Trainable Active Contour Model for Histological Image Segmentation

A.V. Khvostikov[1,A], A.S. Krylov[2,A], I.A. Mikhailov[3,B], P.G. Malkov[4,B]

[A] Laboratory of Mathematical Methods of Image Processing, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University
[B] Department of Pathology, University Medical Center, Lomonosov Moscow State University

[1] ORCID: 0000-0002-4217-7141, khvostikov@cs.msu.ru
[2] ORCID: 0000-0001-9910-4501, kryl@cs.msu.ru
[3] ORCID: 0000-0001-8020-369X, imihailov@mc.msu.ru
[4] ORCID: 0000-0001-5074-3513, pmalkov@mc.msu.ru

Abstract

Lesions analysis of mucous glands, which depends on the glands segmentation in histological images, is an important task of surgical pathology. This paper presents a hybrid method of glands object segmentation in histological images, based on the trainable active contour model. The hybrid method combines the use of both modern convolutional neural networks and classical methods of mathematical image processing. Also, within this hybrid method a special postprocessing algorithm is implemented, which allows to correctly segment stucked glands in the image. The proposed method was tested on PATH-DT-MSU dataset and demonstrated good results. The average value of IoU for all test images is 0.81.

Keywords: image segmentation, hybrid algorithms, convolutional neural networks, active contours, histological images.

1. Introduction

Currently, the use of machine learning and deep learning methods based on convolutional neural networks has become one of the main directions in computer processing and analysis of medical images [1, 2]. However, this technique also has a number of shortcomings (tight dependency of the solution to the dataset on which the neural network was trained and the unpredictability of diagnostic results even with a slight deviation from this dataset, lack of solution justifications with standard medical criteria, insufficient quality control of the used images, insufficient dependency analysis of diagnostic results on the level of image artifacts). Thus, the development of hybrid methods for processing and analyzing medical images, that use both machine-learning methods and the classical methods of mathematical image processing, becomes very important as they allow to significantly help in solving the above problems.

Although the idea of these hybrid methods is quite new, a number of examples of such algorithms can be found in literature. In [3] a hybrid approach uniting a convolutional neural network and a classical non-local denoising algorithm is used in the problem of suppresion of additive Gaussian noise in images, and in [4] a bundle of classical graph cut algorithm and convolutional neural network is used to train the convolutional neural network for semantic image segmentation using weakly supervised data, etc.

In this paper, hybrid algorithms for analyzing medical images will be considered for the problem of mucous glands segmentation in histological images.

Algorithms that perform semantic segmentation of glands in histological images [5] allow in some cases to divide adjacent glands, however, in general, the performed segmentation is

not perfect and needs further improvements. The principal improvement is the transition from the task of semantic segmentation to the task of instance segmentation for detecting individual glands in the image.

Among the neural network based algorithms used for instance segmentation, two main approaches that are well proven in practice are worth highlighting. The first one is Mask R-CNN [6], developed by Facebook AI Research. Based on the input image Mask R-CNN generates assumptions about the areas where the object may be located, predicts the class of the object, refines the bounding box, and generates the pixel mask of the object. A fundamentally different instance segmentation method is implemented by Deep Watershed model [7]. It uses the idea of the classical watershed algorithm and modern deep learning approach to create an energy map of the image, where individual objects are represented in the form of energy pools. Performing a cut at one energy level allows to get related components corresponding to segmented objects.

Mask R-CNN and Deep Watershed architectures are instance segmentation architectures of general type and have their disadvantages when applied to the segmentation of glands in histological images. In particular, Mask R-CNN requires a large amount of data for training and Deep Watershed is prone to perform incomplete segmentation of objects and lose details near objects boundaries. In addition, none of these architectures uses information about the shape of segmented objects, which could be extremely useful in the case of histological structures (the boundaries of the glands are mostly smooth and in a large number of cases are close to ellipses).

To work with this kind of information an active contour model [8], which represents a variational method of finding boundaries in an image, can be a very good solution. In this model the problem of finding an object boundary is formulated as finding a contour on which the specified energy functional reaches its minimum.

The main disadvantage of the classical active contour model is the manual selection of the parameters of the contour for each image and the use of low-level image characteristics when constructing the energy functional. In attempt to combine the good generalization ability of convolutional neural networks and the flexibility of the classical model of active contours, this paper considers the trainable active contour model [9] and a hybrid method of glands segmentation in histological images, that is based on this model.

2. Trainable active contour model

We will consider an active contour as a polygon $\mathbf{y} = (\mathbf{u}, \mathbf{v})$ consisting of L nodes $\mathbf{y}_s = (u_s, v_s) \in \mathbb{R}^2$, where each node $s \in \overline{1, L}$ represents one of the nodes of the discretized contour.

