AUTOMATIC SEGMENTATION OF CORNEAL ENDOTHELIAL CELLS USING ACTIVE CONTOURS

Kamil Charłampowicz[1], Daniel Reska[2], Cezary Boldak[2]

[1] Student of Faculty of Computer Science, Bialystok University of Technology, Białystok, Poland

[2] Faculty of Computer Science, Bialystok University of Technology, Białystok, Poland

Abstract: We present a new method for segmenting the corneal endothelial cells from microscopic images. It uses multiple active contours initialized by adaptive thresholding and limited with their growing to not overlap. Thanks to the inherent characteristics of the active contour both outcomes can be achieved: cell quantity and delimitation. The tool implementing this approach is built within the MESA framework - an environment for developing and evaluating segmentation techniques. The accuracy is estimated on the base of real microscopic cell images segmented manually.

Keywords: image processing, cell segmentation, active contour

1. Introduction and background

Cell segmentation, very important in many biomedical applications, is not a trivial task. Many approaches to this problem have been proposed, but the most often general techniques fail when applied to specific cell shapes and targeted methods perform the best. One example are the corneal endothelial cells, which play important role in the human visual mechanism and are of great interest for physicians [1]. The healthy cells have a characteristic regular hexagonal form (Fig. 1). With time or in presence of pathologies their number drops and their shape deforms, so their analysis can supply important diagnostic and monitoring information. Unfortunately, their standard microscopic imaging does not always provide a good quality of images. Specifically, near the image borders the focus can be lost and the illumination is not homogeneous. The noise is also present and the cell shape can be irregular. Several dedicated approaches have been proposed to segment such the cells, apart from the manual delineating. Vincent and Masters [2] firstly detect the cell centers by

Advances in Computer Science Research, vol. 11, pp. 47-60, 2014.

a dome extractor based on morphological grayscale reconstruction and then let the marker-driven watershed segmentation to extract the binary (so unstructured) result. Mahzoun at al. [3] focus on the cell edges, detecting them (including "tricorn" points) with specially designed convolution filters. Then they complete the shapes with manually initialized active contours, so the approach is not automatic but can provide the cell structure. Sanchez-Marin [4] proposes a fully automatic method using the Gauss low-pass filtering to remove the intensity gradient and also to close gaps in the cell borders, then thresholding and skeletonization (to obtain the one-pixel wide net) followed by some improvement (e.g. pruning) of the final cell structure. The result cells can be smoothed with the active contour. The method seems sensitive to the scale on the Gauss filtering stage. Khan et al. [5] apply the Frangi algorithm (originally detecting vessels) to images filtered by consecutive low-pass Gauss filters in order to track the cell borders. These borders are then improved by mathematical morphology operations and finally thinned to get the one-pixel wide net. The result is not structured - it consists of the border pixel collection. Bullet et al. [6] use composition of four steps to automatically detect the cell contours: FFT band-pass filtering to remove the intensity information and noise, binarization with the mean value threshold, watershed segmentation on the distance map and finally Voronoi diagram segmentation providing the final contours. All the above operations are performed within the standard image processing tool – ImageJ. The most recent work by Piorkowski and Gronkowska-Serafin [1] proposes two approaches. Both of them analyze local configurations of pixel levels in order to mark the border (intensity valley) between cells. The detected contours are then improved with the mathematical morphology.

This work is organized as follows. The next section describes the proposed segmentation methods in details and reveals some implementation aspects. Section 3. presents the method evaluation on real images of the corneal endothelium cells. The last section summarizes the article and proposes some ideas to improve the segmentation.

2. Method

The proposed algorithm is composed of the four steps:

- preprocessing,
- binarization,
- detection of the cell center points,
- segmentation of the cell contours.

All of them are automatic and do not require an operator interaction. Each steps is described in details in the following subsections.

