Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 141 (2018) 327–334

Procedia Computer Science

www.elsevier.com/locate/procedia

The 8th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2018)

Red blood cell segmentation by thresholding and Canny detector

Fatimah Al-Hafiz*, Shiroq Al-Megren and Heba Kurdi

Department of Computer Science, King Saud University, Riyadh, Saudi Arabia

Abstract

A computer visualization system that can detect, count and classify cells in bioimages is needed, and the segmentation of cells plays a vital role in this type of system. In the medical field, blood cell testing is among the most important types of clinical examinations. Manual blood cell examination methods using microscopic devices are more time consuming than automatic methods and require radiologists with more technical skills. However, the development of an effective and fully automatic segmentation process for RBCs (red blood cells) remains challenging. This paper proposes an automatic segmentation algorithm for RBCs that automatically computes the threshold image using boundary-based methods after enhancing the local and global details of the output using morphological operations to segment red blood cells in bioimages. The proposed segmentation method exhibited an average accuracy rate of 87.9% in a public human red blood cell dataset. Furthermore, a comparison of this method with gold segmentation and two other methods typically used for this purpose demonstrated that the proposed method was highly robust and outperformed the other methods.

Keywords: Red blood cells; Boundary-based segmentation; Gradient magnitude; Canny operator

1. Introduction

A blood sample image is a bioimage captured by an electron microscope that allows scientists to detect, classify and observe RBCs (red blood cells), WBCs (white blood cells), platelets and plasma. In 1970, the light microscope was the

* Corresponding author. Tel.: +966 548 287 341
E-mail address: fatimahsaid1989@gmail.com.

1877-0509
10.1016/j.procs.2018.10.193

first type of bioimaging device to use a computer as an analytical tool for cell imaging [1]. Currently, due to the increasing memory capacity and computational speed of digital image processing software, new methods are being developed to acquire and extract useful information from cell populations.

Recognizing, tracking, counting and classifying blood cells by image processing techniques allows experts to investigate diseases, observe cell interactions, and distinguish between normal and malignant cells. These functions depend on the ability of the system to segment the cells in bioimages [2]. RBC segmentation in bioimages usually requires a prefiltering process to prepare the image before applying the segmentation technique. However, accurate cell segmentation is difficult to achieve not only because of the large percentage of variability in cell shapes, densities, backgrounds and colors among different bioimages [1], but also because of three main challenges associated with this process. First, the contrast between the RBCs and the background is very low. Second, the grayscale distribution in the bioimages is not uniform. Third, overlapping sets of cells must be isolated. [3]

Many types of image segmentation exist, including histogram-based, region-based, boundary-based techniques, and region-based techniques [4]. Histograms partition the image using threshold measurements [5]. This is the simplest used to segments the image into N parts. Each part of the histogram has a value of λ_1, λ..., λ_N, where λ_k is the number of pixels in an image with a grayscale of k ($0 \leqslant k \leqslant 255$) [6]. In practice, histogram-based techniques used alone result in poor segmentation of red blood cells and are typically used in the preprocessing stage to enhance the contrast between the red blood cells and the background [1].

Boundary-based techniques rely on the edge detection process, which determines the cell edges based on changes in intensity. The Sobel operator, Prewitt operator, and Robert and Canny detection operators are examples of boundary-based segmentation techniques. Region-based techniques are started by one or more seed points and carried out the region based on similarities in the image. Feature extracted from seed point are matter factors could be segment full elements. [6]

This paper is subdivided as follows. The second section describes related work. The third section describes the system model and its implementation. The fourth section describes the methodology, including the materials, procedure, and evaluation methods. The results, discussion, conclusion and future work are included in the final section.

2. Literature review

Many researchers have proposed segmentation methods for different applications. However, RBC segmentation methods differ based on the aim of RBC segmentation. In a recent study, J. M. Sharif et al. [7] used these methods for counting purpose; these authors focused on RBCs after eliminating WBCs (white blood cells) by combining morphological operations and then applied a marker-controlled watershed algorithm to address the overlapping issues of RBC. Another study involving counting was performed by M. Maitra et al. [9] who extracted the features of RBCs based on their shape and size and used the circular Hough Transform method to isolate the cells from the rest of the image of the blood sample. Furthermore, R. Wang and B. Fang [11] proposed a method for recognizing whether red blood cells are normal by recovering the 3D shape surface feature and segmenting each cell using a multiscale surface fitting segmentation algorithm.

The threshold-based segmentation process is the most commonly used technique for segmentation and detection in bioimaging [12]. An automated segmentation technique was proposed by Sadeghian et al. [13] that applied a Fogbank histogram to minimize bioimage noise, and an improved watershed algorithm was used to detect the shapes of individual cells. In addition, Jiahua et al. [14] used Otsu's threshold iteratively after finding a circular histogram on the base of the least square method.

Boundary-based techniques have undergone many improvements to produce better segmentation. The method proposed by the math work community [8] applies the boundary-based technique after computing the threshold based on the Sobel operator. The algorithm gain segmentation results in the loss of some edges because it manages bioimages with a specified gray level.

Several studies [3], [8], [15], [16], and [17] describe methods that use Sobel edge detector, but Sobel edge detector uses two convolution kernels to identify the edge of an image, resulting in the detection of noise and an imprecise edge location [1]. Due to these limitations, we propose a new approach using a Canny operator after removing noise and preparing the image by enhancing the local details of red blood cell images.

3. System design

We designated a model for the cell segmentation algorithm using four basic steps.

