

Synergy of Convolutional Neural Networks and Geometric Active Contours

Arkadiusz Tomczyk^[0000-0001-9840-6209],
Oleksandr Pankiv^[0000-0002-5062-5813],
Piotr S. Szczepaniak^[0000-0002-9973-0673]

Lodz University of Technology
Institute of Information Technology
Wólczańska 215, 90-924 Łódź, Poland

Abstract. *Hybrid approach to machine learning techniques could potentially provide improvements in image segmentation results. In this paper, a model of cooperation of convolutional neural networks and geometric active contours is proposed and developed. The novelty of the approach lies in combining deep neural networks and active contour model in order to improve CNN output results. The method is examined on the image segmentation task and applied to the detection and extraction of nuclei of HL60 cell line. The model had been tested on both 2-D and 3-D images. Because of feature learning characteristics of convolutional neural networks, the proposed solution should perform well in multiple scenarios and can be considered generic.*

Keywords: *convolutional neural networks (CNN), geometric active contours (GAC), image segmentation, biomedical applications, machine learning.*

1. Introduction

Accurate image segmentation methods are of paramount importance for a wide range of applications such as evaluating digitized pathologic specimens [1], plant recognition [2] and autonomous driving. As a result, automating image segmentation processes became an active research field.

Process, during which every pixel of the image is labelled based on some feature or belonging to a particular logical group is usually called image segmentation. The most basic example of it is thresholding [3], which effectively labels every pixel as one of two groups based on, for example, the brightness of the given element. Many more sophisticated methods can be used for image segmentation,

such as Convolutional Neural Networks [4], Active Contours [5, 6, 7], and others [8].

Convolutional Neural Networks are a specialized type of neural networks, used mostly for data of shape of N -dimensional grid, where at least one layer uses convolution operation during propagation. Layer mentioned above is generally a set of filters trained by a network. When propagating forward through such layer, as the output we get a set of activation maps, indicating responses of each filter for every pixel of the input. Training of such layer is based on modifying filters in it so the activation would occur in the regions of interest. Convolutional Neural Networks are widely used for solving various classes of problems, one of which is image segmentation.

Another possible group of methods that can be used for image segmentation are Active Contour Models. Kass introduced the first model of this group called snakes [9] in 1988. Since then, many researchers are working on various methods of extracting object contour from images. In [10], authors propose a fast algorithm for one of such approaches, which grew quite popular over time. The library [2] used in the following research adopts a variation of this approach in its core, which results in high performance.

Most of the studies focus on perfecting a single approach or model to achieve a more satisfying result. In this paper, we propose a slightly unconventional approach, based on the synergy of Convolutional Neural Networks and Geometric Active Contour model [5]. Such an approach allows utilizing smoothness and continuity of resulting contour provided by the Geometric Active Contour. In the training stage, Convolutional Neural Network is trained to detect, in one variant, certain regions of interest, in other – boundaries of these regions. This approach allows testing both region-based and edge-based models. During testing, the output of the aforementioned neural network becomes an input for a segmentation based on Geometric Active Contour model. The resulting output is a monochromatic image where regions of interest are marked with white colour. For experiments, we chose dataset [1] containing 3-D scans of the HL60 cell line.

The paper is organized as follows. The second section describes the structure of Convolutional Neural Networks prepared and the process of their training, the third section contains a description of Geometric Active Contour library we used. Section four describes the proposed mechanism of combining Convolutional Neural Networks with Geometric Active Contours. The last, fifth section contains experimental results and discussion.

2. Convolutional Neural Networks

For image segmentation, we use encoder-decoder [11, 12] Convolutional Neural Network architecture. Following approach enables us to perform the first segmentation step without changing the image resolution and as a result – losing details vital in the second step. Example of the architecture mentioned above is demonstrated on Figure 1.