2.1 Preprocessing

Our input images are noisy (Fig. 1), what is often the case while analyzing the medical data. The intensity levels of the cell bodies and borders are not constant, even locally. To alleviate these problems we apply the Gauss low-pass convolution filter to the raw images (as like it is common in many segmentation frameworks). This filtering extends also the high gradient zones located on the cell borders and makes them more "visible" to the active contours segmenting the final cell shapes.

2.2 Binarization

In order to distinguish the cell bodies (more intensive) from the cell borders (darker) the most obvious solution is to use the thresholding technique. But as it has been already mentioned above, the illumination of the input images is not equal in all the image regions – one can observe the intensity gradient superimposed on the local intensity distribution (Fig. 1). This artifact makes impossible to binarize the input with the simple thresholding with one global threshold.

One of the possible to employ techniques that has been used in our work is the *adaptive thresholding*. As the intensity variation coming from the varying illumination has very low frequency and can be neglected locally, the threshold is calculated and used also locally. The entire image is divided on square regions and the mean intensity values are calculated for all of them. For each such region its center pixel is given the local threshold equal to the mean intensity of this region, incremented by few levels to avoid detecting some ghost structures in more or less homogeneous areas. The rest of pixels obtain their local threshold by simple linear interpolation. Then for all the pixels in the loop: each one is marked "black" (cell body) if its intensity is above its own local threshold and it is marked "white" (cell border) if the intensity is below it. Example of such the binarization is presented Fig. 1.

2.3 Detection of the cell center points

The healthy corneal cells have they shape close to hexagonal, and even if it is not ideal, it can be approximated by a circle. After the binarization the next loop (going also through all the pixels) tries to localize the biggest (locally) circular regions containing only "black" pixels. A special condition promotes bigger regions if two or more of them overlap – it prevents from locating multiple circle regions inside one cell what can happen because of the order of visiting the pixels (the loop simply increments the pixel coordinates). This condition is effective for the most of cases but it sometimes fails, especially for elongated cells. In order to eliminate such the

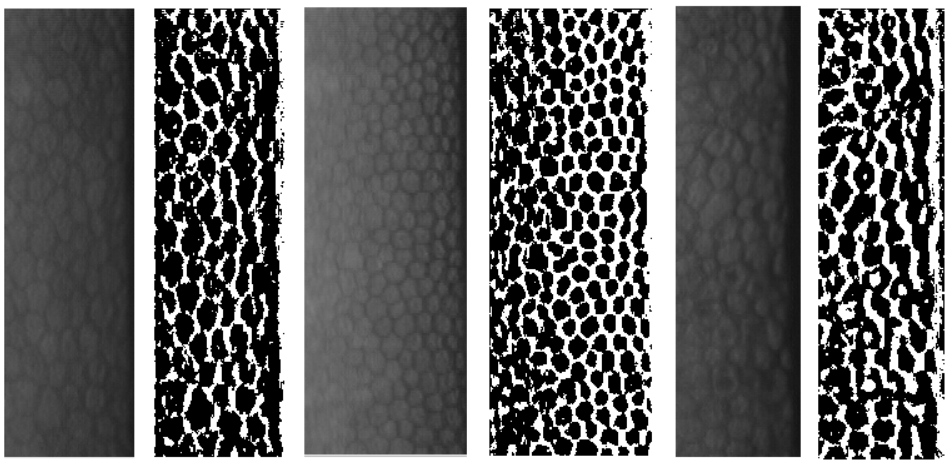

Fig. 1. Example images of the corneal endothelial cells and effects of binarization by adaptive thresholding (after the Gauss filtering)

situations all pairs of detected neighbor circles are examined to verify if the area separating them belongs also to the cell body (Fig. 2). If it is so the regions are merged to form a single cell body representation.

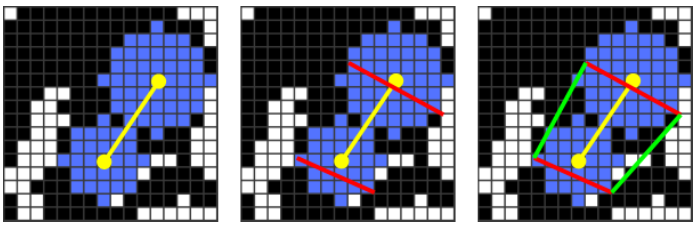

Fig. 2. Merging neighbor circle regions inside a single cell

The effect of this procedure is shown on Fig. 3. One can observe several merged circles inside elongated cells.

