



Graph based method for cell segmentation and detection in live-cell fluorescence microscope imaging



Katarzyna Hajdowska^a, Sebastian Student^a, Damian Borys^{a,*}

^a Department of Systems Biology and Engineering, Silesian University of Technology, Akademicka 16, Gliwice 44-100, Poland

ARTICLE INFO

Keywords:

Microscope image processing
Cell segmentation
Hough transform
Watershed segmentation
Graph cut segmentation

ABSTRACT

Live-cell fluorescence image segmentation is an essential step in many studies, including in drug research and other contexts where keeping cells alive is crucial. Several segmentation algorithms and programs have been previously proposed; however, they do not work sufficiently well on top-down pictures with overlapping cells. Our proposed algorithm, called GRABaCELL, utilizes Graph Cut, Watershed segmentation and Hough Circular Transform to improve automatic segmentation and counting living cells. We also introduce a modified accuracy metric to determine the quality of segmentation in terms of the number of cells detected in the image. The GRABaCELL method results are vastly better in visual assessment, by both Dice index and modified accuracy metric, than all other compared methods maintaining not only a high value of these indices but also a relatively small spread.

1. Introduction

Cells and their internal structures can be observed by using many different forms of light microscopy [33]. Bright-field microscopy registers contrast as the absorption of light transmitted through the sample in dense areas of specimen [49]; the contrast can be increased by staining the specimen [33]. Phase-contrast microscopy is an answer to bright field microscopy's problem with staining. However, it requires a specialized condenser and more expensive lenses. It takes advantage of the slowing down of light as it passes through biological specimens [39]. In fluorescence microscopy, the specimen with fluorochrome stains is illuminated with short-wavelength light, then examined through a barrier filter absorbing the light used for illumination but transmitting the fluorescence [40]. The choice of microscopy method depends on the observation goal and type of specimen - as, for example, staining the sample (its fixation) usually kills the cells, which sometimes is of little importance. In other contexts, it is crucial to keep cells alive.

Image segmentation is the process of partitioning an image into multiple objects. According to some criteria, pixels in one object are similar to one another, such as colour, intensity or texture - and dissimilar to pixels in other objects. Segmentation has a plethora of applications, ranging from noise filtering [12], medical applications [32], locating objects in satellite images [47], face recognition [15] to fingerprint recognition [5].

Cell segmentation, which refers to labelling each cell on a given image, is useful for determining the efficacy of drugs and locating tumours or other pathologies [53,25]. Manual counting is time-consuming, especially for a series of images; therefore, finding an automated solution with high accuracy is necessary. This step is essential in drug discovery for assessing the efficacy of drug treatment. It allows us to determine whether a drug destroys cells affected by sickness with appropriate equipment and analysis. The necessity to keep cells alive forces us to use low fluorescence microscopy, which worsens the images' quality. They possess variable levels of intensity, noise and numerous local artefacts, which complicates the task. The medium is highly autofluorescent. Moreover, cells are highly mobile and tend to overlap and form clumps, especially when the number of cells in the chamber increases. Their varying diameter adds to the challenge.

1.1. Literature review

There exist a multitude of cell segmentation algorithms: region growing [7], seeded watershed [55], K-Means Clustering [14], Expectation–Maximization Method [14], active contours [17] and Min Graph Cut [34], among others, each suitable for different types of images. There have been many previous attempts to segment cells using more complex methods. For example, Wang et al. [50] taught deep convolutional neural networks (DCNN) Euclidean distance transform (EDT) on a

* Corresponding author at: Department of Systems Biology and Engineering, Silesian University of Technology, Akademicka 16, Gliwice 44-100, Poland.
E-mail address: damian.borys@polsl.pl (D. Borys).

mask corresponding to an input image. Then trained a region with CNN was used to detect individual cells in the EDT image, and subsequently performed a final watershed segmentation on output images of the previous steps. Sadanandan et al. also utilized DCNN trained on results produced by standard automated methods of CellProfiler, while handling variations in focal depth by collecting bright-field data at three consecutive focus depths [42]. Kostrykin et al. introduced an approach based on implicitly parameterized shape models [24]. Rundo et al. decided first to apply a bilateral filter to preserve edge sharpness while smoothing the input, and then utilize watershed transform and morphological filtering [41]. Student et al. [45] make use of a support vector machine (SVM) classification system. On the other hand, Alilou et al. [2] suggest the use of reshaped templates. They combine both pixel- and object-level features for detecting regions of interest. Simple-shaped templates were used for coarse segmentation and a content-based template reshaping algorithm to refine final segmentation. Xu et al. [54] make another proposition that combines the iterative deformation idea of active contours with optimization of graph cuts. Those results in contours being the global optimum within a contour neighbourhood and eliminate the bias of graph cuts for a shorter boundary. Farhan et al. [16] suggest another approach in the form of a cell cytoplasm segmentation framework. Firstly, by applying a multi-scale approach, it separates cell cytoplasm from the background. Subsequently, extraction of features is performed, and a classifier is trained to distinguish between either outline or non-outline pixels to detect cell outlines. Lastly, outlines are refined so that they form a closed contour around each cytoplasm. Wang [51] makes a different suggestion of a semi-automatic segmentation method for a variety of cells or nanoparticles. It requires manual calibration of the threshold selection and determination of the segmentation cases for each type of cell or nanoparticle image. The remaining steps - gradient image formation, threshold selection and quantification by iterative morphological erosion - are automatic. Gamarra et al. [19] suggest a different approach – to combine Marker-Controlled Watershed with Split and Merge based on Watershed: during the split phase, clusters are identified based on a cell's characteristics, like size and convexity, and separated with re-checking for over-segmentation using the cell's features. Pandey et al. [31] have used the generative adversarial networks (GANs), which can cope with small datasets. Those networks are employed to create a binary mask, which is incorporated into a second GAN to perform conditional generation of the image. The produced *image-mask* pairs are used to improve the performance of conventional image segmentation approaches. Al-Kofahi et al. [1] approach the problem differently: the foreground of the image is extracted automatically via graph-cuts-based binarization, then seed points are obtained through distance-map-based adaptive scale selection constrained Laplacian-of-Gaussian filtering. Later on, seeds are used to perform initial segmentation, improved by graph-cut step with graph colouring and alpha expansion. Song et al. [43] suggest training and using a cascade sparse regression chain model to obtain locations of objects and boundaries of their clusters, while Dimopoulos et al. [13] opt for graph cuts, which detect cell boundaries through directional cross-correlations and later incorporate spatial constraints. Wang et al. [48] utilize a multi-path dilated residual network explicitly designed to detect small dense objects, like cell nuclei. Hilsenbeck et al. [20] make use of a divide and conquer approach to calculate an optimal set of non-overlapping regions, based on likelihoods obtained by SVM via feature vector of all external regions. Blin et al. [8] offer a solution based on labelling the nuclear envelope, which uses a tree-structured ridge-tracing method and shape ranking according to the trained classifier to segment non-heterogenous, non-spherical and densely packed nuclei. Lu et al. [27] introduce a CNN to learn feature representations of single cells without labelled training data for fluorescence microscopy. Mueller et al. [30] created an algorithm based on sparse component analysis to separate cell nuclei, fibrous components, and adipose components and next, the circle transform to quantitatively measure the size and density of overlapping features to detect residual

disease in tumour margin. Robinson et al. [38] present a method based on Markov random fields (MRFs) on 3D stack image data of prostate cancer cells co-cultured with cancer-associated fibroblasts (CAFs). Ko et al. [22] present an adaptive attention window (AAW) based microscopic cell nuclei segmentation method. AAW operates on a luminance map to create an initial attention window, which is then reduced by quad-tree to be similar in size to the real region of interest. Region segmentation is performed within the AAW, and later region clustering and removal are performed to produce results with only regions of interest. Chalfoun et al. [10] suggest a novel approach to morphological watershed segmentation to produce accurate results for phase contrast, bright field, fluorescence microscopy and binary images. They utilized histogram binning to minimize over-segmentation by reducing image noise and a geodesic distance mask to detect shapes of individual cells, as opposed to linear cell edges of typical watershed-based algorithms. Zhao et al. [56] make use of fused two-channel (green-channel and red-channel) images with seed-based cluster splitting and k-means algorithm to segment individual nuclei with fuzzy boundaries or in a clump.

Segmentation is one technique for cell counting; the other is contour extraction, as suggested by Wiseman et al. [52], however, the quality of pictures in cell segmentation is too low to produce sufficient contours. Moreover, the cells often overlap.

1.2. Aim of the study

This study aims to determine a solution for the automatic segmentation and localization of cells. We utilized Circular Hough Transform to detect cells and watershed and graph cut segmentation to segment them into individual objects. We believe that the existing methods described above are not sufficient for a case where cells exist in semi-3D space, and the only pictures available for the algorithms are top-down pictures. Therefore, we propose our solution - a new GRAPh Based Algorithm for CELL detection and segmentation that we have called GRABaCELL.

2. Materials and methods

2.1. Image sets and experiment description

For this study, we have used three image data sets. Those images are described further in this paragraph. Moreover, in our work, we can distinguish two main phases. The former, where the algorithm was built (further described as algorithm development), uses the first data set and compares our method with other algorithms. The latter - the algorithm validation was based on two other data sets to show the robustness of this method. The details of the algorithm are presented in the Section 2.2 and the metrics used for quantifying the segmentation results are described in 2.3.

2.1.1. Algorithm development

In the first phase of the algorithm development, the images of HeLa cells were used and the accuracy and efficiency have been tested by comparison with other methods and programs.

HeLa image set. The images used are grayscale images obtained by fluorescence microscopy. They are pictures of cervical cancer cells' fluorescently stained nuclei growing in one of 24 chambers in a suitable 3D environment with a continuous flow of medium [46]. Unfortunately, since the images were obtained with low fluorescence microscopy to keep the cells alive, they have mediocre quality and local artefacts, i.e. perfectly black points. Each image stack is a set of pictures of a prolonged experiment conducted in one chamber. We utilized wide-field microscopy for validation.

