the basis of the Topographical Objects Database provided by the Polish Head Office of Geodesy and Cartography. We conducted a thorough analysis of the dataset at the image and object levels. We report training results for UNet with two different backbones. We plan to further expand the dataset by adding annotations for other computer vision tasks, such as axis-aligned and oriented object detection. We also intend to use BDOT10k to develop methods for analyzing images containing tiny objects, such as two-stage tiny object detection.

Acknowledgment

The research was supported by the Ministry of Education and Science as part of the "Doktorat Wdrożeniowy" program (DWD/5/0203/2021).

References

[1] Lin T.Y., Maire M., Belongie S., Hays J., Perona P., Ramanan D., Dollár P., Zitnick C.L., *Microsoft coco: Common objects in context*, [In:] *European conference on computer vision*, Springer, pp. 740–755.

[2] Ding J., Xue N., Xia G.S., Bai X., Yang W., Yang M.Y., Belongie S., Luo J., Datcu M., Pelillo M., et al., *Object detection in aerial images: A large-scale benchmark and challenges*, *IEEE transactions on pattern analysis and machine intelligence*, 2021, vol. 44, no 11, pp. 7778–7796.

[3] Lam D., Kuzma R., McGee K., Dooley S., Laielli M., Klaric M., Bulatov Y., McCord B., *xview: Objects in context in overhead imagery*, *arXiv preprint arXiv:1802.07856*, 2018.

[4] Cheng G., Yuan X., Yao X., Yan K., Zeng Q., Han J., *Towards large-scale small object detection: Survey and benchmarks*, *arXiv preprint arXiv:2207.14096*, 2022.

[5] Cheng G., Han J., Zhou P., Guo L., *Multi-class geospatial object detection and geographic image classification based on collection of part detectors*, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2014, vol. 98, pp. 119–132.

[6] Waqas Zamir S., Arora A., Gupta A., Khan S., Sun G., Shahbaz Khan F., Zhu F., Shao L., Xia G.S., Bai X., *isaid: A large-scale dataset for instance segmentation in aerial images*, [In:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019*, pp. 28–37.

[7] Van Etten A., Lindenbaum D., Bacastow T.M., *Spacenet: A remote sensing dataset and challenge series*, *arXiv preprint arXiv:1807.01232*, 2018.

[8] Lu X., Ji J., Xing Z., Miao Q., *Attention and feature fusion ssd for remote sensing object detection*, *IEEE Transactions on Instrumentation and Measurement*, 2021, vol. 70, pp. 1–9.

[9] Yang X., Yang J., Yan J., Zhang Y., Zhang T., Guo Z., Sun X., Fu K., *Scrdet: Towards more robust detection for small, cluttered and rotated objects*, [In:] *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8232–8241.

[10] Kos A., Majek K., Belter D., *Where to look for tiny objects? ROI prediction for tiny object detection in high resolution images*, [In:] *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, IEEE, pp. 721–726.

[11] Li K., Wan G., Cheng G., Meng L., Han J., *Object detection in optical remote sensing images: A survey and a new benchmark*, *ISPRS journal of photogrammetry and remote sensing*, 2020, vol. 159, pp. 296–307.

Beacon-based Swarm Search and Rescue

Sunil Ratnayake[1][0009−0009−2765−637X], **Maksym Figat**[2][0000−0002−1898−0540]

Warsaw University of Technology
[1]*The Institute of Computer Science*
[2]*Institute of Control and Computation Engineering*
Nowowiejska 15/19 00-665 Warsaw, Poland
sunil.ratnayake.stud@pw.edu.pl, maksym.figat@pw.edu.pl

DOI:10.34658/9788366741928.64

1. Introduction

Searching and rescuing missing people is extremely challenging, especially underground, as seen in the extreme Tham Luang cave rescue, where the trapped boys were in a flooded cave up to 4km from the entrance. Rescuing the boys required a spectacular deployment of human and equipment resources, with time – a key resource critical to the success of the operation – running out. The task facing the rescuers was extremely difficult and dangerous. If it had been possible to develop a robotic system consisting of a swarm of robots [1] able to withstand difficult conditions (water, lack of communication, other external factors), the rescue operation could have taken much less time (than 18 days) and, above all, the risk of loss of life for the rescuers could have been minimised.

In 2021, DARPA proposed the Subterranean Challenge [2], where participants were tasked with solving a search problem in dangerous and hard-to-reach areas. In the robotic systems developed [3], participants had three types of robots (wheeled, walking (e.g. Spot), and flying) with complex designs and high costs. Developing a swarm of robots from these would be very expensive. It is better to equip the robotic system with two types of robots: 1) a large number of low-cost mobile robots, and 2) a smaller number of expensive specialised robots. The former would be tasked with locating the searched person, and the latter, based on the local information gathered by the former, would build a global map, manage the rescue operation and ultimately perform the rescue.

The number of robots used, and therefore the unit price of the robot, is extremely important, as can be seen in the example of the Sniffy Bug [4] project, where a swarm of low-cost robots was used to find a gas leak. It is better to design a system of 10 simple robots and a single specialised robot than 2 specialised robots. In the first case, the 10 robots will quickly cover the area and then send

the specialised robot to the target. In the second case, it will take much longer to complete the task. And time is a resource.

2. Proposed solution

We propose to develop a robot swarm that aims to find and transport an object placed in a dynamically changing environment. The swarm consists of heterogenous robots with a dynamically assigned role: scout or beacon. The scout's task is to find the object, while the beacon's task is to construct a local model of the environment and to navigate the robots with the scout role.

In the initial phase, the scout moves through the environment while creating a local representation, i.e. it creates a weighted graph in which the nodes are intersections and the weights associated with the edges denote the distance between intersections. When it encounters an intersection (at a suitable distance from the beacon) where no robot is present, the scout becomes the beacon and then manages the local movement of the following robots. When two robots meet (scout-scout, scout-beacon), the robots update their knowledge of the environment – scout stores a much smaller graph representation than beacon (the size of the graph created is a parameter for both types of robots). When scout encounters a beacon, the beacon directs scout in a direction that increases the swarm knowledge of the environment.

The role of beacons is very important. Beacons act as signposts. A scout moving between beacons does not need to have equivalent knowledge of the environment. All it needs is the information from the beacon in which direction to move and any knowledge to update the graph (e.g. if the environment has changed). If no other robot appears at the beacon after a certain amount of time, the beacon switches to scout mode and moves to the next known beacon. This allows the whole swarm to move around the extended environment.

3. Conclusions

The proposed approach is based on a swarm of robots moving in a dynamically changing environment. We use homogeneous robots with dynamically assigned roles that form graphs representing the local state of the environment. The interacting robots update the state of the environment and the beacons guide the local movement of the swarm to cover the searched environment space in the fastest way. The system will be developed using a methodology based on the agent-based approach presented in [5].

References

[1] Brambilla M., Ferrante E., Birattari M., Dorigo M., *Swarm robotics: A review from the swarm engineering perspective*, *Swarm Intelligence*, 2013, vol. 7, pp. 1–41.

[2] DARPA/TTO, *DARPA Subterranean Challenge*, (access: 14-07-2023). `https://www.subtchallenge.com/`

[3] Chang Y., Ebadi K., Denniston C.E., Ginting M.F., Rosinol A., Reinke A., Palieri M., Shi J., Chatterjee A., Morrell B., Agha-mohammadi A.a., Carlone L., *Lamp 2.0: A robust multi-robot slam system for operation in challenging large-scale underground environments*, *IEEE Robotics and Automation Letters*, 2022, vol. 7, no 4, pp. 9175–9182.

[4] Duisterhof B.P., Li S., Burgués J., Reddi V.J., de Croon G.C.H.E., *Sniffy bug: A fully autonomous swarm of gas-seeking nano quadcopters in cluttered environments*, 2021.

[5] Figat M., Zieliński C., *Parameterised robotic system meta-model expressed by hierarchical petri nets*, *Robotics and Autonomous Systems*, 2022.

Intelligent Anticipatory Mobile Robot Networks for Autonomous Fruit Harvesting

Andrzej M. J. Skulimowski[1,2][0000−0003−0646−2858],
Masoud Karimi[1,2][0000−0002−0770−1796]

[1]*AGH University of Science and Technology,*
Decision Science Laboratory
al. Adama Mickiewicza 30, 30-059 Kraków, Poland
[2]*Progress and Business Foundation*
International Centre for Decision Sciences and Forecasting
ul. Juliusza Lea 12B, 30-048 Kraków, Poland
{ams, karimi}@agh.edu.pl

DOI:10.34658/9788366741928.65

Abstract. *A relevant class of decision problems solved by autonomous robots consists in deriving consensus strategies for coordinated group task performance. This paper presents preference models for the above consensus problems termed anticipatory networks (AN). By definition, an AN is a multidigraph where temporally ordered agents are linked by a causal relation. Another partial order relation is anticipatory feedback which expresses preferences regarding some future decisions. We will present an application of the above model to coordinating fruit harvesting by autonomous robot teams. A graded freedom of the decision choice allows the robots to achieve desirable efficiency of the harvesting process.*
Keywords: *autonomous robots, intelligent decision agents, coordination, multicriteria decision theory, anticipatory networks*

1. Introduction

Coordination and cooperation are two important concepts in multi-robot systems (MRS), and while they are related, they considerably differ in the way they are dealt with when planning robot team operations. Coordination refers to the process of managing interactions between multiple robots in order to achieve a common goal. This can involve tasks such as task allocation, communication, and formation path planning. In coordination, the performance of individual tasks of each robot depends on the decisions of an agent termed coordinator which ensure the achievement of a common objective of the MRS. The emphasis in coordination is on managing the interactions between the robots to achieve the desired outcome by limiting the degree of autonomy of all or some robots. The team objective in the

coordination process may be conflicting with the individual criteria of each robot. Cooperation, on the other hand, refers to the process where robots work together towards a common goal which is recognized as own goals by each robot as well. In cooperative MRS, each robot has a specific role to play, and the actions of each robot are both independent – as selected without any commands issued by an external agent – and interdependent, because the robots take into account the other agents' actions. The emphasis in cooperation is on reaching a self-organization status ensuring the alignment of each robot's actions with the overall objective of the MRS. Any conflicts between the common and individual goals are solved by robots themselves without an external intervention.

Both coordination and cooperation are essential for effective MRS, specifically for the teams of autonomous robots harvesting fruits. Covered crops are working environments where coordination is possible and ensures higher yields compared to spontaneous task allocation by the same robot teams. The reason for that is the coordinator can take into account the information about fruit distribution in the entire plantation and optimize the allocation of tasks, while individual robots know only their immediate neighborhood and places visited before. This advantage of coordination vanishes when communication fallouts or other difficulties with acquiring and exchanging information disturb the transfer of commands from the coordinator to fruit picking robots. As a remedy, we apply anticipation, which replaces the knowledge of working conditions and robot actions by their expected future values derived from robot decision algorithms and the assumption of their rationality. Specifically, the harvesting problem is formulated as a multi-level multicriteria optimization problem embedded in an anticipatory network (AN) [1].

In this article we show how the theory of anticipatory robotic decisions can be applied to formulate new principles of cooperation and compromise decision selection by autonomous harvesting robots. Section 2 presents the related work on robot coordination and the basic notions of the anticipatory network theory. Then, in Section 3 we will present an application of the above model to establish efficient collaboration of teams consisting of autonomous fruit harvesting robots and human supervisors. We will point out the relation of this real-life problem to the theory of cooperative systems, robot preferences, anticipatory robotics, and discrete-event systems control. In the same section we will briefly outline the software architecture that has been implemented within an ongoing research project aimed at construction and deployment of teamed intelligent and fully autonomous anticipatory strawberry picking robots. First, a nested simulation application has been used to model the anticipatory behavior of an autonomous fruit harvesting robot team. This application was then adopted to robot coordination and control, which made possible finding an optimal formation evolution strategy for the team of four robots. In the concluding Section 4 we will outline the development prospects of the theory of anticipatory robotic decisions and its further applications.

2. Related research: coordination, anticipation, autonomy

Coordination problems for MRS have been studies by numerous authors, with a rapid growth of the number of published papers over last two decades, cf. [2, 3]. Various definitions and approaches used by different authors can be summed up as follows: coordination is focused on managing the interactions between robots with limiting their degree of autonomy, while cooperation aligns robot activities towards a common goal in a way that preserves their decision freedom. Coordination involves explicit protocols and rules related to a diversified spectrum of agent actions, although most researchers focus either on coordinating mobility [4], including the multiple agent path planning (MAPP) [5] and robot formation control [6] or on the multiple robot task allocation (MRTA) [7, 8]. Coordination of multiple robots refers to a decision-making process that directly depends on robot characteristics and the infrastructure of the working environment. The coordinator aims to achieve the common goal of the team that cannot be reached by each single robot in isolation or at least the achievement of this goal would not be optimal. In general, except the common goal, robots may specify and strive to reach individual targets or tasks [9]. Technically, coordination and cooperation algorithms are often combined with reinforcement learning enhanced with deep neural networks [10].

2.1. Anticipatory networks

The anticipatory coordination model assumes that robots are endowed with autonomous decision-making capacity and the knowledge of other robot decision algorithms. Moreover, the preferences of autonomous anticipatory decision agents modeled as AN nodes fulfill the following two conditions, namely:

- some temporally ordered agents are linked by an acyclic causal relation r
- when making their decisions, some agents take into account anticipated solutions of certain future problems solved by other agents; these are indicated by another acyclic relation termed anticipatory feedback (AF).

Both relations are mutually weakly asymmetric, i.e. for any two nodes A and B in an AN it holds

$$A(AF)B \Rightarrow BRA, \tag{1}$$

where R is the transitive closure of r. The above assumption ensures that the preferences regarding future decisions of an agent A can only be expressed by an agent B when B can actually influence the decisions of A by the relation r directly or indirectly by its superposition. The combination of possible influences and preferences regarding future decisions within an MRS allows the robot team developer to construct a multidigraph of decision problems linked causally and by

AF relations. Such multidigraphs are termed *anticipatory networks* [1], see the formal definition (see Def. 1) below.

ANs as models of robot formations share some common rules with the graph-based formation control [11]. On the other hand, the ANs generalize earlier models of consequence anticipation in multicriteria decision problem solving [12] and extend the theory of anticipatory systems of Rosen [13] towards applications in autonomous robotics. Specifically, cooperating robot formations can be modeled as evolving (timed) anticipatory networks driven by a discrete event system with a virtual supervisor [14]. Each agent A modeled as an AN node selects an action from the set of feasible decisions U taking into account multiple optimization criteria $F = (F_1, \cdots, F_N)$, its own individual preference structure P, and a set of further preference relations $\mathrm{AF}_1, \cdots, \mathrm{AF}_k$ regarding actions to be selected by some agents in the future. The set of nondominated decisions with respect to F is defined as

$$\Pi(U, F, P) := \{u \in U : \forall v \in U \; [F(v) \leq_P F(u) \Rightarrow F(v) = F(u)]\} \tag{2}$$

where the preference relation $\leq_P$ is a partial order in $F(U)$ and fulfills the condition $F(x) \leq F(y) \Rightarrow F(x) \leq_P F(y)$. The values of F in the latter inequality are compared first with respect to the natural coordinatewise partial order in $\mathbb{R}^N$.

Now, let R_i and R_j be agents modeled as AN nodes that solve the multicriteria problems (U_k, F_k, P_k), for $k = i, j$. The causal relation modeling the agent's R_i influence on future decisions of another agent R_j is defined by the multifunction

$$\varphi_{i,j} : U_i \to\to U_j, \tag{3}$$

which indicates the new choices allowed for R_j. Furthermore, the *anticipatory feedback* relation AF between the nodes R_j and R_i is defined by pointing out which future decisions of R_j are desirable by R_i. Namely, agent R_i specifies the set $V_{j,i} \subset U_j$ that contains such desirable decisions. With the above notions we can provide now a formal definition of an anticipatory network of autonomous agents.

Definition 1. *Let Q be a certain set of autonomous agents. An anticipatory agent network is a multidigraph with nodes corresponding to agents and their decision problems (one problem per node) linked by a connected acyclic causal influence relation r defined by non-empty-valued multifunctions* (3) *in the following way:*

$$\forall R_i, R_j \in Q : \; [R_i r R_j \Leftrightarrow \exists\, \varphi_{i,j} : U_i \to\to U_j]. \tag{4}$$

Moreover, there is a non-trivial anticipatory feedback relation AF in Q such that for at least one pair of nodes $R_i, R_j \in Q$ it specifies the subset $V_{j,i} \subset U_j$ of decisions of R_j desired from the point of view of R_i and such that the condition (1) *is satisfied.*

2.2. A classification of autonomous decision agents

Following the definitions provided in [15], below we will present the decision autonomy which allows us to quantify the decision freedom lost by agents in the anticipatory coordination process compared to the decisions made when cooperating without external constraints. The autonomy levels are classified by assigning the agent to one of four classes of freedom of choice (FOC) or artificial decision creativity which are defined within the conceptual apparatus of multicriteria decision theory [15] as follows:

- 1st order FOC: agents capable of freely choosing decisions from a given set of alternatives,
- 2nd order FOC: agents endowed with the FOC of 1st order that can independently expand the scope of their decisions (remove or release constraints),
- 3rd order FOC: agents endowed with the FOC of 2nd order capable of independently changing the purpose of their action, the goals to be reached, or any other objectives,

The 4th order FOC is the artificial decision creativity. the class which has not yet been applied in MRS coordination, but will be a subject of future research: Another taxonomy comprising four levels of autonomy was proposed in [9]. This paper lists the centralized, negotiation, agreement, and emergent autonomy, while the decision freedom ranges from externally imposed actions (lowest autonomy) to fully self-determined actions (highest autonomy).

3. Anticipatory robot teams for strawberry harvesting

This section presents a real-life case, where the above theory has been applied to design coordination rules for a team of autonomous fruit harvesting robots. Specifically, a team of n robots is harvesting strawberries in a polytunnel with *m* gutters. The goal of coordination is to replace a human coordinator for each of the robots by autonomous coordination performed by one of them. Then, an employee supervises the entire team instead of the separate performance of single robots. The virtual supervision and coordination explores the above concepts with AN formations modelling the team work. The evolution of the formation is driven by a discrete event system, where agent's state defines this agent's position in the next AN formation. Moreover, robots can access a common knowledge base which allows them to optimize harvest by moving to the sites where fruit picking is most efficient due to the density of ripe strawberries. Optimization with respect to five criteria is performed within the coordination and harvesting processes. Each robot selects nondominated decisions based on the knowledge of the other robots positions and features within a certain AN subgraph termed the anticipation zone.

Below we present an example of an AN formation with 7 harvesting robots.

Example 1. Figure 1 shows an AN of seven robots, R_i, $i = 0, 1, \cdots, 6$, with 4 anticipation zones I_k, causal influences (red arcs) and anticipatory feedbacks (blue arcs). All robots except R_6 influence some other robots by the multifunctions $\varphi_{i,j}$. The AF relations are denoted by $f_{j,i}$. The anticipation zones I_k, $k = 0, 3, 4, 5$ determine the domain of knowledge of a robot. For example, the robot R_0 just knows about robots, R_1,R_2, R_5, because these are included in I_0, while the robot R_1 does not know anything about other robots except itself although it defines preferences regarding the decisions of R_4 and R_6. A robot may define an AF to the other robot outside of its anticipation zone but computing the decisions ensuring best satisfaction of this AF will not be possible [16]. The goal of coordination is to select decisions that ensure the satisfaction of a maximum number of AFs.

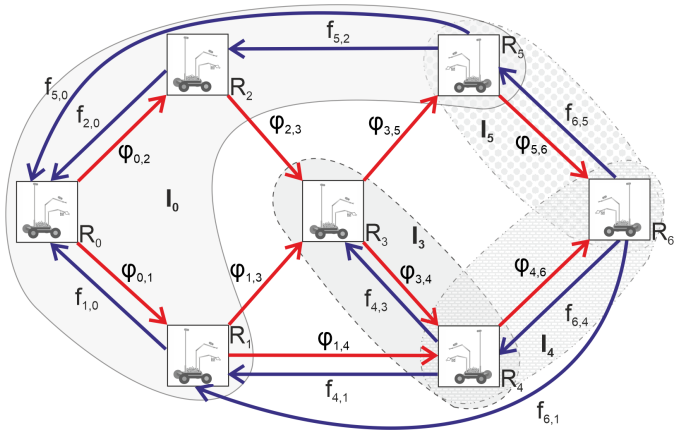


Figure 1. A team of 7 robots coordinated in an anticipatory network formation. Source: own work.

The behavior of intelligent harvesting robots in AN formations in a covered crop environment can be characterized by the 2nd autonomy level (cf. Sect. 2.2). Coordination reduces this autonomy measure on the factor $c(\varphi(u))/c(U)$, where $c(\varphi(u))$ is the number of decisions allowed to the subordinated agent after the coordinator's command u is issued. The set U contains decisions available to the same robot in a cooperating formation without coordination. Thus, the autonomy reduction of all robots can be regarded as a measure of coordination quality, according to the principle that coordination goals should be achieved with least possible interventions. On the other hand, more intensive reduction of autonomy may bear more expenses on part of the coordinating agent. Figure 2 shows a prototype autonomous strawberry harvesting robot which is capable of working in anticipatory formation teams. All above features have been implemented first within a realistic simulation framework, which allows the robotic solution designer to determine

optimal working parameters and to optimize the harvest. Then, the software was adopted to control harvesting robots.

Figure 2. An autonomous strawberry harvesting robot (*construction: UR Kraków*). Source: own work.

4. Conclusions

Anticipatory networks turned out an efficient formation model for autonomous harvesting robots. The overall harvest can be optimized with respect to five criteria at two levels, three of them describing the team performance and two governing robot individual behavior. The criteria at the team level refer to total harvest yield, harvest efficiency, and the picked fruits quality (all to be maximized). The remaining two lower-level individual criteria describe the accrued damage and energy consumption of each robot and are to be minimized.

However, AN-based coordination is not restricted to horticulture and may be applied in all situations where the communication difficulties can be remedied with the knowledge of decision algorithms by all robots in the team. The second prerequisite is the rationality of agents defined as the selection of a nondominated decision by each agent. In case of group decision making so defined rationality implies reaching a cooperative (Pareto) equilibrium of all team criteria, and – obviously – it excludes an adversarial agent behavior. The scope of applications of anticipatory coordination includes inspection robots working in harsh conditions of a mine [14]. The plans of future deployment include the coordination of search and rescue as well as space exploratory robots [16].

Acknowledgment

This research has been financed in part by the European Regional Development Fund (ERDF), contract No. POIR.01.01.01-00-0173/21-00.

References

[1] Skulimowski A.M., *Anticipatory network models of multicriteria decision-making processes*, *International Journal of Systems Science*, 2012, vol. 45, no 1, pp. 39–59, doi: 10.1080/00207721.2012.670308.

[2] Fierro R., Chaimowicz L., Kumar V., *Autonomous Mobile Robots Sensing, Control, Decision Making and Applications*, chap. Multi-Robot Cooperation, CRC Press, Boca Raton, 2018, pp. 417–460.

[3] Lin P., Liu J., Jin P.J., Ran B., *Autonomous vehicle-intersection coordination method in a connected vehicle environment*, *IEEE Intelligent Transportation Systems Magazine*, 2017, vol. 9, no 4, pp. 37–47, doi: 10.1109/mits.2017.2743167.

[4] Gonzalez E., la Rosa F.D., Sebastian A., Angel J., Sebastian J., *A control agent architecture for cooperative robotic tasks*, [In:] *Multi-Robot Systems, Trends and Development*, InTech, 2011, doi: 10.5772/13018.

[5] Surynek P., *Problem compilation for multi-agent path finding: a survey*, [In:] *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence Organization, doi: 10.24963/ijcai.2022/783.

[6] de Queiroz M., Cai X., Feemster M., *Formation Control of Multi-Agent Systems*, Wiley, 2018, doi: 10.1002/9781118887455.

[7] Khamis A., Hussein A., Elmogy A., *Multi-robot task allocation: A review of the state-of-the-art*, [In:] *Cooperative Robots and Sensor Networks 2015*, Springer International Publishing, 2015, pp. 31–51, doi: 10.1007/978-3-319-18299-5_2.