2.4 Final segmentation of the cell contours – active contours

The initial approximation of the cell bodies by the circular regions (or their unions) are further deformed to fit the actual cell borders using the well known *active con-*

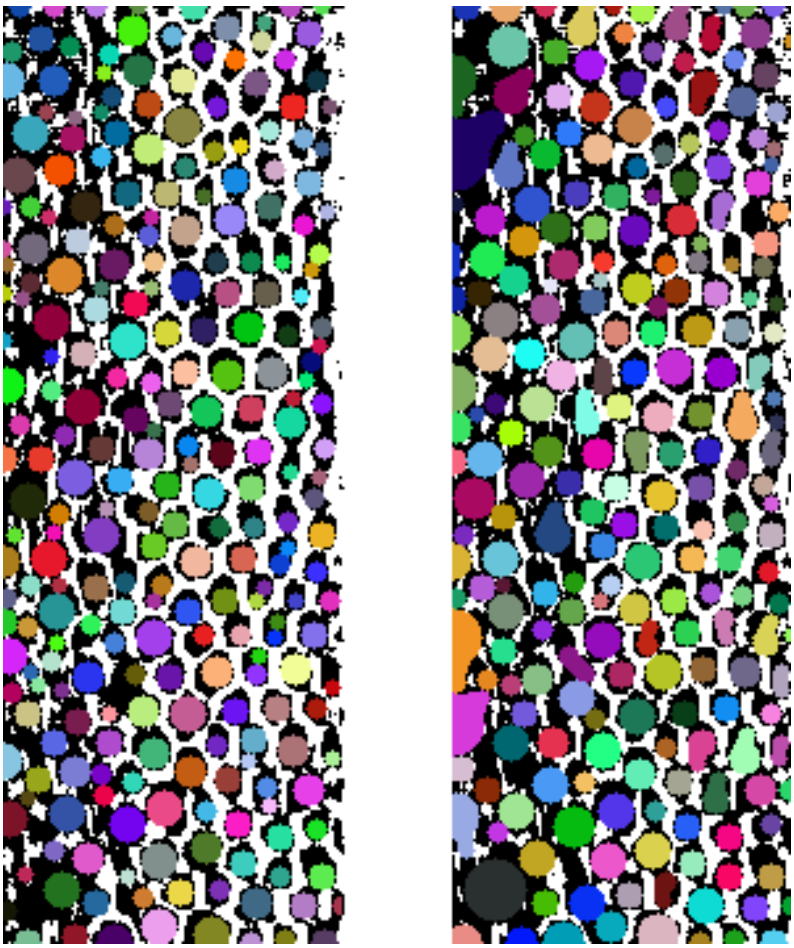

Fig. 3. Comparison of the cell body detection: on the left before merging the neighbor circles, on the right – after this procedure (colors are not corresponding on the two images)

tours – snake [7]. We use here its discrete version where the evolving curve minimizing its energy is represented by the collection of points given the local energy. Each previously marked region defines (by its external points) the initial form of the snake. All the snakes evolve on the original image with the full intensity range – its binary form is used only for the initialization. We use three energies and one balloon force (described below) that are applied to each potential (being examined) snake point (snaxel) position. The following notation is used: p – examined position of the snaxel, p_{prev} and p_{next} – neighbor snaxels (fixed while p is examined), $v_{prev} = \overrightarrow{p\ p_{prev}}$ and $v_{next} = \overrightarrow{p\ p_{next}}$) – vectors from p to its neighbors.