The pictures are TIFF stacks made of 163 images with 16-bit color depth, one image taken approximately every hour for one week. Individual images have varying intensity levels, i.e., the dark background in one can be brighter than the bright foreground in others. They also possess high levels of 'salt and pepper' noise.

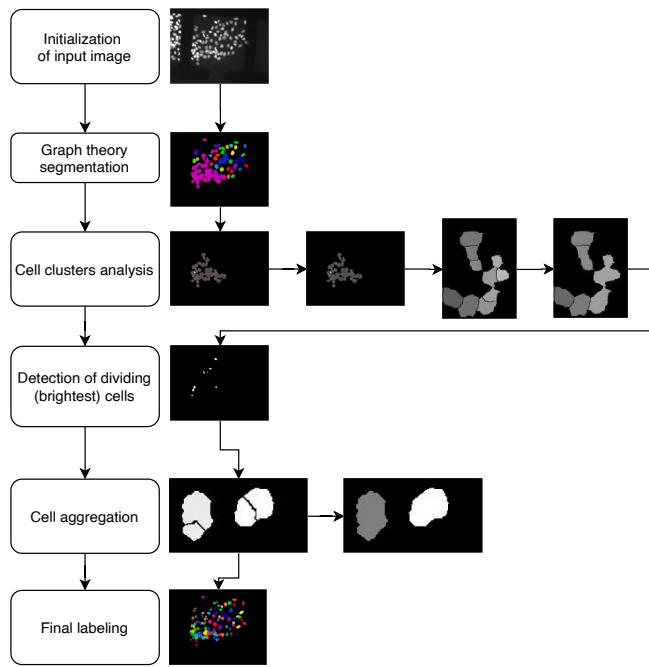


Fig. 1. Block diagram of GRABaCELL method.

The cells are of cervical cancer tumours, from the oldest and most commonly used human cell line, HeLa [35]. HeLa cells are remarkable as they grow exceptionally fast, doubling every 20 to 24 h. Their abnormal vigor is due to telomerase enzyme. Normally, telomeres at DNA's ends are shortened with each cell division, which results in ageing and cell death. However, in HeLa cells, enzymes rebuild telomeres, effectively avoiding death and making them immortal. They are also highly mutated, as they possess 76 to 80 chromosomes, compared to a usual number of 46 [11].

Compared algorithms. Several commonly used algorithms and programs were compared with the GRABaCELL algorithm to check the quality of segmentation and detection:

- watershed method, as described by Monteiro et al. [28];
- Bayes Graph-Cut Model of Beheshti et al. [6];
- Otsu method as it was implemented by Bailey at [4];
- CellProfiler with RobustBackground as a thresholding method [26]
- ImageJ's Trainable Weka Segmentation plugin [3].

2.1.2. Algorithm verification

To verify the proposed algorithm and to show the robustness of the method, additional datasets were tested.

The code also ran on two additional sets of images:

- 21 images of HCT116 cells, which are human colon cancer cells [36],
- 38 images selected from the training set of Data Science Bowl (DSB) dataset, which was a Kaggle-run competition [21]. The algorithm works on oval- and circle-shaped nuclei.

2.2. Proposed method – GRABaCELL

The original code for minimum graph cut segmentation, which is part of the proposed method, was created by Ravindra Gadde and Raghu Yalamanchili at [18].

The algorithm begins with initializing the input image by increasing its contrast, then proceeds with graph theory segmentation, using nuclei centers detected by Circular Hough Transform as source nodes. Radius of the found circles can be optimised by analyzing one random image from the dataset and choosing the mode of radius values. Subsequently, each segmented object is analyzed to determine whether there is more than one center detected by Circular Hough Transform. Should that be the case, watershed segmentation would be required. The algorithm then checks for dividing (brightest) cells, which are further processed. Lastly, segmented objects are aggregated, if too small, or removed altogether, and then labeled. The individual steps are described in more detail below. see Fig. 1.

2.2.1. Initialization of input image

The contrast of the input image is first increased, according to the limits, derived from the bottom 1% and the top 1% of all pixel values. The results of this intensity stretching, along with the original input, are shown in Fig. 2. The input can be either an RGB or a grayscale image.

2.2.2. Graph theory segmentation

The graph cut segmentation is performed, with the background pixel being the maxmin of the input image (to exclude artifacts) and the object pixels being in fact centers of the nuclei, as identified by Circular Hough Transform. Afterwards, to remove objects too small to possibly be cell nuclei, morphological closing and opening of an image, and binary removal are executed.

Circular Hough Transform.

One of the common problems in computer image processing is determining the location, number, or orientation of particular objects in an image, with shapes other than straight lines. The solution of this problem is the Hough Transform, which can be described as a transformation of feature points from x,y-plane to the parameter space.

Equation of a circle in x, y-plane is defined as:

$$r^2 = (x - a)^2 + (y - b)^2 \quad (1)$$

where r is radius of a circle, and a, b are its centers. The circle's parametric representation is:

$$x = a + r * \cos(\theta) \quad (2)$$

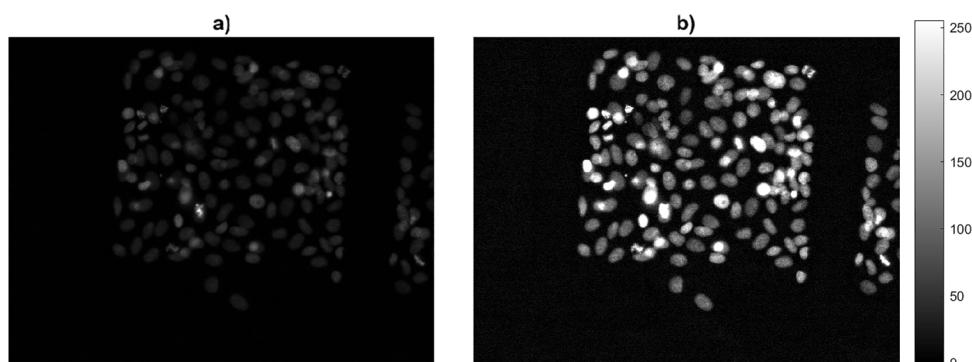


Fig. 2. Input image a) with display scaled to increase visibility, with results b) of stretching intensity values.

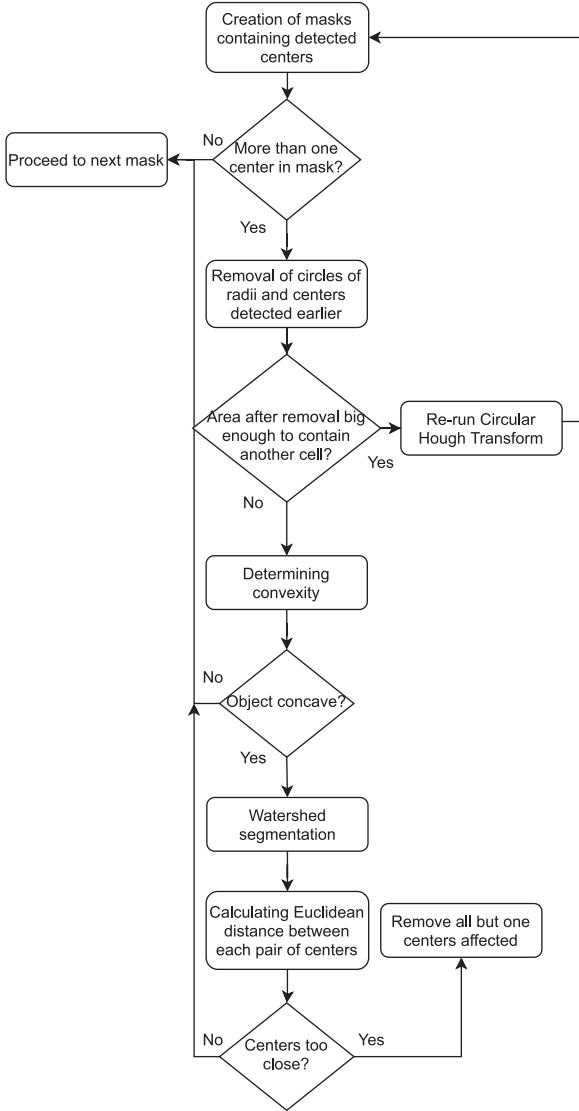


Fig. 3. Sub-diagram for cell cluster analysis.

$$y = b + r * \sin(\theta) \quad (3)$$

with θ being an angle. The total number of parameters is three, therefore the parameter space for a circle will belong to R^3 , as opposed to a line's R^2 with its two parameters of θ and ρ :

$$\rho = x * \cos(\theta) + y * \sin(\theta) \quad (4)$$

where θ represents angle and ρ represents length.

The Hough Transform, in general, can be used to detect any shape, however, with its complexity increasing with the rising number of parameters needed to describe some shape, it is only really considered for shapes with parameters belonging to R^3 at most. In addition, to simplify the parametric representation of a circle, the radius can be assumed to have a constant value, or a limited range [37]. Each point in the geometric space generates a circle in the parameter space. The intersection point of all circles in the parameter space represents the center of the sought circle [29].

Graph cuts and computer vision

An undirected graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is defined as a set of nodes (vertices \mathcal{V}) and a set of undirected edges (\mathcal{E}) that connect these nodes. Each edge is assigned a nonnegative weight (cost). There also exist two special nodes called terminals (a source and a sink). A graph cut is a

partition of a graph, so that the terminals become separated. Cost of a cut is defined as the sum of the weights (costs) of the severed edges:

$$|C| = \sum_{e \in C} w_e \quad (5)$$

A minimum graph cut is a cut with minimum cost.

Graph cut formalism is well suited for the segmentation of images (or N-dimensional volumes), with nodes representing pixels (or voxels) and edges being any neighborhood relationship between those pixels (voxels). There are two additional nodes: a source, which is an "object" terminal, and a sink, that is a "background" terminal. The weights of edges depend on which type of nodes they are connecting [9].

2.2.3. Cell clusters analysis

Subsequently, preparations for cell cluster analysis are made. Centers of nuclei and their radii are detected using Circular Hough transform with higher sensitivity settings than in the previous step and labeling of detected objects is done. Those centers are utilized to create a mask with maximum brightness in their place, and, later on - to create masks with maximum brightness in place of the centers of nuclei in the object with their label number equal to the iteration number. Matrices with objects of label number equal to iteration number are also made.