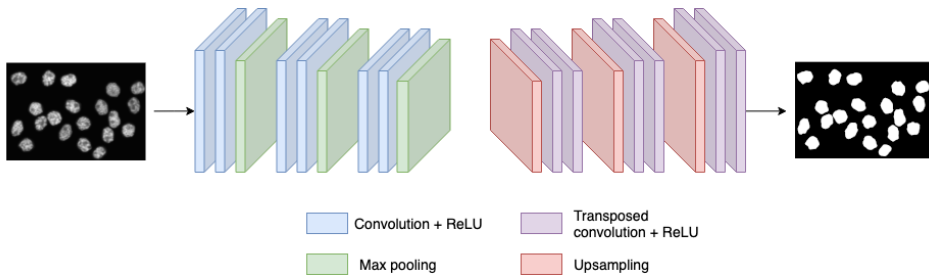


Figure 1. Example encoder-decoder CNN architecture

The entire encoder-decoder network can logically be split into two halves: encoder part, containing Convolutional and Pooling layers, and decoder part containing Transposed Convolutional and Upsampling layers.

In this work, we consider two approaches to interpreting input data. The first approach ignores the fact that dataset consists of 3-D images and considers every slice as a single independent image. The second approach extends this logic to analysing sets of spatially connected slices to give Convolutional Network more logical background of neighbouring data. With that in mind, we prepared two neural networks.

First neural network implemented was designed to process 2-D slices of scan one by one and contained 18 layers: 9 encoder layers and 9 decoder layers. Encoder half contained three logical blocks of layers. Every block had two convolutional layers and one max-pooling layer. The decoder is divided into three logical blocks as well, although the order of layers in single block and order of blocks is opposite to encoder. In the decoder, a block contains one upsampling layer followed by two deconvolution layers.

The second neural network, designed for processing the number of neighbouring slices of 3-D scans had 12 layers: 6 encoder and 6 decoder layers, respectively. The logical structure of the following neural network mostly resembles the previously discussed layout, with one minor difference: all the operations inside layers are now operating on 3-D matrices. The number of layers has been reduced in favour of performance.

In both cases, convolutional layers use a leaky rectified linear unit that can be expressed with function from Equation 1:

$$f(x) = \frac{1}{2} \cdot (1.0 + leak) \cdot x + \frac{1}{2} \cdot (1.0 - leak) \cdot |x| \quad (1)$$

where x is a numerical value of a pixel and $leak = 0.2$. Different variations of this function are widely used in encoder-decoder architectures.

In the following publication, we compare two approaches to training. For the input image on Figure 2 expected output in one training strategy is the edge of cell nuclei (Figure 3), in other – whole region of interest (Figure 4).

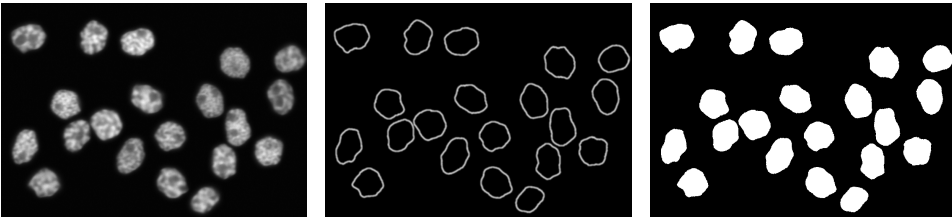


Figure 2. Example input image

Figure 3. Output for edge objective

Figure 4. Output for region objective

Such an approach to training allowed us to test both region-based and edge-based Geometric Active Contour Models. For edge-based models, we used outputs from the network trained to find borders of cell nuclei. For the region-based approach – outputs containing cell nuclei marked as white regions. Such a strategy allowed us to test different approaches and analyse the difference between them.

3. Geometric Active Contours

Geometric Active Contour, in general, is a model of a deformable spline, used for delineating an object outline on an image. In this article, we use the library named OFELI (as an acronym for Open, Fast and Efficient Level set Implementation) which uses a modification of Shi-Karl fast algorithm for level-set curves [2]. The following library is implemented in C++ programming language and provides us with:

- Abstract implementation of Shi-Karl algorithm [10].
- Chan-Vese [13] model implementation (region-based);
- Geodesic model [5] implementation (edge-based).