According to the classical active contour model [8] the energy functional can be defined as:

$$E(\mathbf{y}, x) = \sum_{s=1}^{L} \left[D(\mathbf{x}, (\mathbf{y}_s)) + \alpha \left| \frac{\partial \mathbf{y}}{\partial s} \right|^2 + \beta \left| \frac{\partial^2 \mathbf{y}}{\partial^2 s} \right|^2 \right] + \kappa \mathbf{n}(\mathbf{y}_s) \tag{1}$$

where $D(\mathbf{x}) \in \mathbb{R}^{U \times V}$ represents external energy, which depends on the source image $\mathbf{x} \in \mathbb{R}^{U \times V \times d}$ of size $U \times V$, $\alpha \in \mathbb{R}$ is the membrane term, $\beta \in \mathbb{R}$ is the thin plate term, $\kappa \in \mathbb{R}$ is the baloon term. The notation $\mathcal{F}(\mathbf{x}, (\mathbf{y}_s))$ corresponds to the value of $\mathcal{F}(\mathbf{x})$ at position $\mathbf{y}_s = (u_s, v_s)$. A function depending on the gradient of the image is used as an external energy, the remaining parameters are manually selected for each image.

The main difference of the trainable active contour model [9] from the classical model is that the energy functionals are determined independently at each point of the image and are predicted by a segmentational convolutional neural network. Within this model, the energy functional is defined as:

$$E(\mathbf{y}, \mathbf{x}) = \sum_{s=1}^{L} \left[D(\mathbf{x}, (\mathbf{y}_s)) + \alpha(\mathbf{x}, (\mathbf{y}_s)) \left| \frac{\partial \mathbf{y}}{\partial s} \right|^2 + \beta(\mathbf{x}, (\mathbf{y}_s)) \left| \frac{\partial^2 \mathbf{y}}{\partial s^2} \right|^2 \right] + \sum_{u,v \in \Omega(\mathbf{y})} \kappa(\mathbf{x}, (u, v)), \tag{2}$$

where $D(\mathbf{x}) \in \mathbb{R}^{U \times V}$ represents external energy that depends on the input image $\mathbf{x} \in \mathbb{R}^{U \times V \times d}$ of size $U \times V$, $\alpha(\mathbf{x}) \in \mathbb{R}^{U \times V}$ is the membrane term, $\beta(\mathbf{x}) \in \mathbb{R}^{U \times V}$ is the thin plate term, $\kappa(\mathbf{x})$ is the baloon term, and $\Omega(\mathbf{y})$ is the area inside polygon $\mathbf{y}$.

The field of external energy $D(\mathbf{x})$ defines the regions of the image, where the active contour should move. In other words $D(\mathbf{x})$ should be small near the boundaries of segmented objects and it should obtain large values in other areas of image. During inference the contour is moved in the direction of steepest descent $-\nabla D(\mathbf{x}) = - \left[\frac{\partial D(\mathbf{x})}{\partial u}, \frac{\partial D(\mathbf{x})}{\partial v} \right]$.

In the case of trainable active contour model the external energy field as well as internal fields $\alpha(\mathbf{x})$, $\beta(\mathbf{x})$ and ballon term $\kappa(\mathbf{x})$ are predicted using a convolutional neural network at each pixel of the image.

According to [9] the derivative of internal contour energy E_{int} with respect to the coordinates of polygon $\mathbf{y}_s = (u_s, v_s) \in \mathbb{R}^2$ can be calculated as:

$$\frac{\partial E_{int}}{\partial \mathbf{y}} = (A(\alpha) + B(\beta))\mathbf{y}, \tag{3}$$

where $A(\alpha)$ is a tri-diagonal matrix and $B(\beta)$ is a penta-diagonal matrix.
The ballon term can also be represented in energy form:

$$E_\kappa = \sum_{u,v \in \Omega(\mathbf{y})} \kappa(u, v). \tag{4}$$

Shifting the node $\mathbf{y}_s$ by $\Delta y = (\Delta u_s, \Delta v_s)$ leads to changes of energy E_κ. The derivative of E_κ is

$$\frac{\partial E_\kappa}{\partial u_s} = \frac{1}{v_{s-1} - v_s} \int_{h=0}^{v_{s-1} - v_s} h\kappa(h)dh + \frac{1}{v_{s+1} - v_s} \int_{h=0}^{v_{s+1} - v_s} h\kappa(h)dh \tag{5}$$

in the case of movement by Δu_s and

$$\frac{\partial E_\kappa}{\partial v_s} = \frac{1}{u_{s-1} - u_s} \int_{h=0}^{u_{s-1} - u_s} h\kappa(h)dh + \frac{1}{u_{s+1} - u_s} \int_{h=0}^{u_{s+1} - u_s} h\kappa(h)dh \tag{6}$$

in the case of movement by Δv_s.

The energy of contour (2) can be split into internal energy E_{int}, consisting of $\alpha(x)$ and $\beta(x)$, and external energy E_{ext}, consisiting of $D(x)$ and $\kappa(x)$.