1. *Image energy* given by:
$$E_{image}(p) = I(p), \tag{1}$$
where $I(p)$ is the intensity level in the p position. Since the cell borders are darker (lower intensity), the snake tends to them.
2. *Internal energies* controlling the snake appearance:
   - *smoothness energy* promoting the linear location of snaxels and expressed as a length of sum of two normalized vectors v_{prev} and v_{next} (for the collinear points it has its minimum equal to zero):
$$E_{smoothness}(p) = \left| \frac{v_{prev}}{|v_{prev}|} + \frac{v_{next}}{|v_{next}|} \right|; \tag{2}$$
   - *regularity energy* favorizing the equal distance between all the snaxels and expressed as the two square differences between a mean inter-snaxel distance avg and the length of vectors v_{prev} and v_{next}, divided by the square avg:
$$E_{regularity}(p) = \frac{(avg - |v_{prev}|)^2 + (avg - |v_{next}|)^2}{avg^2}. \tag{3}$$
.
3. *Balloon force* [8] responsible for pushing out the contours (initially located inside the cells). This force does not influence the p position optimization (with the above energies) – instead it shifts all the examined neighborhood of p by a vector $v_{balloon}$ perpendicular to the difference $v_{next} - v_{prev}$ and directed outward. Its length is the method parameter.

Evolution of snakes The process of minimization of the snake energy is decomposed on the separate snaxels – no global energy is formulated and for every snaxel p the local energy $E(p) = E_{image}(p) + E_{smootheness}(p) + E_{regularity}(p)$ is analyzed independently. In every iteration each snaxel neighborhood (shifted by $v_{balloon}$ as described

above) is examined and if a position with lower local energy is found – the snaxel is moved there. Because there are multiple evolving contours on the image (one for each cell) we impose a condition forbidding each snake to grow inside any other one. To avoid the curve discretization problems two procedures are introduced to keep the inter-snaxel distance reasonable (few pixels):

- each two neighbor snaxels too close one to another are replaced by a single one;
- if two neighbor snaxels are too distant one from another a new one is added in the middle of them.

The evolution stops when there is no more moving snaxels (all snakes reached their local optima). Two examples of automatic segmentations are given on Fig. 4.

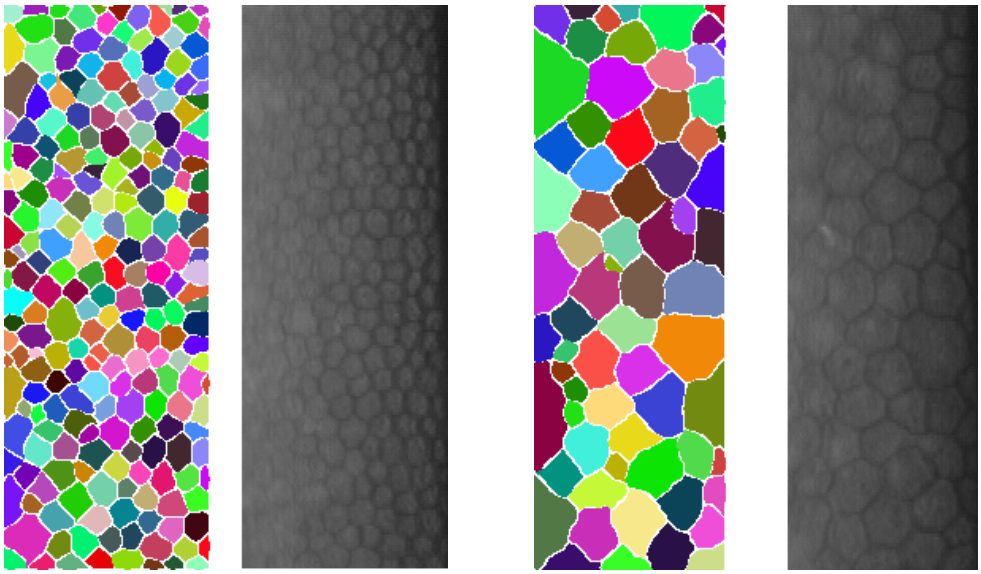

Fig. 4. Examples of the automatic cell segmentation

2.5 Implementation issues

The entire method has been implemented within the MESA framework [9]. It helps in designing and evaluating of new segmentations methods based on the deformable models. Its on-line version is accessible (mesa.wi.pb.edu.pl) – however its desktop version was used here (incorporation of the presented approach is planned). All the

standard operations (e.g. image loading, visualization, segmentation procedure) are offered there. The user is only responsible for defining the basic elements of his snake (writing new Java classes). In this work all three custom components were used:

- *model* replacing the standard one since multiple contour evolves in the image instead of the single one;
- *energies* defining the three ones as described in Section 2.4;
- *extensions* responsible for correcting the snake topology (adding and removing snaxels).

The segmentation of a single image was of order of seconds (after some optimization on controlling of the overlapping snakes) on a standard PC machine.

3. Experimental validation

Verification of quality of the medical image segmentation is not a trivial task. The most important problem is usually lack of the ground truth location and shape of the segmented object. A manual delineating by an expert can provide such a reference, but inter- and also intra-operator variabilities are very often significant. It is also time-consuming and tedious. Nevertheless, this procedure is sometimes the only possibility to quantitatively asses the precision of the method. In our work this approach was also selected and one of the authors (not being a medical expert) manually drew the cell contours of two chosen images (Fig. 5 and 6). He took into account only well visible cells. Then the presented method segmented the cells on the same images. Both segmentations took 10 snake iterations.

Having such the reference we choose two segmentation quality indices (after [10]):

- Overlap Error (OE)

$$OE(A,B) = 100(1 - (|A \cap B|/|A \cup B|)), \tag{4}$$

  where A and B are the pixel sets (in our work representing the cell bodies: segmented by our method and manually marked by the operator), value 0 characterizes two sets completely overlapping, value 100 – two separate sets;
- Relative Volume Difference (RVD)

$$RVD(A,B) = 100((|A| - |B|)/|B|), \tag{5}$$

  A and B meaning as above, value 0 means that the two compared sets have the same size (and says nothing about the set overlapping), negative value indicates undersegmentation, positive one – oversegmentation.

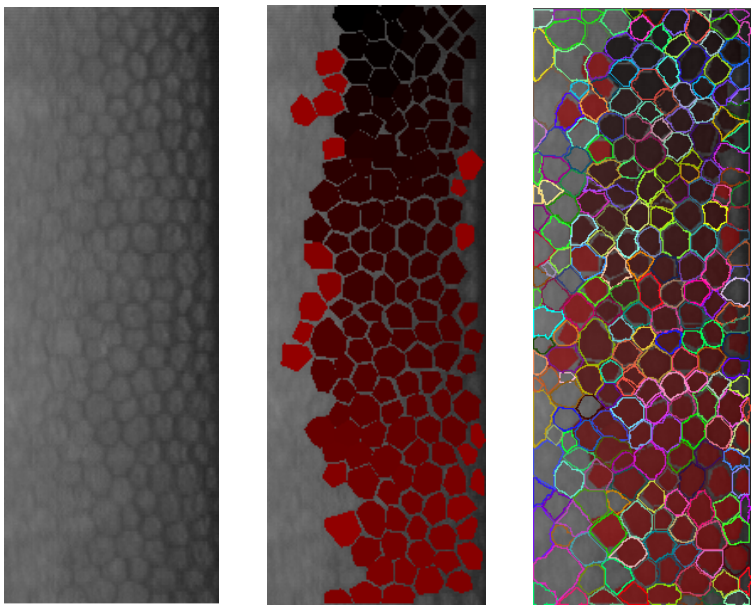

Fig. 5. Input image 1 (left), its manual segmentation – about 150 cells (center) compared with the automatic segmentation (right)

These indices are defined to be used with a single segmented object. In our cases there are multiple cells, so two problems are to be resolved. The first one is to find the correspondence between cells in two such images. This is done by finding for each reference cell an automatically segmented one with the highest overlapping ratio. The second problem is to gives one value (or at least few ones) characterizing the whole image and not separate values for each cell. So we give here: mean, median, minimum and maximum values for every index. The results are presented in Tables 1 and 2. The value distributions are also presented Fig. 7 and 8.