[8] Wei C., Ji Z., Cai B., *Particle swarm optimization for cooperative multi-robot task allocation: A multi-objective approach*, *IEEE Robotics and Automation Letters*, 2020, vol. 5, no 2, pp. 2530–2537, doi: 10.1109/lra.2020.2972894.

[9] Mariani S., Cabri G., Zambonelli F., *Coordination of autonomous vehicles*, *ACM Computing Surveys*, 2021, vol. 54, no 1, pp. 1–33, doi: 10.1145/3431231.

[10] Wang D., Deng H., Pan Z., *MRCDRL: Multi-robot coordination with deep reinforcement learning*, *Neurocomputing*, 2020, vol. 406, pp. 68–76, doi: 10.1016/j.neucom.2020.04.028.

[11] Desai J.P., *A graph theoretic approach for modeling mobile robot team formations*, *Journal of Robotic Systems*, 2002, vol. 19, no 11, pp. 511–525, doi: 10.1002/rob.10057.

[12] Skulimowski A.M.J., *Solving vector optimization problems via multilevel analysis of foreseen consequences*, *Foundations of Control Engineering*, 1985, vol. 10, no 1, pp. 25–38.

[13] Nickerson J.V., *Anticipatory systems: philosophical, mathematical, and methodological foundations*, *International Journal of General Systems*, 2012, vol. 41, no 8, pp. 867–871, doi: 10.1080/03081079.2012.726322.

[14] Skulimowski A.M.J., *Anticipatory control of vehicle swarms with virtual supervision*, [In:] *Lecture Notes in Computer Science*, Springer International Publishing, 2016, pp. 65–81, doi: 10.1007/978-3-319-51969-2_6.

[15] Skulimowski A.M.J., *Freedom of choice and creativity in multicriteria decision making*, [In:] *Knowledge, Information, and Creativity Support Systems*, Springer Berlin Heidelberg, 2011, pp. 190–203, doi: 10.1007/978-3-642-24788-0_18.

[16] Skulimowski A.M.J., *Coordination of autonomous mobile robot teams with anticipatory networks*, [In:] *The International Work Conference on Artificial Neural Networks*.

Evolution of Robotic System Specification Methodology

Maksym Figat[0000−0002−1898−0540], **Cezary Zieliński**[0000−0001−7604−8834]

Warsaw University of Technology
Institute of Control and Computation Engineering
Nowowiejska 15/19 00-665 Warsaw, Poland
maksym.figat@pw.edu.pl, cezary.zielinski@pw.edu.pl

DOI:10.34658/9788366741928.66

Abstract. *Design of robotic systems is a challenging task. More than 30 years ago some members of our team have embarked on a quest to find a general methodology for the design of any robotic system. Here we present the results that have been obtained thus far – a Robotic System Specification Methodology (RSSM). The foundation of RSSM is a metamodel – the scaffolding of any robotic system. Appropriate definition of the parameters of the metamodel transforms it into a model of a particular system, thus providing its specification, which in turn is translated into the control system code.*
Keywords: *Robotic System Specification, Robotic System Architecture*

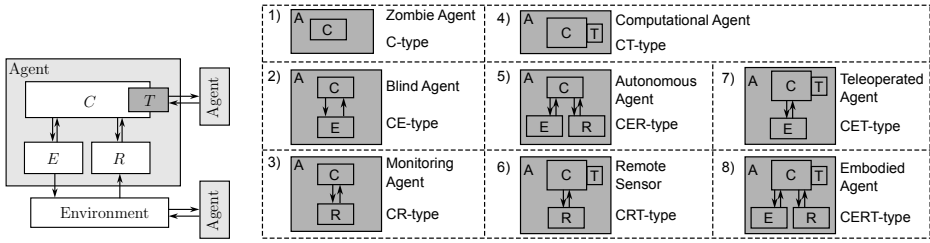
1. Introduction

Robotic systems are composed of robots and possibly some supplementary devices. Robotic System Specification Methodology (RSSM) is based on Model Driven Engineering (MDE) [1, 2]. Its founding concept is the embodied agent [3, 4, 5] and it follows the general MDE approach to system creation [6, 7]. This paper describes the underlying concepts and the evolution of thought that led to its development – more thoroughly presented in [5].

The remainder of the paper is divided into four sections. Sec. 2 presents the first layer of the RSSM, i.e. the architectural pattern for robotic systems (the following layers are explained in more detail in [8, 9, 10, 11]). Sec. 3 summarises the evolution of the methodology of designing robotic systems. Sec. 4 presents the current stage of the development of RSSM. Sec. 5 concludes the article.

2. Robotic system architectural pattern

Systematic design of robotic systems requires adequate specification tools, that rely on a well selected body of concepts and relations between them – an ontology.



(a) (b)

Figure 1: (a): General structure of an embodied agent submerged in its ambience; (b): Types of agents depending on their internal composition, i.e.: C, *E*, *R* and T. Source: own work.

The below presented concepts can be expressed precisely by diverse mathematical means, however here we focus on an introductory general formulation. The cornerstone of this approach is the concept of an embodied agent acting in a physical environment, thus the necessity of a physical body, i.e. embodiment. An embodied agent (Fig. 1a) uses its receptors *R* to gather the information about the state of the environment and influences that environment by using its effectors *E*. The control system C of the agent uses its knowledge about the task to coordinate the activities of its effectors and receptors in order to execute this task. The embodied agent can cooperate with other agents communicating with them utilising transmission buffers T or via the environment. In all its glory the embodied agent is composed of four components: C, *E*, *R* and T. By removing from the embodied agent any of the following three components: *E*, *R* or T, different types of agents emerge, as presented in Fig. 1b. Each type of the agent has different capabilities and is utilised for a different purpose [5]. Since the other types of agents emerge from the embodied agent, further discussion focuses on the embodied agent (Fig. 1a). Embodied agents are the building blocks of robots. A robotic system is composed of one or more robots and possibly auxiliary devices.

The internal structure of an embodied agent results from the observation that the following natural feedback loop exists: data from the environment is gathered by receptors, it is subsequently processed by the control system that is aware of the task to be executed, and hence it produces the commands driving the effectors which influence the environment, thus closing the loop. The monolithic control system C of an embodied agent has to be decomposed into subsystems: a single control subsystem *c* as well as zero or more virtual effectors *e* and virtual receptors *r*. Virtual effectors *e* drive real effectors *E*, while virtual receptors *r* aggregate information from real receptors *R*. The control subsystem acquires information from the real receptors via virtual receptors and commands real effectors via virtual effectors. The mentioned subsystems interact with each other through communica-

tion channels. Those communication channels transmit data from the output buffer of one subsystem to the input buffers of another subsystem. Agents interact directly by establishing communication channels between their control subsystems. Subsystems possess internal memory. The contents of the buffers and the memory express the concepts that are used to formulate subsystem tasks.

Each of the subsystems of the control system C executes its own task. Each task is defined as a set of behaviours and a mechanism switching between them. A behaviour iteratively: acquires from the communication channels data into input buffers, using that data and the contents of the internal memory as arguments computes a transition function, then places the results into the internal memory and the output buffers so that they can be dispatched through the communication channels to the other subsystems. The duration of each iteration defines the subsystem sampling time. The behaviour terminates when a predicate defining the terminal condition is satisfied. Then a next behaviour must be selected by the behaviour switching mechanism. The overall activity of the system results from the execution of the individual tasks of subsystems and the interactions between them.

The above mentioned concepts form an ontology for describing robotic systems, both their structure and their activities. Hence two different aspects of a single ontology can be distinguished: a) the ontology defining concepts forming a robotic system structure and b) the ontology defining concepts required to specify the robotic system activity. More details are available in [10, 11].

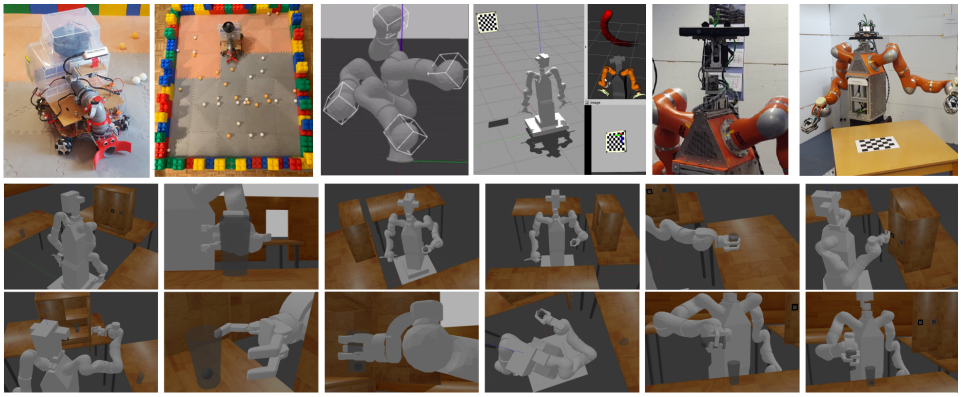


Figure 2: Experiments performed using the RSSM methodology. Source: own work.

3. Evolution of robotic system design methodology

In our early work, i.e. in MRROC++, a robotic system was composed of the following concepts: Effector Driver Process (virtual effector), Virtual Sensor Pro-

cess (virtual receptor), Effector Control Process (control subsystem), Master Process (system coordinator), and trajectory generator (transition function). Their utility was tested on the design of several robot controllers, e.g. industrial robot having a serial-parallel manipulator structure [12, 13]. Subsequently, those concepts were integrated into an embodied agent [14] and enabled the creation of multi-agent systems [15]. Further research concentrated on diverse definitions of transition functions and their iterative compositions, i.e. behaviours. This included: robots utilising position-force control [16], mobile robots [17], visual servoing [18], integration of vision and force control [19]. Besides studying fixed structure robotic control systems also variable structure ones have been designed [20]. Such systems are necessary whenever the numerosity of tasks that the system has to execute is unknown and is large, thus task executing modules have to be exchanged by acquiring new ones from a cloud repository. Switching of behaviours of a subsystem of an agent is usually executed by a finite state automaton (FSA). Implementation of communication between such subsystems required a whiteboard [8]. As the universality of embodied agents was validated [8] further research was directed at the simplification of their specification, thus the concept of hierarchic FSA (HFSA) was introduced [21]. The concept of hierarchy is also employed by Robotic System Hierarchic Petri Nets (RSHPN) [10]. However RSHPN represents inter/intra-agent communication in a more natural way. Treating RSHPN as a metamodel of a robotic system, RSSM could be established, which was subsequently verified on simple, however diverse, robotic systems presented in [9, 10, 11] (Fig. 2).

4. Robotic System Specification Methodology

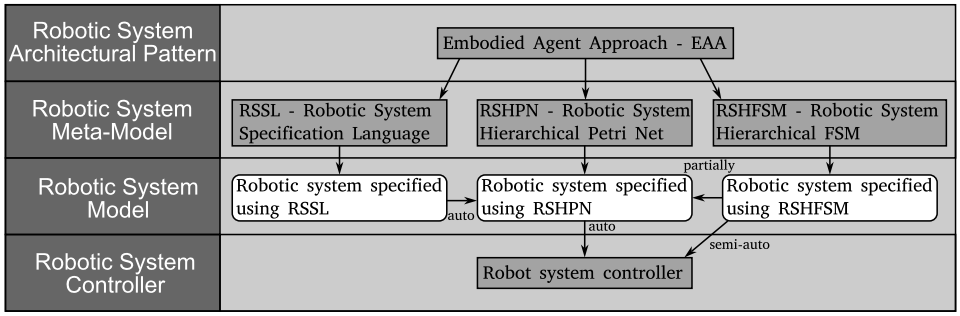


Figure 3: Stages of robotic system software development, where auto denotes automatic generation/transformation. The white rounded blocks are defined by the designer, the others have been already developed or are generated automatically. Source: own work.

RSSM has been developed in four stages, as presented in Fig. 3. In the first stage the domain concepts stemming from robotics and software engineering were

identified. Those concepts play a fundamental role in the specification of a general robotic system structure and its activities (Sec. 2). The choice of a specific subset of those concepts influences the definition of a robotic system meta-model. Up till now three different meta-models (and thus variants of the presented approach) have been formulated: 1) defined in terms of Hierarchic Finite State Machines (HFSM) [8] (RSHFSM), 2) defined in terms of Hierarchical Petri Nets (HPN) [9, 10] (RSHPN), and 3) defined in terms of Robotic System Specification Language (RSSL) [11]. According to the ISO/IEC/IEEE 24765:2010 [22] standard, a meta-model defines the elements used to specify a model. Most often in the literature, the meta-model is treated as a modelling language. Therefore, each of the above three meta-models can be also viewed as a domain specific language for modelling robotic systems. RSSL is a textual modeling language, while RSHFSM and RSHPN are graphical modelling languages. On the other hand, the parameterised RSHPN can be understood as a surrogate model of the model under development, requiring an appropriate determination of the parameters. The RSHPN and RSSL enable holistic definition of a robotic system activity, while in the case of RSHFSM meta-model [8] the communication between subsystems is treated as an implementation detail. In the third stage, using any of the three meta-models, a specification of a particular robotic system is produced, and thus the model of this particular system is created. In the fourth stage this model is used to generate the code of the robotic system controller in a general purpose programming language, e.g. C++, Python. Out of those four stages the system designer uses the tools provided by the second stage to create in the third stage a particular specification. The controller is produced as a result of a model-to-text transformation. The first two stages had been done when the meta-model was conceived, thus do not have to be repeated. The produced tools are used by the system developer only in stage three. Stage four is done automatically, hence the developer focuses only on expressing the system model based on any of the three meta-models – what is the subject of stage three.

Unavoidably HPNs are complex, thus direct system specification in terms of a RSHPN is cumbersome, therefore, instead of manually creating a model expressed by RSHPN, it is necessary to generate it automatically. This requires that the RSHPN meta-model is first appropriately parameterised and that parameters specifying the robotic system are provided [10]. For this purpose the RSSL language was developed. RSSL compiler transforms the specification expressed in RSSL into RSHPN. RSSL representation is much more compact than the RSHPN one, thus facilitating system specification and automatic implementation.

5. Conclusions

The RSSM described in this article resulted from a long evolution of robotic system design methods elaborated at our laboratory. This evolution not only improved and extended the concepts underlying RSSM but tested their utility on fairly complex systems. Thus RSSM, being the pinnacle of this evolution, certainly can be used for the purpose of specifying and generating controllers of diverse complex robotic systems. The mentioned concepts are fairly independent of each other thus they support decomposition facilitating robot system design. RSSM provides the metamodel (pattern) being the scaffolding for any robotic system. The parameters of the metamodel, when appropriately defined, transform this metamodel into a model of a particular system.

The RSSM is an original method of specifying controllers for any robotic system. The created universal parametric metamodel of robotic systems enables both the verification of system properties and the generation of controller code. The presented test systems (Fig. 2) produced by the proposed RSSM convince of its usefulness. The tools created significantly improve the design procedure.

References

[1] Brugali D., *Model-driven software engineering in robotics: Models are designed to use the relevant things, thereby reducing the complexity and cost in the field of robotics*, *IEEE Robotics & Automation Magazine*, 2015, vol. 22, no 3, pp. 155–166, doi: 10.1109/mra.2015.2452201.

[2] Nordmann A., Hochgeschwender N., Wrede S.B., *A survey on domain-specific languages in robotics*, [In:] *Simulation, Modeling, and Programming for Autonomous Robots*, 7, pp. 75–99.

[3] Brooks R.A., *Intelligence without reason*, [In:] *International Joint Conference on Artificial Intelligence*.

[4] Kornuta T., Zieliński C., *Robot control system design exemplified by multi-camera visual servoing*, *Journal of Intelligent &; Robotic Systems*, 2013, vol. 77, no 3-4, pp. 499–523, doi: 10.1007/s10846-013-9883-x.

[5] Kulczycki P., Korbicz J., Kacprzyk J. (eds.), *Automatic Control, Robotics, and Information Processing*, Springer International Publishing, 2021, doi: 10.1007/978-3-030-48587-0.

[6] de Araújo Silva E., Valentin E., Carvalho J.R.H., da Silva Barreto R., *A survey of model driven engineering in robotics*, *Journal of Computer Languages*, 2021, vol. 62, p. 101021, doi: 10.1016/j.cola.2020.101021.

[7] Brambilla M., Cabot J., Wimmer M., *Model-Driven Software Engineering in Practice*, Springer International Publishing, 2017, doi: 10.1007/978-3-031-02549-5.

[8] Zieliński C., Figat M., Hexel R., *Communication within multi-FSM based robotic systems*, *Journal of Intelligent &; Robotic Systems*, 2018, vol. 93, no 3-4, pp. 787–805, doi: 10.1007/s10846-018-0869-6.

[9] Figat M., Zielinski C., *Robotic system specification methodology based on hierarchical petri nets*, *IEEE Access*, 2020, vol. 8, pp. 71617–71627, doi: 10.1109/access.2020.2987099.

[10] Figat M., Zieliński C., *Parameterised robotic system meta-model expressed by hierarchical petri nets*, *Robotics and Autonomous Systems*, 2022, vol. 150, p. 103987, doi: 10.1016/j.robot.2021.103987.

[11] Figat M., Zielinski C., *Synthesis of robotic system controllers using robotic system specification language*, *IEEE Robotics and Automation Letters*, 2023, vol. 8, no 2, pp. 688–695, doi: 10.1109/lra.2022.3229231.

[12] Zieliński C., Szynkiewicz W., Mianowski K., Nazarczuk K., *Mechatronic design of open-structure multi-robot controllers*, *Mechatronics*, 2001, vol. 11, no 8, pp. 987–1000, doi: 10.1016/s0957-4158(00)00038-6.

[13] Zieliński C., Mianowski K., Nazarczuk K., Szynkiewicz W., *A prototype robot for polishing and milling large objects*, *Industrial Robot: An International Journal*, 2003, vol. 30, no 1, pp. 67–76, doi: 10.1108/01439910310457733.

[14] Zieliński C., *Formal approach to the design of robot programming frameworks: thebehavioural control case*, *Bulletin of the Polish Academy of Sciences – Technical Sciences*, 2005.

[15] Zieliński C., *A unified formal description of behavioural and deliberative robotic multi-agent systems*, *IFAC Proceedings Volumes*, 2003, vol. 36, no 17, pp. 405–412, doi: 10.1016/s1474-6670(17)33428-6.

[16] Zieliñski C., Winiarski T., *Motion generation in the MRROC++ robot programming framework*, *The International Journal of Robotics Research*, 2009, vol. 29, no 4, pp. 386–413, doi: 10.1177/0278364909348761.

[17] Janiak M., Zieliński C., *Control system architecture for the investigation of motion control algorithms on an example of the mobile platform rex*, *Bulletin of the Polish Academy of Sciences Technical Sciences*, 2015, vol. 63, no 3, pp. 667–678, doi: 10.1515/bpasts-2015-0078.

[18] Staniak M., Zieliński C., *Structures of visual servos*, *Robotics and Autonomous Systems*, 2010, vol. 58, no 8, pp. 940–954, doi: 10.1016/j.robot.2010.04.004.

[19] Staniak M., Winiarski T., Zielinski C., *Parallel visual-force control*, [In:] *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, doi: 10.1109/iros.2008.4650654.

[20] Zieliński C., Stefańczyk M., Kornuta T., Figat M., Dudek W., Szynkiewicz W., Kasprzak W., Figat J., Szlenk M., Winiarski T., Banachowicz K., Zielińska T., Tsardoulias E.G., Symeonidis A.L., Psomopoulos F.E., Kintsakis A.M., Mitkas P.A., Thallas A., Reppou S.E., Karagiannis G.T., Panayiotou K., Prunet V., Serrano M., Merlet J.P., Arampatzis S., Giokas A., Penteridis L., Trochidis I., Daney D., Iturburu M., *Variable structure robot control systems: The RAPP approach*, *Robotics and Autonomous Systems*, 2017, vol. 94, pp. 226–244, doi: 10.1016/j.robot.2017.05.002.

[21] Zieliński C., *Specification of agent based robotic systems using hierarchical finite state automatons*, [In:] *Advances in Intelligent Systems and Computing*, Springer International Publishing, 2020, pp. 465–476, doi: 10.1007/978-3-030-50936-1_39.

[22] *ISO/IEC/IEEE International Standard – Systems and software engineering – Vocabulary*, ISO/IEC/IEEE 24765:2010(E), pp. 1–418, 2010.

Improving RGB-D Visual Odometry with Depth Learned from a Better Sensor's Output

Aleksander Kostusiak

[1]*Poznan University of Technology*
Institute of Robotics and Machine Intelligence
ul. Piotrowo 3A, 60-965 Poznań, Poland
AleksanKostu@gmail.com

DOI:10.34658/9788366741928.67

Abstract. *This paper compares the results obtained from an indoor Visual Odometry (VO) system with RGB-D images provided by a Kinect v1 camera against those achieved by a VO with enhanced depth channel. For this purpose, we have used two classic image inpainting methods and a deep-learning approach for scene depth estimation employing Kinect v2 depth maps as reference data. The ability to enhance lower-quality data is crucial to reduce the cost of VO applications because higher-quality information can be infused through deep learning in systems using budget sensors.*
Keywords: *visual odometry, RGB-D sensors, inpainting, deep learning*

1. Introduction

The not-so-recent introduction of commodity RGB-D cameras allowed the development of new Visual Odometry (VO) and Simultaneous Localization and Mapping (SLAM) systems. The Kinect v1, available on the market since 2010, is inexpensive and allows sufficient measurements for many indoor localisation systems. However, the structured-light principle applied in this sensor and several similar devices, e.g., Intel's RealSense family, results in frequent depth artifacts and large no-data areas in the depth images. The ongoing progress has given us newer cameras like the Kinect v2 or Kinect Azure, which allow us to achieve better results and to enhance measurements from worse (often older) sensors to increase overall performance. We are using deep learning with the Kinect v2 images used as ground truth data for training to demonstrate this. Whereas inpainting of missing image data was already possible with classic algorithms, such as Telea [1] or Navier-Stokes [2], the advent of deep learning and open-sourced learning frameworks allows us to achieve better results, thanks to learning the dependencies between the RGB images available in all RGB-D sensors and the paired depth images. Hence, to enhance the depth information provided by Kinect v1 to achieve better frame-by-frame trajectory estimation, we are employing deep learning techniques and Kinect v2 depth frames as the training examples.

2. RGB-D visual odometry pipeline

Our research system[1] is a simple VO pipeline, following the feature-based approach to camera tracking (Fig. 1). Firstly, we detect, describe, and cross–match keypoints (using AKAZE detector/descriptor) with the corresponding depth information on two consecutive frames. Next, we filter out bad matches using the RANSAC procedure twice, with the minimal distance thresholds and the inliers to outliers ratio optimised, as described in [3], with the Particle Swarm Optimisation (PSO) algorithm. The AKAZE detector threshold is optimised further with PSO and Evolutionary Algorithm [3]. Finally, we estimate the transformation between those two frames and update the camera pose adding the frame-to-frame estimates head-to-tail.

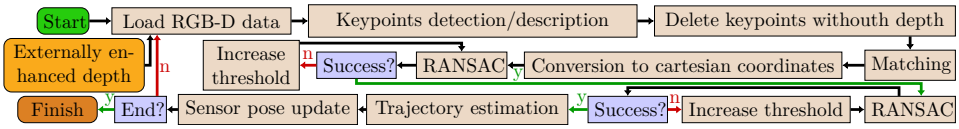


Figure 1. Block scheme of the simple VO system used in this research. Source: own work.

3. Dataset characteristics

In the experiments, we used the *PUTKK* [4] dataset containing 8 different trajectories, recorded by Kinect v1 and v2 cameras paired and moved together. The registered sequences consist of 60 – 2855 frames. The collection of all images from both Kinects and the motion-capture system (used for ground truth retrieval) have been time-synchronized. A more detailed description of the dataset and the test environment is given in [4], where Kraft et al. demonstrate that the missing depth data areas in Kinect v1 are an essential source of problems for VO systems because of the reduced number of useful point features.

4. Classic inpainting methods for depth estimation

The missing depth areas in Kinect v1 images can be substituted by depth values estimated upon the existing neighboring depth areas. To this end, the Telea [1] or Navier-Stokes [2] algorithms included in the OpenCV library can be applied, which require image masks that mark regions to be inpainted.