Whereas E_{int} depends only on contour $\mathbf{y}$, and considering (6) the step of contour movement can be calculated as:

$$\mathbf{y}^{t+1} = \mathbf{y}^t - \frac{dE_{ext}}{d\mathbf{y}^t} - (A + B)\mathbf{y}^{t+1}. \tag{7}$$

Finally, according to [9] the contour movement is described as:

$$\mathbf{y}^{t+1} = (I + A + B)^{-1} \left(\mathbf{y}^t - \frac{dE_{ext}}{d\mathbf{y}^t} \right). \tag{8}$$

To make the iteration process (8) more robust, we limit the contour movement to some maximal value v, defining the number of pixels each node of the contour can be shifted along each axis per iteration. We denote the desired contour movement as $\Delta \mathbf{y}$:

$$\Delta \mathbf{y}^t = (I + A + B)^{-1} \left(\mathbf{y}^t - \frac{dE_{ext}}{d\mathbf{y}^t} \right) - \mathbf{y}^t. \tag{9}$$

After that, we represent the contour movement taking into account the introduced restrictions:

$$\mathbf{y}^{t+1} = \mathbf{y}^t + v\tanh(\Delta \mathbf{y}^t). \tag{10}$$

To make the movement of the active contour smoother and more stable, we apply the idea of momentum [10] that is widely used in machine learning. In particular, we use the weighted sum of the calculated movement on the current iteration (17) and the movement from the previous iteration:

$$\mathbf{y}^{t+1} = \mathbf{y}^t + (1 - \mu)(v \tanh(\Delta \mathbf{y}^t)) + \mu(\mathbf{y}^t - \mathbf{y}^{t-1}), \tag{11}$$

where μ is momentum. Note that if $\mu = 0$ formula (11) turns into (10).

As a result, the movement of the active contour is described by (9), (11).

Thus, the scheme of the algorithm based on the described trainable active contour model is the following. Using the input image $\mathbf{x}$, the segmentational convolutional neural network predicts energy fields $D(\mathbf{x}), \alpha(\mathbf{x}), \beta(\mathbf{x}), \kappa(\mathbf{x})$. After that according to the initial position of the contour and the predicted energy fields in accordance with (3, 9, 11) the active contour is expanded, which leads to the object segmentation.

It is also worth noting that the process of contour expansion can be implemented directly within the network. Thus, the source image and the initial contour y^0 are used as neural network inputs, and at the output we get all contour representations as a $[L, 2, n]$ tensor, where n is the number of iterations over time. This allows us to consider the described steps as the use of a convolutional neural network with structured prediction [11]. Such neural networks got this name because the output of the network represent a complex structured data, not a single value or image (as in the case of neural networks used for classification and segmentation tasks).

The main drawback of the described active contour model with structured prediction is the impossibility to construct any ground-truth data. That is, with a fixed number of L nodes on the contour, the boundary of the object can be sampled in almost an unlimited number of ways, which makes a standard comparison of the neural network prediction with ground-truth data using L_2 norm completely meaningless.

In [9] the authors propose a method for training a neural network with structured prediction, which is based on generating negative examples of incorrect segmentation to supplement positive examples of correct segmentation. For all pairs contour/image from training dataset and using a task loss function $\rho(\mathbf{y}, \hat{\mathbf{y}})$ they build a functional based on max-margin formulation. After that the contour that deviates most from the ground-truth is found for the fixed weights of the neural network. This contour is used to calculate the subgradient of the maximum distance functional, which makes it possible to find the required change of the network weights. As a result, the proposed method allows to simultaneously reduce the energy of the ground-truth contour $E(\mathbf{y}^i, \mathbf{x}^i; \omega)$ and increase the energy of the contour most different from the desired one. Thus, the neural network is trained to predict contours that are close to the ground-truth according to task loss function ρ.

The general scheme of the proposed hybrid method is shown in Fig.1.
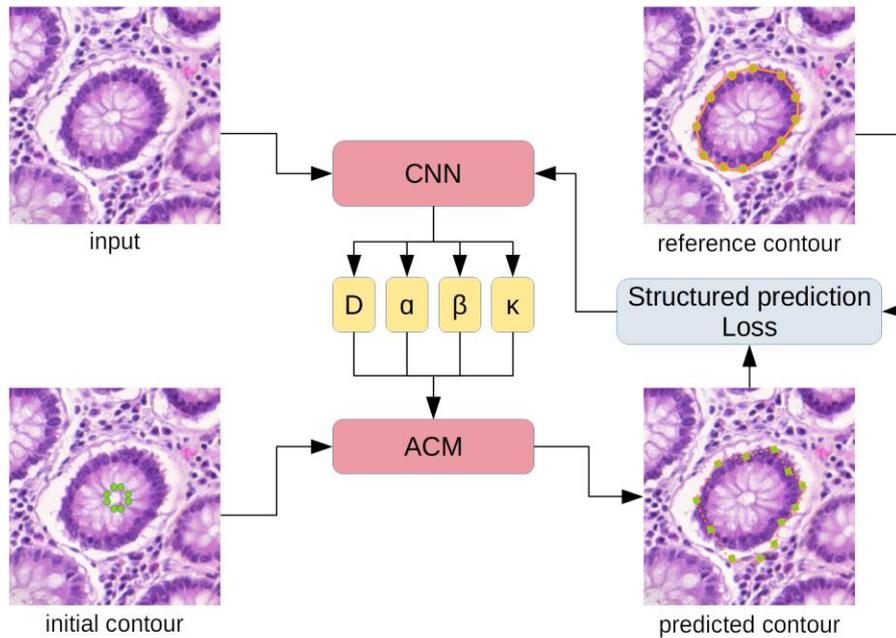
Fig. 1. The scheme of trainable active contour model.