Table 1. Overlap Error statistics for the two selected images

Image	Median	Mean	Minimum	Maximum
Image 1 (150 cells)	22.34	24.92	7.12	86.18
Image 2 (56 cells)	22.41	26.85	7.79	71.47

As it can be observed, for the most of cases the cells are detected correctly. However one can see also some problems:

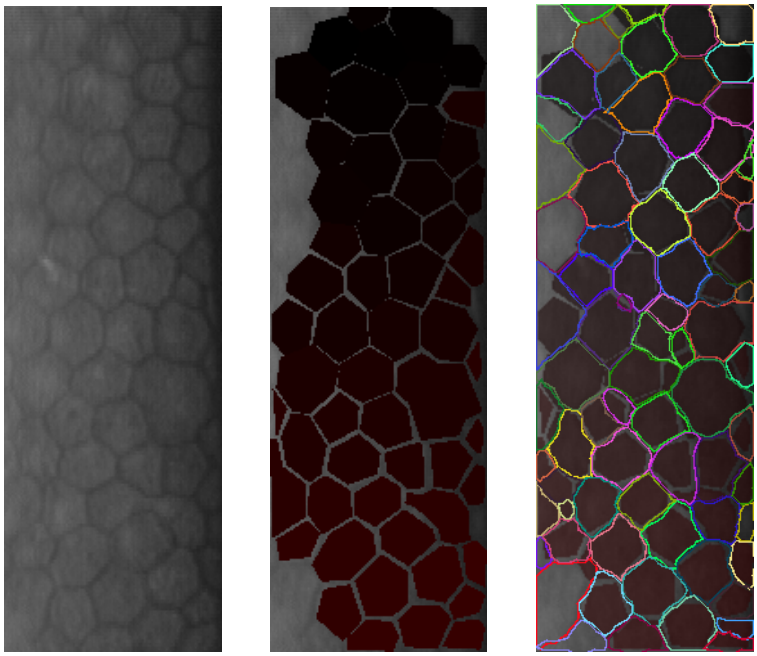

Fig. 6. Input image 2 (left), its manual segmentation – about 50 cells (center) compared with the automatic segmentation (right)

Table 2. Relative Volume Difference statistics for the two selected images

Image	Median	Mean	Minimum	Maximum
Image 1 (150 cells)	29.76	39.07	-48.83	363.83
Image 2 (56 cells)	5.35	35.25	-62.14	452.34

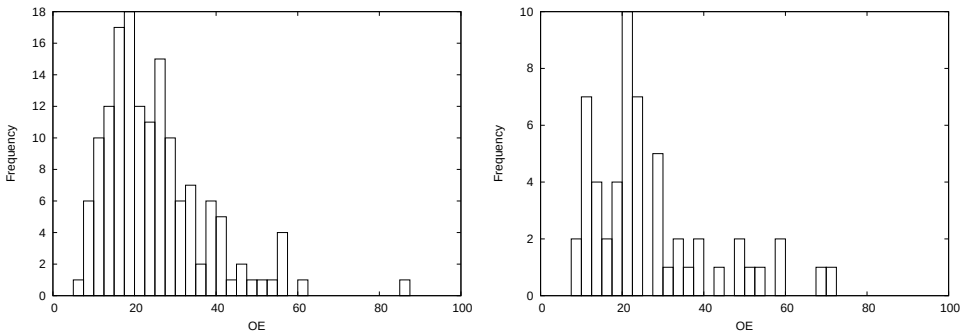

Fig. 7. Overlap Error histograms for image 1 (left) and image 2 (right)