Telea uses the Fast Marching Method [5] to select sequent pixels to be inpainted. Then all known points in a given neighborhood (encircled with a chosen

[1]https://github.com/VVilk/RGBDVisualOdometryParticleSwarmOptimization

by-user radius) are used to compute a weighted average. The computed weights are chosen to propagate pixel depth values and sharp details of the image.

Navier-Stokes employs a fluid dynamics model to propagate the image Laplacian in the isophotes direction. The isophote lines in the inpainting region must be parallel to the level curves of the smoothness of the image intensity.

5. Deep learning for scene depth estimation

For the purpose of deep learning-based depth estimation we employ the Monodepth model [6], which follows the U-net architecture by using ResNet 18 with weights pre-trained on ImageNet, followed by up-scaling depth decoder and skip connections in between them.

We used the FastAi v1 framework to fine-tune this network with Euclidean RMSE error as a loss function. We have changed its first layer to accept 4-channel RGB-D input instead of standard 3-channel RGB images. For training, we used images from *PUTKK*: Kinect v1 as input and Kinect v2 (appropriately transformed to the view of v1) as ground truth. To augment data, we have chosen only affine (horizontal and vertical) transformations and rotation by 90°, as other techniques had a negative effect. We learned with momentum, with the learning rate decreasing in the direction of the first layers. We have learned only the last few classifier layers for the first few cycles. Then we fine-tuned the rest of the model.

6. Experiments and results

All experiments presented in this paper were performed using the *PUTKK* dataset sequences. For visual assessment of the enhanced depth frames see Fig. 2. Depth images inpainted with classic algorithms do not differ much. They do not restore most of the thin chair legs and are blurred in some areas. Objects in the depth images with learned depth values have smooth boundaries, present more plausible chair legs, and are overall more eye-pleasing, even though not all the details visible in the RGB images are reconstructed. We used *putkk_Dataset_5*

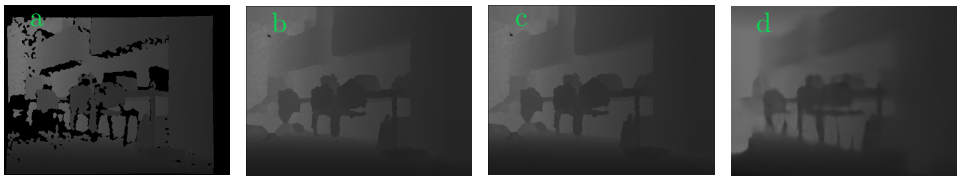


Figure 2. Depth maps: a) original, b) Navier-Stokes (NS), c) Telea, d) learned. Source: own work.

sequences for choosing the best inpainting radius for classic algorithms and for

parameter optimisation of the VO system using enhanced depth information. Parameters of the VO pipeline with enhanced depth frames were optimised using the approach from [3].

We used the popular ATE and RPE error metrics defined in [7] to asses the performance, computing the root mean squared errors for both metrics. The ATE needs trajectories to be aligned and takes the difference between the estimated and ground-truth camera poses. The RPE shows only the local differences between the estimated and ground-truth path. Tab. 1 presents ATE RMSE [7] and RPE RMSE results. For classical methods, we only show those for the best inpainting radius.

Table 1. Results for no inpainting, Telea and Navier-Stokes inpainting with radius 3, and learned depth with optimised parameters

	putkk_Dataset_5_Kin_1			
Metric	no inpainting	NS	Telea	learned depth
ATE RMSE [m]	0.198	0.201	0.204	0.059
Trans. RPE RMSE [m]	0.012	0.011	0.012	0.037

We can see the superiority of using learned depth images in VO tasks. To verify if this is not due to the optimization of parameters, we used other sequences from the *PUTKK* dataset. Tab. 2 collects the results for the first three sequences. The verification step shows that using enhanced depth information can

Table 2. Results for three *PUTKK* sequences not used for training nor optimisation of parameters.

Metric	no inpainting	NS	Telea	learned depth
	putkk_Dataset_1_Kin_1			
ATE RMSE [m]	0.596	1.112	0.443	0.275
Trans. RPE RMSE [m]	0.009	0.021	0.016	0.029
	putkk_Dataset_2_Kin_1			
ATE RMSE [m]	0.677	1.148	0.616	0.817
Trans. RPE RMSE [m]	0.010	0.030	0.033	0.044
	putkk_Dataset_3_Kin_1			
ATE RMSE [m]	1.145	0.934	1.092	0.807
Trans. RPE RMSE [m]	0.012	0.033	0.030	0.036

sometimes worsen the results–which is the case of *putkk_Dataset_2_Kin_1*, which also includes the learned approach. That may be due to specific scene characteristics differing from the scene appearing in the sequence used for training. Fig.

3 shows ATE plot (black lines represent the ground truth trajectory, blue lines are the estimated one, and red segments the Euclidean errors) and RPE plots for *putkk_Dataset_1*.

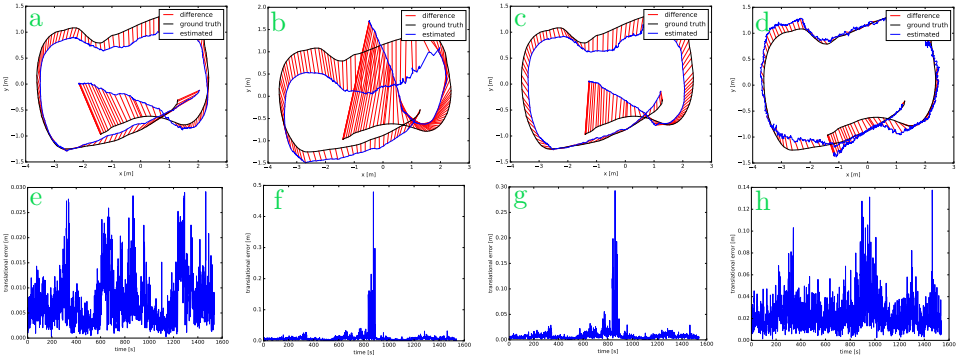


Figure 3. First row: ATE error plots, second: translational RPE plots of *putkk_Dataset_1_Kin_1* for VO system working with (a, e) no inpainting, (b, f) NS, (c, g) Telea, (d, h) learned depth. Source: own work.

7. Conclusions

We demonstrated that by employing a deep learning technique, it is possible to incorporate better sensor information into budget RGB-D VO systems. Our experiments show that deep learning allows for better results in the VO task than with the classic inpainting approaches that do not explore the RGB image context. Training the learned model jointly with the depth and RGB images and using a better, more complete depth image as ground truth for the loss function enables the model to infuse dependencies between the appearance of the objects and the no-depth areas in the Kinect v1 depth maps.

An issue in the proposed approach is that the deep learning model overfits to a given environment, and may not be suitable for different ones. Also, the keypoint detector parameters found using PSO/EA with the enhanced data differ from those obtained with original depth information allowing for detecting more keypoints. Such parameters also do not generalize well across different scenes.

References

[1] Telea A., *An image inpainting technique based on the fast marching method*, *Journal of Graphics Tools*, 2004, vol. 9, no 1, pp. 23–34, doi: 10.1080/10867651.2004.10487596.

[2] Bertalmio M., Bertozzi A., Sapiro G., *Navier-stokes, fluid dynamics, and image and video inpainting*, [In:] *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, pp. I–355, doi: 10.1109/CVPR.2001.990497.

[3] Kostusiak A., Skrzypczynski P., *On the efficiency of population-based optimization in finding best parameters for RGB-D visual odometry*, *J. Autom. Mob. Robotics Intell. Syst.*, 2019, vol. 13, no 2, pp. 5–14.

[4] Kraft M., Nowicki M., Schmidt A., Fularz M., Skrzypczynski P., *Toward evaluation of visual navigation algorithms on RGB-D data from the first- and second-generation kinect*, *Mach. Vis. Appl.*, 2017, vol. 28, no 1-2, pp. 61–74.

[5] Sethian J.A., *A fast marching level set method for monotonically advancing fronts*, *Proceedings of the National Academy of Sciences*, 1996, vol. 93, no 4, pp. 1591–1595, doi: 10.1073/pnas.93.4.1591.

[6] Godard C., Aodha O., Firman M., Brostow G., *Digging into self-supervised monocular depth estimation*, [In:] *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3827–3837.

[7] Sturm J., Engelhard N., Endres F., Burgard W., Cremers D., *A benchmark for the evaluation of rgb-d slam systems*, [In:] *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, ISBN 978-1-4673-1737-5, pp. 573–580, doi: 10.1109/IROS.2012.6385773.

Mixing Synthetic and Real-world Datasets Strategy for Improved Generalization of the CNN

Kamil Młodzikowski, Dominik Belter

Institute of Robotics and Machine Intelligence,
Poznan University of Technology,
60-965 Poznań, Poland

DOI:10.34658/9788366741928.68

Abstract. *In this paper, we deal with the problem of supervised training neural networks with an insufficient number of real-world training examples. We propose a method that at the beginning trains the neural network using a relatively simple synthetic dataset. In the following epochs, we add more challenging and real-life images to the training dataset. We compare the proposed strategy with other methods of using artificial and real-world datasets for training the neural network. The obtained results show that the proposed strategy allows for obtaining the neural network with higher generalization capabilities than competitive methods.*
Keywords: *deep learning, robot perception, articulated objects*

1. Introduction

When trying to learn new skills, people tend to start with easy, straightforward examples, increasing the difficulty in time. Such a strategy can also be helpful while working with deep neural networks. While training a model, a rich, robust, and balanced dataset is of great importance. In a typical scenario, we have a large synthetic dataset that can be used to train the neural network. However, the obtained neural network does not generalize well on the data from the real robot. On the other hand, we can have access to the dataset with a small number of real-life examples. Training on the limited dataset results in an overfitted neural network. The most popular strategy is to train the neural network on the dataset containing synthetic and real data at the same time. In this paper, we check if this strategy is a good choice.

In this research, we focus on the problem of mixing artificial with real-world data to achieve the best training outcome using the two sets. We propose three data mixing strategies, compare their influence on the training process and test the results on validation data to check which one provides the most generalized outcome.

1.1. Related Work

Our problem can be also treated as multi-task learning. One problem is to work on artificial data and one is to work on real-world data. Multi-task architectures in the field of computer vision have conventionally been constructed with a shared global feature extractor, consisting of convolutional layers, followed by distinct output branches for each task. The subsequent tasks use the output of the previous task as input, allowing for interdependent learning [1].

Another approach is to adjust and refine the simulated data to look more realistic. It can be achieved using GAN models [2]. The human brain is capable of continual learning through synaptic consolidation, which reduces the flexibility of synapses that are critical to previously learned tasks. In order to replicate this in artificial neural networks, the authors of [3] have developed an algorithm that constrains vital parameters to remain in proximity to their previous values.

Most existing methods that implement rehearsing for continual learning, primarily in the context of image classification, rely on reusing a subset of previously seen data during the training process. iCaRL [4] utilizes sets of representative images. When presented with new data for previously unseen classes, iCaRL modifies its feature extraction process and updates the exemplar set accordingly. OCS [5] leverages three selection strategies to obtain a core set that promotes generalization by discarding outliers and minimizing interference with previous tasks. On the other hand, the authors of [6], propose a new approach based on random undersampling, which allows them to preserve the entirety of past training data for retraining the model on future problems.

In [7], the network is trained on synthetic data by simulating the robot's camera view. Subsequently, the network is augmented with randomly initialized parameters and further trained on real-world robot manipulation tasks. A different approach is proposed in [8], where the main idea is to create a diverse dataset of artificial learning scenarios by randomly varying the environment, allowing for transfer learning to reality with minimal real images required for adjustment.

2. Modulated dataset mixing

We propose a deep learning method that utilizes linear, incremental mixing of real-world and synthetic data. We verify the proposed strategy on the problem of axis rotation segmentation on the RGB-D images. The problem of segmenting an axis of rotation on an image is challenging, partially because of insufficient real-world data. Collecting more of real-world data is time-consuming and requires precise measuring.

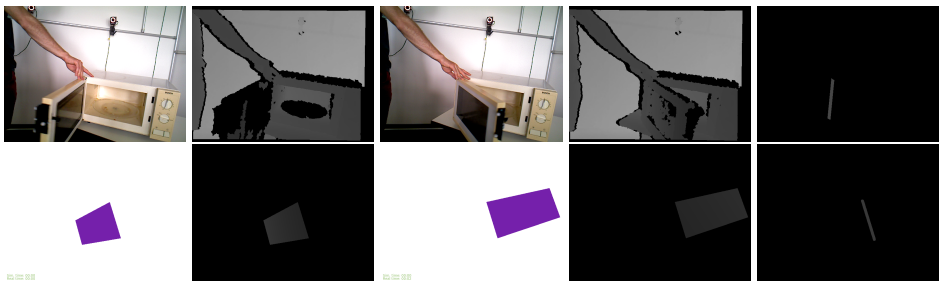


Figure 1: Example RGB-D pairs from the RBO (top) and synthetic (bottom) datasets. From left to right: RGB image of the first position, its depth image, RGB image of the second position, its depth image, and the axis of rotation. Source: own work.

2.1. Datasets

2.1.1. Real-world data (RBO Dataset)

To train the neural network, we use the real-life RBO Dataset [9]. It contains objects with rotational joints, precisely measured using motion capture systems. However, the data is redundant, as in our case usable sequences are only recorded from one perspective. Also, not many objects are available. We selected 20000 RGB-D pairs of images from the dataset to use in our tests. An example is presented in Fig. 1. The CNN trained only on this dataset is working well on similar objects, but does not generalize well [10]. Increasing the number of real-world examples would improve the generalization capabilities of the neural network, but it requires access to many unique objects and a lot of time for precise measuring.

2.1.2. Generated dataset

The synthetic dataset contains pairs of RGB-D images of rectangular planes rotating around one of the edges. We generated 20000 pairs to use in our tests. Example RGB-D images are presented in Fig. 1.

2.2. Deep neural network architecture

Our method was developed and tested using architecture presented in [10]. We use 3D U-Net [11] with a pair of RGB-D images, captured before robotic interaction with an articulated object and after rotating the object, as an input. The output from the CNN is a single image with a segmented axis of rotation.

2.3. Scenarios

We propose 4 dataset-mixing methods:

synth.→real – the CNN is trained on the synthetic images at the beginning and these images are gradually replaced by real ones.

synth.→real and synth. – the CNN is trained on the synthetic images at the beginning and we gradually add real images to the training set

real and synth.→real – the CNN is trained on the mixture of synthetic and real images at the beginning and we gradually remove synthetic images from the training set

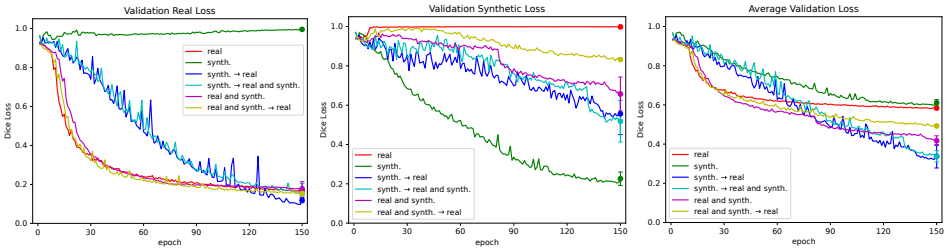
real and synth. – the CNN is trained on the mixture of synthetic and real images

We also train the network on only real and only synthetic data for comparison.

3. Tests and results

To compare the dataset mixing methods, we performed training the network 3 times for 150 epochs per scenario. Training CNN takes an average of 35 hours, and the whole testing takes about 630 hours. The network was evaluated separately on a synthetic validation set and on the real-world validation set. The average of these two validations was also calculated. To measure the performance of a network, the Dice Loss [12] was used.

After 150 epochs of training (Fig 2), the synth.→real and synth. and synth.→real scenarios achieved the best results on real and average validation loss, both reaching the average Dice Loss of 0.337. However the standard deviation of synth.→real and synth. is smaller since it achieved more consistent results. All the results are presented in Table 1.



(a) Validation performed on the real-world dataset. (b) Validation performed on the synthetic dataset. (c) Average value of both validation losses.

Figure 2: Training progress validated on real-world (a), synthetic (b) datasets, and the and the mixture of real and synthetic images. Source: own work.

We also performed tests on previously unseen sequences from the RBO Dataset. To quantitatively evaluate the results of the segmentation we compute the error an-

gle $\overline{e}^{\text{proj}}_{\text{axis}}$ described in [10]. The results are presented in Table 2. The synth.→real and synth. and synth.→real scenarios also achieved the best results in these tests.

Table 1: Error metric (Dice Loss) for the segmentation images on real validation set $\overline{e}_{\text{real}}$, synthetic validation set $\overline{e}_{\text{synth.}}$ and average of these two $\overline{e}_{\text{avg.}}$ at 150th epoch for all the training scenarios.

	real	synth	real & synth.	real & synth.→real	synth→real	synth→real & synth.
$\overline{e}_{\text{real}}$	0.169	0.996	0.178	0.153	0.119	0.157
σ_{real}	0.001	0.002	0.036	0.014	0.013	0.047
$\overline{e}_{\text{synth.}}$	0.998	0.226	0.657	0.832	0.556	0.518
$\sigma_{\text{synth.}}$	0.002	0.034	0.086	0.001	0.106	0.106
$\overline{e}_{\text{avg.}}$	0.584	0.611	0.417	0.493	**0.337**	**0.337**
$\sigma_{\text{avg.}}$	0.001	0.016	0.025	0.007	**0.059**	**0.029**

Table 2: Error angle $\overline{e}^{\text{proj}}_{\text{axis}}$ between the projection of the ground truth axis on the image plane and the direction given by the segmentation results [10] for the segmentation images on real-world test dataset at 150th epoch for all the training scenarios.

	real	synth	real & synth.	real & synth.→real	synth→real	synth→real & synth.
$\overline{e}^{\text{proj}}_{\text{axis}}$	0.527	1.241	0.563	0.703	**0.405**	**0.424**

4. Conclusion

In this paper, we propose dataset mixing methods that have a significant impact on the final model performance. The modulated mixing method helps with training a neural network with limited access to real-world data. We propose to start training with the synthetic dataset. With this strategy, the neural network learns quickly to solve the simplified problem. Then, we gradually introduce real and more challenging data. As a result, we obtain the best result on synthetic and real images when compared to other training strategies.

In the future, we are going to test the proposed strategy on the other popular problems in robotics that suffer from the limited number of real-world training examples.

Acknowledgment

The work was supported by the National Science Centre, Poland, under research project no UMO-2019/35/D/ST6/03959.

References

[1] Crawshaw M., *Multi-task learning with deep neural networks: A survey*, *CoRR*, 2020, doi: 10.48550/arXiv.2009.09796.

[2] Shrivastava A., Pfister T., Tuzel O., Susskind J., Wang W., Webb R., *Learning from simulated and unsupervised images through adversarial training*, [In:] *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2242–2251, doi: 10.1109/CVPR.2017.241.

[3] Kirkpatrick J., Pascanu R., Rabinowitz N., Veness J., Desjardins G., Rusu A.A., Milan K., Quan J., Ramalho T., Grabska-Barwinska A., Hassabis D., Clopath C., Kumaran D., Hadsell R., *Overcoming catastrophic forgetting in neural networks*, *Proceedings of the national academy of sciences*, 2019, vol. 114, no 13, pp. 3521–3526.

[4] Rebuffi S.A., Kolesnikov A., Sperl G., Lampert C.H., *icarl: Incremental classifier and representation learning*, [In:] *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5533–5542, doi: 10.1109/CVPR.2017.587.

[5] Yoon J., Madaan D., Yang E., Hwang S.J., *Online coreset selection for rehearsal-based continual learning*, [In:] *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, OpenReview.net.

[6] Zamorski M., Stypułkowski M., Karanowski K., Trzciński T., Zieba M., *Continual learning on 3d point clouds with random compressed rehearsal*, *Computer Vision and Image Understanding*, 2023, vol. 228, p. 103621, ISSN 1077-3142, doi: https://doi.org/10.1016/j.cviu.2023.103621.

[7] Rusu A.A., Vecerik M., Rothörl T., Heess N., Pascanu R., Hadsell R., *Sim-to-real robot learning from pixels with progressive nets*, 2016, doi: 10.48550/ARXIV.1610.04286.
https://arxiv.org/abs/1610.04286

[8] Tobin J., Biewald L., Duan R., Andrychowicz M., Handa A., Kumar V., McGrew B., Ray A., Schneider J., Welinder P., Zaremba W., Abbeel P., *Domain randomization and generative models for robotic grasping*, [In:] *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3482–3489, doi: 10.1109/IROS.2018.8593933.

[9] Martín-Martín R., Eppner C., Brock O., *The RBO dataset of articulated objects and interactions*, *The International Journal of Robotics Research*, 2019, vol. 38, no 9, pp. 1013–1019.

[10] Młodzikowski K., Belter D., *CNN-based joint state estimation during robotic interaction with articulated objects*, [In:] *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 78–83, doi: 10.1109/ICARCV57592.2022.10004277.

[11] Cicek O., Abdulkadir A., Lienkamp S.S., Brox T., Ronneberger O., *3D U-Net: Learning dense volumetric segmentation from sparse annotation*, 2016, doi: 10.48550/ARXIV.1606.06650.

[12] Sudre C.H., Li W., Vercauteren T., Ourselin S., Cardoso M.J., *Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations*, [In:] *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Springer International Publishing, 2017, pp. 240–248, doi: 10.1007/978-3-319-67558-9_28.

NeRF-based RGB-D Images Generation in Robotics – Experimental Study

Bartłomiej Kulecki[0000−0002−2820−8212], **Dominik Belter**[0000−0003−3002−9747]

Poznan University of Technology
Institute of Robotics and Machine Intelligence
Piotrowo 3A, 60-965 Poznań, Poland
bartlomiej.kulecki@put.poznan.pl

DOI:10.34658/9788366741928.69

Abstract. *Multiple learning-based algorithms in robotics require collecting RGB-D images of the scene from various viewpoints. These procedures are time-consuming, so many methods are trained using synthetic images. Recently, a Neural Radiance Fields (NeRF) model of the scene was proposed. Moreover, recent methods show that this model can be trained in minutes. This opens the possible applications in robotics for training the systems to reconstruct scenes, grasp objects or estimate their 3D poses using RGB-D images generated from a small number of input images. In this paper, we verify the quality of RGB-D images generated by the Instant Neural Graphics Primitives implementation of NeRF. We compare the obtained results from the Instant NeRF with the ground-truth RGB-D images obtained from the Kinect Azure and images generated from the point cloud model of the scene. The results show that the difference between generated RGB-D images and ground truth images is small, especially near the object.*
Keywords: *robot perception, image synthesis*

1. Introduction

Many machine learning methods in robotics for grasping [1] and scene reconstruction [2, 3] require large datasets to train the neural network. An RGB-D camera has to be moved around the scene and objects to collect a large number of training images. Because this procedure is time-consuming synthetic datasets of 3D objects like ShapeNet [4] are utilized to generate synthetic training images. However, this approach causes problems with transferring the neural network to the real robot that uses data from real sensors. Other methods create point cloud-based models of the environment generated using an RGB-D camera and reference tracking system [5]. Then, the points from the point cloud are projected on the new camera pose to generate synthetic RGB-D images.

In this paper, we propose and verify a hybrid approach that requires collecting a small number of real images of the scene and utilizes the neural-based method to

generate images of the scene from other viewpoints. Generative Adversarial Network (GAN) is a popular tool for the generation of synthetic images [6] but it is mainly used to generate images of objects that do not exist. Our goal is to generate the images of the scenes used to train the robotic perception system. Recently, a Neural Radiance Fields (NeRF) was proposed [7]. This method uses a small set of input images to learn the neural scene model and later uses the model to generate the images of the scene from other camera poses. In this paper, we verify the accuracy of the generated RGB and depth images for the popular Kinect Azure RGB-D camera. In this paper, we compare the images generated by the NeRF model and the images generated from the point cloud to show the capability of using NeRF as a potential replacement for synthetic and point cloud-based generators of RGB-D images.