3. Applying trainable active contour model for gland segmentation in histological images

We will consider the issue of using the trainable active contour model for the problem of glands segmentation in histological images (the initialization step of active contours requires more detailed research and is not included in this work). In this paper, we use PATH-DT-MSU histological dataset [5], which consists of 20 full-frame sections of colon. To assess the quality of the segmentation, we use Jaccard similarity index also known as intersection over union or IoU.

Previously described trainable active contour model can be used to simultaneously operate with only one contour in the image. Therefore, both in the case of training and evaluating the model, all of the original histological images are divided into square patches of a fixed size, using information about the initial positions of the active contours. The size of the patch was chosen empirically so that it exceeded the size of the gland presented in PATH-DT-MSU data set, and was chosen to be 512 pixels. Also, the formation of patches is performed in a way so that the gravity center of the initial position of the contour coincides as far as possible with the center of the patch (except the glands that lay close to the image boundary).

The number patches used for training is enlarged using data augmentation. Data augmentation is performed by turning a patch at a random angle, applying random reflection and scale changing, applying nonlinear distortion and changes in brightness. Only those sequences of these random transformations are considered, during which the ground-truth contour corresponding to the patch does not outstep the patch (mainly the rotations are concerned).

In this paper a modified version of U-Net [12] is used as a segmentational convolutional neural network for the trainable active contour model. The network depth (the number of convolutional blocks in the encoding part of the network) is chosen to be 5, a normalization operation has been added inside each block. To simplify the task and speed up the learning process, it was decided to feed the network with patches not of the original size, but reduced to 128×128 resolution. The main difference of the used network from the original U-Net is the number of output tensors, which is equal to 4. In particular, after the network calculates the final tensor at the end of the decoding path, four independent convolutions are applied, comparing to the one that is used in U-Net to reduce the number of channels to 2. This allows to obtain 4 result maps from a single input image: $D(\mathbf{x}), \alpha(\mathbf{x}), \beta(\mathbf{x}), \kappa(\mathbf{x})$. All calculations for contour expansion based on the initial location and the predicted energy maps are performed inside the network, thereby forming a network with structured prediction, the training procedure for which was described earlier.

Also, in the current problem of glands segmentation in histological images due to the limited amount of training data, it was decided to use the patch-averaged value of $\alpha(\mathbf{x})$ instead of the individual value at each point of the patch.

Presently, the computation scheme that moves the contour based on the predicted energy maps is implemented only for a single contour and is not adapted to work in batch mode. Therefore, choosing the large batch size results in the low training speed of the network. To solve this problem we use group normalization [13] instead of batch normalization inside each convolutional block. This approach allows to leave the batch size small (in the current configuration, it is equal to 2) while maintaining a network training speed comparable to the use of batch normalization with a batch of 16.

The network is trained using Adam (*Adaptive Moment Estimation*) optimizer [14]. It should be noted that due to the fact that this neural network does not use a smooth loss function, and weights are optimized based on the subgradient method, training with a gradual decrease of the learning rate (by a fixed number of iterations or upon reaching a plateau) gives poor results. In this case a cyclical learning rate is much more suitable. To train the network we use the scheme of changing the learning rate from [15], in which the learning rate curve with respect to an iteration is represented as oscillations with a fast linear increase and a smooth cosine decrease with a gradual decrease in amplitude.

The parameters of the trainable active contour model for the current task were chosen as follows: the number of nodes in the contour $L = 40$, the number of iterations $N = 40$, the maximum distance each contour node can be shifted during one iteration $v = 2$, contour moment $\mu = 0.2$, the initial positions of the contours were set as circles with a radius of 5 pixels around the gravity center of the ground-truth contour of each gland.

The results of the glands segmentation in histological images performed with the proposed trainable contour model on PATH-DT-MSU dataset using all the described above techniques are shown in Fig. 2. The training process is visualized in Fig. 5 in the form of IoU dependency from the training epoch on training and test data. In addition, Fig. 3 demonstrates the visualization of some of the predicted energy maps.