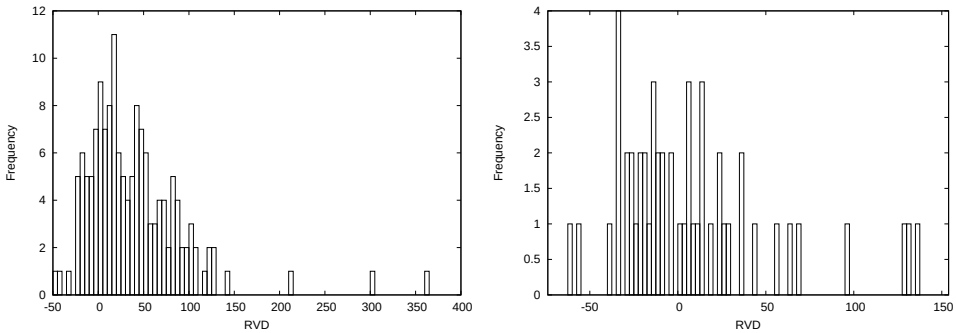

Fig. 8. Relative Volume Difference histograms for image 1 (left) and image 2 (right)

- sometimes one cell is segmented as two separate contours;
- some contours do not reach the cell borders, stopping their evolution inside the cells;
- some contours do not stop on the cell borders and they overpass them.

The first problem means that the detection of the cell centers does not work correctly there, marking two separate circles inside one cell and not merging them. Since the snake topology is fixed it results in two segmented cells inside the actual one. The next two problems show how hard is to balance the snake energies. When the balloon force is too weak (having too low weight in comparison to other forces, i.e. image and internal ones) to push the snake enough to reach the border, the contour evolution stops prematurely. But in the same time this weight is too high in other situations and it pushes the contour too much what results in overpassing the border. This behavior will be investigated more to improve the method.

The problems are reflected by the Overlapping Error values (especially the median and mean ones) what leaves a place to further working on the method. The positive Relative Volume Error shows a slight oversegmentation, but it is not very important.

In order to give even introductory comparison with other approaches (to be continued in future works) we took a single image from [1] and applied our method. The results can be visually compared Fig. 9.

4. Conclusions and future works

In this article we have presented the automatic method segmenting the corneal endothelial cells form microscopic images. The first results are promising, it detects

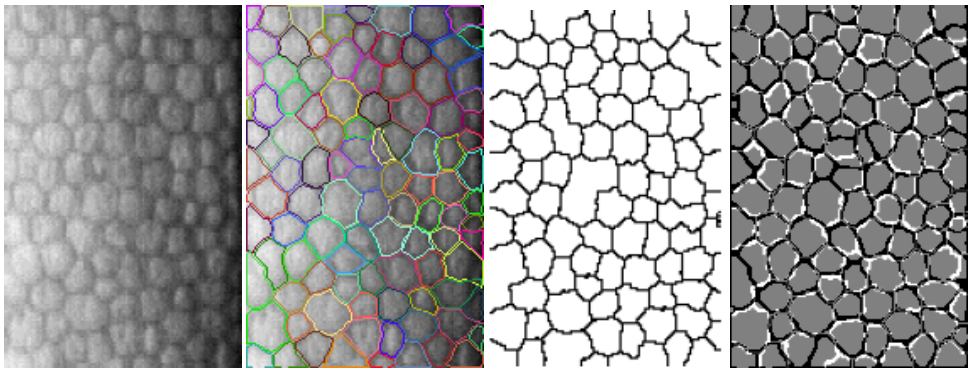

Fig. 9. Comparison of the presented method and [1]: original image (first), our segmentation superimposed on the original image (second), binary result from [1] (third), two segmentations on one image (fourth) - in gray common area, in black our segmentation, in white - [1].

even little visible cells and works quite fast (order of seconds on a standard machine). The implementation, for the moment limited to the standalone version, is realized in the technology (MESA environment) that allows an easy porting to the on-line web application and can be make available for the public use.