2. Experiments description

The purpose of the experiments is to compare the NeRF-based scene modeling method with a method that generates images from the set of point clouds obtained from a real sensor moved around the scene. In this work, we use the Microsoft Kinect Azure camera, which returns high-quality color and depth images. In the first stage of the experiment, we collected data from the Kinect Azure RGB-D camera. A scene consists of a non-uniform background and a single object selected from the YCB Dataset. Then, we took 55 camera images of the object from various viewpoints. The images covered a full 360-degree range around the scene, the distance from the object varied in the range from 0.4 to 0.65 meters, and the camera height was in the range from 0.4 to 0.6 meters. The recorded data includes color images, depth images, and point clouds. The camera poses are determined for the collected images using the general-purpose Structure-from-Motion (SfM) tool named COLMAP [8]. We randomly divide the dataset into the train and test sets (44/11 images). The pipeline simultaneously estimates transformations that describe camera positions relative to the object's local coordinate system and camera parameters.

To train the NeRF model, we used the computationally efficient implementation named *instant-ngp* [9]. The main advantage of this version over the original NeRF implementation [7] is Multiresolution Hash Encoding of the input data, which increases the accuracy of the model, reduces the dimension of the neural network, and speeds up the learning and inference process. We used a set of 44 training images and estimated camera poses to train the model. This set is sufficient for rendering images of the scene from new perspectives. The selected implementation enables training with depth supervision, so we used camera depth images as an additional input in this training mode.

For comparison, we use collected point clouds, aligned and transformed to a

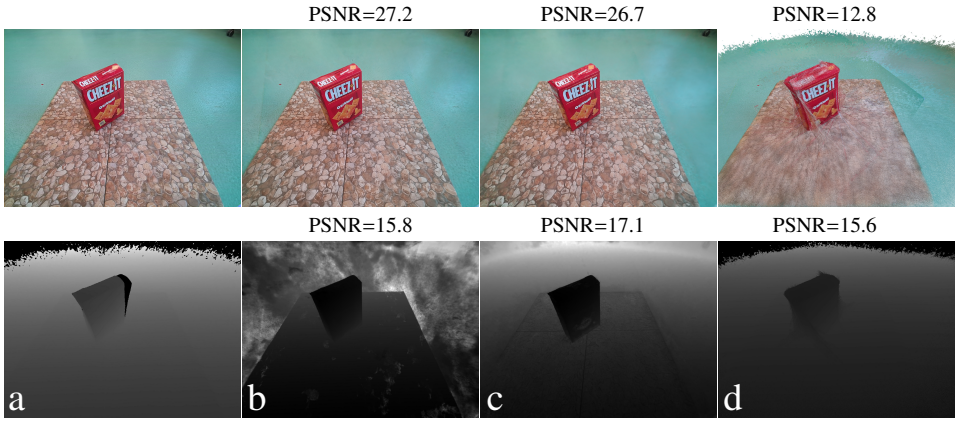


Figure 1. Example results of generating novel views: color (top row) and depth (bottom row) images from the camera – ground truth (a), the NeRF model (b), the NeRF model trained with depth supervision (c), and images generated from the point cloud model (d). Source: own work.

Table 1. Image similarity metrics for test images generated from point cloud (PCL), NeRF, and NeRF trained with depth supervision.

	RGB images						Depth images					
	PCL		NeRF with DS		NeRF		PCL		NeRF with DS		NeRF	
	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
MAE ↓	0.1553	0.019	0.0331	0.005	**0.0305**	0.006	0.1114	0.060	0.1834	0.018	**0.1023**	0.032
RMSE ↓	0.2162	0.021	0.0491	0.008	**0.0454**	0.009	0.2054	0.046	0.2078	0.017	**0.1765**	0.045
PSNR ↑	13.3335	0.853	26.2897	1.362	**27.0399**	1.822	13.9833	2.071	13.6786	0.755	**15.3522**	2.262
SRE ↑	57.9119	0.450	64.3818	0.695	**64.7581**	0.947	34.2305	1.820	34.0782	1.018	**34.9149**	0.759

global frame using COLMAP-determined camera poses. As a result, the individual point clouds of the analyzed object (from the training set) overlap to form a more complete model of the object. In the next step, we use the camera parameters to generate new views from the desired camera pose (from the test set).

3. Results

The example test sample is illustrated in Fig. 1. We show the color and depth images generated by the NeRF-based model, the NeRF-based model trained with depth supervision and obtained by projecting accumulated point cloud on the image. The images captured with the RGB-D camera are used as ground truth.

We can observe in Fig. 1d that the color image obtained from the point cloud is blurred in many places, and the object is slightly distorted. This comes from

the inaccurate depth measurements at the edges of the object and the limited precision of camera position estimation, which affects the shape of the final model. The RGB image generated by NeRF (Fig. 1b) is much more realistic, especially in regions with multiple visual features. The image generated by the NeRF contains more details, the ground structure is preserved, and the object's shape is accurate and consistent. The quality of the NeRF-generated images decreases on the background, where the texture is homogenous and does not contain visual features. Test images have better quality when the camera position is closer to the configurations included in the training set. Otherwise, the color image is blurred.

For all generated test images we calculate various similarity metrics by comparing the generated images to the reference camera images (ground truth). To increase the reliability of the results obtained for depth images, the metrics are calculated only for places where the ground truth (depth camera) images have values other than zero. The results obtained for the analyzed scene (mean values from 11 test samples) are shown in Tab. 1. It can be seen that the quality of color images is better for those generated by NeRF, as evidenced by lower MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) values compared to the PCL-based method. The PSNR (Peak Signal-to-Noise Ratio) and SRE (Signal to Reconstruction Error) metrics achieve higher values for the NeRF model. The NeRF-based method preserves the sharp edges of the object while the PCL-based method produces distorted objects.

From the point of view of robotics and especially object manipulation, depth images carry more relevant information than color images. The depth images generated from the NeRF model have many incorrect depth estimates in the plain regions (blue floor) without visual features. This drawback can be eliminated by training the NeRF model with depth supervision. In this mode, generated depth and RGB images have fewer unwanted patches. Although, the number of details in color images is lower than in those from the NeRF model trained without input depth data. Also, training with depth supervision oversharpens the details, such as the floor structure, and produces an offset in plain parts of the generated depth images. Even though these images (Fig. 1c) can look more accurate, the resulting offset causes lower average metric scores for this method.

4. Conclusions and future work

In this paper, we compared two methods of generating novel views of the scene. Color images generated from the model based on Neural Radiance Fields are much more accurate than images obtained from the point cloud. Also, depth images from NeRF have higher quality than PCL-based, and there is a possibility to improve it by training with depth supervision like in [10, 11]. The results show that NeRF can be used in robotics to generate realistic images of the scene for new

camera poses. In the future, we plan to tune or modify the depth supervision training of the NeRF model to eliminate the offset error. We will study the relationship between the amount of learning data, training time, the complexity of object geometry, and output quality. Our main goal is to use the NeRF's learning capabilities on a small amount of data to generate a dataset for training the grasping method from a single camera view.

References

[1] Yu Y., Cao Z., Liang S., Geng W., Yu J., *A novel vision-based grasping method under occlusion for manipulating robotic system*, *IEEE Sensors Journal*, 2020, vol. 20, no 18, pp. 10996–11006, doi: 10.1109/JSEN.2020.2995395.

[2] Staszak R., Kulecki B., Sempruch W., Belter D., *What's on the other side? A single-view 3D scene reconstruction*, [In:] *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 173–180.

[3] Park J.J., Florence P., Straub J., Newcombe R., Lovegrove S., *Deepsdf: Learning continuous signed distance functions for shape representation*, [In:] *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 165–174.

[4] Chang A.X., Funkhouser T., Guibas L., Hanrahan P., Huang Q., Li Z., Savarese S., Savva M., Song S., Su H., Xiao J., Yi L., Yu F., *ShapeNet: An Information-Rich 3D Model Repository*, Tech. Rep. arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[5] Wietrzykowski J., Belter D., *Stereo plane R-CNN: Accurate scene geometry reconstruction using planar segments and camera-agnostic representation*, *IEEE Robotics and Automation Letters*, 2022, vol. 7, no 2, pp. 4345–4352.

[6] Karras T., Laine S., Aittala M., Hellsten J., Lehtinen J., Aila T., *Analyzing and improving the image quality of StyleGAN*, [In:] *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, doi: 10.1109/CVPR42600.2020.00813.

[7] Mildenhall B., Srinivasan P.P., Tancik M., Barron J.T., Ramamoorthi R., Ng R., *NeRF: Representing scenes as neural radiance fields for view synthesis*, [In:] A. Vedaldi, H. Bischof, T. Brox, J.M. Frahm (eds.), *Computer Vision – ECCV 2020*, Springer, Cham, pp. 405–421.

[8] Schönberger J.L., Frahm J.M., *Structure-from-motion revisited*, [In:] *Conference on Computer Vision and Pattern Recognition (CVPR), 2016*.

[9] Müller T., Evans A., Schied C., Keller A., *Instant neural graphics primitives with a multiresolution hash encoding*, *ACM Trans. Graph.*, 2022, vol. 41, no 4, pp. 102:1–102:15.

[10] Deng K., Liu A., Zhu J.Y., Ramanan D., *Depth-supervised NeRF: Fewer views and faster training for free*, [In:] *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12872–12881.

[11] Dey A., Ahmine Y., Comport A.I., *Mip-NeRF RGB-d: Depth assisted fast neural radiance fields*, *Journal of WSCG*, 2022, vol. 30, no 1-2, pp. 34–43.

Predictive User Interface for Emerging Experiences

Paweł Kapusta[0000−0002−3527−7208]], **Piotr Duch**[0000−0003−0656−1215]

[1]*Lodz University of Technology*
Institute of Applied Computer Science
Stefanowskiego 18, 90-537 Łódź, Poland
pawel.kapusta@p.lodz.pl
piotr.duch@p.lodz.pl

DOI:10.34658/9788366741928.70

Abstract. *This research paper focuses on the use of predictive techniques to improve interaction with user interfaces in emerging experiences such as Virtual Reality, Augmented Reality, Metaverse, and touchless kiosks and dashboards. We propose the concept of intelligent snapping, which uses gaze tracking, head-pose tracking, hand tracking, as well as gesture recognition and hand posture recognition to catch the intent of the person rather than the actual input.*
Keywords: *Virtual Reality, Augmented Reality, User Experience, artificial intelligence*

1. Introduction and methodology

Virtual reality (VR), augmented reality (AR), and touchless interfaces are becoming increasingly popular for a variety of applications, including gaming, education, and healthcare. However, interacting with these systems using touch gestures or hand (controller) tracking can be inaccurate and imprecise, leading to frustration and decreased user engagement. To address these challenges, predictive user interfaces (PUIs) have been proposed to improve the interaction between the user and the virtual world, though many of the previous works were mostly concentrated on using gaze tracking [1] in order to predict the user's action.

The proposed PUI utilizes a combination of techniques to predict the user's intent and provide more efficient and accurate interactions. We use a multi-modal approach and combine hand posture recognition and gesture recognition, gaze tracking, head pose tracking and full body tracking with a neural network based on Transformer architecture. This allows us to infer the actual "intent" of the user and create a predictive and not reactive experience, as well as provide more efficient and accurate interactions in VR, AR, and touchless interfaces.

The approach is a natural progression of the work on Redirection Techniques (RETs) in VR, that allows exploring vast virtual environments while being constrained by physical room [2] [3]. They prove that it is possible to deceive the player in regard to rotation and translation mapping. Our work expands on this concept by introducing advanced trajectory prediction in order to achieve intelligent redirection of hand or controller movement mapping that is imperceptible to the player. This makes it possible to eliminate jarring object snapping in VR. By capturing the intent, it is also possible to build truly predictive User Interfaces by calculating the probability heatmap for all the UI elements. Moreover, by capturing the intent, it is possible to predict more advanced actions in real-time, such as changing gears in a driving simulator or reloading a weapon in an FPS and make these actions seem natural to the player.

References

[1] Karaman Ç.Ç., Sezgin T.M., *Gaze-based predictive user interfaces: Visualizing user intentions in the presence of uncertainty*, *International Journal of Human-Computer Studies*, 2018, vol. 111, pp. 78–91.

[2] Brument H., Marchal M., Olivier A.H., Argelaguet Sanz F., *Studying the Influence of Translational and Rotational Motion on the Perception of Rotation Gains in Virtual Environments*, [In:] *SUI 2021 - Symposium on Spatial User Interaction*, Virtual Event, United States, pp. 1–12, doi: 10.1145/3485279.3485282.

[3] Paris R.A., McNamara T.P., Rieser J.J., Bodenheimer B., *A comparison of methods for navigation and wayfinding in large virtual environments using walking*, [In:] *2017 IEEE Virtual Reality (VR)*, IEEE, pp. 261–262.

Semantic Segmentation for Autonomous Drone Delivery SUADD'23 Challenge

Anna Mrukwa[1,2][0000−0002−2886−8472], **Karol Majek**[2][0000−0002−1351−8496]

[1]*Silesian University of Technology, Department of Data Science and Engineering, Gliwice, Poland*
[2]*Cufix, 05-825 Grodzisk Mazowiecki, Poland*

DOI:10.34658/9788366741928.71

Abstract. *The popularity of drones as well as other different flying devices remains undeterred for several years now, with various industries recognizing their usefulness in a range of applications. However, the effectiveness of such systems is heavily dependent on real-time autonomous surface identification. The goal of this work is to evaluate recently published dataset dedicated to Unmanned Aircraft Systems. We performed experiments using several semantic segmentation neural network architectures. We outline possible improvements for future research and promising results for attention-based solutions in the field.*
Keywords: *Autonomous Unmanned Aircraft Systems, Semantic Segmentation*

1. Introduction

With the increased usage of various flying systems for a diverse range of tasks such as terrain mapping, surveillance or goods delivery, the need for efficient autonomous control arises. The growing number of such solutions makes the remote control by a man nearly impossible to manage. Yet, the aircraft system must be provided with a way to understand its location and to ensure a safe landing on proper terrain. Semantic segmentation of the images provided by the camera attached to the flying machine can be used for that.

Currently there are only a few semantic segmentation datasets for UAV such as UAViD [1], Semantic Drone Dataset[1] and a recently published dataset for the Scene Understanding for Autonomous Drone (SUAD) Semantic Segmentation & Depth Estimation challenge[2].

In this paper we shortly analyse the SUAD Semantic Segmentation dataset, and study the performance of models with different UNet-based architectures.

[1]http://dronedataset.icg.tugraz.at

[2]Available on the challenge website: https://www.aicrowd.com/challenges/scene-understanding-for-autonomous-drone-delivery-suadd-23

2. Methodology

2.1. Data acquisition and preprocessing

We used the SUAD Semantic Segmentation dataset. It entails greyscale photos from 412 drone flights, with several frames recorded per flight with Above Ground Level (AGL) ranging between 5 and 25 m, as well as greyscale segmentation masks. The ground truths masks are coded as consecutive classes with per-pixel values from 0 to 15 and 255 for the unknown class. For both training and validation we resize images to 512px squares, as the original image sizes across the datasets vary up to even 150 pixels per dimension. Images and masks were augmented via shift, scale and rotation. The pixel values were normalized using mean and standard deviation both equal to 0.5.

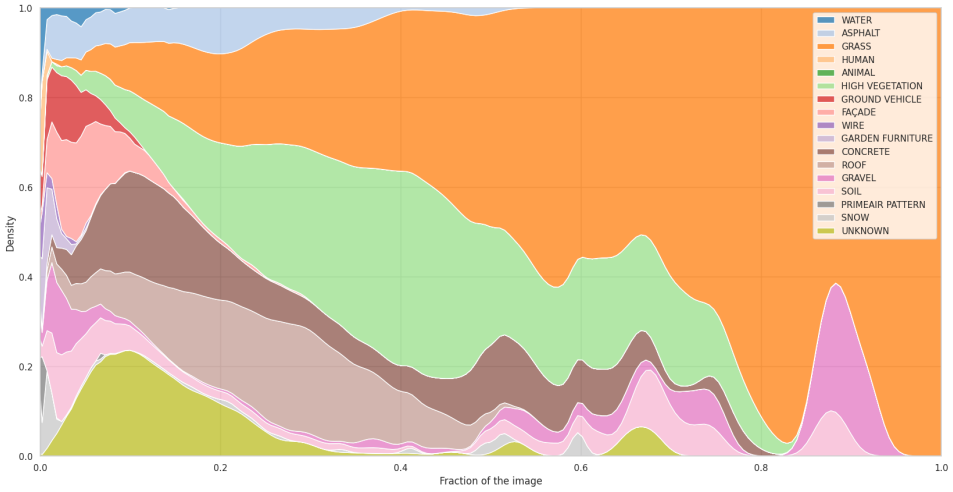


Figure 1. The contribution of the classes to the pixels percentage of the individual images rescaled to 512x512. Source: own work.

After analyzing the underlying ground truth categories, it can be seen that the most frequent class is *GRASS*, and the least frequent one is *ANIMAL*. The data are highly imbalanced, as can be seen in Fig. 1. Some of the classes occur only in a few photos but then may take a considerable portion of the photos – bodies of water or snow in the winter, obstructing the actual class of the surface below. Moreover, there is a significant share of the pixels labeled as unknown, not taken into account during the evaluation We do not consider them while computing loss during training. All of the above may lead to the model's failure in identifying living creatures or vehicles, as the number of images depicting them is significantly low.

2.2. Semantic Segmentation Networks Architectures

As the baseline architecture, we use the UNet [2] architecture, with ResNet-34 serving as the encoder. While the encoder downsamples the image, a similar upsampling procedure is added in parallel with the pooling process and the results of the appropriate layers on both sides are concatenated, creating a U-like structure. This enables general context understanding as well as more focus on finer details than in the image classification models.

The next architecture, U^2Net [3], is an improved version of the UNet structure described above. Rather than focusing on using even more complex architectures as the basis or linking several UNet instances together, as in the previous attempts of improving the model's results, this model introduces a rather different concept – the residual U-block is nested into the U-like structure. This allows the extraction and aggregation of the features more effectively. Both the *Lite* and *Full* versions of the structure are used in the experiments.

Since the usage of the transformers has been proven to be successful for the computer vision tasks [4], XUNet[3] model is proposed. This structure aims to combine the benefits of the nested U-structure and the attention layers. This solution should hence not only perform on par with the commonly used semantic segmentation models, but also need fewer computational resources.

2.3. Experiments

All the models were trained for 200 epochs, with the early stopping mechanism set to 10 on the validation loss. The analysis of the obtained training and validation loss curves indicates that the longer training may produce better results in the future work. RAdam optimizer was used, with the learning rate set to $3e^{-4}$. In all experiments we use Cross-Entropy Loss.

The validation set was extracted from the provided data, whereas the challenge organizers conducted the test set's evaluation remotely. Therefore, the evaluation of the models' quality is performed in two steps: firstly, locally, by observing the accuracy on the validation set, then on the unknown dataset via calculation of the Dice Index and mean Intersection-Over-Union (mIOU).

3. Results

While the accuracy of the models calculated on the validation dataset is quite high (as presented in Table 1), the other values that come from the external evaluation are extremely low. As can be seen, the XUNet architecture undoubtedly has the best performance across all the scores. Taking a closer look at the validation photos and the consecutive masks shown in Fig. 2, the issue presented by

[3]The implementation code is available here: https://github.com/lucidrains/x-unet.

the external scores stems partially from the lack of correct identification of smaller objects, which are bleeding into the background. In the case of the best architecture, some outline of humans can be already observed in the photos. Based on the experiment results, we do not observe overfitting of large models. Therefore the proposed training schedule does not guarantee minimum validation loss. While for the attention-based and the biggest architectures the segmentation results seem to be a little more consistent in the content of the whole photo, the other models are producing much more artifacts in the picture. However, the best models are performing worse in terms of memory usage (full U^2Net) or the inference time (XuNet). Considering application in an UAV it may be beneficial to use U^2Net Lite variant to reduce required memory.

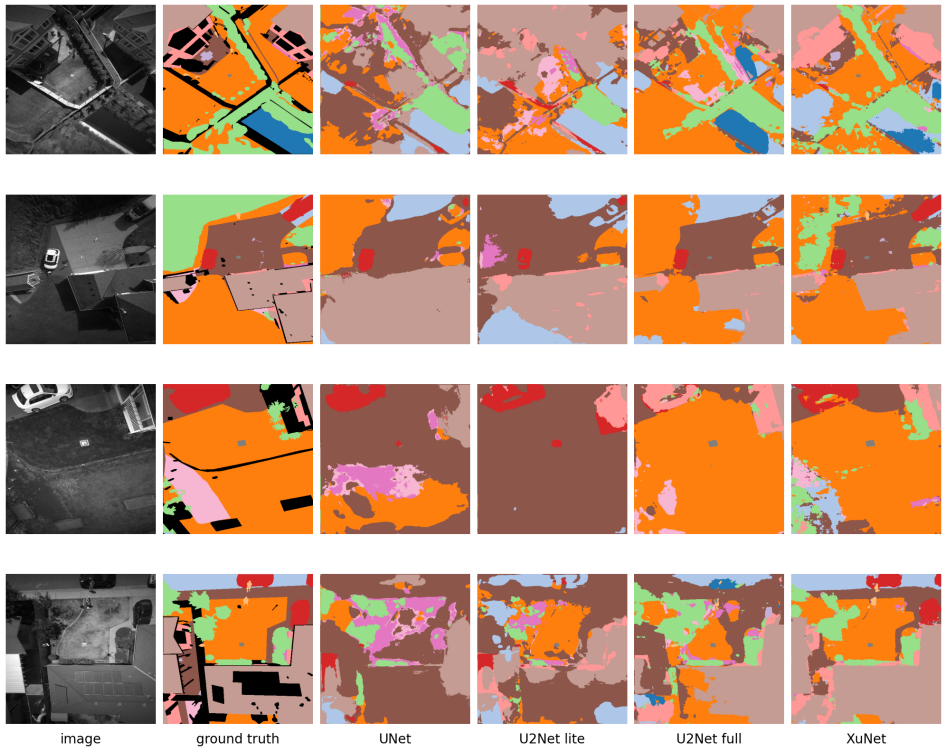


Figure 2. Qualitative comparison of the methods. Source: own work.

4. Conclusions and future work

The comparison of three different neural network architectures for semantic segmentation was presented. While the results provided by the XUNet architecture are of highest quality and the structure succeeds to be smaller than the other

Table 1. Results of models' evaluation; Inference time measured on a single image, using Nvidia RTX A4000 GPU, tested using 16bit floating point precision in PyTorch.

Model			Scores		
Name	**Parameters count**	**Infer time (s)**	**Dice Index**	**mIOU**	**Accuracy**
UNet	24.4 M	0.010	0.07	0.12	0.81
U^2*Net* lite	1.2 M	0.028	0.06	0.11	0.82
U^2*Net* full	44.2 M	0.028	0.24	0.16	0.94
XUNet	18.2 M	0.126	0.38	0.29	0.93

ones (as shown in Table 1), there is still the room for the improvement. To use models in UAV on computationally limited platform, models should be further optimized for a dedicated GPU using e.g. Nvidia TensorRT framework. The next step in the process of providing a functional model for the autonomous flight process would be increasing the length of the training. Further augmentations of the datasets to avoid overfitting with the particular focus placed on the photos with the smaller objects may solve the potential problem of neglecting living creatures in the segmentation, greatly improving the safety of the system. As a future work we consider also the comparison of the performance of models on a dedicated hardware e.g. Nvidia Jetson series.

References

[1] Lyu Y., Vosselman G., Xia G.S., Yilmaz A., Yang M.Y., *UAVid: A semantic segmentation dataset for UAV imagery*, *ISPRS Journal of Photogrammetry and Remote Sensing*, 2020, vol. 165, pp. 108–119.

[2] Ronneberger O., Fischer P., Brox T., *U-Net: Convolutional networks for biomedical image segmentation*, *ArXiv*, 2015, vol. abs/1505.04597.

[3] Qin X., Zhang Z.V., Huang C., Dehghan M., Zaiane O.R., Jägersand M., *U2-Net: Going deeper with nested u-structure for salient object detection*, *ArXiv*, 2020, vol. abs/2005.09007.

[4] Dosovitskiy A., Beyer L., Kolesnikov A., Weissenborn D., Zhai X., et al., *An image is worth 16x16 words: Transformers for image recognition at scale*, *ArXiv*, 2020, vol. abs/2010.11929.