Eventually, if there is more than one center on the mask, the rest of the cluster analysis continues; otherwise, the algorithm proceeds to the next object. Then, the circles of radii and centers detected earlier are subtracted from the matrix with the object of label number equal to the iteration number, and area of the result is then compared with the area of the smallest circle drawn to determine if there are still centers to detect. If the area of this circle is greater, then Circular Hough transform is run on the matrix with the remaining area. Convexity of a circle is also determined.

Afterwards, watershed segmentation is executed on each labeled, concave object, with the peak points being detected by a very sensitive Circular Hough transform. It is possible to obtain multiple centers of one nucleus this way; therefore, Euclidean distances between each pair of centers are calculated and, if too close, all but one affected are removed.

Matrices with the results of segmentation of all detected objects are then aggregated, and a morphological opening is applied to smooth out the outcome.

The sub-diagram of Fig. 3 pictures the algorithm of cell cluster analysis.

Watershed segmentation.

Watershed segmentation works on an image reimagined as a topographic surface, where intensity represents an elevation in the landscape, with maxima of gradient magnitude being peaks and minima valleys. The surface is then flooded, with water rising from the minima, ultimately forcing the creation of a dam to prevent the two catchment basins from merging. Eventually, only the tops of the dams are visible above the water line, which corresponds to the watershed segmentation algorithm's boundaries. Therefore, an image is understood as catchment basins separated by watersheds - areas from which all rainwater flows into one reservoir or river [55].

Every pixel of the resulting image can be assigned to either of three categories:

- low-lying valleys - minima,
- high-altitude ridges - watershed lines,
- slopes - catchment basins [23].

Instead of letting the water rise from every minimum, water may be allowed to rise only from seeded minima to prevent over-segmentation. However, fully automatic foreground seeding is often tricky. Two seeds in one object lead to incorrect results, as both seeds strive to be divided [7].

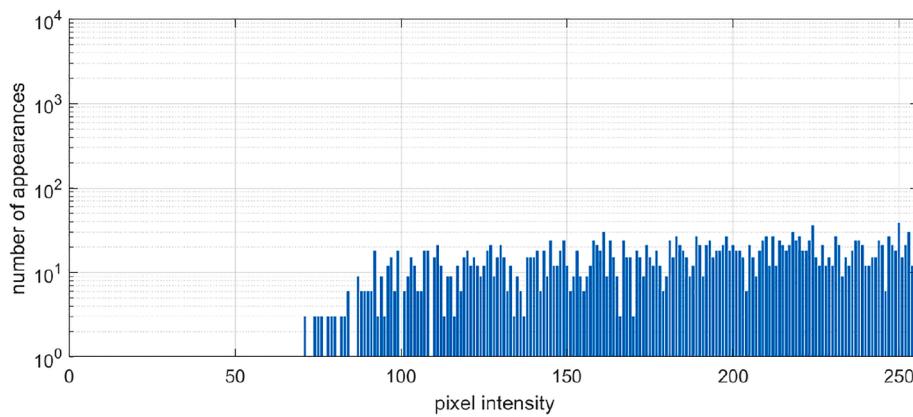


Fig. 4. Distribution of pixel intensity values for one exemplary dividing cell, after removing 0's of background.

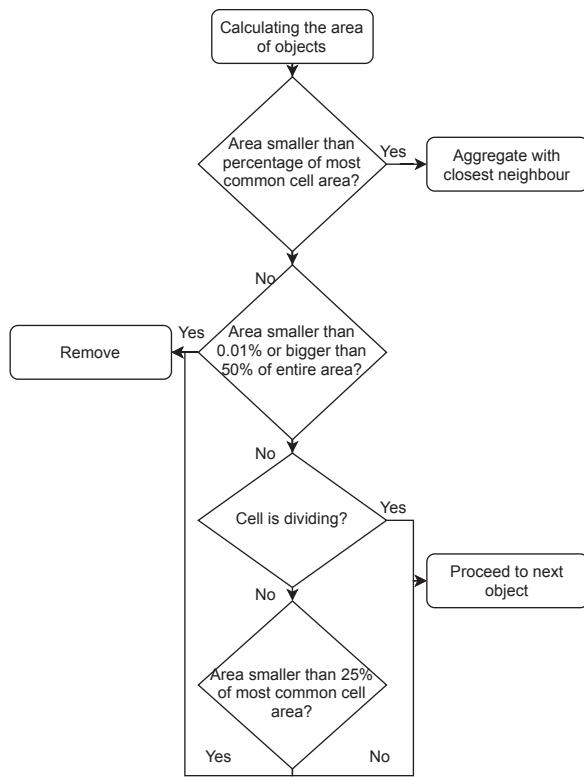


Fig. 5. Sub-diagram for cell aggregation.

2.2.4. Detection of dividing cells

To detect the dividing cells, that are the brightest in an image due to an increased amount of DNA, all objects are labeled, and the mode value of their histograms is obtained. If the mode is located near the bins with the lowest or highest possible values of the histogram, the object is removed from the resulting matrix for further processing: again, every object that was extracted is labeled and then divided into two, based on the pixel brightness values: dividing cells are the top 10% of pixel values, and the bottom 90% are the cells underneath. However, if the area of the brightest cell is more than half of the total area of the object, it is assumed that there is only one cell in the segmented object. The reason for such threshold values is the shape of the histogram, as can be noted on Fig. 4.

2.2.5. Cell aggregation

Objects are aggregated to their closest neighbour if their area is smaller than a certain percentage of the area taken by the circle of the

most common radius (since nuclei are closer to being oval than circular). Objects with an area smaller than 0.01% or bigger than 50% of the entire area of an image are removed altogether, as are objects smaller than a quarter of the area of the circle of the most common radius that was not detected to be dividing cells. We chose 0.01% and 50% as those producing the best results; however, the user is free to change these parameters.

The sub-diagram of Fig. 5 pictures the algorithm of cell aggregation.

2.2.6. Final labeling

In the last step, the segmentation results are transformed into a binary matrix and then multiplied by the original image to produce the final results. Afterwards, the program proceeds to save the original image, colored results after watershed segmentation, final colored results, final results as a binary image, and the final results as a grayscale image.

2.3. Quality measures

The quality measures are used to quantitatively describe the quality of segmentation on various types of input images, using numerous segmentation algorithms and programs.

2.3.1. Dice index

Dice index, also known as Sørensen-Dice coefficient, is the most commonly used statistical tool to gauge the similarity between two samples. It is defined as:

$$DSC = \frac{2|X \cap Y|}{|X| + |Y|} \quad (6)$$

with $|X|$ and $|Y|$ being the number of elements in each set. It can also be written as:

$$DSC = \frac{2TP}{2TP + FP + FN} \quad (7)$$

using the definition of true positive (TP), false positive (FP), and false negative (FN). Its values are in the range of [0, 1], with 1 indicating no difference between the samples and 0 suggesting no similarities found between them.

The samples compared were segmentation results done by hand (gold standard) and results of the GRABaCELL method.

2.3.2. Modified accuracy metric

Another metric is also proposed, with its equation as follows:

$$M = \frac{TP - (FP + FN)}{TP + FN} \quad (8)$$

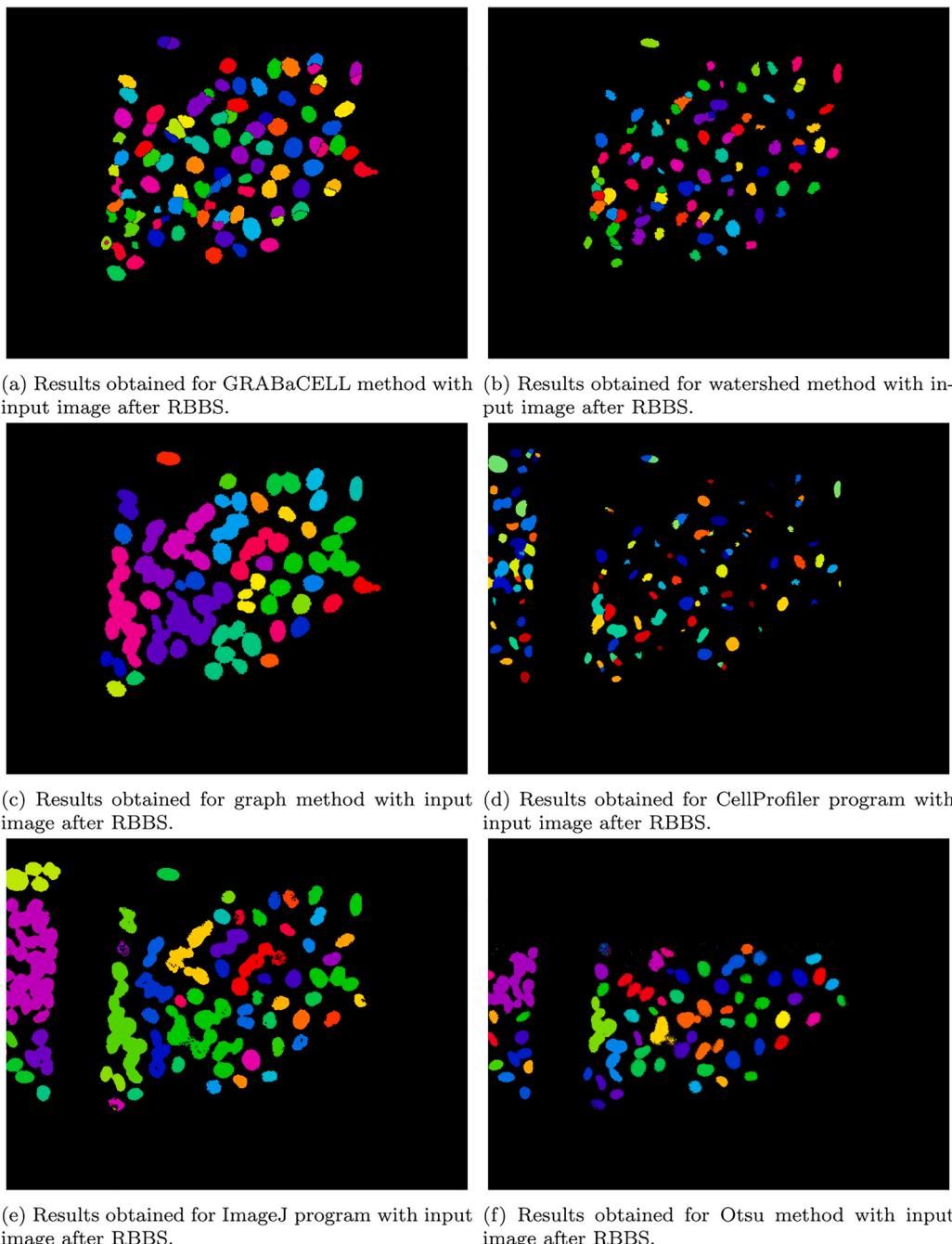


Fig. 6. Results for compared algorithms, for input image after RBBS.