However there is still some work that can be done here. The results have been not yet seriously compared with the other existing methods and it is difficult to say where our approach is between them in term of quality of segmentation. Only two images (of rather good quality) was used to quantitatively assess the segmentation accuracy and the tedious manual cell delineation was done by the non expert. The parameter choice has not been extensively studied as well. Some of the parameters seem not critical (the constant in the adaptive thresholding) but others (the weights balancing the snake energies, especially the balloon one) have to be adjusted carefully and even then fail in some configurations. An iterative incrementation of the balloon force could solve the problems of not reaching some cell borders and overpassing others in the same image.

5. Acknowledgments

This work was supported with the Bialystok University of Technology Grants S/WI/2/2013 and W/WI/5/2014. We would like also to thank Jolanta Gronkowska-Serafin from Pomeranian Medical University for the cell images and Adam Piórkowski from AGH University of Science and Technology for making us interested in this problem and for other help.

References

[1] Piórkowski, A., Gronkowska-Serafin, J.: Towards Automated Cell Segmentation in Corneal Endothelium Images. Image Processing & Communications Challenges 6 Advances in Intelligent Systems and Computing 313, 2015, pp 179–186.

[2] Vincent, L.M., Masters, B.R.: Morphological image processing and network analysis of cornea endothelial cell images. Proc. SPIE vol. 1769, Image Algebra and Morphological Image Processing III, 1992, pp. 212–226.

[3] Mahzoun, M., Okazaki, K., Mitsumoto, H., Kawai, H., Sato, Y., Tamura, S., Kani, K.: Detection and complement of hexagonal borders in corneal endothelial cell image. Medical Imaging Technology 14(1), 1986, pp. 56–69.

[4] Sanchez-Marin F.J.: Automatic segmentation of contours of corneal cells. Computers in Biological and Medicine, 29(4), 1999, pp. 243–58.

[5] Khan, M.A.U., Niazi, M.K.K., Khan, M.A., Ibrahim, M.T.: Endothelial cell image enhancement using non-subsampled image pyramid. Information Technology Journal 6(7), 2007, pp. 1057–1062.

[6] Bullet, J., Gaujoux, T., Borderie, V., Bloch, I., Laroche, L.: A reproducible automated segmentation algorithm for corneal epithelium cell images from in vivo laser scanning confocal microscopy. Acta Ophthalmologica 92(4), 2014, pp. e312–6.

[7] Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1 (4), 1988, pp. 321–331.

[8] Cohen, L.D.: On active contour models and balloons. CVGIP: Image Understanding, 53(2), 1991, pp 211–218.

[9] Reska, D., Jurczuk, K., Boldak, C., Kretowski, M.: MESA: Complete approach for design and evaluation of segmentation methods using real and simulated tomographic images. Biocybernetics and Biomedical Engineering 34(3), 2014, pp. 146–158.

[10] Badakhshannoory, H., Saeedi, P.: A model-based validation scheme for organ segmentation in CT scan volumes. IEEE Transactions on Biomedical Engineering, 58(9), 2011, pp. 2681–2693.

AUTOMATYCZNA SEGMENTACJA KOMÓREK ŚRDBŁONKA ROGÓWKI OKA PRZY POMOCY AKTYWNYCH KONTURÓW

Streszczenie: W artykule zaprezentowano autorską automatyczną metodę segmentacji komórek śródbłonka rogówki oka z obrazów mikroskopowych. Metoda używa wielu aktywnych konturów zainicjalizowanych wewnątrz komórek za pomocą adaptacyjnego progowania i ograniczonych w swoim rozroście tak, aby nie pokrywać się. Metoda został zaimplementowana w środowisku MESA przeznaczonym do rozwoju i ewaluacji technik segmentacji. Jakość segmentacji została oszacowana na rzeczywistych obrazach mikroskopowych w odniesieniu do ręcznie zaznaczonych konturów komórek.

Słowa kluczowe: przetwarzanie obrazów, segmentacja komórek, aktywny kontur

Artykuł zrealizowano w ramach pracy statutowej Politechniki Białostockiej nr S/WI/2/2013 oraz pracy nr W/WI/5/2014.