Semi-formal Methods for Security Informed Safety Assessment of Robotic Systems

Vyacheslav Kharchenko[1][0000−0001−5352−077X],
Artem Abakumov[1][0000−0001−9359−097X],
Sergiy Yakovlev[2][0000−0001−6736−371X]

[1]*National Aerospace University "Kharkiv Aviation Institute"*
17 Chkalova Street, 61070 Kharkiv, Ukraine
v.kharchenkoi@csn.khai.edu, a.abakumov@khai.edu
[2]*Lodz University of Technology*
Institute of Information Technology
Politechniki 8, 93-590 Łódź, Poland
sergiy.yakovlev@p.lodz.pl

DOI:10.34658/9788366741928.72

Robotic industrial systems (RIS) are an aim of cyber attacks on vulnerabilities of RIS software and hardware. The intensity and variety of such attacks are increasing, first of all, due to the use of commercial components for the development/integration of RISs, information about the vulnerabilities of which is published and can be collected from various sources to enhance attack means. To evaluate the risks of successful cyber attacks, it is needed to collect and apply different techniques of assessment. The objective of the investigation is to suggest a method of risk-oriented assessing cyber security and safety of RISs and a choice of countermeasures to assure an acceptable risk of critical failures.

Based on the state-of-the-art analysis, it is suggested classification scheme and approach to complexing analytical techniques [1, 2] such as STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privileges), IMECA (Intrusion Modes and Effect Criticality Analysis), ATA (Attack Tree Analysis), F&VIT (Faults and Variabilities Injection Testing), R(S)DD (Reliability, Security& Safety Block Diagrams) and so on, and experimental methods, first of all, penetration testing (PT). Another particularity of the approach is the security-informed safety assessment of RISs. A method of combining IMECA analysis with penetration testing for the SIS assessment of RISs has been suggested to get the final risks of critical failures caused by hardware and software faults and attacks on vulnerabilities of components and their configurations described with the help of the functional IDEF model.

This paper addresses RIS architecture using a collaborative robot as an example. The set of vulnerabilities and their potential impact on the robotic system un-

der cyberattacks was assessed using the IMECA technique and summarized in the criticality matrix. Further steps and the necessary tools in performing penetration testing of the RIS and choice of countermeasures according to criteria "costs-SIS" are discussed. Besides, it is analyzed the features of implementing a scheme "AI powered attacks against AI powered protection" for the RISs.

The main particularity of the method is joining reliability and cybersecurity challenges to assess the safety of RISs. This is implemented by considering the different reasons, effects and criticality of RIS failures, including cyber attacks on vulnerabilities.

References

[1] Chlup S., Christl K., Schmittner C., Shaaban A.M., Schauer S., Latzenhofer M., *THREATGET: Towards automated attack tree analysis for automotive cybersecurity*, 2023, vol. 14, no 1, p. 14, doi: 10.3390/info14010014.

[2] Abakumov A., Kharchenko V., *Combining IMECA analysis and penetration testing to assess the cybersecurity of industrial robotic systems*, [In:] *2022 12th International Conference on Dependable Systems, Services and Technologies (DESSERT), 2022*, pp. 1–7, doi: 10.1109/DESSERT58054.2022.10018823.

Using Publicly Available Building Data to Improve 3D Map

Krzysztof Krygiel[1][0000−0003−0017−0299],
Karol Majek[1][0000−0002−1351−8496],
Janusz Będkowski[2][0000−0003−2630−1947]

[1]*Mobile4.pro sp. z o.o.*
02-621 Warsaw, Poland
[2]*Institute of Fundamental Technological Research,*
Polish Academy of Sciences,
02-106 Warsaw, Poland

DOI:10.34658/9788366741928.73

Abstract. *In this paper, we address the problem of 3D Map accuracy. No access to RTK GPS or LIDAR leads to poor accuracy of the map. High-rise buildings cause even greater trajectory errors. We used artificial intelligence methods to integrate publicly available building data and show that it can improve map accuracy from monocular camera and inaccurate GPS receiver. The main novelty is a method of building elevation detection in sparse point cloud data. We match detected elevations with building data and use modified bundle-adjustment algorithm to improve the map. We show that proposed approach decreases the trajectory error.*
Keywords: *SLAM, 3D Map, Building elevation detection*

1. Introduction and related work

As maps are crucial for many systems there is a lot of approaches to build them. Map accuracy highly depends on sensors used to capture the data. Using available solutions with access to low quality devices results in a non accurate map. One way to improve the quality of a map is to invest in expensive equipment. We show that it is also possible to improve quality by making use of publicly available 3D building data.

Several researchers investigate possibility of using external information in similar applications. Olga Vysotska and Cyrill Stachniss [1] proposed modification to graph-based SLAM and an advanced optimization framework of factor graph. They used information about the layout of building and fused it with the sensor data recorded by the robot.

Our approach does not require utilization of a laser scanner. We propose a solution which may improve quality of a *3D Map* built even with bottom-end devices.

Our goal was to be able to work on a map generated with ORB-SLAM [2] using monocular camera (smartphone) and low quality GPS data. We decided to use building 3D models published by the Polish Head Office of Geodesy and Cartography, but any other source containing similar data might be exploited such as [3]. The main novelty is the method of building elevation detection in a sparse point cloud and matching them with proper building elevations from external source. Having such a mapping we are able to minimize the trajectory position error and correct position of 3D points.

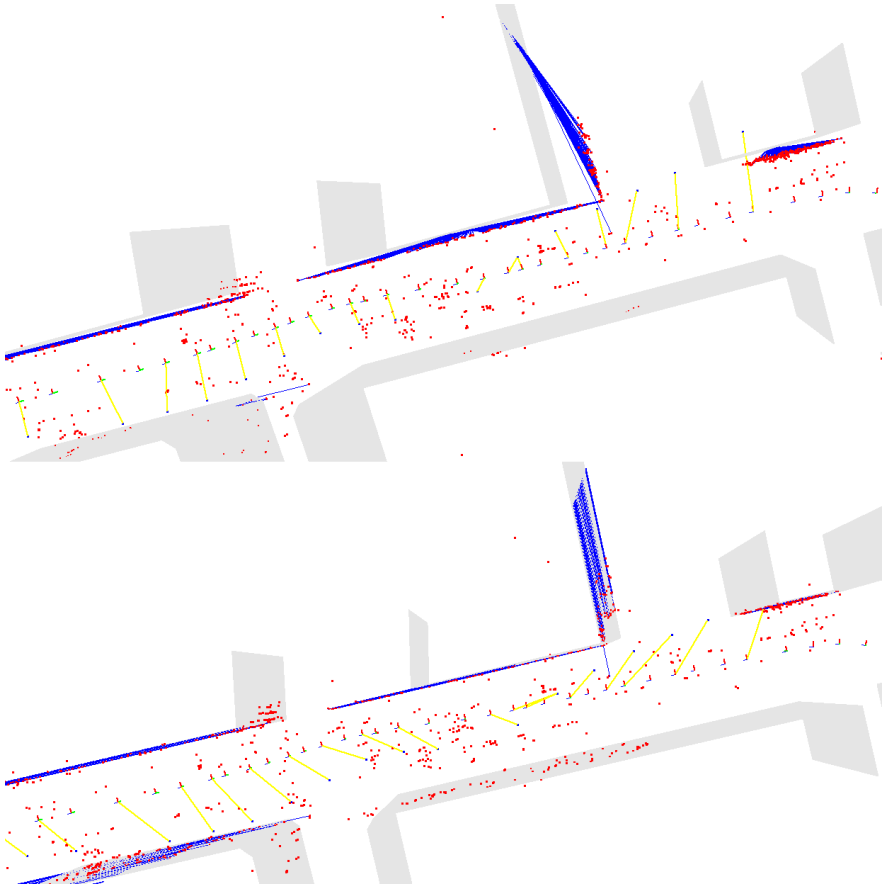


Figure 1. Visualization of data before (top) and after optimization; Map points in red, elevations planes in gray, yellow lines connect camera poses with *GPS* positions, blue lines connect points with the center of the elevation. Source: own work.

2. Proposed system

2.1. Data preparation

We propose a system which consists of 3 steps. In first step we prepare a map which is going to be optimized. As we wanted to prove that even low quality input might be used we decided to capture video using smartphone. Same phone was used as *GPS* receiver. Collected images were processed by a SLAM system in which map points are generated from key points detected with *ORB* detector. Generated output consisting of camera poses and sparse point cloud was transformed to world coordinate system. Rotation, translation and scale parameters were calculated using recorded *GPS* positions.

To improve the map we decided to use building 3D models. We processed building data to remove roofs, walls not visible from the street and elevations which are not vertical. Such preprocessing reduces the number of wrong matches between point cloud and external models. Figure 1 presents data prepared for showing error in the positioning of the model in 3D space.

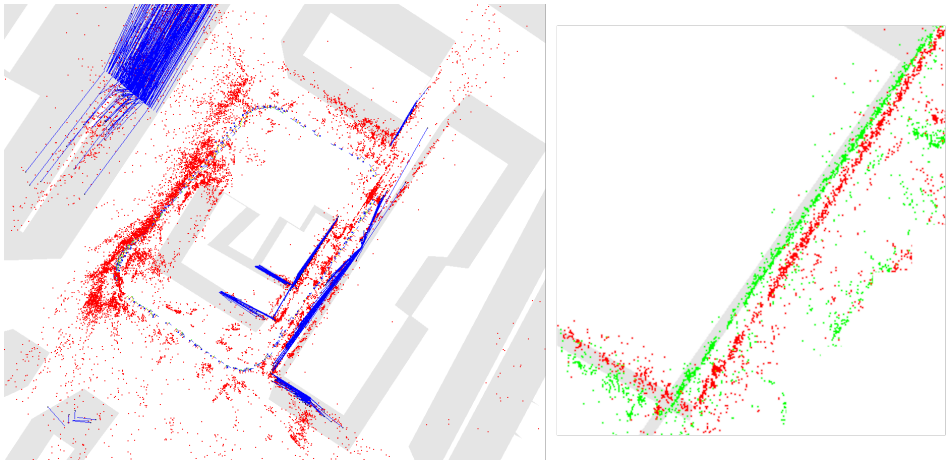


Figure 2. Map from Poznan University of Technology campus; Left: whole system before optimization, right: zoomed-in view of map points before optimization (red) and after optimization (green). Source: own work.

2.2. Detecting elevations in a point cloud

In the second step we aim to solve most critical aspect of whole solution which impacts heavily on final results. We need to find such sparse cloud point subsets which are with a certain degree of freedom coplanar and are close to planes (walls) obtained from external source. In urban environment there are more points fulfill-

ing these requirements (eg. billboards) so it is important to focus only on points covering large area which create a plane perpendicular to the earth's surface.

We decided to start with external elevations and try to find for each of them points in a point cloud. We process all elevations using only points which are in a close proximity. We use PCL implementation of RANSAC [4] to find planes among selected points. We filter resulting planes by the number of points, covered area, rotation relative to reference elevation. Finally we assign the points creating selected planes to the external plane from building elevation.

2.3. Optimization

In order to align a map with external data we decided to prepare optimization problem and use *Cholesky* solver as described in [5]. *"Camera reprojection error"* and *"motion model and relative poses"* conditions presented in [5] were used. We need to add new rows to design matrix containing information about detected building elevations.

Our optimization problem modifies camera poses $[R_i, T_i]$ (rotation and translation) and global position of each map point $P_j^g = (x_j^g, y_j^g, z_j^g)$. We need to prepare equation which calculates residual for each match between a point P_j^g and an elevation E_k.

As building elevation is a plane the easiest way is to represent plane E_k using plane equation:

$$A_k x + B_k y + C_k z + D_k = 0$$

Using this representation, it can be seen that for a point P_j^g in global reference system, residual is only dependent on point location. For each match between a map point P_j^g and external plane E_k we need to create one row in design matrix and set just 3 values A_k, B_k, C_k (in columns responsible for x_j^g, y_j^g, z_j^g).

Because we process only building elevations all C_k parameteres are zeros. In consequence, with this approach we can only optimize x_j^g and y_j^g global point coordinates. The way we use in order to fix altitude (z_j^g) is to get street elevation from external source for any camera pose and add modification to GPS positions taking into account on what height above the road level camera was set. Figure 1 presents results after using described method.

2.4. Results

Using building elevation from an external source allowed us to minimize the trajectory error. Elevation points were moved properly and all other components were transformed as well. In order to prove that such an approach decreases trajectory error we performed following experiments. We compared *Absolute Trajectory Error* for maps obtained using photos taken at Poznan University of Technology

campus. Translation part of camera poses were checked against *RTK GPS* data. Only 2D data were used, without the altitude part. After *SLAM* the mean *ATE* was close to 2.1 meter. Using proposed algorithm we were able to decrease it to 1.5 meter. In Figure 2 we show the whole experiment before optimization and one wall before and after optimization. Another way to show that camera pose location is improved is to use it to project building elevation corners into the picture. Having building corners in world coordinate system from used external source and knowing camera pose we may draw building contour as shown in Figure 3. It proves that not only translation of camera pose is improved but also the orientation part of poses.

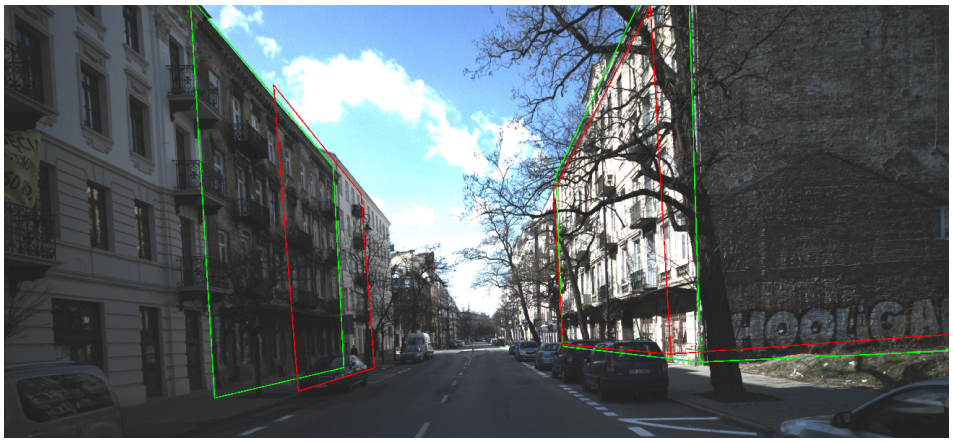


Figure 3. Projecting real object into picture; Red lines were calculated using camera pose before optimization and green with optimized data. Source: own work.

3. Conclusion and future work

In this paper, we presented a solution that allows to align a map to any external data containing 3D building models. By detecting building elevations in a point cloud and extending optimizer to contain additional conditions we are able to decrease Absolute Trajectory Error (ATE) comparing to ground truth. Main advantage of our solution is fact that it might be used with low-end devices and doesn't require lot of computation. As results strongly depend on quality of planes detection and assigning them to building elevations there is still wide scope for improvement. We focused on a single method in this step but it is worth of exploring how to find more points lying on elevations and how to eliminate wrong matches. It should allow for a significant improvement.

Acknowledgment

The research leading to these results has received funding from POIR.01.01.01-00-0494/20 "Development and verification of the automatic location and 3D visualization of the selected objects in urban environment technology together with people flow modeling".

References

[1] Vysotska O., Stachniss C., *Improving slam by exploiting building information from publicly available maps and localization priors*, *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 2017, vol. 85, no 1, pp. 53–65, doi: 10.1007/s41064-017-0006-3.

[2] Campos C., Elvira R., Rodriguez J.J.G., Montiel J.M.M., Tardos J.D., *ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM*, *IEEE Transactions on Robotics*, 2021, vol. 37, no 6, pp. 1874–1890, doi: 10.1109/tro.2021.3075644.

[3] OpenStreetMap contributors, *Planet dump retrieved from https://planet.osm.org* , 2017, (access: 17-07-2023). https://www.openstreetmap.org

[4] Rusu R.B., Cousins S., *3D is here: Point Cloud Library (PCL)*, [In:] *IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Shanghai, China 2011.

[5] Będkowski J., *Large-Scale Simultaneous Localization and Mapping*, Springer, 2022.

Chapter 9

Young.AI

Domain Editors:

1. Arkadiusz Tomczyk, Lodz University of Technology
2. Jakub Walczak, Lodz University of Technology
3. Stanisław Kaźmierczak, Warsaw University of Technology

AloneKnight – Enabling Affective Interaction within Mobile Video Games

Paweł Jemioło[1][0000 – 0001 – 5962 – 4043], **Krzysztof Świder**[1],
Dawid Storman[2][0000 – 0001 – 6643 – 1389],
Weronika T. Adrian[1][0000 – 0002 – 1860 – 6989]

[1]*AGH University of Science and Technology*
al. Mickiewicza 30, 30-059 Krakow, Poland;
{pawljmlo,wta}@agh.edu.pl, krzysztof.swider@hotmail.com
[2]*Chair of Epidemiology and Preventive Medicine;*
Department of Hygiene and Dietetics; Faculty of Medicine Jagiellonian University Medical College; dawid.storman@doctoral.uj.edu.pl

DOI:10.34658/9788366741928.74

Abstract. *Artificial intelligence is used in various contexts, including video games, where it can enhance the game design and adapt content to players' emotional states through affective computing. In this paper, we present an example of an affective mobile game and compare participants' opinions after playing two versions of the game, with and without an affective loop. The game was developed using Unity. In the affective version, physiological data is recorded and analysed to detect emotions based on facial expressions and electrodermal activity, which then affects the game. The study with 11 participants showed positive feedback for the game with affective loop.*
Keywords: *video games, mobile, artificial intelligence, affective computing*

1. Introduction

Artificial intelligence (AI) is applied in various contexts nowadays. Not only is it useful at workplaces but also accompanies people during leisure activities. One such example may be video games. Designers may utilise AI when developing new apps: for writing code [1], graphics production from text [2], enhancing creative processes, i.e. generating ideas regarding plots with large language models like GPT [3], controlling behaviour of non-player characters [4], and many others.

Since Rosalind Picard proposed a new research area [5], i.e. affective computing, which aims at creating emotion-aware applications, scientists made significant progress within this field. Much effort has been put into providing new models for emotion and affect recognition [6, 7] from different modalities, e.g. physiological signals or behavioural cues. Such models may then be used to adapt game contents

to the player's current state influencing it according to specification. We call this framework an *affective loop*.

Despite these advances, their usage when it comes to video games is limited mostly to the academic field [8]. Authors have focused on personal computers [9] as they have remained the most significant platform for video games for years. However, it is no longer true since mobile gaming is now more widespread among players [10]. At the same time, companies invest tremendous amounts of money in virtual and augmented reality games to provide even more immersive environments in the future. Still, people are not as interested in these environments as one would have expected [11].

One reason might be that too little effort is put into taking care of players' emotions when they play games as this very medium focuses on evoking specific feelings in its recipients. In this paper, we present an example of an affective mobile game. We transfer our experience in affective games for personal computers [9] to mobile platforms. We compare the opinions of participants playing two versions of the game, with and without affective loop.

2. AloneKnight

2.1. Game plot

The player is a knight named Arthur. Their goal is to escape from a haunted castle. On each level, they have to reach the door within a limited time frame. Multiple opponents, i.e. bats and dragons, try to disturb the player. The knight is equipped with the sword, which can be used to defeat them. For completing each level, the player is granted points, which are then summed up and presented as a total score in the end. The number of points that are awarded to the player depends on remaining time and slain monsters.

2.2. Game implementation and assets used

Our game was developed using Unity in two versions (with and without affective elements, see next sections). The application uses several free graphic assets from the Unity Asset Store, i.e. *2D Animated Fantasy Knight*, *Dragon and Princess Pack*, *Medieval pixel art asset*, *Fantasy Wooden GUI*, and *Free Monster Creature Battler + Animation Pack*.

Additionally, we used free sound effects from Freesound and a track called *It Is Coming* by David Fesliyan (free license for non-commercial use).

The player uses their thumbs to press onscreen buttons to control the game. The mobile device used for launching the game should be held horizontally and the player could not cover the front camera of the phone as it is used in the affective version of the game.

2.3. Measurements

To collect physiological signals, we utilised BITalino (r)evolution kit. In this game, we collected data regarding the electrodermal activity of the player. Two electrodes were placed on the palms of the subjects. Additionally, we took photos of users to assess their facial expressions using Microsoft Cognitive Services.

2.4. Affective loop

We introduced three game mechanics to close the affective loop within the application. By doing so, we aimed at enhancing the user experience from gameplay, as their state (assessed using electrodermal activity and facial expressions) affects events within the game.

Figure 1. Game screen with red filter added. Source: own work.

The first affective mechanic is accomplished using a simple colour filter put on the game depending on the emotion detected using Microsoft service. It is triggered every five seconds by taking a photo. Based on the obtained result, the adequate filter is applied: yellow for joy, blue for sadness, red for anger, dark grey for fear, green for surprise, orange for disgust, transparent for a neutral state, and light grey when no face was detected. The effect is presented in Figure 1.

The second affective mechanic bases on electrodermal activity. It is triggered every two seconds. If the currently registered value is greater than the previous one, the player's field of view decreases, and increases in a different situation.

To prevent losing the visibility completely, a minimum field of view has been established, see Figure 2. This mechanic is designed to motivate players to remain calm, as higher readings of electrodermal activity may indicate being aroused [12].

The last affective mechanic involves changing the speed of movement of bats, again based on measurements of electrodermal activity. In this case, the difference between a particular recording and the average value of the parameter for a given level (and a player) is calculated. If the sample value is greater than this average, the bats fly faster, and vice versa. When the reading is the same as the average, the bat flies at the default speed.

Figure 2. Game screen with a reduced field of view. Source: own work.

3. Evaluation

3.1. Procedure

We used OnePlus 6 with Android 10 and BITalino (r)evolution kit to conduct the experimental evaluation. The study consisted of several steps. Firstly, we informed the participant about the purpose of the study and obtained consent to participate in the study. Then, the participant played the version of the game without the affective loop and answered general questions about the game (Game Experience Questionnaire [13]). They used a scale from 1 (*I completely disagree*) to 5 (*I completely agree*). After this step, we connected electrodes from the BITalino device and the participant played the version of the game with the affective loop enabled. Next, they answered questions about the affective loop (regarding perceived differences, i.e. *The game with affective loop turned on gave me more impressions*, *Affective mechanics made this version of the game more interesting*). Again, they used a scale from 1 (*I completely disagree*) to 5 (*I completely agree*). Finally, we thanked the subjects for their willingness to participate in the study.

3.2. Participants

In the study, 11 participants (36% women) took part. Both experienced players and those who do not play at all took part in the experiment.

3.3. Results and discussion

Based on the GEQ questionnaire results, it can be concluded that the game was positively evaluated by the study participants. The category with a notably high score was *positive affect* (*mean* = 4.23). This result, along with the low ratings for questions regarding *negative affect* (*mean* = 1.85), indicates that the participants enjoyed the experiment. The game achieved fairly high scores in categories such as *immersion* (*mean* = 3.95) and challenge (*mean* = 3.75). It suggests that the game was difficult, and the players paid full attention to it.

Participants rather agree that the game with the affective loop turned on gave them more impressions (*mean* = 4.18). The same applies to the statement that the new mechanics made the game more interesting (*mean* = 4.18).

Subjects rated all implemented affective mechanics similarly ($mean_{m1}$ = 3.6, $mean_{m2}$ = 3.72, $mean_{m3}$ = 3.82). From the average ratings, it appears that the participants liked the third mechanic the most. One possible explanation is that it was the easiest one to notice as it significantly impact the difficulty of the game in contrast to the colour filter mechanic, which was only a visual addition. At the same time, it was not as challenging as the change of the field of view, especially when the level required many precise jumps.

4. Conclusions and future works

In this study, we showed that adding affective mechanics to the game might improve the way that it is perceived by players. Two versions of a game were compared and the majority of players preferred the one with additional mechanics enabled. In this prototype, we reasoned about the affective states of the players using simple rules. However, our preliminary engine might be easily replaced with a more advanced AI model. In the future, we want to carry out extended evaluations with more subjects within different experimental setups.

References

[1] Dakhel A.M., Majdinasab V., Nikanjam A., Khomh F., Desmarais M.C., Ming Z., et al., *GitHub Copilot AI pair programmer*, *arXiv*, 2022.

[2] Ding M., Zheng W., Hong W., Tang J., *Cogview2: Faster and better text-to-image generation via hierarchical transformers*, *arXiv*, 2022.