Unlike snakes, level-set curves are represented by higher-dimension function Φ defined for the whole image. For example, in the case of 2-D image function Φ would be a 3-D function, and curve would be represented with her zero level:

$$\Gamma = \{(x, y) \mid \Phi(x, y) = 0\} \quad (2)$$

This approach allows detecting multiple objects as a curve is just a zero level of a function Γ , and its form and topology are not limited in any way.

4. Cooperation Mechanism of CNN and GAC

Both Convolutional Neural Network and Geometric Active Contour can be considered a rather typical solution for image segmentation problem. Nevertheless, none of them is a perfect choice. Convolutional Neural Networks are nowadays widely used in various image-related machine learning tasks. Their main disadvantage is simple: to achieve high precision, they need a vast training dataset, which might be a problem in some cases. For example, in biomedical applications, datasets are usually relatively small and attempt to create a big training dataset would face challenges with both gathering enough data and labelling it by an expert. In the case of Geometric Active Contours, we do not face such issues, as they are not typically trainable. Nevertheless, figuring out precise parameters to use in this model is quite complicated and, in some particular cases, virtually impossible process.

The expected advantage of the idea behind the combination of these models lies in the possibility of supporting each other to obtain more accurate results. Convolutional Neural Network can solve Geometric Active Contour problem of picking the right parameters for a particular group of images. At the same time, Geometric Active Contour can get rid of inaccuracies in the outputs of Convolutional Neural Network.

The proposed approach consists of the two following steps:

1. Convolutional Neural Network is used to find locations or borders of cell nuclei.
2. Geometric Active Contour is processing the resulting image in order to select regions of interest on the image.

In the first step, Convolutional Neural Network gets in one case – 2-D image, and in another case – a subset of neighbouring 2-D "slices" which form a 3-D image. The output is spatially identical to an input image which contains the normalized output of the neural network. Depending on task Convolutional Network

was trained for output contained borders (Figure 3) of cell nuclei or regions (Figure 4) belonging to it.

In the second step, the output image of the neural network was thresholded and passed to the corresponding Geometric Active Contour model. After the evolution of contour pixels inside it were marked with white, and all the others – black colour.

5. Results

The Convolutional Neural Network experiments were performed with Tensorflow framework on GeForce GTX 780 Ti GPU. The training set included 24 3-D images, each containing 128 2-D spatial slices totalling in 3072 2-D images.

One of the challenges during the training process that might be worth mentioning was training Convolutional Neural Network to detect edges of nuclei. Unlike in the case of regions, edges do not occupy a significant amount of pixels on image, and as a result, mean squared error for blank (all pixels are black, no activations) result was not too high which complicated training process. In order to solve this problem, ground truth mask values were multiplied by 10^4 .

To evaluate and compare the results of classical approaches with our idea, we chose popular statistical measures [14]: accuracy (3), precision (4) and recall (5).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$precision = \frac{TP}{TP + FP} \quad (4)$$

$$recall = \frac{TP}{TP + FN} \quad (5)$$

where:

- *TP* (True Positive) – number of pixels correctly classified as belonging to the area of interest;
- *TN* (True Negative) – number of pixels correctly classified as not belonging to the area of interest;
- *FP* (False Positive) – number of pixels falsely classified as belonging to the area of interest;
- *FN* (False Negative) – number of pixels falsely classified as not belonging to the area of interest;

Test data set included six 3-D scans, each containing 128 2-D spatial slices totalling in 768 2-D images. Six approaches have been considered:

1. Region-based Geometric Active Contour model using 2-D slices as input.
2. Edge-based Geometric Active Contour model using 2-D slices as input.
3. 2-D Convolutional Neural Network using 2-D slices as input.
4. 3-D Convolutional Neural Network using groups of spatially connected 2-D slices as input.
5. 2-D Convolutional Neural Network trained to find regions with cell nuclei combined with region-based Geometric Active Contour model using 2-D slices as input.
6. 2-D Convolutional Neural Network trained to find edges of cell nuclei combined with edge-based Geometric Active Contour model using 2-D slices as input.
7. 3-D Convolutional Neural Network trained to find regions with cell nuclei combined with region-based Geometric Active Contour model using groups of spatially connected 2-D slices as input.
8. 3-D Convolutional Neural Network trained to find edges of cell nuclei combined with edge-based Geometric Active Contour model using groups of spatially connected 2-D slices as input.