where $TP + FN$ is a total number of objects segmented by hand, TP stands for the number of cells with their centers inside cells segmented by hand, and $FP + FN$ means the absolute difference between the total number of objects segmented by the GRABaCELL method and the total number of objects segmented by hand. Therefore, the metric can also be expressed as:

$$M = TPR - modFPR - FNR \quad (9)$$

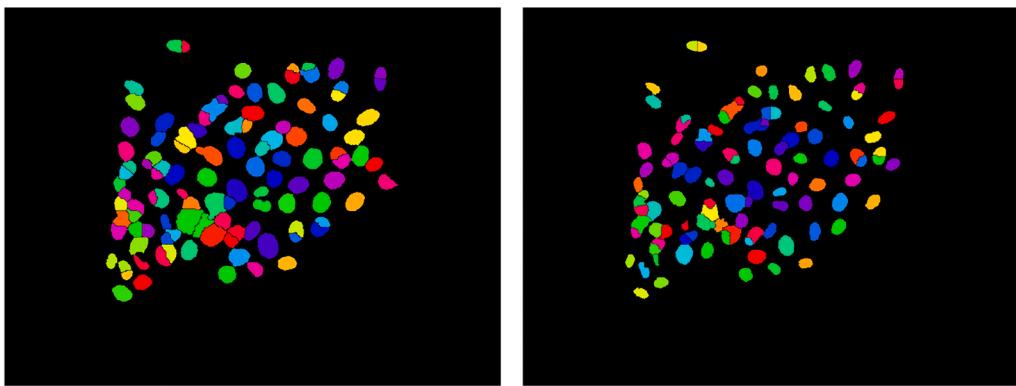
if we assume TPR is true positive rate, FNR – false negative rate, and $modFPR$ is modified false positive rate, with $TP + FN$ (as opposed to FPR 's $FP + TN$) in the denominator. The reason for such a modification is to enable us to quantitatively score well-classified cells and badly classified background. It is important to have as low amount of false positives as possible, as it is easier to divide one cell into two than aggregate a cell

incorrectly segmented into two separate objects.

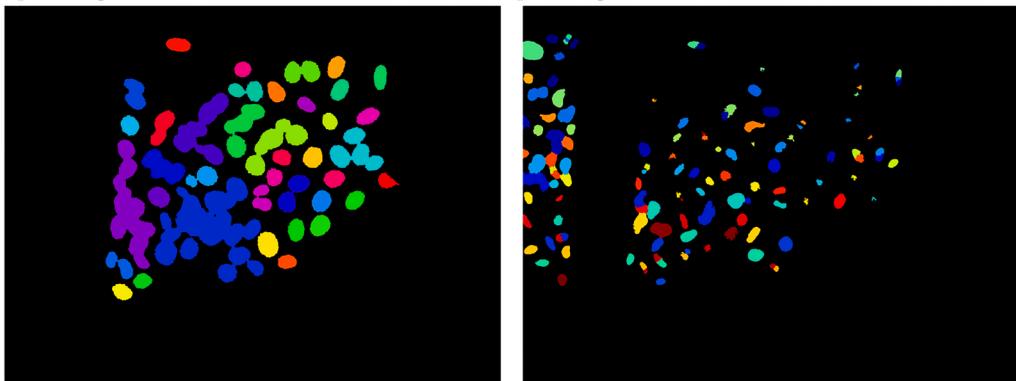
Values of the metric are in the range of $[0, 1]$, and the closer the value is to 1, the closer results are to the gold standard. The metric was introduced to quantitatively compare the results with the gold standard.

2.4. Noise removal

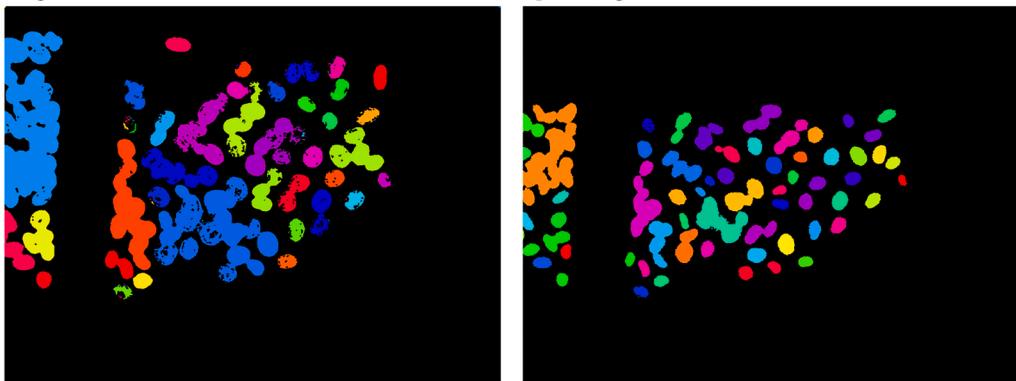
In order to cancel out the noise of input images, two methods were performed – in one, a median filter of size 3, so with 3-by-3 neighborhood is applied, to reduce salt and pepper noise by forcing the median value of the neighborhood in its place, in the other, Rolling Ball Background Subtraction (RBBS) [44] is performed, which removes large spatial variations of background by subtracting a local background value for every pixel averaged over a large ball.



(a) Results obtained for GRABaCELL method with input image after median filter. (b) Results obtained for watershed method with input image after median filter.



(c) Results obtained for graph method with input image after median filter. (d) Results obtained for CellProfiler program with input image after median filter.



(e) Results obtained for ImageJ program with input image after median filter. (f) Results obtained for Otsu method with input image after median filter.

Fig. 7. Results for compared algorithms, for input image after median filter.

3. Results and discussion

The exemplary results of algorithms compared are shown in Fig. 6 and Fig. 7. Input images were denoised, using Rolling Ball Background Subtraction and median filtering. The proposed method, GRABaCELL, produces the best results visually, as it was able to not only segment clumps of cells but also cover most of the cells' area. Watershed's results are similar; however, most of the cells' area was segmented out, which is more prevalent in input denoised by RBBS. The graph method in both cases did not manage to segment clumps of cells. CellProfiler segmented out a large number of cells in the middle. ImageJ managed to find most of the cells; however, it did not segment clumps; it also left out centers of several objects, which is more visible in the image after median filtering. The Otsu method, too, could not segment cell clumps.

Table 1
Values of Dice index for all methods.

method	average value	standard deviation
GRABaCELL algorithm - RBBS	0.7740	0.0855
GRABaCELL algorithm - median	0.7397	0.1276
watershed - RBBS	0.6217	0.1409
watershed - median	0.6928	0.1894
graph - RBBS	0.3436	0.1419
graph - median	0.3811	0.1208
CellProfiler - RBBS	0.5486	0.1766
CellProfiler - median	0.5489	0.1861
ImageJ - RBBS	0.4590	0.2058
ImageJ - median	0.3103	0.2084
Otsu - RBBS	0.5880	0.0869
Otsu - median	0.5822	0.0899

Table 2

Wilcoxon signed rank test with 5% significance level results for Dice index for proposed GRABaCELL method; yellow color indicates failure to reject the null hypothesis, while the number is p-value.

-	GRABaCELL RBBS	GRABaCELL median
GRABaCELL RBBS	1	1.8E-05
GRABaCELL median	1.8E-05	1
watershed RBBS	6.86E-08	0.014
watershed median	0.004	0.033
graph RBBS	4.55E-09	3.06E-09
graph median	4.55E-09	3.06E-09
CellProfiler RBBS	4.55E-09	3.06E-09
CellProfiler median	4.55E-09	5.71E-09
ImageJ RBBS	4.55E-09	7.3E-09
ImageJ median	4.55E-09	1.92E-08
Otsu RBBS	4.55E-09	3.06E-09
Otsu median	4.55E-09	3.06E-09

The GRABaCELL method is developed on Matlab on a PC with a 4.60 GHz CPU and 32 GB RAM, taking an average of 0.5 min to process an image. Images with higher numbers of cells take longer than those with fewer – however, no image took longer than 2 min to process. The script was run on the whole set of 163 images, and to quantitatively compare the performance, 33 images were selected, segmented by hand and cells in them counted.

In Table 1, it can be observed that as measured by the Dice index, results obtained for the GRABaCELL method are the best on both types of input images. However, watershed segmentation runs on input image after median filtering is close behind the proposed algorithm run on the same input image type. Moreover, the Wilcoxon signed-rank test with a 5% significance level indicates a failure to reject the null hypothesis of zero median, as is in Table 2.