(a)

(b)

(c)

(d)
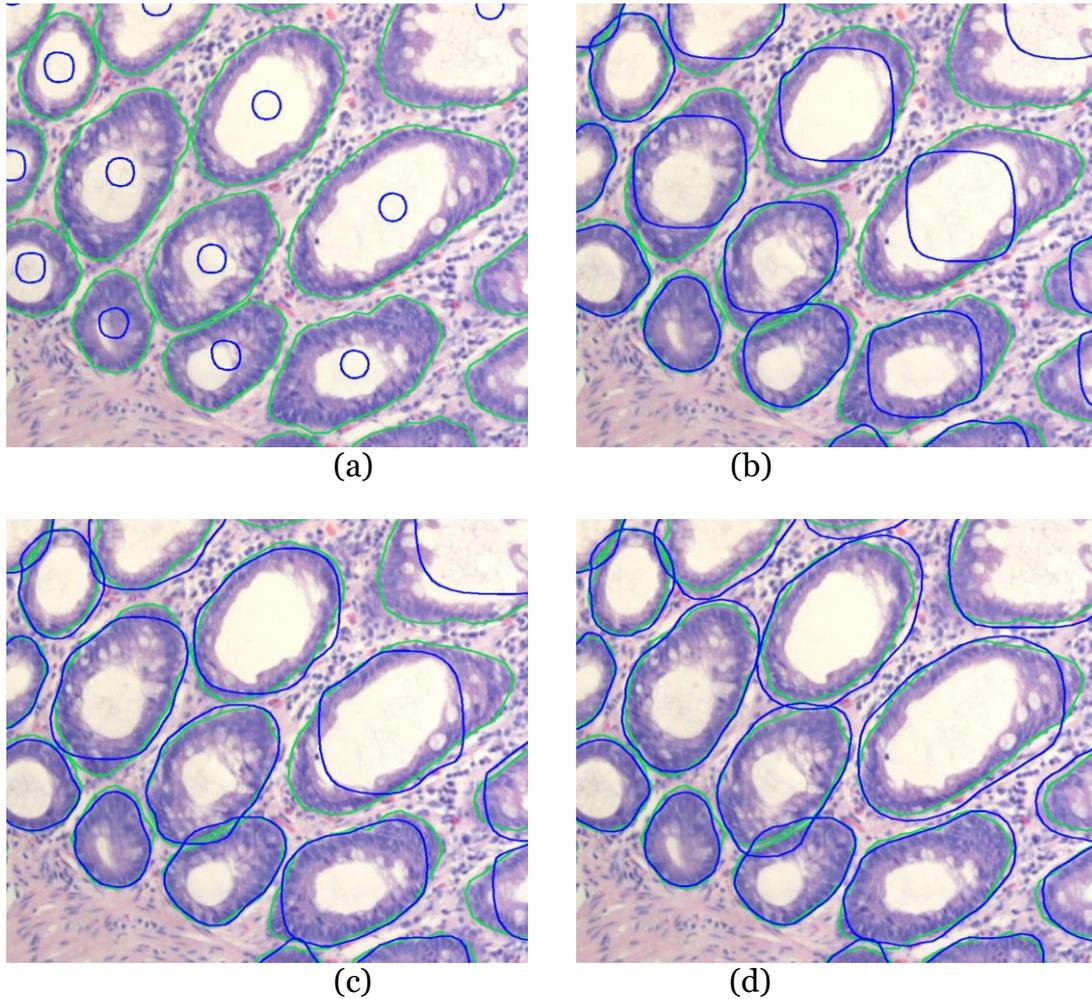
Fig. 2. Glands segmentation in the test image from PATH-DT-MSU dataset. The ground-truth contours of glands are green, the predicted contour is blue. (a), (b), (c), (d) are the states of contours at iteration 1, 20, 30 and 40.

(a)