In order to calculate values for each approach, the resulting images were compared with ground truth images from the data set. Summarized results are presented in Table 1.

Table 1: Summarized results for test set

Segmentation method	precision	recall	accuracy
Region-based GAC	35,39%	88,88%	84,95%
Edge-based GAC	40,75%	83,85%	87,75%
2-D CNN (regions)	79,84%	77,92%	94,15%
3-D CNN (regions)	83,78%	80,31%	96,95%
2-D CNN (regions) + GAC	51,58%	78,63%	91,20%
2-D CNN (edges) + GAC	73,58%	98,73%	96,83%
3-D CNN (regions) + GAC	54,51%	80,43%	92,31%
3-D CNN (edges) + GAC	79,96%	92,26%	97,33%

According to our results, a combination of both 2-D and 3-D Convolutional Neural Networks with region-based Geometric Active Contour model did not improve results. On the other hand, in case of synergy between 3-D Convolutional Neural Networks with edge-based Geometric Active Contour, we see a slight improvement of results. Even though 3-D neural network achieved the highest precision, the same network combined with Geometric Active Contour shows the highest accuracy.

6. Discussion and Conclusion

In this paper, we propose an unconventional method of using Convolutional Neural Networks with Geometric Active Contours. The approach starts with a traditional image segmentation with a neural network which is being improved by Geometric Active Contour model. Comparative experiment results suggest that this approach has a strong potential to improve segmentation accuracy.

Future work may include an extension of the method to cooperate directly with Convolutional Neural Network during the learning process. Another plausible extension lies in combining other types of neural networks with Geometric Active Contours in order to improve experimental results.

References

- [1] Svoboda, D., Kozubek, M., and Stejskal, S. Generation of digital phantoms of cell nuclei and simulation of image formation in 3d image cytometry. *Cytometry Part A*, 75A(6):494–509, 2009.
- [2] Suta, L., Bessy, F., Veja, C., and Vaida, M. F. Active contours: Application to plant recognition. *Intelligent Computer Communication and Processing (ICCP)*, 2012. URL <http://ieeexplore.ieee.org/document/6356183/>.
- [3] Yuheng, S. and Hao, Y. Image segmentation algorithms overview. *CoRR*, abs/1707.02051, 2017. URL <http://arxiv.org/abs/1707.02051>.
- [4] Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016. URL <http://arxiv.org/abs/1606.02147>.
- [5] Caselles, V., Kimmel, R., and Sapiro, G. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

- [6] Tomczyk, A. and Szczepaniak, P. S. Adaptive potential active contours. *Pattern Analysis and Applications*, 14:425–440, 2011.
- [7] Tomczyk, A. Image segmentation using adaptive potential active contours. *Computer Recognition Systems 2 (CORES), Advances in Soft Computing*, 45:148–155, 2007.
- [8] Tomczyk, A. and Szczepaniak, P. S. Adaptive potential active hypercontours. *Artificial Intelligence and Soft Computing - ICAISC 2006*, pages 692–701, 2006.
- [9] Kass, M., Witkin, A., and Terzopoulos, D. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988. ISSN 1573-1405. doi:10.1007/BF00133570. URL <https://doi.org/10.1007/BF00133570>.
- [10] Shi, Y. and Karl, W. C. A real-time algorithm for the approximation of level-set-based curve evolution. *IEEE Transactions on Image Processing*, 17(5):645 – 656, 2008. URL <http://ieeexplore.ieee.org/document/902291/>.
- [11] Badrinarayanan, V., Kendall, A., and Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv:1511.00561v3*, 2016. URL <https://arxiv.org/abs/1511.00561>.
- [12] Noh, H., Hong, S., and Han, B. Learning deconvolution network for semantic segmentation. *arXiv:1505.04366v1*, 2015. URL <https://arxiv.org/abs/1505.04366>.
- [13] Chan, T. F. and Vese, L. A. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266 – 277, 2001. URL <http://ieeexplore.ieee.org/document/902291/>.
- [14] Powers, D. Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation. *Mach. Learn. Technol.*, 2, 2008.