[3] Taecharungroj V., *"What Can ChatGPT Do?" Analyzing Early Reactions to the Innovative AI*, *Big Data and Cognitive Computing*, 2023, vol. 7, no 1, p. 35.

[4] Kopel M., Hajas T., *Implementing AI for non-player characters in 3D video games*, [In:] *Intelligent Information and Database Systems: 10th Asian Conference, ACIIDS 2018, Dong Hoi City, Vietnam, March 19-21, 2018, Proceedings, Part I 10*, Springer, pp. 610–619.

[5] Picard R.W., *Affective computing*, MIT press, 2000.

[6] Jemioło P., Storman D., Mamica M., Szymkowski M., Orzechowski P., *Automated Affect and Emotion Recognition from Cardiovascular Signals – A Systematic Overview Of The Field*, [In:] *Proceedings of the 55th Hawaii International Conference on System Sciences*.

[7] Jemioło P., Storman D., Mamica M., Szymkowski M., Żabicka W., Wojtaszek-Główka M., Ligęza A., *Datasets for Automated Affect and Emotion Recognition from Cardiovascular Signals Using Artificial Intelligence – A Systematic Review*, *Sensors*, 2022, vol. 22, no 7, p. 2538.

[8] Robinson R., Wiley K., Rezaeivahdati A., Klarkowski M., Mandryk R.L., *"Let's Get Physiological, Physiological!" A Systematic Review of Affective Gaming*, [In:] *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pp. 132–147.

[9] Jemioło P., Giżycka B., Nalepa G.J., *Prototypes of arcade games enabling affective interaction*, [In:] *Artificial Intelligence and Soft Computing: 18th International Conference, ICAISC 2019, Zakopane, Poland, June 16–20, 2019, Proceedings, Part II 18*, Springer, pp. 553–563.

[10] Wijman T., *The global games market will generate $ 152.1 billion in 2019 as the u.s. overtakes china as the biggest market*, (access: 17-07-2023).
https://newzoo.com/resources/blog/the-global-games-market-will-generate-152-1-billion-in-2019-as-the-u-s-overtakes-china-as-the-biggest-market

[11] Vanian J., *Metaverse off to ominous start after VR headset sale shrank in 22*, 2022, (access: 17-07-2023).
https://www.cnbc.com/2022/12/28/metaverse-off-to-ominous-start-after-vr-headset-sales-shrank-in-2022.html

[12] Fowles D.C., *The three arousal model: Implications of Gray's two-factor learning theory for heart rate, electrodermal activity, and psychopathy*, *Psychophysiology*, 1980, vol. 17, no 2, pp. 87–104.

[13] IJsselsteijn W.A., de Kort Y.A., Poels K., *The game experience questionnaire*, *Eindhoven: Technische Universiteit Eindhoven*, 2013, pp. 3–9.

AMUseBot: Towards Making the Most out of a Task-oriented Dialogue System

Iwona Christop, Kacper Dudzic, Mikołaj Krzymiński

Adam Mickiewicz University
Faculty of Mathematics and Computer Science
Uniwersytetu Poznańskiego 4, 61-614 Poznań, Poland
{iwochr2, mikkrz1}@st.amu.edu.pl, kacper.dudzic@amu.edu.pl

DOI:10.34658/9788366741928.75

Abstract. *This paper presents AMUseBot, a task-oriented dialogue system designed to assist the user in completing multi-step tasks. Taking into consideration that the fundamental issues with such systems are poor user ratings and high rates of uncompleted tasks, the main goal of the project is to keep the user focused and provide engaging conversations. We approach these problems by the introduction of dynamic multimodal communication and graph-based task management.*
Keywords: *dialogue systems, virtual assistants, conversational AI, artificial intelligence*

1. Introduction

Recently, interest in dialogue systems and virtual assistants has grown significantly. This pertains in particular to task-oriented dialogue agents that support the user in completing concrete tasks via conversation. Despite the rapid development, such systems still suffer from fundamental problems, such as poor user ratings and a high rate of uncompleted tasks. One of the main reasons for these issues is the difficulty of keeping users focused when dealing with complex or multi-step tasks [1].

These challenges provoked the development of AMUseBot – a task-oriented dialogue system whose main purpose is to assist the user in the domain of cooking. To enhance the user experience and decrease the high rate of uncompleted tasks, two novel approaches were combined – dynamic multimodal communication and graph-based task management. As a result, the user's attention is better maintained through interaction with multiple senses, while a visualization element and optimization of the flow of conducted tasks help ensure user satisfaction and incentivize goal achievement.

The primary objective of this paper is to describe the design of AMUseBot to date. The following section contains a survey of similar task-oriented dialogue

systems designed for the 2021 edition of the Alexa Prize TaskBot Challenge and their flaws [2]. The next part provides an overview of the technical implementation of the dialogue agent and high-level descriptions of its modules. The concluding section contains notes on future work.

2. Related Work

Progress in the field of task-oriented dialogue systems has been constant over the past several years. There are many dialogue agents on the market, each with its own strengths and weaknesses. Some were created for specific domains, while others provide interactions more general in nature. Considering the domain and characteristics of AMUseBot, a good point of reference could be the dialogue agents created as part of the Alexa Prize TaskBot Challenge – a competition focused on creating systems that guide users through multi-step complex tasks in the domains of cooking and DIY [2].

First place in the 2021 edition of the Alexa Prize TaskBot Challenge belonged to GRILLBot, developed by the University of Glasgow. Its key features include a new task representation mechanism called TaskGraphs and multi-modal elements such as videos and images to help users navigate through tasks. In the final months of the challenge, GRILLBot achieved an average rating of 3.86/5.0, which showed that the approach holds great potential for future improvements [3].

Second place in the competition went to TWIZ by NOVA School of Science and Technology. The system improves the clarity of complex tasks by decomposing them into simpler steps illustrated with images and videos. Additionally, the system keeps users engaged by providing a 3D visual preview of the task, introducing task-specific curiosities, and accounting for the user's cognitive load. The system exploits the multimodal capabilities of Alexa devices, and user feedback has shown that features such as curiosity facts and videos were highly valued [4].

Ohio State University placed third with its TacoBot, which overcomes several challenges related to a lack of in-domain training data, domain shift, and noisy user utterances through data augmentation strategies, annotation of real user conversations, actionable design guidelines, and an automated end-to-end test suite. As a result, TacoBot achieves improved natural language understanding and search, flexible dialogue management, engaging responses, and efficient issue identification. In the semifinals, TacoBot achieved a decent average rating of 3.55/5.0 and a task completion rate of 20.08% [5].

3. System Overview

AMUseBot follows the conventional dialogue system architecture and includes several modules responsible for specific natural language processing tasks. Fur-

thermore, two additional modules are introduced to control the visualization and storage of cooking recipes as shown in Figure 1 below. The system combines both machine-learning and rule-based modules.

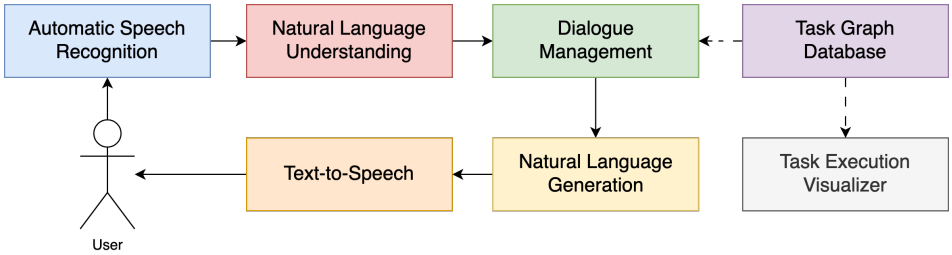


Figure 1. System architecture. Source: own work.

Automatic Speech Recognition The system is able to guide the user through the selected recipe using both voice and text conversation. The spoken language is converted into text input by the Automatic Speech Recognition (ASR) module. The core of the component is Whisper, a general-purpose speech recognition model developed by OpenAI, which utilizes a deep neural network trained on large-scale multilingual speech datasets to perform speech-to-text transcription. The model operates on spectrogram representations of the speech obtained by transforming the raw audio waveform with a sequence of Fourier transforms. The spectrogram is then fed into the Whisper model, which produces a sequence of phoneme or character-level transcriptions [6].

Natural Language Understanding The process of understanding the user's utterances by the Natural Language Understanding (NLU) module is treated as a classification task handled by a fine-tuned large language model. The module labels utterances with dialogue acts representing the user's actions, and slot values specifying the elements of the conversation that the system internally keeps track of. For better performance, two additional strategies were employed during the fine-tuning of the language model used by the module: data augmentation of the training dataset of user utterances, and supplying of additional information constituting the global context of the dialogue being conducted, such as previous turns of the conversation and previously identified intents and slot values.

Dialogue Management The Dialogue Management module is responsible for controlling the conversation flow and managing the interaction between the user and the system. The component consists of an internal ontological description of the supported domains, maintains a dialogue state representing the

current state of the conversation, and uses a dialogue policy model to decide upon the system's reactions. The module utilizes reinforcement learning techniques [7] and a large language model-based user simulator.

Natural Language Generation The Natural Language Generation (NLG) module generates natural language outputs in response to the user's inputs or system events. It is implemented as a template-based component that produces text responses relevant to the conversation based on appropriate dialogue data and context.

Text-to-Speech The final component of the system is the text-to-speech (TTS) module, which converts text generated by the NLG module into an audible speech signal. It is based on Google's Text-to-Speech technology, which uses deep neural networks to generate natural-sounding speech and offers multiple languages and voices, which makes it a versatile tool for dialogue system development [8].

Graph-based Task Management The main novel feature of the AMUseBot system is graph-based task management, which organizes the dialogue flow and provides visual hints to the user. A task graph consists of consecutive steps that guide the user toward the successful completion of a cooking recipe along with alternative steps that can be undertaken and suggested voice commands to trigger them, as shown in Figure 2. By providing a clear representation of task structures, such visualization can improve the user experience and increase the rate of completed tasks.

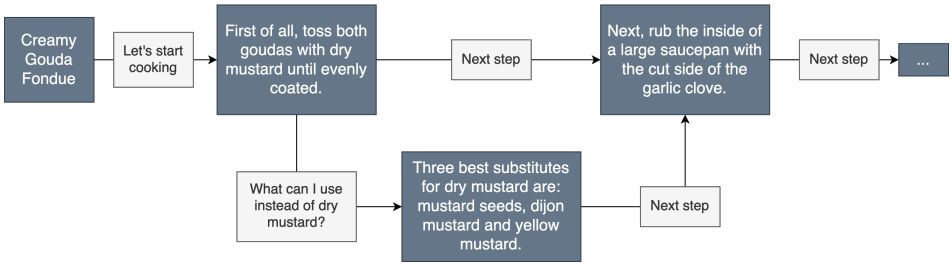


Figure 2. Dialogue flow concept graph. Source: own work.

4. Conclusions

Future work will focus on improving the performance and effectiveness of the AMUseBot system. To reduce transcription errors, algorithms for noise reduction will be applied and phonetic similarity will be used. The NLU module will be

expanded to include an additional model for named entity recognition. As for the intent and slot recognition model from the same module, its training dataset will be further augmented with synthetic utterances containing relevant entities, such as kitchen utensils and food products. A partial overhaul of the NLG module is also planned to integrate an autoregressive large language model for the purpose of generating the model's answers to very ambiguous or open-ended user questions, which cannot be feasibly covered by the existing template-based component.

Acknowledgments

First and foremost we are extremely grateful to the supervisor of the project, Marek Kubis, Ph.D., for his insightful comments and patience. His excellent experience and substantive advice made the system and paper possible. The authors would also like to acknowledge the invaluable contributions of Adrian Charkiewicz, Julian Zabłoński, and Dominik Zaleśny to this project. Their expertise, insights, and dedication were instrumental in the development of the solutions and the completion of this paper.

AMUseBot is being developed as part of the "Research and Development Project" course for students of the Computer Science MS degree program at Adam Mickiewicz University in Poznań. The course is financed by the AI Tech government project [9].

References

[1] Gottardi A., Ipek O., Castellucci G., Hu S., Vaz L., Lu Y., Khatri A., Chadha A., Zhang D., Sahai S., Dwivedi P., Shi H., Hu L., Huang A., Dai L., Yang B., Somani V., Rajan P., Rezac R., Johnston M., Stiff S., Ball L., Carmel D., Liu Y., Hakkani-Tur D., Rokhlenko O., Bland K., Agichtein E., Ghanadan R., Maarek Y., *Alexa, let's work together: Introducing the first alexa prize taskbot challenge on conversational task assistance*, 2022, doi: 10.48550/ARXIV.2209.06321.

[2] *Alexa Prize TaskBot Challenge*, 2023, (access: 17-07-2023). https://www.amazon.science/alexa-prize/taskbot-challenge/2021

[3] Gemmell C., Mackie I., Owoicho P., Rossetto F., Fischer S., Dalton J., *Grillbot: An assistant for real-world tasks with neural semantic parsing and graph-based representations*, *arXiv.org*, 2022, doi: 10.48550/arXiv.2208.14884.

[4] NOVA University Lisbon, *Twiz: A conversational task wizard with multimodal curiosity-exploration*, [In:] *Alexa Prize TaskBot Challenge Proceedings*, (ac-cess: 17-07-2023). https://www.amazon.science/alexa-prize/proceedings/twiz-a-conversational-task-wizard-with-multimodal-curiosity-

[5] Chen S., Chen Z., Deng X., Lewis A., Mo L., Stevens S., Wang Z., Yue X., Zhang T., Su Y., et al., *Bootstrapping a user-centered task-oriented dialogue system*, *arXiv.org*, 2022, doi: 10.48550/arXiv.2207.05223.

[6] Radford A., Kim J.W., Xu T., Brockman G., McLeavey C., Sutskever I., *Robust speech recognition via large-scale weak supervision*, 2022, doi: 10.48550/ARXIV.2212.04356.

[7] Sutton R.S., Barto A.G., *Reinforcement Learning: An Introduction*, A Bradford Book, Cambridge, MA, USA, 2018, ISBN 0262039249.

[8] *Text-to-Speech: Lifelike speech synthesis*, 2023, (access: 17-07-2023). https://cloud.google.com/text-to-speech

[9] *Website of the AITech project*, (access: 17=07-2023). https://www.gov.pl/web/aitech

Hierarchical Distributed Cluster-based Method for Robotic Swarms

Bartłomiej Mastej[1][0009−0006−6180−176X], **Maksym Figat**[2][0000−0002−1898−0540]

Warsaw University of Technology
[1]*Institute of Telecommunications*
[2]*Institute of Control and Computation Engineering*
Nowowiejska 15/19, 00-665 Warsaw, Poland
bartlomiej.mastej@gmail.com, maksym.figat@pw.edu.pl

DOI:10.34658/9788366741928.76

Abstract. *The growing interest in autonomous systems inspired by nature has led to a major shift towards swarm robotics. The main characteristics of swarms are independence from global knowledge, scalability and relatively low cost. However, the design of a swarm system is still a challenging task. Most of the existing research focuses on the task-specific solutions, which are hardly applicable to other solutions. Therefore, in this paper we present the method that provides a general guideline for the design of the swarm systems. The approach is verified in the simulation of the letter formation task.*

Keywords: *swarm robotics, swarm intelligence, multilayer cluster approach*

1. Introduction

Swarm robotics is a field of science that deals with large numbers of individual, relatively simple robots that are used to solve tasks collectively, without access to global knowledge. Due to the simplicity of the individual robot, the cost of the swarm system is low. They are inspired by biological societies such as fish, birds and ants. Swarms generally avoid direct communication. Instead, they use stigmergy. Swarms are highly scalable. The more robots used, the more efficient the system should be. Despite its unique advantages, swarm robotics is still in its infancy. The main advantages of swarm robotics: lack of global knowledge, decentralisation, simplicity create problems for engineers.

Although there has been a lot of research into swarm robotics, there has been little focus on design methodologies. As stated in [1] the most common design method is behaviour based. It focuses on designing the individual behaviour of the robot and then on its influence on the collective behaviour. This approach allows to solve such simple tasks as aggregation [2], collective movement [3][4]. However,

without the global knowledge, it is not possible to perform advanced tasks, e.g.: shape formation [5][6]. In advanced problems, there is a need to self-divide the task among the swarm members, as not all of them need to perform the same subtask at the same time.

Therefore, in recent works [3, 7, 8] focused on swarming robotics, a hierarchy has been proposed. However, these solutions either lack dynamic structure [3] or, despite the hierarchy, the abstraction layer is flat [7][8]. As for today, there are no design methods that can be easily applied to different tasks.

2. Contribution

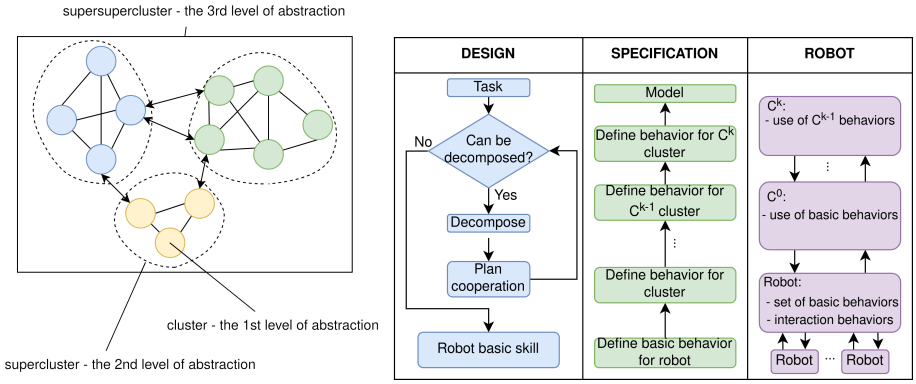
If we analyse a complex organism such as the human body, we can see that it is decentralised. The most primitive unit of the human body system is the cell. Proper cells work together and form the tissues. These tissues also work together to form the organs. Finally, the organs make up the human body. Inspired by this phenomenon, we have come up with a novel method to create a hierarchical, dynamic and distributed swarm. In terms of swarm robotics, the most primitive unit of the system is the individual robot. When a group of robots cooperate together, we treat them as a cluster. However, these clusters can also cooperate with each other to form superclusters. Furthermore, we can flexibly add the abstraction layers as needed. The general idea of the method is presented in Fig 1a. The advantage of this approach is that robots can perform different tasks at each level of abstraction, allowing a complex task to be solved more efficiently. Moreover, the hierarchy simplifies the design approach of the swarms.

In this paper, we present a method facilitating the design process of the swarm systems. We verify its usefulness in the experiments.

3. Method

Let's define the set of robots which perform the same basic task. Such a set is called a cluster C^0. A cluster C^k (from the k-th abstraction layer) is composed of cooperating clusters C^{k-1} (from $k-1$-th abstraction layer). Inside each C^k there must be considered two types of communication: intercluster and intracluster.

Designing process Fig. 1 shows a general idea of the design procedure using the multilayer cluster method. In the design phase the top-down approach can be used. Whereas, during the specification process one can apply the reverse process (bottom-up) to obtain the behavioral model of each robot inside the swarm. Each abstraction layer can be treated as a separate computational agent [9]. Thus, it may be possible to develop the cluster behaviours based on the design methodology for robotic systems [10].



(a) (b)

Figure 1: General cluster idea (Fig. 1a). Design procedure diagram (Fig. 1b). Source: own work.

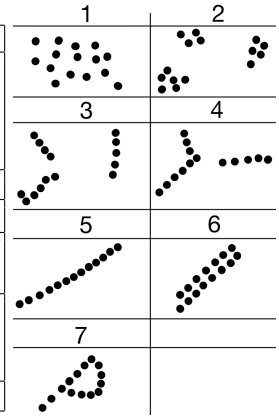
Intracluster communication In order to reach a common agreement between the robots within the cluster, a certain type of communication needs to be established. The communication within the cluster is similar to the flat wireless sensor network (WSN). As a result, different well-known routing protocols can be used within the cluster, assuming the anonymity of the agents, e.g.: the flooding routing algorithm – where information spreads incredibly quickly around the cluster, or first-past-the-post voting – a method of finding consensus.

Intercluster communication The clusters can communicate directly with each other through the robots that are on their edge. Therefore, the common agreement between the clusters can be achieved on the basis of the edge robots. Nevertheless, the edge robots can be considered as representatives of the whole cluster of the given hierarchy level. This means that the inside robots can contribute to the decision making process either directly or indirectly. Direct influence occurs when robots reach a common agreement through intra-cluster communication. While the indirect influence occurs when an edge robot makes a decision (based on its own knowledge of the current cluster state) without any consultation with its neighbours.

Swarm algorithms To validate the approach, it was decided to tackle an advanced problem – the formation of a letter. Therefore, it was necessary to modify some existing well-known swarm algorithms and to introduce new ones. Fig. 2 presents swarm algorithms used to tackle the problem.

1. Random motion
2. Aggregation: Cockroach algorithm [2] with the first-past-the-post algorithm to create a cluster among closely aggregated robots. Also used a custom-built algorithm called Timer Based Phase Switching (TBPS) to make a decision to end aggregation.
3. Collective movement: Leader-follower approach in use.
4. C^1 **creation:** The first C^0 to find the attraction point creates C^1.
5. Line formation: Each C^0 that detects the C^1 merges into the line and enters the C^1 (variation of TBPS is used).
6. Self-division: Bisection – every second robot takes a step forward. These robots form the new common C^0 (while still in same C^1) and maintain the line formation inside C^0.
7. Letter creation: Algorithms for "P", "V" and "W".

(a)



(b)

Figure 2: Algorithms description (Fig. 2a). Letter creation overview (Fig. 2b). Source: own work.

4. Experiments

The task was to align the swarm of robots into a given letter (as presented in the video[1]). Robots are assumed to be simple. They can move in any direction, observe its close environment and communicate with other robots by stigmergy. In the developed simulator each robot is represented by a dot. The colour of the dot represents the id of the swarm. Due to the complexity of the task it was decided to divide it into phases. A swarm of 40 robots was used in the simulation. Fig. 3 presents transitions from the random state (a) to the formation of the clusters (b). The subfigures (b) and (c) show the beginning of the collective movement. This is followed by the formation of the line. The highlighted cluster in (e) and (f) denotes the supercluster. One can observe how other robots merge into the cluster. Subsequently, in (h) self-division into two clusters is shown, although they are in the same cluster. Finally, in the last row the process of letter formation is shown. On (j) there is a letter "P", on (k) – "V" and on (l) – "W".

5. Conclusions

This article presented the general idea of the design approach for swarm robotic systems. It enables to create hierarchical, dynamical and distributed systems. Although the resulting letters are not perfect, unlike other existing approaches, no global knowledge was used. This proves that by decomposing the complex problem it is possible to solve complex task autonomously by the swarm. It can be

[1] Video of the robotic swarm forming the letter: https://youtu.be/mBGUdEeF6mU

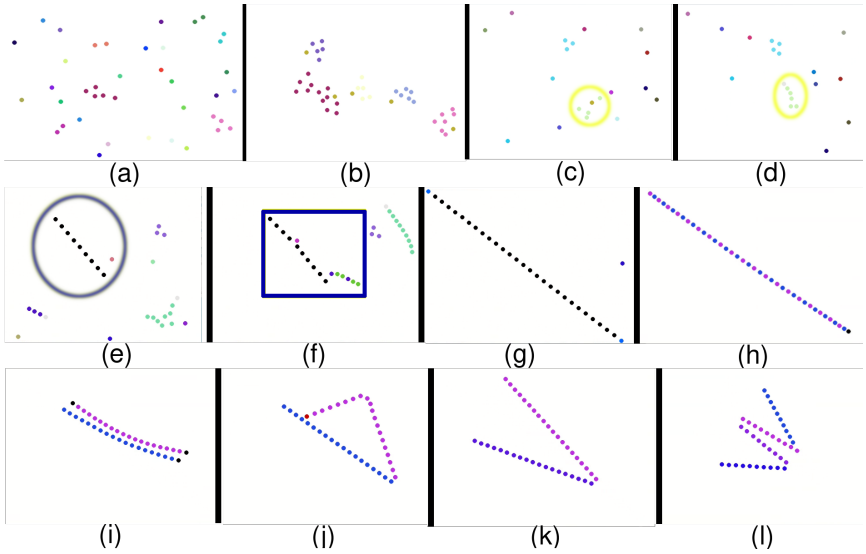


Figure 3: Letter creation experiments. Source: own work.

observed that the proposed method contributes to scalability, as depending on the abstraction layer it is easier to design the swarm behaviour that is scalable. The proposed approach can be extended to include automatic design methods, e.g. reinforcement learning – to train the model that decomposes the task into subtasks and assigns them to clusters. Furthermore, it would be beneficial to express the multi-layered cluster approach following the embodied agent approach and the robotic system design methodology [10, 9].