(b)
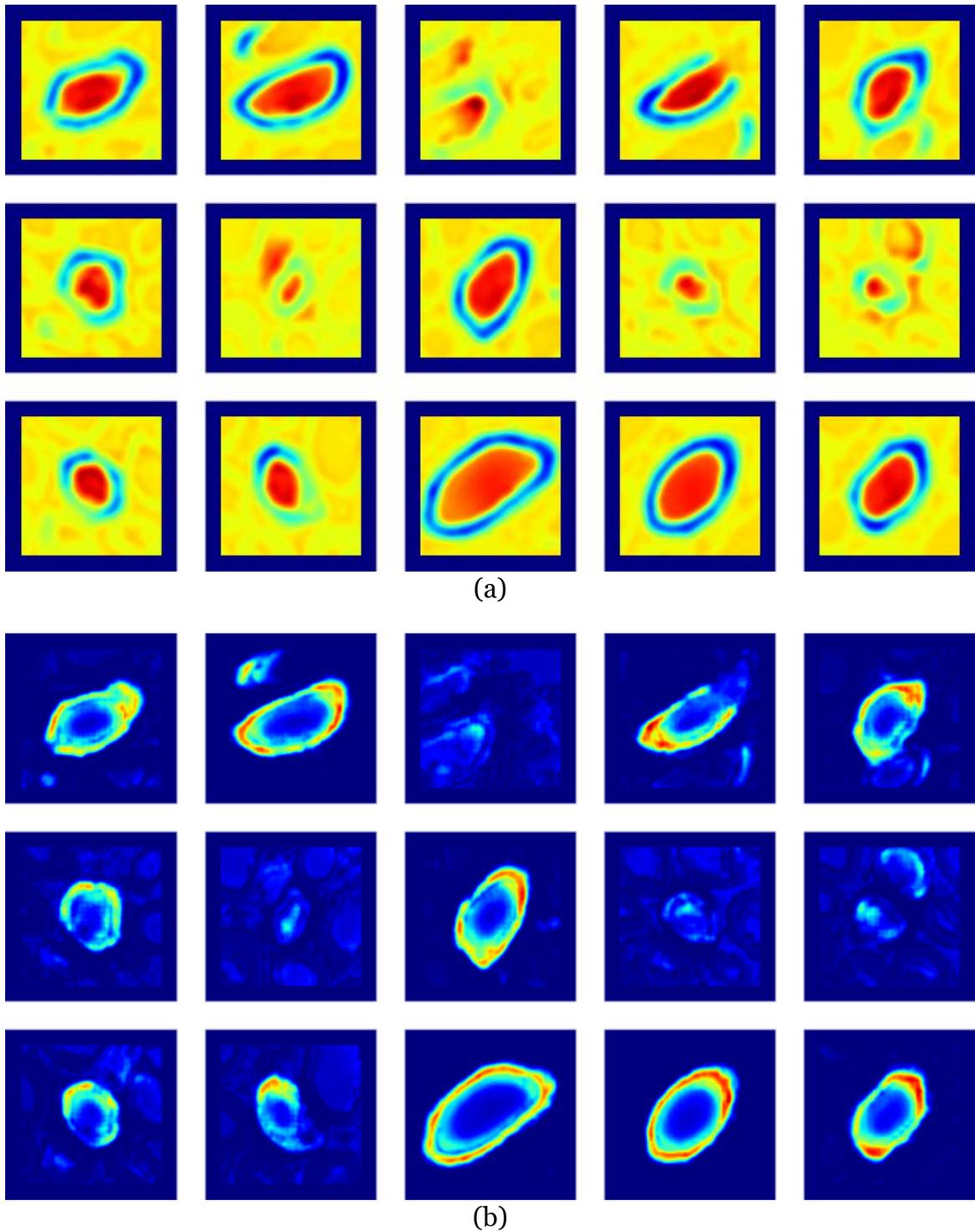
Fig. 3: Visualization of the predicted energy fields for several patches of the test image from PATH-DT-MSU dataset: (a) is $D(\mathbf{x})$ energy field, (b) is $\kappa(\mathbf{x})$ energy field.

4. Postprocessing segmentation results

The described algorithm for segmentation of individual glands in histological images has one major drawback. Namely, the segmentation of each gland with the active contour extending from its original position is performed independently and does interact with other glands present in the image. Due to the fact that the segmentation of each individual gland with the active contour model may not be ideal, in some areas of the image the resulting contours of several glands may overlap. This effect is obviously a segmentation error. To eliminate the described effect, we propose to use a special postprocessing algorithm.

We consider a set $\mathcal{Y}$ of N active contours, each of them consists of L nodes and M iterations over time. Thus, $\mathbf{y}_i^t(k)$ stands for the coordinated of k-th node of i-th active contour at time step t. To prevent the possibility of contour overlapping we propose a simple but at the same time effective postprocessing collision resolution algorithm (Alg.1).

1: **procedure** REGULARIZE($\mathcal{Y}$)

        ◦ processed set of contours $\mathcal{Y} = \{\mathbf{y}_i, i = \overline{1,N}\}$

2: **for** t = 2 **to** M **do**

        ◦ iteration over time

3:   shuffle($\mathcal{Y}$)