Nevertheless, there is no dispute that the GRABaCELL method run on input image after RBBS is vastly better than any other algorithm, as seen in boxplots represented in Fig. 8 and 9. The Wilcoxon signed-rank test indicates rejection of the null hypothesis of zero median with all other

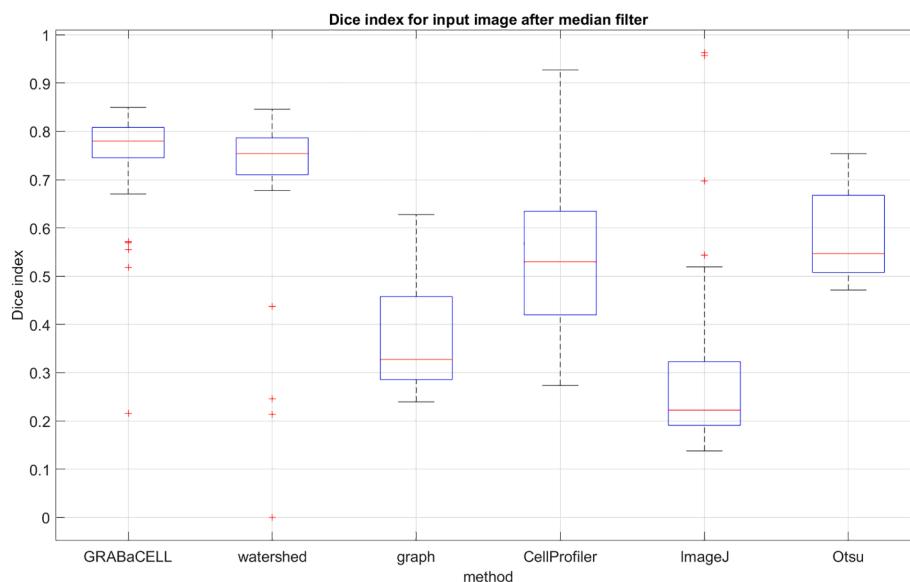
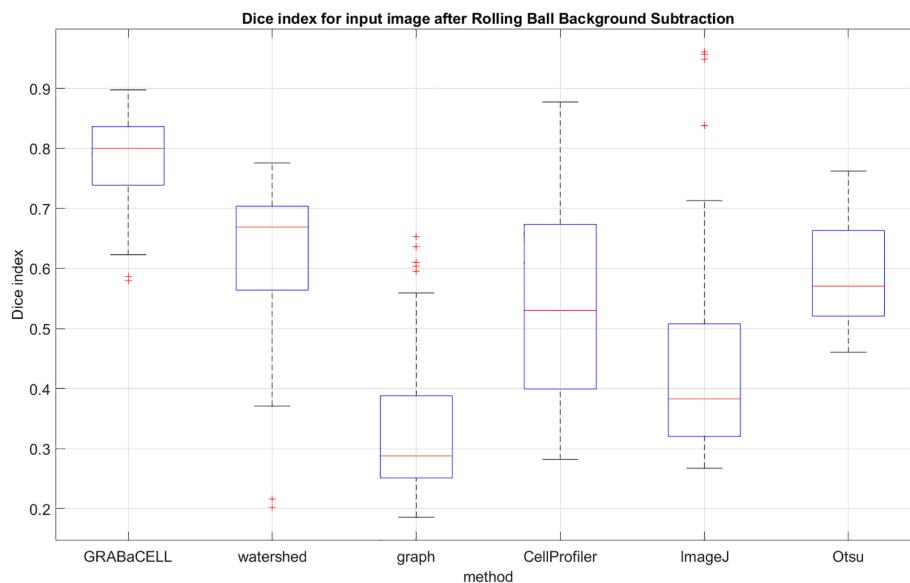
**Fig. 8.** Dice index for input image after median filter.**Fig. 9.** Dice index for input image after RBBS.

Table 3

Values of modified accuracy metric for all methods.

method	average value	standard deviation
GRABaCELL algorithm - RBBS	0.9048	0.0560
GRABaCELL algorithm - median	0.9013	0.0574
watershed - RBBS	0.7758	0.1849
watershed - median	0.6269	0.2500
graph - RBBS	0.6188	0.2015
graph - median	0.5417	0.2159
CellProfiler - RBBS	0.5561	0.2326
CellProfiler - median	0.5210	0.2488
ImageJ - RBBS	0.3393	0.2840
ImageJ - median	0.3317	0.2312
Otsu - RBBS	0.6631	0.1927
Otsu - median	0.4920	0.2323

methods other than the same method run on input image after median filtering. The test results for the proposed GRABaCELL method can be seen in Fig. 2.

It should be noted that CellProfiler and ImageJ are also capable of producing good results, as can be indicated by the presence of ImageJ's outliers and CellProfiler's upper whiskers approaching the Dice index's perfect score of 1. Their average score is so low because the images' brightness was inhomogeneous - on some, the brightness of the background was higher than the brightness of the foreground on others. Watershed also generated good results on median filtered images; however, the presence of extremely low in value outliers greatly undervalued the average score.

When it comes to modified accuracy metric, the proposed method also produces the best results, as seen in Table 3. However, the average and standard deviation for input image after median filtering and input image after RBBS are very close in value – so close that Wilcoxon signed-rank test with a 5% significance level indicates a failure to reject the null hypothesis of zero median between them. Nonetheless, the test showed a rejection of the null hypothesis of zero median between all other methods, as can be noted on boxplots in Fig. 10 and 11, since the red line of median did not fit into any other boxplot, but it's own. Test results for

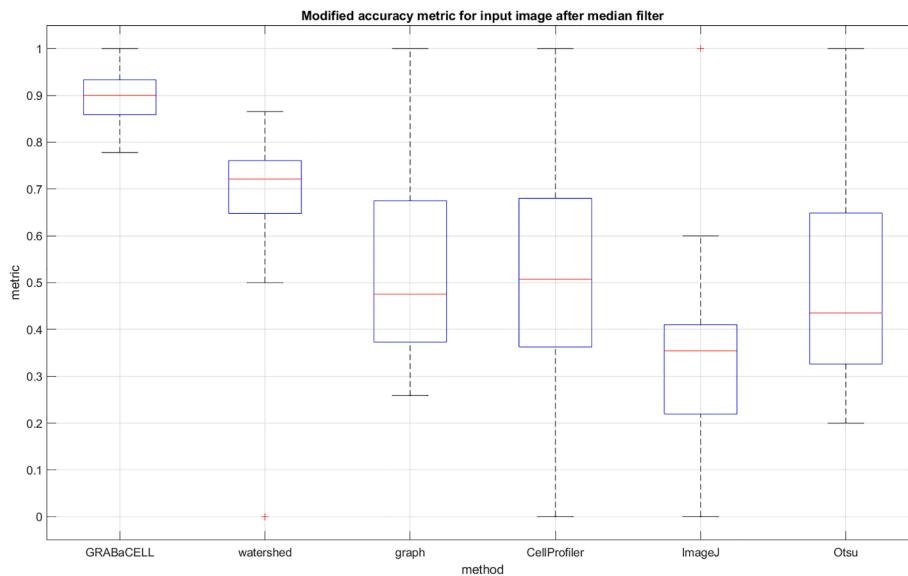


Fig. 10. Modified accuracy metric for input image after median filter. The higher metric value is better (values range from 0 to 1).

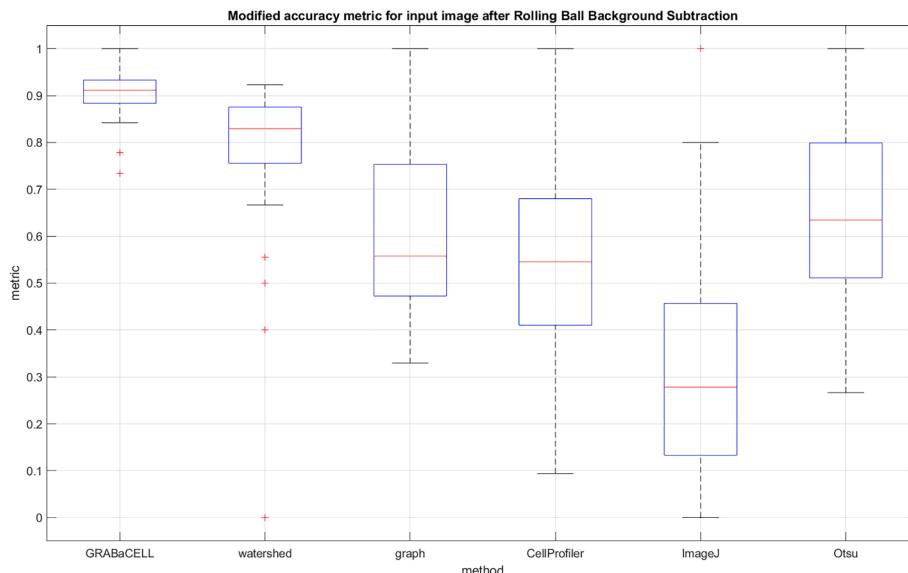


Fig. 11. Modified accuracy metric for input image after RBBS. The higher metric value is better (values range from 0 to 1).

Table 4

Wilcoxon signed rank test with 5% significance level results for modified accuracy index for proposed GRABaCELL method; yellow color indicates failure to reject the null hypothesis, while the number is p-value.

-	GRABaCELL RBBS	GRABaCELL median
GRABaCELL RBBS	1	0.004
GRABaCELL median	0.004	1
watershed RBBS	0.023	0.381
watershed median	1.42E-08	0.00011
graph RBBS	3.06E-09	5.05E-09
graph median	3.06E-09	6.46E-09
CellProfiler RBBS	3.06E-09	3.06E-09
CellProfiler median	3.06E-09	9.32E-09
ImageJ RBBS	1.19E-08	2.15E-09
ImageJ median	4.55E-09	4.39E-08
Otsu RBBS	3.03E-09	1.11E-09
Otsu median	3.03E-09	3.03E-09

the proposed GRABaCELL method are shown in Table 4.

We have also used the GRABaCELL method to produce results for two additional datasets - Kaggle's Data Science Bowl and HCT116 cells, 38 and 21 images, respectively. The results were later compared with the segmentation of HeLa cells. Kaggle provided its own gold standard. However, its quality was discussable at best, which led us to manually segmenting images by hand. As can be seen in Fig. 12, the Dice index

value was high, especially that of HCT116 cells. However, the Kaggle-provided gold standard has the lowest Dice index due to incomplete segmentation of cells, as can be noted in Fig. 14b, that has several missing cells from Fig. 14a, which can be found in Fig. 14c. Modified accuracy metric, presented in Fig. 13, was also high. However, the Kaggle-provided golden standard produced a slightly higher median of metric, much higher third quartile value and much lower first quartile value than those of manual gold standard. However, the Wilcoxon signed-rank test fails to reject the hypothesis of zero median between them. HTC116's results are interesting in being nearly perfect, which is explained by them being images of simple, circular and singular cells, as can be noted in Fig. 15. Those facts prove the robustness of the algorithm.