References

[1] Brambilla M., Ferrante E., Birattari M., Dorigo M., *Swarm robotics: a review from the swarm engineering perspective*, *Swarm Intelligence*, 2013, vol. 7, no 1, pp. 1–41, doi: 10.1007/s11721-012-0075-2.

[2] Correll N., Martinoli A., *Modeling and designing self-organized aggregation in a swarm of miniature robots*, *The International Journal of Robotics Research*, 2011, vol. 30, no 5, pp. 615–626, doi: 10.1177/0278364911403017.

[3] Haghighi R., Cheah C., *Multi-group coordination control for robot swarms*, *Automatica*, 2012, vol. 48, no 10, pp. 2526–2534, doi: 10.1016/j.automatica.2012.03.028.

[4] Nouyan S., Dorigo M., *Chain based path formation in swarms of robots*, [In:] *Ant Colony Optimization and Swarm Intelligence*, Springer Berlin Heidelberg, 2006, pp. 120–131, doi: 10.1007/11839088_11.

[5] Rubenstein M., Cornejo A., Nagpal R., *Programmable self-assembly in a thousand-robot swarm*, *Science*, 2014, vol. 345, no 6198, pp. 795–799, doi: 10.1126/science.1254295.

[6] Wang H., Rubenstein M., *Shape formation in homogeneous swarms using local task swapping*, *IEEE Transactions on Robotics*, 2020, vol. 36, no 3, pp. 597–612, doi: 10.1109/tro.2020.2967656.

[7] Yang H., Li Y., Duan X., Shen G., Zhang S., *A parallel shape formation method for swarm robotics*, *Robotics and Autonomous Systems*, 2022, vol. 151, p. 104043, doi: 10.1016/j.robot.2022.104043.

[8] Gigliotta O., *Equal but different: Task allocation in homogeneous communicating robots*, *Neurocomputing*, 2018, vol. 272, pp. 3–9, doi: 10.1016/j.neucom.2017.05.093.

[9] Figat M., Zieliński C., *Parameterised robotic system meta-model expressed by hierarchical petri nets*, *Robotics and Autonomous Systems*, 2022, vol. 150, p. 103987, doi: 10.1016/j.robot.2021.103987.

[10] Figat M., Zielinski C., *Synthesis of robotic system controllers using robotic system specification language*, *IEEE Robotics and Automation Letters*, 2023, vol. 8, no 2, pp. 688–695, doi: 10.1109/lra.2022.3229231.

Lung Xray Images Analysis for COVID-19 Diagnosis

Anna Kloska[1][0000−0002−5776−5448], **Martyna Tarczewska**[2], **Agata Giełczyk**[2][0000−0002−5630−7461], **Beata Marciniak**[2][0000−0003−1092−1279]

[1]*Faculty of Medicine Ludwik Rydygier Collegium Medicum in Bydgoszcz, Nicolaus Copernicus University in Torun, Bydgoszcz, Poland*
[2]*Bydgoszcz University of Science and Technology, Bydgoszcz, Poland*
agata.gielczyk@pbs.edu.pl

DOI:10.34658/9788366741928.77

Abstract. *Background: The SARS-CoV-2 pandemic began in early 2020. It paralyzed human life all over the world and threatened our security. Thus, proposing some novel and effective approaches to diagnosing COVID-19 infections became paramount. Methods: This article proposes a method for the classification of chest X-ray images based on the transfer learning. We examined also different scenarios of dataset augmentation. Results: The paper reports accuracy=98%, precision=97%, recall=100% and F1-score=98% in the most promising approach. Conclusion: Our research proofs that machine learning can be used in order to support medics in chest X-ray classification and implementing augmentation can lead to improvements in accuracy, precision, recall, and F1-scores.*
Keywords: *COVID-19, image processing, augmentation, artificial intelligence*

1. Introduction

A recently observed SARS-CoV2 pandemic has infected millions of people all over the world and caused numerous problems: health, mental, social and economical. It was confirmed by World Health Organization that more than 200 countries have been affected by the coronavirus pandemic. The infection caused some disease symptoms (cough, fever, fatigue, and respiratory distress), but the pandemic lead to an unprecedented failure in the health services due to the lack of medical staff or overloading entire healthcare systems. However, latest scientific reports suggest that artificial intelligence (AI) could be used to overcome the pandemic crisis including by its usage in: diagnosis, drug development and treatment. In this article we present the baseline COVID-19 detection system based on lung Xray images and its further improvements by selected dataset augmentation techniques.

2. Related work

Two important issues may be raised after the state-of-the-art review: the necessity of using segmentation and the strong tendency to use various transfer learning in lung Xray images analysis. Authors in [1] prepared the comparison of various ResNet architectures: ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152 providing the highest F1-score=94%. In [2] Narin A. et al. implemented some CNN-based transfer learning models like: ResNet50, ResNet101, ResNet152, InceptionV3, and Inception-ResNetV2. The ResNet50 architecture was the most promising that provided the accuracy over 96%. Bargshady G. et al. [3] used Inception V3 for Xray images analysis. They proposed a novel approach to the dataset augmentation based on GAN. Even though it seemed to be very computationally demanding, it enabled to reach the accuracy exceeding 94%. In [4] the role of segmentation was emphasized. Authors proposed COVID-19 Lung segmentation network (CLSeg) inspired by the popular U-Net architecture, consisting the encoder and decoder parts. They proved that segmentation is a crucial step using the heat maps.

3. Proposed method

Figure 1 presents all steps of the proposed pipeline: data, augmentation, preprocessing, and classification. Finally, the method gives the answer of *'true'* for the COVID-19-positive sample and *'false'* for the healthy sample.

In the research we used the dataset `https://www.kaggle.com/datasets/andyczhao/covidx-cxr2` described in [5] for training and validating. For evaluation we used our own dataset available at `https://github.com/UTP-WTIiE/Xray_data.git` which was previously described in [6]. We used also a novel architecture based on two ResNet networks. The first one performs segmentation and is a model pre-trained for lung segmentation whereas the second performs the classification. The parameters for training classification model, ResNet18, were: optimizer – SGD (stochastic gradient descent), loss function – cross entropy, 200 epochs, and batch size 16. Whole experiments were conducted using the PyTorch library.

The augmentation is the key point of this article. We proposed and assessed 5 different approaches to augmentation: 1. **None** – no augmentation methods were used in the baseline approach. It was essential since we want to research the influence of augmentation on the model's prediction; 2. **Blurring** – slightly blurring the images in order to reduce noises and get rid of some unwanted details: motion blur, median blur and Gaussian blur; 3. **Contrast and brightness** – manipulating the contrast and brightness of the image: CLAHE (Contrast Limited Adaptive Histogram Equalization), random brightness modification, random con-

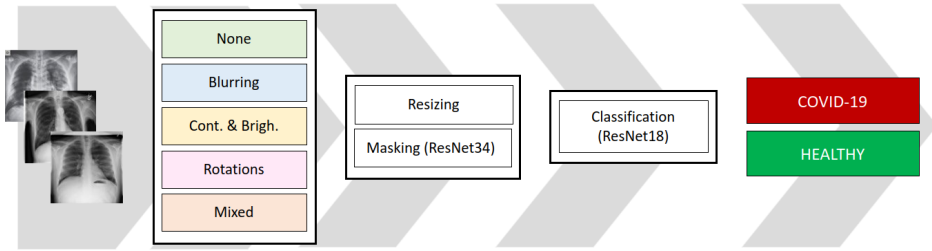


Figure 1. The pipeline of the proposed system (from left side): data used in the research, augmentation, pre-processing (resize and masking by ResNet34), classification (ResNet18) and the final result: COVID-19/Healthy. Source: own work.

trast modification, sharpening; 4. **Rotations** – rotating an image by a fixed angle to simulate different orientations of the image; images were rotated by angle in range $< -3, 3 >$ expressed in degrees; 5. **Mixed** – a mix of all mentioned augmentation methods.

4. Evaluation and results

Each model in this research was evaluated using four validation metrics as follows: Accuracy (Eq. 1), Precision (Eq. 2), Recall (Eq. 3), and F1-score (Eq. 4), which use the below mentioned measures TP – *true positives* – COVID-19 patient classified as sick, FP – *false positives* – healthy patient classified as sick, FN – *false negatives* – COVID-19 patient classified as healthy, and TN – *true negatives* – healthy patient classified as healthy. These metrics can be treated as a golden standard in ML-based studies. They can help also in comparison of the proposed method to the state-of-the-are results. The obtained results were presented in Table 1. The most promising results were highlighted in bold.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{F1-score} = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{4}$$

Table 1. The results obtained for the selected augmentation methods.

Augmentation	Accuracy	Precision	Recall	F1-score
None	0.9032	0.9063	0.9063	0.9063
Blurring	0.9516	0.9677	0.9375	0.9524
Contrast and Brightness	0.9355	0.9667	0.9063	0.935
Rotations	0.9516	0.9677	0.9375	0.9524
Mixed	**0.9839**	**0.9697**	**1.0000**	**0.9846**

5. Conclusions

In this paper we proposed the baseline architecture for lung Xray images-based COVID-19 detection using two ResNet networks. We presented and examined also some powerful augmentation techniques. In the proposed schema the most promising approach was to join all described types of augmentation and to implement them together. Obviously, the proposed schema can be extended in the future. It is possible to add some additional augmentations and, what seems to be more important, to add some explainability. It could help the medics in understanding how does the ML-based system really work.

References

[1] Showkat S., Qureshi S., *Efficacy of transfer learning-based resnet models in chest x-ray image classification for detecting covid-19 pneumonia*, *Chemometrics and Intelligent Laboratory Systems*, 2022, vol. 224, p. 104534.

[2] Narin A., Kaya C., Pamuk Z., *Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks*, *Pattern Analysis and Applications*, 2021, vol. 24, pp. 1207–1220.

[3] Bargshady G., Zhou X., Barua P.D., Gururajan R., Li Y., Acharya U.R., *Application of cyclegan and transfer learning techniques for automated detection of covid-19 using x-ray images*, *Pattern Recognition Letters*, 2022, vol. 153, pp. 67–74.

[4] Zhao H., Fang Z., Ren J., MacLellan C., Xia Y., Li S., Sun M., Ren K., *Sc2net: A novel segmentation-based classification network for detection of covid-19 in chest x-ray images*, *IEEE Journal of Biomedical and Health Informatics*, 2022, vol. 26, no 8, pp. 4032–4043.

[5] Wang L., Lin Z.Q., Wong A., *Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images*,

Scientific Reports, 2020, vol. 10, no 1, p. 19549, ISSN 2045-2322, doi: 10.1038/s41598-020-76550-z.

[6] Giełczyk A., Marciniak A., Tarczewska M., Kloska S.M., Harmoza A., Serafin Z., Woźniak M., *A novel lightweight approach to covid-19 diagnostics based on chest x-ray images*, *Journal of Clinical Medicine*, 2022, vol. 11, no 19, doi: 10.3390/jcm11195501.

On Parameters of Migration in PEA Computing

Sylwia Biełaszek[0000−0001−8947−6272],
Aleksander Byrski[0000−0001−6317−7012]

AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Krakow, Poland
bielsyl@agh.edu.pl, olekb@agh.edu.pl

DOI:10.34658/9788366741928.78

Abstract.

Metaheuristics, such as evolutionary algorithms have been proven to be (also theoretically, see works of Vose [1]) universal optimization methods. Skolicki and DeJong [2] researched impact of migration intervals on island models. In this article, we explore different migration intervals and amounts of migrating indyviduals, complementing Skolicki and DeJong's research. In our experiments we use different ways of selecting migrants and pave the way for further research, e.g. involving different topologies and neighborhoods. Besides sketching out the background and presenting the idea of the algorithm we show the experimental results and discuss them in detail.
Keywords: *parallel evolutionary computing, metaheuristics, migration*

1. Introduction

In our research, we investigated the operation of the island model of evolutionary algorithms on two problems: Rastrigin i Sphere (De Jong's function) on dimention=200 both, using three versions of selection strategies on the source island: 'best', 'max distance' and for comparision: 'random' strategy. We tested the performance of four different migration intervals and four different numbers of migrants in six experimental setups.

2. Preliminary results

We obtained the highest improvement for the Sphere problem with selection of migrants 'best' strategy. while for the Rastrigin problem, the most cases of improvement were for 'max distance' and the greatest improvement in terms of value for 'best' strategy. We can see Sphere results on Fig. 1 (right). The results of one island model is showed as red bar, five island results are green.

		number of migrants			
		1	4	9	12
migration interval	5	4	5	0	0
	15	3	4	4	2
	25	0	2	1	1
	35	0	1	0	0

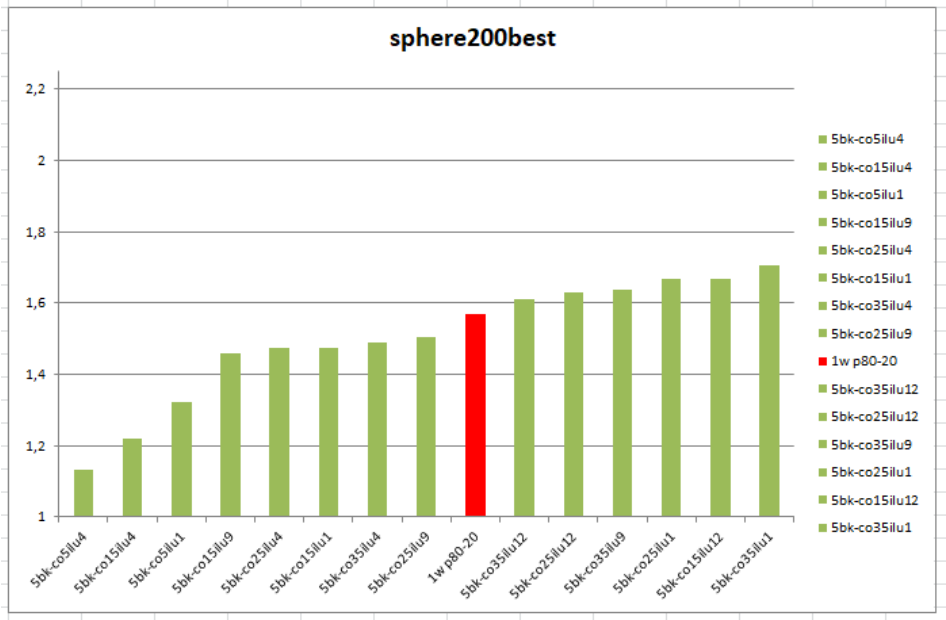


Figure 1. Top: Comparative: number of improvements in six five-island cases in relation to the score of one-island, obtained for the examined intervals and number of migrants, Bottom: Improvement in case of Sphere problem 'best' strategy. Source: own work.

3. Conclusion

Experimental results for the vast majority of experiments showed promising results for our 5-island settings compared to the single-island. Only using a random strategy of selecting migrants, the improvement was almost invisible. In addition, on Fig. 1 (left), we observed an improvement in the results with a small, but not too small, both the migration interval and the number of emigrants. In conclusion, it is worth looking for more accurate relationships between the above-mentioned parameters, topology and population size in the island model in order to obtain efficient versions of large-population evolutionary algorithms.

Acknowledgements

Presented research has been financially supported by Polish National Science Center Grant no. 2019/35/O/ST6/00571 "Parallelization of metaheuristics with desynchronization."

References

[1] Davis L.D., De Jong K., Vose M.D., Whitley L.D., Miller W. (eds.), *Evolutionary Algorithms*, *The IMA Volumes in Mathematics and its Applications*, vol. 111, Springer, New York, NY, 1999, doi: 10.1007/978-1-4612-1542-4.

[2] Skolicki Z., De Jong K., *The influence of migration intervals on island models*, [In:] *Genetic and Evolutionary Computation Conference, GECCO, Washington D.C. 2005*, pp. 1295–1302, doi: 10.1145/1068009.1068219.

On the Importance of the RGB-D Sensor Model in the CNN-based Robotic Perception

Mikołaj Zieliński[1][0000−0003−1107−1149], **Dominik Belter**[2][0000−0003−3002−9747]

[1]*Poznan University of Technology*
Institute of Robotics and Machine Intelligence
ul. Piotrowo 3A, 60-965 Poznań, Poland

DOI:10.34658/9788366741928.79

Abstract. *Mobile and manipulation robots operating indoors use RGB-D cameras as the environment perception sensors. To process data from RGB and depth cameras neural networks are applied. These neural-based systems are trained using synthetic datasets due to the difficulties of obtaining ground truth data on real robots. As a result, the neural model used on the real robot does not produce satisfactory performance due to the differences between the images used during training and the inference. In this paper, we show the importance of depth sensor modeling while training the neural network on a synthetic dataset. We show that the obtained neural model can be used on the real robot and process the data from the real RGB-D camera.*
Keywords: *robotics, RGB-D camera, neural perception*

1. Introduction

Most of the neural-based systems used in robotics are trained on perfect synthetic depth images and do not take into account the sensor noise [1, 2]. As a result, the obtained neural models do not perform well on the data from the real sensors. Rarely, the methods trained on the synthetic datasets consider depth sensor properties when generating a dataset [3, 4] but they focus on relatively old Kinect 360 and Intel RealSense cameras [5]. Nowadays, more accurate and reliable Kinect Azure cameras are available that have different measurement principles and noise properties [6]. In this paper, we propose the model of the Kinect Azure camera. We show the influence of the sensor's modeling on the improved performance of a neural network trained on the dataset augmented by the proposed camera model.

In this research, we train the neural network to reconstruct occluded objects observed on the scene by a mobile-manipulating robot. We provide the RGB-D images of the scene on the input of the neural network, and we train the neural network to remove objects that occlude the object that is related to the task performed by the robot. The occluding object removal aims to improve the performance of

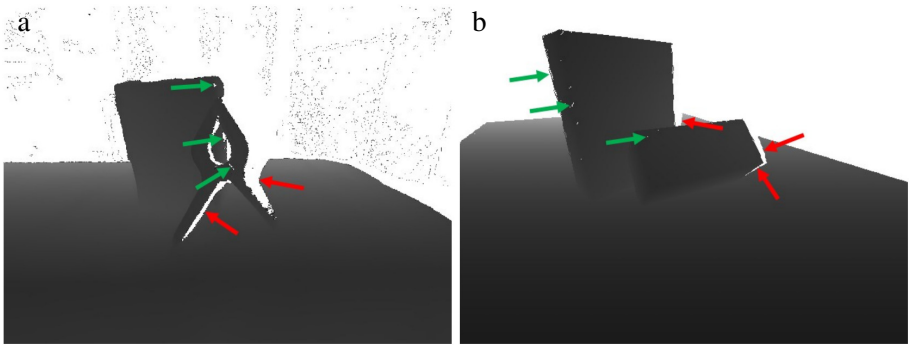


Figure 1. Difference between the depth image from the Kinect Azure (a) and the synthetic image obtained using the proposed model of the depth camera (b). Green arrows show the missing depth measurements on the surface of the object and the red arrows show the missing measurements on the edges of the object. Source: own work.

objects grasping in cluttered scenes [2] and pose estimation during in-hand manipulation. The proposed architecture utilizes a cascade of three separate networks. Each network is based on a U-net structure with the ResNet encoder and is trained separately to focus on different reconstruction tasks. To show the performance of the proposed sensor model, we compare the neural networks trained on the synthetic dataset using "perfect images" and images that contain features introduced by the proposed Kinect Azure model.

2. Sensor model

The Kinect Azure uses amplitude modulation phase-shift-based indirect TOF distance measurement. This method is more accurate than the method used on the Kinect 360 that utilizes an infrared projected pattern and matches the features detected on the infrared images [6]. The difference is well visible on the edges of the objects where the older version of the Kinect sensor does not provide measurements. The same phenomenon exists in the images from the Kinect Azure camera but the source of this phenomenon is different. During training the neural network, we use calibrated RGB and depth images that are aligned to each other. The Kinect Azure projects the depth measurements on the RGB image that is shifted with respect to the depth sensor. As a result, we can observe missing data on the right side of the objects. In Fig. 1, we show the images from the Kinect Azure to illustrate the areas with missing depth data.

During the experiments, we noticed that sensor noise, which causes inaccurate depth pixel values, plays a minor role in the performance of the neural network.

Table 1. Mean error of the occluded object reconstruction [mm] obtained with the neural network trained on the raw synthetic images (baseline) and images augmented with the proposed camera model.

baseline	**with the camera model**
29.8	24.8

Moreover, the sensor noise can be easily modeled during data augmentation. To model the main property of the Kinect Azure, we project the depth pixels $\mathbf{d}$ from the depth camera to the position of the RGB camera $\mathbf{T}_{\text{RGB}}$ and then back to the position of the depth camera. To this end, we have used the pinhole camera model and extrinsic parameters obtained during intrinsic camera calibration $\mathbf{K}$:

$$\mathbf{d}_{\text{i}}^{\text{RGB}} = C^{-1}(\mathbf{T}_{\text{RGB}}^{-1} \cdot \mathbf{K} \cdot \mathbf{d}_i), \tag{1}$$

$$\mathbf{d}_i = \mathbf{K}^{-1} \cdot \mathbf{T}_{\text{RGB}} \cdot C \cdot \mathbf{d}_{\text{i}}^{\text{RGB}}, \tag{2}$$

where $\mathbf{d}_{\text{i}}^{\text{RGB}}$ is the i-th pixel of the depth image projected on the position of the RGB camera and C^{-1} is the inverse pinhole camera model. The proposed procedure given by (1) and (2) adds the effect of missing depth measurements on the surface and the right edges of the objects.

3. Results

To evaluate the scene reconstruction, we compute the MAE between the set of 100 references Kinect Azure images with the removed object and the output from the CNN. The quantitative results are presented in Tab. 1. The obtained results show that when the proposed sensor model is used, the scene reconstruction error is reduced by 16.8% for real sensor data. When the objects are trained and evaluated on synthetic images, the reconstruction error is equal to 17.0 mm and 22.5 mm for the models trained on the raw synthetic images and images generated with the proposed camera model, respectively.

In Fig. 2, we show the example scene reconstructed by the trained neural network. The proposed method improves the depth image computed by the neural network (Fig. 2c) when compared to the results obtained when perfect RGB-D images are used for training the neural network (Fig. 2b).

4. Conclusions

In this paper, we propose the model of the Kinect Azure camera. The presented experiments show that the inference results are improved by training the

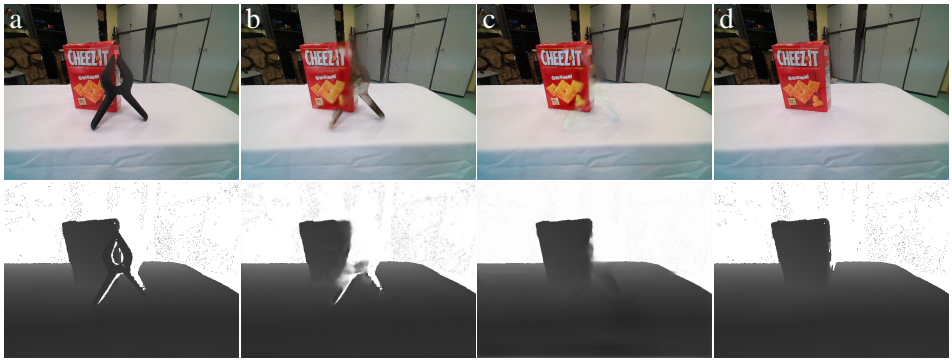


Figure 2. Example results of the occluding object removing method with the CNN trained on the perfect synthetic data (b) and with the proposed sensor model (c): input RGB-D images to the neural network (a), and the reference output (d). Source: own work.

neural network on the data augmented by the proposed camera model. We present quantitative results on a real sensor to show that considering the sensor model during training improves scene reconstruction results. The proposed model of the camera improves the accuracy by 16.8%. We also demonstrate the example reconstruction results that show that the proposed model enhances the reconstruction of RGB and depth images.