        ◦ shuffle contours

4:   **for** i = 1 **to** N **do**

        ◦ iterate over all contours

5:     $P = \{\mathbf{y}_j(t), j = \overline{1, i-1}\}$

        ◦ already processed contours

6:     $Q = \{\mathbf{y}_j(t-1), j = \overline{i+1, N}\}$

        ◦ contours that are not processed yet

7:     **for** k = 1 **to** L **do**

        ◦ iterate over all nodes

8:       $seg1 = (\mathbf{y}_i^{k-1}(t-1), \mathbf{y}_i^k(t))$

        ◦ segment connecting nodes $k-1$ and k

9:       $seg2 = (\mathbf{y}_i^k(t), \mathbf{y}_i^{k+1}(t-1))$

        ◦ segment connecting nodes k and $k+1$

10:     **if** $(seg1 \cup seg2) \cap (P \cup Q) \neq \emptyset$ **then**

        ◦ if there are any intersections

11:       $\mathbf{y}_i^k(t) = \mathbf{y}_i^k(t-1)$

        ◦ do not move this node

12: **return** $\mathcal{Y}$

        ◦ result set of contours without collisions

Alg.1: Collision resolution algorithm

This collision resolution algorithm for the expanding active contours works as follows. For each iteration $t \in \overline{2, M}$ the algorithm randomly shuffles the set of contours, and then alternately examining each contour from the sequence moves only those nodes of the contour (from their position at step $t-1$ to the new position at step t), the movement of which does not lead to the intersection of the processed contour with any of the remaining (Fig. 4). When the current contour is interacting with the one that has already been processed at the current iteration, the processed contour is considered at the updated time step t, and when interacting with the contour that has not yet been processed at the current iteration, the unprocessed contour is considered at the previous time step $t-1$. The random shuffling of contours before each iteration of the algorithm affects the order of contours processing and leads to more robust results.
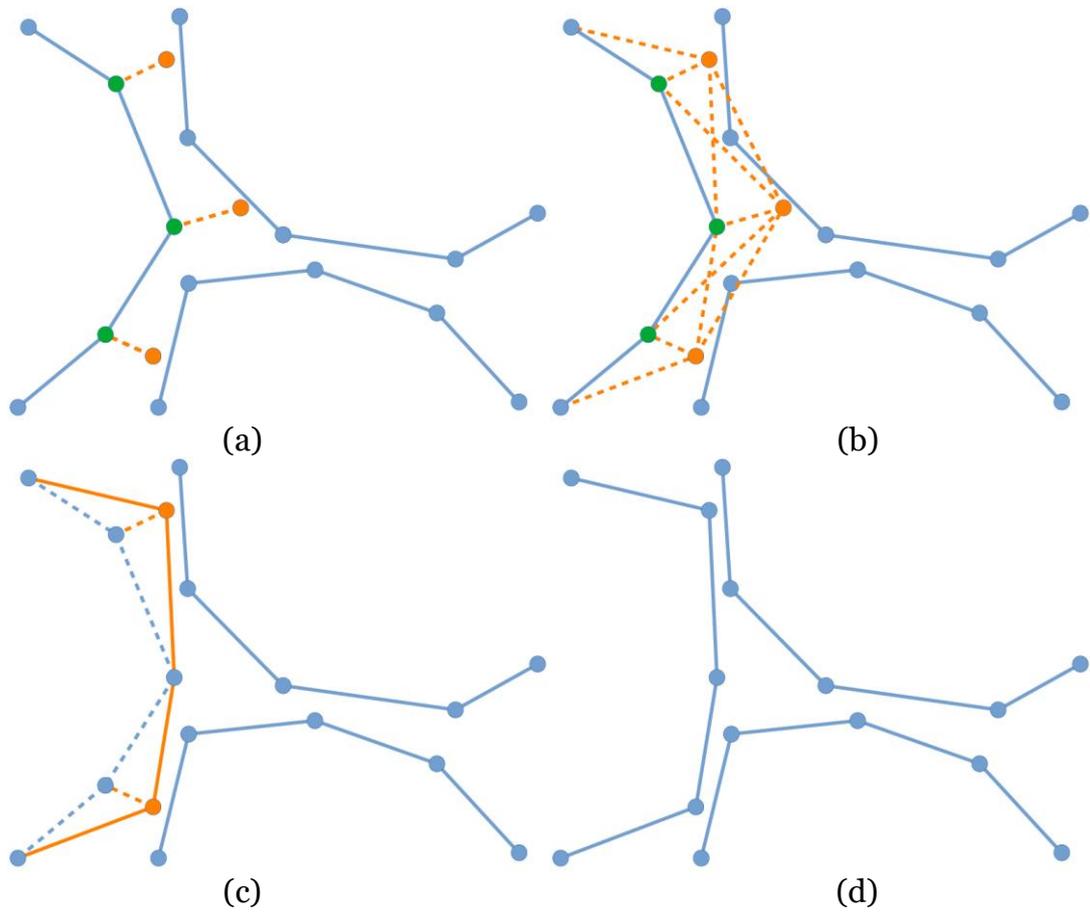
Fig. 4: Interaction of active contours. Turn of the left contour movement. The nodes that are moved at the current iteration are marked in green, their new predicted positions are marked in orange. (a) shows the initial position of the contour, (b) shows possible new positions of the contour nodes and the intersection of its segments with other contours, (c) demostrates the choice of the nodes that can be moved, (d) shows the result of the movement.

Postprocessing the set of active contours with the described collision resolution algorithm makes it possible to ensure that the resulting contours in the image do not intersect. At the same time, since the introduced trainable active contour model assumed to limit the shift of each node of the contour by 2 pixels per iteration, the contours that initially intersected would be located close to each other.
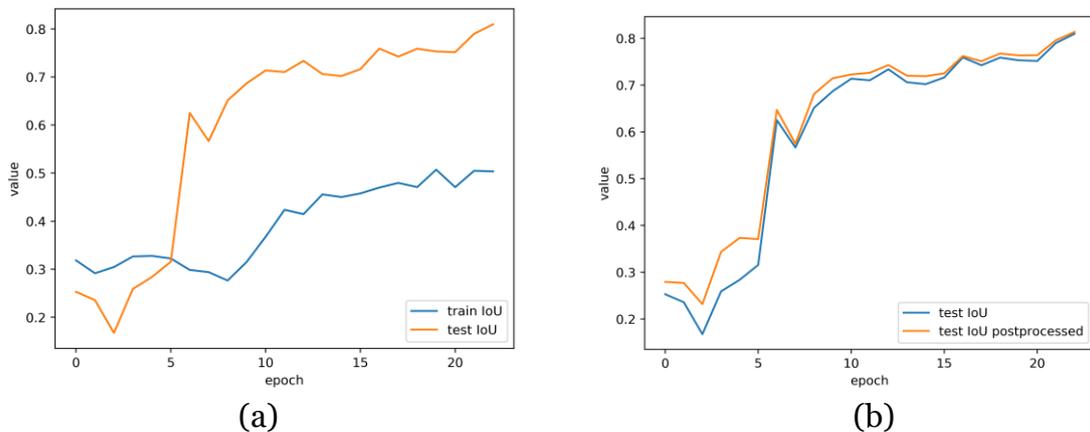
Fig. 5: Training a hybrid trainable active contour model: (a) shows IoU over epoch for training and test data, (b) shows IoU over epoch for test data with and without the use of a collision resolution algorithm.