The algorithm is susceptible to the value of the most common radius. It is recommended to segment images by hand until at least several cells are obtained to calculate the mode. Otherwise, objects can be aggregated with their closest neighbour or deleted altogether if they are much smaller than expected. The value of the most common radius is a design constraint - the radii of all objects should be similar so that they are not removed.

It is important to notice that nearly all compared algorithms could produce good results - only watershed, the previous leader among algorithms compared with the proposed method, was unable to generate a perfect segmentation. However, on average, it performed better than the

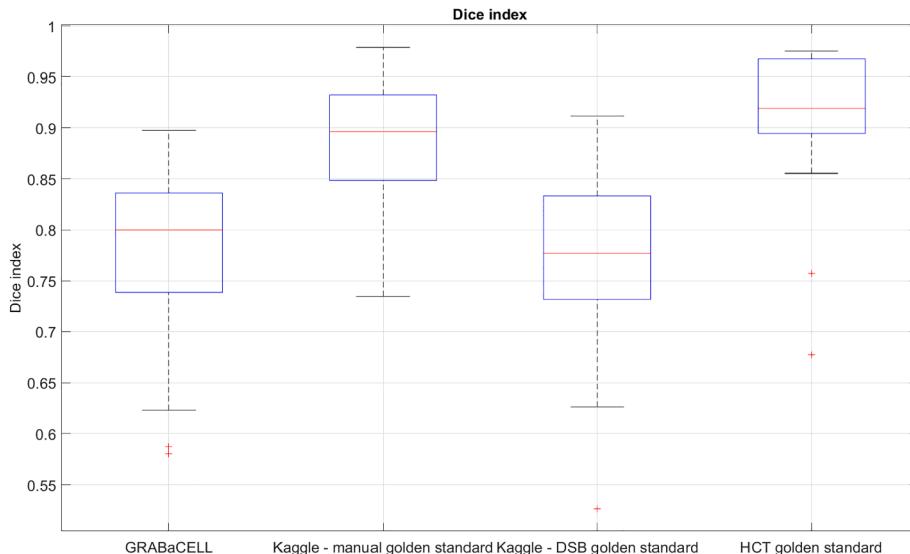
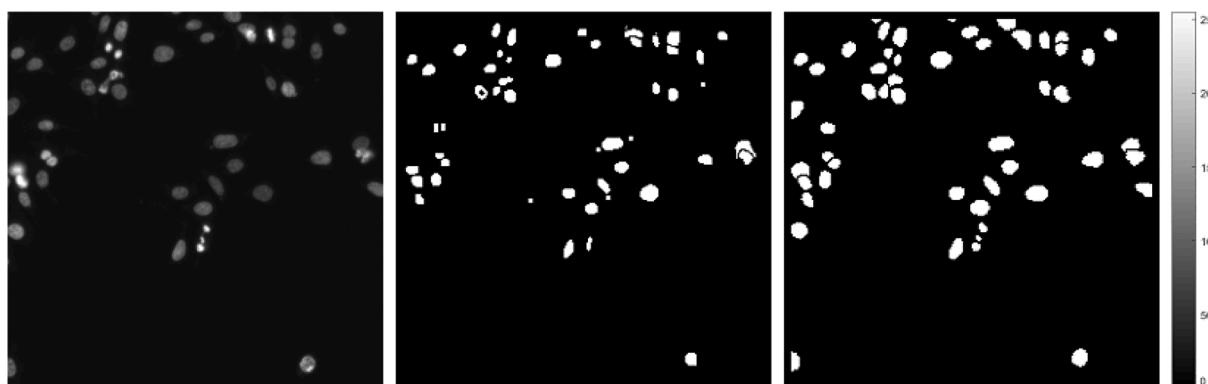


Fig. 12. Dice index for compared sets of images:GRABaCELL with HeLa cells, Kaggle dataset with two golden standards: (manual and DSB) and HCT cells.



(a) Original image. (b) Kaggle golden standard. (c) Manual golden standard.

Fig. 14. Image from Kaggle dataset: a) is original image, b) Kaggle-provided golden standard and c) is manual golden standard.

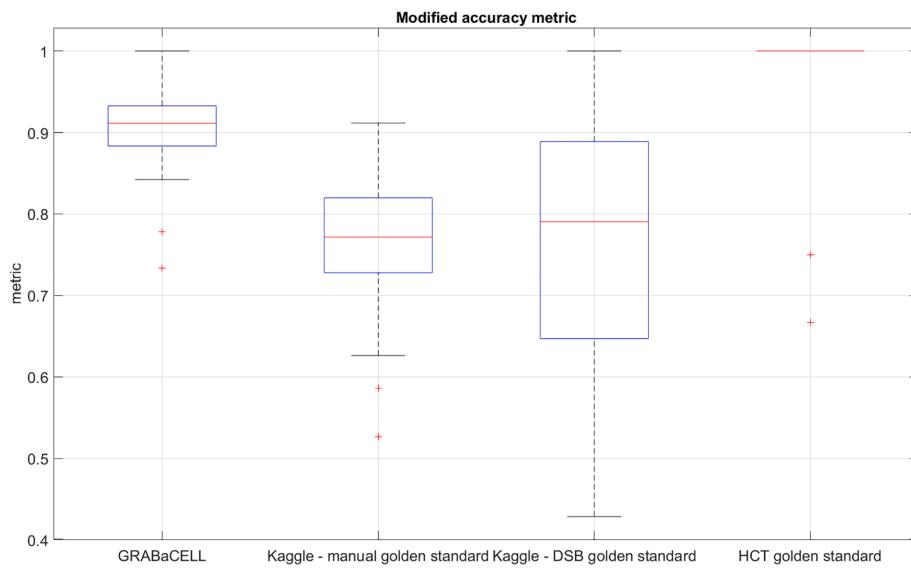


Fig. 13. Modified accuracy metric for compared sets of images: GRABaCELL with HeLa cells, Kaggle dataset with two golden standards: (manual and DSB) and HCT cells.

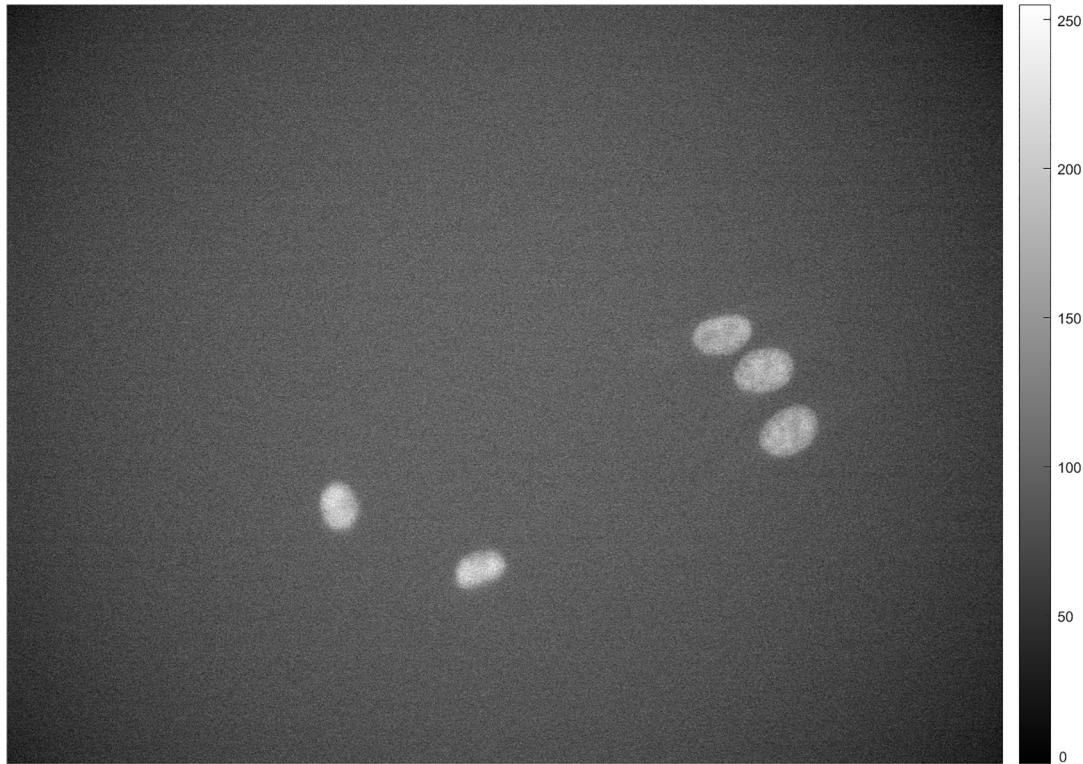


Fig. 15. Image example from the HCT116 dataset.

rest. CellProfiler, ImageJ and Otsu's average score is much lower than their best because of the inhomogeneous brightness of objects. Meanwhile, graph theory segmentation performed best with a minimal number of cells in the image, which skews the metric results, as we treat each image with the same weight.

The percentage mentioned in 2.2.5 was obtained by comparing results for both types of input for values in the range of [20, 140]. The best results happened to be 90% for input image after median filtering and 60% for input image after RBBS and were later used for comparison with other methods and programs.

The results of noise removal are shown on Fig. 16. RBBS removed the

compartment from the image, making it grainy, while median filter only partly denoised it by blurring.

The main difficulty in segmenting images was having to correctly label cells in cell clumps - where cells were partially stacked on top of each other despite being in 2D space - as it made determining exact borders difficult even visually, as can be noted on Fig. 17. Shapes and colors often blended, which added to the difficulty.

It is worth noting that the GRABaCELL method is the slowest compared, with watershed being the fastest, as can be seen in Table 5. However, it is fast enough to process all data obtained during the experiment before the next round of testing is done.