In the future, we are going to propose and evaluate the influence of other popular RGB-D camera models.

Acknowledgment

The work was supported by the National Science Centre, Poland, under research project no UMO-2019/35/D/ST6/03959.

References

[1] Park J.J., Florence P., Straub J., Newcombe R., Lovegrove S., *DeepSDF: Learning continuous signed distance functions for shape representation*, [In:] *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019*, pp. 165–174.

[2] Staszak R., Kulecki B., Sempruch W., Belter D., *What's on the other side? a single-view 3D scene reconstruction*, [In:] *2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2022*, pp. 173–180.

[3] Papon J., Schoeler M., *Semantic pose using deep networks trained on synthetic RGB-D*, [In:] *2015 IEEE International Conference on Computer Vision (ICCV), 2015*, pp. 774–782.

[4] Eitel A., Springenberg J.T., Spinello L., Riedmiller M., Burgard W., *Multimodal deep learning for robust rgb-d object recognition*, [In:] *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015*, pp. 681–687.

[5] Handa A., Whelan T., McDonald J., Davison A.J., *A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM*, [In:] *2014 IEEE International Conference on Robotics and Automation (ICRA), 2014*, pp. 1524–1531, doi: 10.1109/ICRA.2014.6907054.

[6] Tölgyessy M., Dekan M., Chovanec L., Hubinsky P., *Evaluation of the Azure Kinect and its comparison to Kinect v1 and Kinect v2*, *Sensors*, 2021, vol. 21, no 2, ISSN 1424-8220.

On the Selection of a Machine Learning model in TinyML Devices – Preliminary Study

Tobiasz Puślecki, Krzysztof Walkowiak

Wrocław University of Science and Technology
Department of Systems and Computer Networks
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
{tobiasz.puslecki, krzysztof.walkowiak}@pwr.edu.pl

DOI:10.34658/9788366741928.80

Abstract. *The expected development of TinyML-related technologies will necessitate the development of methods for efficient use of energy resources. In this article, we present preliminary study of machine learning (ML) model selection in TinyML devices in order to reach a tradeoff between accuracy and energy consumption. We study various use cases with different ML models. Our research shows that the presented method can improve the TinyML system in terms of operation time at the cost of slightly lower accuracy.*
Keywords: *TinyML, Energy Consumption, Model Selection*

1. Introduction

TinyML is a generic term for fast growing field of machine learning technologies and applications on devices with limited resources. TinyML including hardware, algorithms and software capable of performing on-device sensor data analytics at extremely low power, typically in the mW range and below, and hence enabling a variety of always-on use-cases and targeting battery operated devices [1]. TinyML requires special techniques and tools to create and train ML models that are optimized for size and performance. These models usually have to be greatly simplified compared to ML models designed to run on PCs or servers, because devices with limited resources cannot handle complex models [2].

Reducing the ML model saves energy and fits the model on the board, which is important due to aggressive energy and memory constraints. The same model on different platforms can consume different amounts of electricity [3] [4]. There are use cases where it is possible to specify models for which a larger model is also more effective [5]. This assumption implies also that more efficient model consumes more energy, so selection of models is crucial. In this paper, we present a preliminary study of ML model selection in TinyML power-constrained devices in order to reach a tradeoff between accuracy and energy consumption.

The rest of the paper is organized as follows. Section 2 describes conducted experiments. Section 3 concludes the work.

2. Experiments

The experimental platform is the Arduino Nano 33 BLE Sense with the nRF5 2480 chip, equipped with a Cortex-M4 core clocked at 64 MHz and 256 KB of RAM, 1 MB of Flash memory. All tested ML models are run with exactly the same data (collected earlier) as other models to eliminate external factors and test the models under exactly the same conditions. We assume the availability of a set of models that can be switched. Models can be loaded from non-volatile memory or updated Over-the-Air [6] [7].

It was decided to choose three use cases for the study:

- recognition of gestures made by waving a magic wand [8] from the book Author's database [9] (motion data, labeled as *Magic_X*) – multiclass (10 classes) classfication using convolutional neural networks (test models have successively 3, 2 and 3 convolution layers of different widths) measured by accuracy metric,
- prediction of the sine function with 10% noise added [8] (synthetic case, labeled as *Sinus_X*) a hundred times in one cycle – regression using dense neural networks (test models have successively 3, 3 and 2 dense layers of different widths) measured by MAE (Mean Absolute Error) metric,
- handwriting recognition from the MNIST database [10] (visual data, labeled as *MNIST_X*) – multiclass classfication using dense neural networks (test models have 2 dense layers of different widths) measured by accuracy metric.

All models were implemented in the TensorFlow library (based on delivered library examples), converted to TensorFlow Lite models and implemented on the board using the Arduino library in the Arduino IDE.

The original models for each use case were modified to simplify them by removing layers. Various layers were experimentally removed/modified, verifying that the neural network retains its core classification/regression capability. For the *Magic* use case, the filters parameter and the number of convolution layers were changed – [16, 32, 64], [16, 32], [16, 16, 32]. For the *Sinus-100* use case, the neurons parameter and the number of dense layers were changed – [100, 100], [25, 25], [12]. For the *MNIST* use case, the parameter of neurons was changed – [100], [50]. Table 1 contains measurements of fundamental metrics for particular models depending on the use case. Accuracy indicates the accuracy of the model on the test dataset, latency indicates the average time for a single inference, current indicates the current drawn by the system during inference, and size indicates the size of the header file that contains the converted and quantized TensorFlow Lite model. The measurements given are averages of 10k executions.

Table 1. Measurements of fundamental TinyML metrics for particular models depending on the use case

Magic	Accuracy [%]	Latency [us]	Current [mA]	Size [b]
Magic_A	99.6	76435	18.54	31544
Magic_B	99.3	56160	18.15	10264
Magic_C	97.5	43027	17.95	13824
Sinus-100	MAE	Latency [us]	Current [mA]	Size [b]
Sinus_A	0.0882	144230	19.13	13968
Sinus_B	0.0931	29960	17.76	3904
Sinus_C	0.312	7710	16.59	2216
MNIST	Accuracy [%]	Latency [us]	Current [mA]	Size [b]
MNIST_A	97.94	14196	16.78	84904
MNIST_B	97.09	7194	16.20	43520

We assume that the analyzed TinyML system uses a 18650 type battery with a capacity of 2500mAh, nominal voltage of 5.124V (measured by *USB Tester Voltmeter UM25C*) and a discharge safety (the percentage of battery capacity that is never used) of 20% under normal conditions. Table 2 presents operation time and accuracies for use cases depending on the selection strategy. In the *Best accuracy* strategy we always choose the most accurate model. In *Lowest latency* strategy we always choose the fastest model. In turn, in the *Lowest energy consumption* strategy, we always select a model that consumes the least current.

Table 2. Operation time and accuracy of models depending on selection strategy

	Model/Operation time[hrs]/Evaluation metric		
	Best accuracy	**Lowest latency**	**Low. ener. cons.**
Magic	A/107.87/99.6%	C/111.42/97.5%	C/111.42/97.5%
Sinus-100	A/104.55/0.0882	C/120.55/0.312	C/120.55/0.312
MNIST	A/119.2/97.94%	B/123.46/97.09%	B/123.46/97.09%

Let's assume the order of ML models from the best to the worst (*_A*, *_B*, *_C*). They can be changed by taking into account the percentage of the battery charge. Steps are defined as a percentage threshold of battery charge beyond which you the model is changed and a weaker model is used for inference. Table 3 presents opperation time and weighted accuracies for use cases depending on the selected steps. For example, for the use case *Magic* and *steps* = {66, 33} – Model A is used at first, but when the battery level drops below 66%, then model B is used, and so on. In other words, as the battery discharges, we choose a weaker model that is slightly less effective but will increase the operating time. Of course, the order of

the models can be changed as needed. For example, accuracy may increase over time instead of decreasing if models are in (_*C*, _*B*, _*A*) order instead of (_*A*, _*B*, _*C*) order.

Table 3. Operation time and weighted accuracies for use cases depending on the selected steps

	Operation time [hrs]/Evaluation metric		
	steps={66, 33}	**steps={83, 41}**	**steps={99, 49}**
Magic	109.79/98.8%	110.30/98.61%	110.78/98.42%
Sinus-100	112.53/0.164	114.50/0.182	116.43/0.200
	steps={75}	**steps={50}**	**steps={25}**
MNIST	122.40/97.30%	121.33/97.52%	120.26/97.73%

3. Conclusions

This short article demonstrated that our approach allows to maintain very similar accuracy gaining additional operation time. The use of different steps allows us to try to reach a tradeoff between accuracy and energy consumption. The method presented here can improve the TinyML system in terms of operation time at the cost of slightly less accuracy. In future work, we plan to extend the presented concept with energy harvesting using solar panels and predicting the energy production. This will allow the system to recharge batteries and change tresholds or models in real time. Further work may also include other use cases and models, as well as research into other hardware platforms that are able to meet the strict limitations of TinyML. In addition, we plan to optimize the entire procedure at the meta level, which means defining the model reduction factor as an optimization parameter with respect to the desired running time and acceptable model quality.

References

[1] *TinyML Foundation Organization Homepage*, (access: 25-02-2022). https://www.tinyml.org

[2] Ray P.P., *A review on tinyml: State-of-the-art and prospects*, *Journal of King Saud University – Computer and Information Sciences*, 2022, vol. 34, no 4, pp. 1597–1598, doi: 10.1016/j.jksuci.2021.11.019.

[3] Giordano M., Piccinelli L., Magno M., *Survey and comparison of milliwatts micro controllers for tiny machine learning at the edge*, [In:] *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS), 2022*, pp. 94–97, doi: 10.1109/AICAS54282.2022.9870017.

[4] Piechocki M., Kraft M., Pajchrowski T., Aszkowski P., Pieczynski D., *Efficient people counting in thermal images: The benchmark of resource-constrained hardware*, *IEEE Access*, 2022, vol. 10, pp. 124835–124847, doi: 10.1109/ACCESS.2022.3225233.

[5] Zhang Y., Wijerathne D., Li Z., Mitra T., *Power-performance characterization of tinyml systems*, [In:] *2022 IEEE 40th International Conference on Computer Design (ICCD), 2022*, pp. 644–651, doi: 10.1109/ICCD56317.2022.00099.

[6] Nicolas C., Naila B., Amar R.C., *Energy efficient firmware over the air update for tinyml models in lorawan agricultural networks*, [In:] *2022 32nd International Telecommunication Networks and Applications Conference (ITNAC), 2022*, pp. 21–27, doi: 10.1109/ITNAC55475.2022.9998338.

[7] Sudharsan B., Breslin J.G., Tahir M., Intizar Ali M., Rana O., Dustdar S., Ranjan R., *Ota-tinyml: Over the air deployment of tinyml models and execution on iot devices*, *IEEE Internet Computing*, 2022, vol. 26, no 3, pp. 69–78, doi: 10.1109/MIC.2021.3133552.

[8] Warden P., Situnayake D., *Rendering*, [In:] *TinyML Machine Learning with TensorFlow on Arduino, and Ultra-Low Power Micro-Controllers*, O'Reilly Media, ISBN 9781492052043, 2020, pp. 89–90.

[9] Warden P., *Magic Wand motion database*, 2021, (access: 17-07-2023). https://github.com/petewarden/magic_wand_digit_data

[10] LeCun Y., Cortes C., *MNIST handwritten digit database*, 2010, (access: 17-07-2023). http://yann.lecun.com/exdb/mnist/

Valuing Passes in Actions Leading to the Third Zone on the Pitch with Machine Learning Methods

Mateusz Tylka[1[0000−0003−2270−9494]],
Sebastian Wałęsa [1[0000−0001−8694−2047]],
Kornelia Girejko[1[0000−0001−9729−3863]],
Jakub Kaczmarek[1[0000−0002−4115−2698]],
Bartłomiej Grzelak[2[[0000−0002−6132−651X]],
Tomasz Piłka [1,2[0000−0003−1206−2076]]

[1]*Adam Mickiewicz University in Poznan*
Faculty of Mathematics and Computer Science
{*mattyl*, *jakkac*5, *korgir*, *sebwal*1}*@st.amu.edu.pl*
[2]*KKS Lech Poznań*
Science Department
bartlomiej.grzelak, tomasz.pilka@lechpoznan.pl

DOI:10.34658/9788366741928.81

Abstract. *In football, the ability to make accurate and effective passes to the third zone of the pitch is a key aspect of a team's success. Evaluating these passes can provide valuable information about a team's performance and help coaches and analysts make informed decisions about their tactics and strategies. In this article, we will explore the possibility of using artificial intelligence methods to score passes to the third zone on the field, in comparison to traditional metrics.*
Keywords: *football analytics, player performance, pass values*

1. Introduction

Football is one of the most popular team sports in the world. Matches between top club or national teams attract huge crowds, attract sponsors, and, for the players themselves, are often the target of demanding daily training sessions. It is important for clubs to prepare their players properly and to learn from past matches. To this end, increasingly rich and accurate sets of match data are being collected. We distinguish between event data and tracking data. Event football data describe players' actions with the ball, such as passes, dribbles, interceptions, tackles, and shots. Tracking data provides the exact spatial location of the players

and the ball at all times. Increasingly, artificial intelligence is being used to analyze data. One of the critical elements of football is passing; studies show that teams that pass the ball more accurately are more likely to win matches [1]. Analysis of passing efficiency can therefore provide valuable information about the team's performance and inform the preparation of tactical training before the next matches. It is important to analyze the area of the pitch where passing efficiency is most important. Such a place is the so-called third zone, the end zone, the area of the pitch closest to the opposing team's goal. Actions leading to this zone are crucial in creating scoring opportunities. A successful pass in this zone can lead to a goal. Therefore, analyzing the value of passes leading to zone three can provide valuable information about the effectiveness of a team's attacking strategy.

In this article, we use event data from PKO BP Ekstraklasa matches from the 2020/21 and 2021/22 seasons to show that, by applying machine learning methods, we can propose a method for evaluating passes made in the third zone of the pitch. For this purpose, we propose our metric, the Probability Pass Value (PPV). The metric makes it possible both to evaluate the passes themselves, and the sub-zones in which they are most valuable, and to identify the players who show the highest efficiency in their execution, thus influencing the outcome of the match.

2. Related Work

The use of AI methods in football analysis has gained popularity in recent years due to the large amount of data generated by matches. Machine learning algorithms, such as Random Forests and Support Vector Machines, have been used to analyze football data and identify patterns that can provide insights into team performance [2]. In [3], the authors show that collective actions scored 51.6% of all goals, while individual actions scored 10.5% of goals. In terms of the penultimate action, crosses were more common against organized defenses, while passes in behind the defense or actions such as dribbling or running with the ball had a higher percentage of goals against circumstantial defenses.

The challenges are to determine the optimal number of actions or seconds to look ahead and to measure the similarity between plays. For example, a pass during a slow build-up is likely to have a different value than a pass during a fast counterattack. A third approach is to allocate the reward of a possession sequence (e.g., a goal) starting at a given pitch location to its constituent passes [4]. It's difficult to decide on the optimal weighting scheme to distribute the credit among all the passes in the sequence. Typically, passes at the end of the possession sequence receive more credit than passes at the beginning of the sequence.

3. Data

The experiments conducted in this article used data from the PKO BP Ekstraklasa, polish football league, for the 2020/2021 and 2021/2022 seasons. The data for the experiment comes from sports data provider StatsBomb and includes 240 matches played by 16 teams in the 2020/21 season and 306 matches played by 18 teams in the 2021/22 season. A total of 494260 on-field events were included in the analysis, where each tuple with one event was described on more than 150 features. The data covered events for a total of 754 players.

The data was coded in the Soccer Player Action Description Language (SPADL) format, which represents a match as a sequence of on-the-ball actions $[a_1, a_2, \ldots, a_m]$, where m is the total number of actions that occurred in the match. Each action is a tuple of the same twelve attributes, the description of the attributes and the types of actions in the data are described in the study [5].

4. Machine Learning and football passes metrics

In recent years, the analysis of football matches has moved beyond the statistical analysis of passes and the number and efficiency of shots. Increasingly, machine learning (ML) algorithms are being used to create metrics that evaluate a given situation. The most commonly used metric is Expected Goals (xG), which measures the probability that a shot will result in a goal. Expected Threat (xThreat, xT) is a newer model that estimates the probability of a team scoring a goal based on the current location of the ball and players. Both solutions can be used to assess the quality of passes that lead to shots on goal.

Developments in data collection methods and the ability to apply computational resources are leading to the creation of new metrics that describe events during a match. These are created by data providers, such as the On-Ball-Value (OBV) proposed by StatsBomb [6], or by solutions created in research centers.

An important solution is Valuing Actions by Estimating Probabilities (VAEP) [7, 5], a framework for valuing actions in a football game. Unlike most existing work, it considers all types of actions (e.g. passes, crosses, dribbles, and shots) and takes into account the context of action as well as its possible longer-term effects. Intuitively, an action value reflects the expected impact of the action on the outcome of a match.

To develop the VAEP values, the **XGBoost** (Extreme Gradient Boosting) algorithm was used as a predictive method. XGBoost is a machine learning library used for regression, classification, and ranking problems [8]. The result was obtained by training using data from the first season and then was used to evaluate activities in the 2021/2022 season. Overall, these experiments aimed to evaluate football actions using advanced metrics that take into account various factors, such

as the location and type of action, to gain deeper insights into the quality of play. By utilizing advanced data analysis techniques, researchers and coaches can better understand and optimize team performance.

This study also applied XGBoost, however, to different attributes than VAEP. It is used to evaluate metrics and find key features of passes to the third zone. Firstly, passes that end up in the third zone were isolated. Then all the gathered data was preprocessed and passed as inputs into XGBoost algorithm. They used 0 or 1 as labels, where 0 means an unsuccessful pass and 1 is a pass with a successful result.

5. Methodology

The first step was dividing pitch into zones as shown in Figure 1. The third zone contains central penalty (CP), left penalty (LP), right penalty (RP), central outside (CO), left outside (LO), and right outside (RO). Areas near the corners of the pitch are the left zone (LZ) and right zone (RZ).

Firstly, passes to the third zone were isolated and divided into two groups. Proposal 1 contains passes from the second zone to the third zone. Proposal 2 includes passes that started in the left or right zone and ended in the third zone. Then it was processed and put into a data frame that contains information about passes and zones.

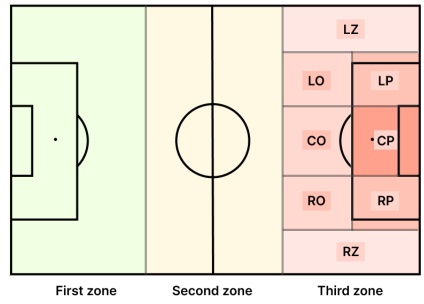
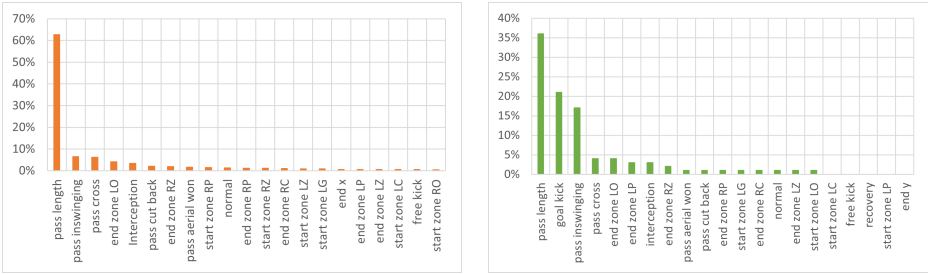


Figure 1: Pitch division. Source: own work.

The training was performed with XGBoost and then predicted the outcome of passes from testing data. Data consisted of 49 attributes (e.g. corner, free kick, goal kick, interception, kick-off, recovery, throw-in, under pressure, the start and end of the pass in each zone, features related to the pass (cross, angle, ground, etc.)). The XGBoost probability of positive pass outcome was used to generate the value of our metric **PPV** – *Probability Pass Value*. The impact of features on the outcome was evaluated and is shown in Figure 2. We interpret this metric as the degree of contribution of the pass to the successive action in the third zone.

6. Results

After predicting outcome of passes for testing data accuracy was equal to 98 percent. We used feature_importances function from XGBoost which outcome is shown in figure 2. This function returns values for features indicating which



(a) Proposal 1 (b) Proposal 2

Figure 2: Features importance. Source: own work.

features had the greatest impact on the prediction results. In conlusion in both proposals the lenght of the pass is the key factor to succesful pass.

In order to compare players from the last season we divided metrics by minutes played in whole season. The results are shown in table 1.

Table 1: Players with highest PPV per 90 minutes for both proposals

Player	Mins	VAEP	OBV	PPV	xT
Proposal 1					
Josué Pesqueira (Legia Warszawa)	2668	0.95	0.15	3.47	0.028
Ivi López (Raków Częstochowa)	2606	0.79	0.1	3.32	0.025
Krystian Getinger (Stal Mielec)	2911	0.69	0.08	2.72	0.012
Dawid Abramowicz (Radomiak Radom)	3267	0.7	0.1	2.37	0.002
Proposal 2					
Josué Pesqueira (Legia Warszawa)	2668	1.17	0.05	2.67	0.029
Filipe Nascimento (Radomiak Radom)	2104	0.94	0.04	2.52	0.005
Tom Hateley (Piast Gliwice)	1844	0.81	0.03	2.29	0.002
Joel Pereira (Lech Poznań)	1959	0.75	0.03	2.21	0.045

The developed method shows that for passes from the indicated zones into the third zone, we should evaluate them differently than what we know from xG, xThreat, or VAEP. The results obtained were reviewed and assessed by people with expertise in the field, who indicated that the values were consistent with their assessment of PKO BP Ekstraklasa players passing in the third zone of the pitch and that the method itself was worthy of further development. In further work, we plan to develop the method by taking into account the position of the players on the pitch. In further work, we plan for this method to be advanced by taking into account the position of players on the field.

7. Conclusion

In conclusion, this article aims to evaluate the value of passes leading to the third zone on the field using AI methods. Analyzing these passes can provide valuable insights into team performance, especially in the attacking phase of the game. By using machine learning algorithms, we can identify patterns in the data and assess the effectiveness of passes leading to a shot on goal and the players who contribute most to such actions. This research can be useful for coaches, analysts, and players to improve team performance and develop effective attacking strategies.

Acknowledgment

The authors would like to thank the KKS Lech Poznań club for providing data for the study and their support during the study.

References

[1] Lago-Peñas C., Dellal A., *Ball possession strategies in elite soccer according to the evolution of the match-score: the influence of situational variables*, *Journal of Human Kinetics*, 2010, vol. 25, no 2010, pp. 93–100, doi: doi:10.2478/v10078-010-0036-z.

[2] Bialkowski A., Lucey P., Carr P., Yue Y., Sridharan S., Matthews I., *Identifying team style in soccer using formations learned from spatiotemporal tracking data*, [In:] *2014 IEEE International Conference on Data Mining Workshop*, pp. 9–14, doi: 10.1109/ICDMW.2014.167.

[3] González-Ródenas J., López-Bondia I., Aranda-Malavés R., Desantes A.T., Sanz-Ramírez E., *Technical, tactical and spatial indicators related to goal scoring in european elite soccer*, 2020, doi: 10.14198/jhse.2020.151.17.

[4] Brooks J., Kerr M., Guttag J., *Using machine learning to draw inferences from pass location data in soccer*, *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 2016, vol. 9, no 5, pp. 338–349, doi: 10.1002/sam.11318.

[5] Decroos T., Bransen L., Van Haaren J., Davis J., *Actions speak louder than goals: Valuing player actions in soccer*, [In:] *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 1851–1861, doi: 10.1145/3292500.3330758.

[6] StatsBomb, *Introducing on-ball value (OBV)*, (access: 17-07-2023). https://statsbomb.com/articles/soccer/introducing-on-ball-value-obv/

[7] Van Roy m., Robberechts p., Decroos T., Davis J., *Valuing on-the-ball actions in soccer: A critical comparison of xT and VAEP*, [In:] *Computer Science 2020*, AI in Team Sports Organising Committee, corpusID: 226289375.

[8] *XGBoost documentation*, (access: 17-07-2023).https:/xgboost.readthedocs.io/en/stable/

ISBN 978-83-66741-92-8