The described postprocessing algorithm improves the quality of segmentation (Fig. 6). Thus, the value of the object IoU value on test images after applying the collision resolution algorithm increases by an average of 0.01 (Fig. 5). In this case, the final IoU for all test images from PATH-DT-MSU dataset is 0.81.

(a)	(b)
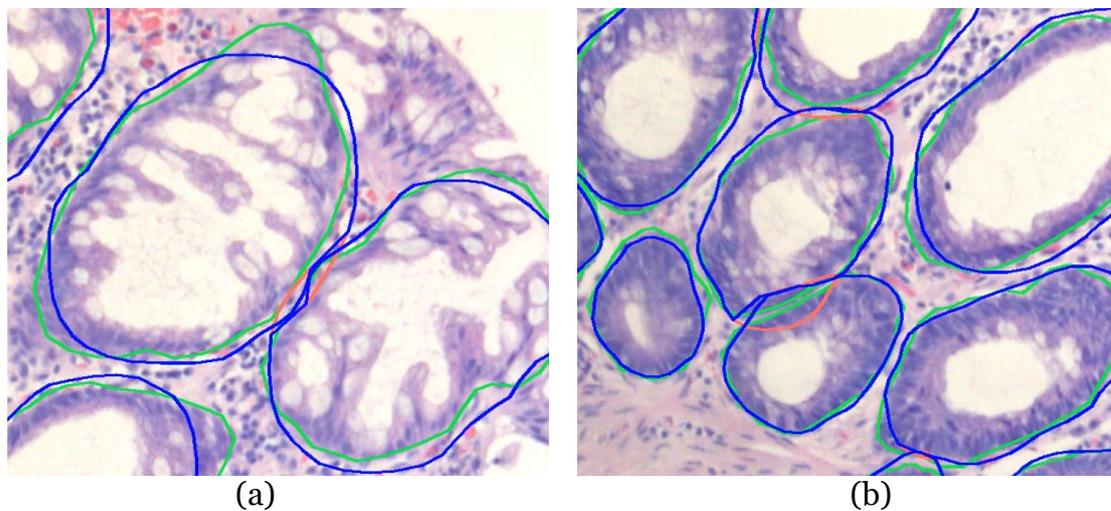
Fig. 6: Postrpocessing with collision resolution algorithm. The ground-truth contours of the glands are marked in green, the active contours being trained with the collision resolution algorithm are shown in blue, the initial parts of the contours that have changed after applying the algorithm are marked in red.

5. Conclusion

This paper presents a hybrid method for glands segmentation in histological images based on the trainable active contour model. A separate algorithm is also proposed for postprocessing of the segmentation results obtained with active contours. The proposed methods were tested on PATH-DT-MSU dataset and demonstrated good segmentation results.

Further studies will include the development of active contour initialization algorithms designed for histological images, which will allow us to construct a fully automatic trainable method for histological images segmentation.

Acknowledgements

References

[1]   Mihalj Bakator and Dragica Radosav. Deep learning and medical diagnosis: A review of literature. *Multimodal Technologies and Interaction*, 2(3):47, 2018.

[2]   Riccardo Miotto, Fei Wang, Shuang Wang, Xiaoqian Jiang, and Joel T Dudley. Deep learning for healthcare: review, opportunities and challenges. *Briefings in bioinformatics*, 19(6):1236–1246, 2017.

[3]   Cristóvão Cruz, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Nonlocality-reinforced convolutional neural networks for image denoising. *IEEE Signal Processing Letters*, 25(8):1216–1220, 2018.

[4]   Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3159–3167, 2016.

[5]   Automatic mucous glands segmentation in histological images / A. V. Khvostikov, A. S. Krylov, I. A. Mikhailov et al. // ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. — 2019. — Vol. 42, no. 2/W12. — P. 103–109.

[6]   Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[7]   Min Bai and Raquel Urtasun. Deep watershed transform for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5221–5229, 2017.

[8]   Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[9]   Marcos D. et al. Learning deep structured active contours end-to-end //*arXiv preprint arXiv:1803.06329*, 2018.

[10]   Qian N. On the momentum term in gradient descent learning algorithms //Neural networks. – 1999. – V. 12. – №. 1. – pp. 145-151.

[11]   Hannes Schulz and Sven Behnke. Structured prediction for object detection in deep neural networks. In *International Conference on Artificial Neural Networks*, pages 395–402. Springer, 2014.

[12]   Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[13]   Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[14]   Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15]   Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.