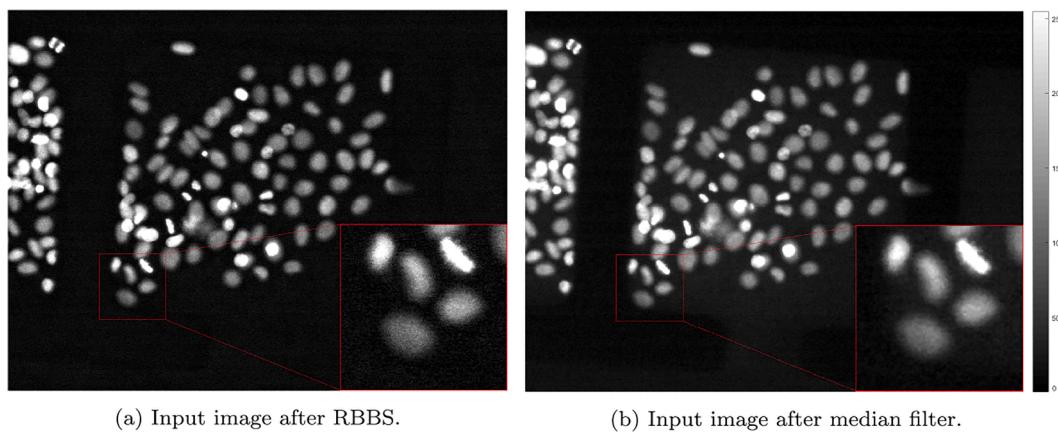
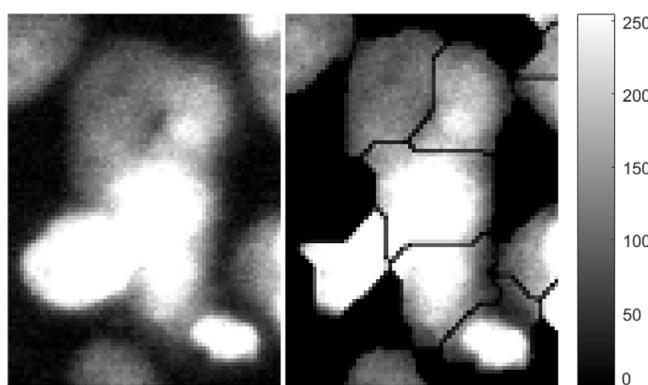
**Fig. 16.** Results of noise removal.**Fig. 17.** Clump of cells with GRABaCELL segmentation.

Table 5
Total time elapsed for all compared methods and programs,
for all 33 images.

method	time [s]
GRABaCELL algorithm	1566
watershed	38
graph	167
CellProfiler	90
ImageJ	175
Otsu	439

4. Conclusions

In this paper, we suggested a metric for calculating the segmentation quality and presented a method for cell segmentation and counting. The novelty was in combining Circular Hough transform, calculating convexity of objects with watershed and graph cut segmentation, and modifying accuracy metric, commonly used for determining the quality of segmentation.

We demonstrated that the GRABaCELL method produces remarkably good results for the Dice index and our suggested metric. We also compared the outcome of the proposed method to three commonly used segmentation methods (watershed, graph-based and Otsu) and two widely incorporated programs (CellProfiler and ImageJ). We proved our method outperforms them in both metrics, maintaining the high value of these indices and their relatively small spread. Code is available at <https://github.com/khajd/cell-segmentation>.

It is worth noting that the GRABaCELL algorithm performs vastly better than commercially used programs like CellProfiler and ImageJ, generating output images much closer to the gold standard. Although it

might not be useful for diagnostic tests due to the time required for segmentation of all images, it excels at drug tests thanks to superb accuracy.

CRediT authorship contribution statement

Katarzyna Hajdowska: Software, Methodology, Data curation, Validation, Writing - original draft. **Sebastian Student:** Methodology, Validation, Resources, Writing - review & editing. **Damian Borys:** Conceptualization, Methodology, Validation, Writing - original draft, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Calculations were performed on the Ziemowit computer cluster in the Laboratory of Bioinformatics and Computational Biology, created in the EU Innovative Economy Programme POIG.02.01.00–00-166/08 and expanded in the POIG.02.03.01-00-040/13 project. Data analysis was partially carried out using the Biostest Platform developed within Project PBS3/B3/32/2015 financed by the Polish National Centre of Research and Development (NCBiR). This work was partially supported by the Silesian University of Technology Grants No. 02/010/RGH19/0161 (DB), 02/040/RGJ21/1011 (SS) and internal SUT grant 02/040/BK_21/1010 (BK-215/RAu1/2021) (KH).

References

- [1] Y. Al-Kofahi, W. Lassoued, W. Lee, B. Roysam, Improved automatic detection and segmentation of cell nuclei in histopathology images, *IEEE Trans. Bio-medical Eng.* 57 (4) (2010) 841–852, <https://doi.org/10.1109/TBME.2009.2035102>.
- [2] M. Alilou, V. Kovalev, V. Taimouri, Segmentation of cell nuclei in heterogeneous microscopy images: a reshapable templates approach, *Comput. Med. Imaging Graph.* 37 (7–8) (2013) 488–499, <https://doi.org/10.1016/j.compmedim.2013.07.004>.
- [3] I. Arganda-Carreras, V. Kaynig, C. Rueden, K.W. Eliceiri, J. Schindelin, A. Cardona, H. Sebastian Seung, Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification, *Bioinformatics (Oxford, England)* 33 (15) (2017) 2424–2426, <https://doi.org/10.1093/bioinformatics/btx180>.
- [4] S. Bailey. Teaching notebook for total imaging newbies, 2018. URL:<https://www.kaggle.com/stkbailey/teaching-notebook-for-total-imaging-newbies>. (accessed: 01.08.2020).
- [5] S. Bana, D. Kaur, Fingerprint recognition using image segmentation, *Int. J. Adv. Eng. Sci. Technol.* 5 (1) (2011) 12–23.
- [6] M. Beheshti, J. Faichney, A. Gharipour, Bio-Cell Image Segmentation Using Bayes Graph-Cut Model. In 2015 International Conference on Digital Image Computing:

