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.

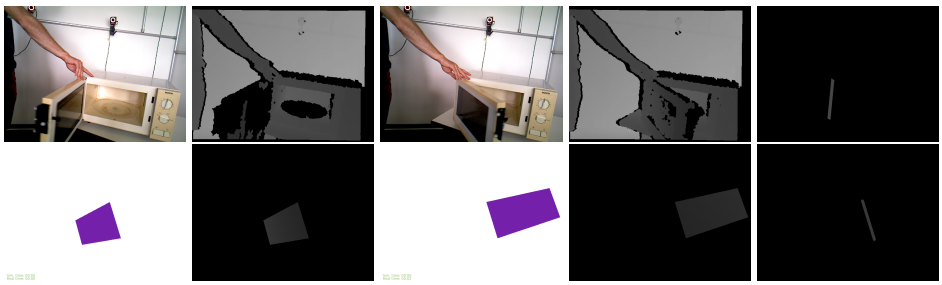


Figure 1: Example RGB-D pairs from the RBO (top) and synthetic (bottom) datasets. From left to right: RGB image of the first position, its depth image, RGB image of the second position, its depth image, and the axis of rotation. Source: own work.

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

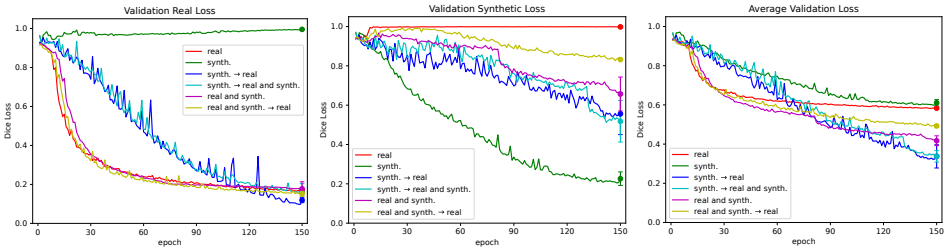
real and synth. – the CNN is trained on the mixture of synthetic and real images

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.