- Techniques and Applications (DICTA), pages 1–5, 1 2015. doi: 10.1109/DICTA.2015.7371241.
- [7] E. Bengtsson, J. Lindblad, Robust cell image segmentation methods, *Pattern Recogn. Image Anal.* 14 (2) (2004) 157–167.
 - [8] G. Blin, D. Sadurska, R. Portero Migueles, N. Chen, J.A. Watson, S. Lowell, Nessys: A new set of tools for the automated detection of nuclei within intact tissues and dense 3D cultures, *PLoS Biology* 17 (8) (2019), <https://doi.org/10.1371/journal.pbio.3000388> e3000388.
 - [9] Y. Boykov and M.-P. Jolly, Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. Proceedings. Eighth IEEE International Conference on Computer Vision: 2001. Vol. 1, 1: 105–112, 07 2001. doi: 10.1109/ICCV.2001.937505.
 - [10] J. Chalfoun, M. Majurski, A. Dima, C. Stuelten, A. Peskin, M. Brady, FogBank: a single cell segmentation across multiple cell lines and image modalities, *BMC Bioinformatics* 15 (431) (2014) 12, <https://doi.org/10.1186/s12859-014-0431-x>.
 - [11] A. del Cariou. The good, the bad, and the HeLa, 2014. URL:<https://berkeleysciencereview.com/article/good-bad-hela/>. (accessed: 05.11.2020).
 - [12] I. Despotović, V. Jelača, E. Vansteenikste, W. Philips, Noise-robust method for image segmentation, in: *International Conference on Advanced Concepts for Intelligent Vision Systems*, Springer, 2010, pp. 153–162.
 - [13] S. Dimopoulos, C.E. Mayer, F. Rudolf, J. Stelling, Accurate cell segmentation in microscopy images using membrane patterns, *Bioinformatics* (Oxford, England) 30 (18) (2014) 2644–2651, <https://doi.org/10.1093/bioinformatics/btu302>.
 - [14] X. Du, S. Dua. Segmentation of fluorescence microscopy cell images using unsupervised mining. *Open Medical Informatics J.*, 4: 41, 5 2010. doi: 10.2174/1874431101004020041.
 - [15] N. El abadi. Detection and Segmentation of Human Face. *Inte. J. Adv. Res. Computer Commun. Eng.*, 4: 90–94, 02 2015. doi: 10.17148/IJARCCE.2015.4220.
 - [16] M. Farhan, P. Ruusuvuori, M. Emmenlauer, P. Rämö, C. Dehio, O. Yli-Harja, Multi-scale Gaussian representation and outline-learning based cell image segmentation, *BMC bioinformatics* 14 (Suppl 10) (2013) S6, <https://doi.org/10.1186/1471-2105-14-S10-S6>.
 - [17] P.M. Ferreira. Microscopy image segmentation by active contour models. PhD thesis, Universidade Nova de Lisboa, 2014.
 - [18] R. Gadde and R. Yalamanchili. Teaching notebook for total imaging newbies, 2011. URL:<https://masterravi.wordpress.com/2011/05/24/interactive-segmentation-using-graph-cutsmatlab-code/>. (accessed: 04.08.2020).
 - [19] M. Gamarra, E. Zurek, H.J. Escalante, L. Hurtado, H. San-Juan-Vergara, Split and merge watershed: A two-step method for cell segmentation in fluorescence microscopy images, *Biomed. Signal Process. Control* 53 (2019) 101575, <https://doi.org/10.1016/j.bspc.2019.101575>.
 - [20] O. Hilsenbeck, M. Schwarzsicher, D. Loeffler, S. Dimopoulos, S. Hasteirer, C. Marr, F. Theis, T. Schroeder, FastER: A User-Friendly tool for ultrafast and robust cell segmentation in large-scale microscopy, *Bioinformatics* (Oxford, England) 33 (2017), <https://doi.org/10.1093/bioinformatics/btx107>.
 - [21] Kaggle. 2018 Data Science Bowl, 2018. URL:<https://www.kaggle.com/c/data-science-bowl-2018>. (accessed: 29.04.2021).
 - [22] B. Ko, M. Seo, J.-Y. Nam, Microscopic cell nuclei segmentation based on adaptive attention window, *J. Digital Imaging* 22 (3) (2009) 259–274, <https://doi.org/10.1007/s10278-008-9129-9>.
 - [23] A. Kornilov, I. Safonov, An Overview of Watershed Algorithm Implementations in Open Source Libraries, *J. Imaging* 4 (10) (2018) 123, <https://doi.org/10.3390/jimaging4100123>.
 - [24] L. Kostrykin, C. Schnörr, K. Rohr, Globally optimal segmentation of cell nuclei in fluorescence microscopy images using shape and intensity information, *Medical Image Anal.* 58 (2019) 101536, <https://doi.org/10.1016/j.media.2019.101536>.
 - [25] J. Kumar, G.R. Kumar, V.K. Reddy, Review on image segmentation techniques, *Int. J. Sci. Res. Eng. Technol. (IJSRET)* 3 (2014) 992–997.
 - [26] Y.S. Lau, L. Xu, Y. Gao, R. Han, Automated muscle histopathology analysis using Cell Profiler, *Skeletal muscle* 8 (1) (2018) 32, <https://doi.org/10.1186/s13395-018-0178-6>.
 - [27] A.X. Lu, O.Z. Kraus, S. Cooper, A.M. Moses, Learning unsupervised feature representations for single cell microscopy images with paired cell inpainting, *PLoS Comput. Biol.* 15 (9) (2019) e1007348, <https://doi.org/10.1371/journal.pcbi.1007348>.
 - [28] A.C.B. Monteiro, Y. Iano, R.P. França, Detecting and counting of blood cells using watershed transform: An improved methodology, in: Y. Iano, R. Arthur, O. Saotome, V. Vieira Estrada, H.J. Loschi (Eds.), *Proceedings of the 3rd Brazilian Technology Symposium*, Springer International Publishing, Cham, 2019, pp. 301–310.
 - [29] C. More, A. Patil, Circular Hough Transform For Detecting And Measuring Circles of Object, *Int. J. Recent Innovation Trends Comput. Commun.* 3 (2015) 563–566.
 - [30] J.L. Mueller, Z.T. Harmany, J.K. Mito, S.A. Kennedy, Y. Kim, L. Dodd, J. Geradts, D. G. Kirsch, R.M. Willett, J.Q. Brown, N. Ramanujam, Quantitative Segmentation of Fluorescence Microscopy Images of Heterogeneous Tissue: Application to the Detection of Residual Disease in Tumor Margins, *PloS one* 8 (6) (2013) e66198, <https://doi.org/10.1371/journal.pone.0066198>.
 - [31] S. Pandey, P.R. Singh, J. Tian, An image augmentation approach using two-stage generative adversarial network for nuclei image segmentation, *Biomed. Signal Process. Control* 57: 101782 (2020) 1, <https://doi.org/10.1016/j.bspc.2019.101782>.
 - [32] D.L. Pham, C. Xu, J.L. Prince, Current methods in medical image segmentation, *Ann. Review Biomed. Eng.* 2 (1) (2000) 315–337.
 - [33] A. Pinidiyaarachchi. Digital Image Analysis of Cells: Applications in 2D, 3D and Time. PhD thesis, Universitetsbiblioteket, 2009.
 - [34] J. Qi, B. Wang, N. Pelaez, L. Rebay, R.W. Carthew, A.K. Katsaggelos, L.A. N. Amaral, *Drosophila* eye nuclei segmentation based on graph cut and convex shape prior, in: *2013 IEEE International Conference on Image Processing*, 2013, pp. 670–674.
 - [35] R. Rahbari, T. Sheahan, V. Modes, P. Collier, C. Macfarlane, R.M. Badge, A novel L1 retrotransposon marker for HeLa cell line identification, *Biotechniques* 46 (4) (2009) 277–284.
 - [36] A. Rajput, I. Dominguez San Martin, R. Rose, A. Beko, C. Levea, E. Sharratt, R. Mazurchuk, R.M. Hoffman, M.G. Brattain, and J. Wang, Characterization of HCT116 human colon cancer cells in an orthotopic model. *The Journal of surgical research*, 147 (2): 276–81, 6 2008. doi: 10.1016/j.jss.2007.04.021.
 - [37] M. Rizon, H. Yazid, P. Saad, A.Y. Md Shakaff, A.R. Saad, M. Sugisaka, S. Yaacob, M. Mamat, and M. Karthigaya. Object Detection using Circular Hough Transform. *American Journal of Applied Sciences*, 2 (12): 1606–1609, 1 2005. doi: 10.3844/ajasp.2005.1606.1609.
 - [38] S. Robinson, L. Guyon, J. Nevalainen, M. Toriseva, M. Åkerfelt, M. Nees, Segmentation of Image Data from Complex Organotypic 3D Models of Cancer Tissues with Markov Random Fields, *Plos one* 10 (12) (2015) e0143798, <https://doi.org/10.1371/journal.pone.0143798>.
 - [39] M. Rohde, 4 - microscopy, in: F. Rainey, A. Oren (Eds.), *Taxonomy of Prokaryotes*, volume 38 of *Methods in Microbiology*, Academic Press, 2011, pp. 61–100, <https://doi.org/10.1016/B978-0-12-387730-7.00004-8>.
 - [40] B.F. Rost. *Fluorescence microscopy*, volume II By F.W.D. Rost Cambridge University Press, Cambridge and New York (1995). *Scanning*, 18 (8): 593–593, 1996. doi: 10.1002/scia.4950180810. URL:<https://onlinelibrary.wiley.com/doi/abs/10.1002/scia.4950180810>.
 - [41] L. Rundo, A. Tangherloni, D.R. Tyson, R. Betta, C. Militello, S. Spolaor, M.S. Nobile, D. Besozzi, A.L.R. Lubbock, V. Quaranta, G. Mauri, C.F. Lopez, P. Cazzaniga, ACDC: Automated cell detection and counting for time-lapse fluorescence microscopy, *Appl. Sci.* 10 (18) (2020) 6187, <https://doi.org/10.3390/app10186187>.
 - [42] S.K. Sadanandan, P. Ranefall, S. Le Guyader, C. Wählby, Automated training of deep convolutional neural networks for cell segmentation, *Sci. Rep.* 7 (1) (2017) 7860, <https://doi.org/10.1038/s41598-017-07599-6>.
 - [43] J. Song, L. Xiao, and Z. Lian. Contour-Seed Pairs Learning-Based Framework for Simultaneously Detecting and Segmenting Various Overlapping Cells/Nuclei in Microscopy Images. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 27 (12): 5759–5774, 12 2018. doi: 10.1109/TIP.2018.2857001.
 - [44] S.R. Sternberg, Biomedical Image Processing, *Computer* 16 (1) (1983) 22–34, <https://doi.org/10.1109/MC.1983.1654163>.
 - [45] S. Student, M. Danch-Wierzchowska, K. Gorczewski, D. Borys, Automatic segmentation system of emission tomography data based on classification system, in: F. Ortuño, I. Rojas (Eds.), *Bioinformatics and Biomedical Engineering*, Springer International Publishing, Cham, 2015, pp. 274–281, https://doi.org/10.1007/978-3-319-16483-0_28.
 - [46] S. Student, S. Milewska, Z. Ostrowski, K. Gut, I. Wandzik, Microchamber microfluidics combined with thermogelable glycomicrogels - Platform for single cells study in an artificial cellular microenvironment, *Mater. Sci. Eng.: C* 119 (2020) 02, <https://doi.org/10.1016/j.msec.2020.111647>.
 - [47] P. Ulmas, I. Liiv. Segmentation of Satellite Imagery using U-Net Models for Land Cover Classification. *arXiv preprint arXiv:2003.02899*, 2020.
 - [48] E.K. Wang, X. Zhang, L. Pan, C. Cheng, A. Dimitrakopoulou-Strauss, Y. Li, N. Zhe, Multi-Path Dilated Residual Network for Nuclei Segmentation and Detection, *Cells* 8 (5) (2019) 5, <https://doi.org/10.3390/cells8050499>.
 - [49] G. Wang and N. Fang, Detecting and tracking nonfluorescent nanoparticle probes in live cells. Methods in enzymology, 504: 83–108, 1 2012. doi: 10.1016/B978-0-12-391857-4.00004-5.
 - [50] W. Wang, D.A. Taft, Y.-J. Chen, J. Zhang, C.T. Wallace, M. Xu, S.C. Watkins, and J. Xing, Learn to segment single cells with deep distance estimator and deep cell detector. *Computers in biology and medicine*, 108: 133–141, 5 2019. doi: 10.1016/j.combiomed.2019.04.006.
 - [51] Z. Wang, A New Approach for Segmentation and Quantification of Cells or Nanoparticles, *IEEE Trans. Industr. Inf.* 12 (3) (2016) 962–971, <https://doi.org/10.1109/TII.2016.2542043>.
 - [52] Y. Wiseman, E. Fredj, Contour extraction of compressed JPEG images, *Journal of Graphic Tools* 6 (3) (2002) 37–42, <https://doi.org/10.1080/1067651.2001.10487544>.
 - [53] W. Wu, A. Chen, L. Zhao, and J. Corso. Brain tumor detection and segmentation in a CRF (conditional random fields) framework with pixel-pairwise affinity and superpixel-level features. *International journal of computer assisted radiology and surgery*, 9: 241–253, 07 2013. doi: 10.1007/s11548-013-0922-7.
 - [54] N. Xu, N. Ahuja, R. Bansal, Object segmentation using graph cuts based active contours, *Comput. Vis. Image Underst.* 107 (3) (2007) 210–224.
 - [55] Y. Zhang, D. Xu, Improved Watershed Algorithm for Cell Image Segmentation, *Adv. Mater. Res.* 464–468 (07) (2012) 546–547, <https://doi.org/10.4028/www.scientific.net/AMR.546-547.546>.
 - [56] M. Zhao, J. An, H. Li, J. Zhang, S.-T. Li, X.-M. Li, M.-Q. Dong, H. Mao, L. Tao, Segmentation and classification of two-channel *C. elegans* nucleus-labeled fluorescence images, *BMC Bioinformatics* 18 (1) (2017) 412, <https://doi.org/10.1186/s12859-017-1817-3>.