3.1. Preprocessing

The preprocessing step is used to create a 2D filter that allows the system to compute the gradient magnitude of the image. First, computation of a 3-by-3 horizontal edge-finding mask and y-derivative approximation mask are performed by applying the following mask:

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (1)$$

Then, computation of the x-derivative by the transposed G_y is performed to identify the vertical edges. Using the result of G_y and its transposed value, the gradient magnitude is computed to locally enhance the appearance of RBC in the images because the internal gradient produces a mask of the internal boundaries each cell and blur the non-important details in the image.

3.2. Determination of the threshold

In general, an object can be easily segmented if it has a different contrast level from the background of each image. However, in most blood sample images, the contrast between red blood cells and their background is low. Use of a constant threshold value for all red blood cell images is not possible because the threshold depends on several factors, such as the microscope type and light conditions. Therefore, we generated an appropriate threshold value for each input image based on its gradient magnitude. First, we computed the gradient value of the image, and then we converted the image into grayscale by the adaptive thresholding method.

3.4. Edge detection process

The Canny operator is an optimal edge detector [4] that uses a grayscale image as the input and produces an output image showing the positions of edges tracked by the discontinuousness of the intensity. The Canny detector is applied here because it satisfies three criteria: a low error rate, edge localization, and displaying one response to a single image [13].

The Canny detector allows the extraction of useful structural information from different angles of RBCs from the image and reduces the amount of data to be processed [18].

3.4. Postprocessing of segmentation

After the Canny detector was applied, morphological operators were used to elimination WBC appearance and to smoothen the RBCs contours. First, the edges were dilated to show lines of high contrast in the image, and open operations were used to isolate the full shape of each segmented cell to ensure that there was no overlap between the regions. The open operation allows for overlapping objects in the image to be separated and can place a gap between connected objects using a line of pixels.

At this point, we improved the ROI by masking it with the original image to fill the gaps in the lines surrounding the red blood cells and to fill the hole inside the background with a random number of pixels. Finally, we enhanced the ROI by using median filter to make RBCs appear as natural as possible.

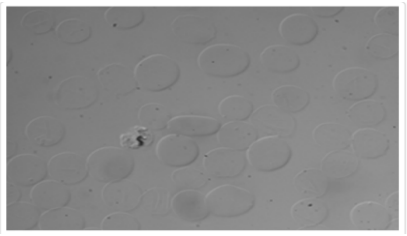

Fig. 1: Original human red blood cell image in DIC format.

4. Experiments

4.1. Configuration process

The experimental validation of this algorithm was performed by using all images in the human red blood dataset [19], which is a public dataset available in the Broad Bioimage Benchmark Collection (BBBC), which is an annotated biological image set used for testing and validation. The BBBC is a collection of freely downloadable microscopy image sets, and each set includes a description of the biological application and some type of ground truth (expected results). The dataset consists of five images in DIC format as shown in Fig. 1 with a size of 512 x 512 pixels, and we applied the method to all images in the dataset and computed the results of each image one-by-one. We converted all images to 8-bit TIFF format before coding. We wrote the code by using MATLAB R2010a, and all experiments were performed on a computer with an Intel® Core™ 2 Duo CPU with 2.10 GHz (2.10 GHz) and 2 GB of memory.

4.2. Procedures

The experimental procedure was performed to achieve two goals. First, we used the same algorithm under different settings. The second goal was to compare the results of the proposed algorithm with those of two other common methods. These two experimental procedures used by [20].

First, our goal was to compare the performance of this algorithm by using a Canny operator in an edge detection process to the performance of this algorithm by using a Sobel operator in an edge detection process. Here, the goal is to measure the precision segmentation of our algorithm by applying the operator along with that of another Canny detector. The second goal was to compare our algorithm with two typical methods. The first method was proposed by Feng-Ming et al. [10] to detect and count cells based on the segmentation process using the Laplacian of the Gaussian method as a first derivative of the edge detector to smooth the edges before the edge detection; these authors applied the watershed algorithm to obtain the final segmentation. The second method is published on the Math Work Community page and is used to detect a cell using image segmentation; They designated this method DCUIS [8] and used this method in the edge detection process by a Sobel operator with a tuned threshold value and improved the result by using erosion and open morphological operations without using a predefined 2D filter to enhance the local details of the images.

4.3. Experimental metrics

By using the following metrics, we evaluated the segmentation performance of this proposed algorithm by comparing the segmented result obtained by this method with the ground truth images available in the BBBC and outlined manually by experts [19].

4.3.1. Confusion matrix

The confusion factors are the true positive (TP), false positive (FP), false negative (FN), and true negative (TN) rates. In the evaluation process, we followed [21] by accepting the results of the image segmentation if TP>0.5; otherwise, the results of the remaining metrics are rejected.

- Sensitivity: the TP rate was determined using the following equation:

$$Sensitivity = \frac{TP}{TP+NF} \quad (2)$$

- Precision: the fallout segmentation was calculated using the FP rate or precision to measure the validity of the edge detection process [20] using the following equation:

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

4.3.2. F-score

This is a parametric curve that allows the trade-off between the error and accuracy [20] to be utilized and is used to evaluate the success of cell segmentation based on the recall and precision as follows:

$$F - Score = \frac{2*(Precision*Sensitivity)}{Precision+Sensitivity} \quad (4)$$

4.3.3. Dice coefficient similarity

This metric, used in [22] to measure the accuracy of segmentation, is based on the Dice coefficient. The Dice metric is a similarity index measure computed by the following equation:

$$D(GT.SI) = (2 * |D \cap G|)/|G| + |S| \quad (5)$$

where a segmentation (S) is matched with a GT cell (G) if and only if it contains more than half of the pixels of that GT cell for each GT cell and its matched segmentation.

4.3.4. Area under the curve (AUC)

To measure the accuracy of the segmentation algorithm, we computed the area under the ROC curve, which is given by:

$$AUC = \int_x^y f(s)ds \quad (6)$$

5. Results and discussion

The results are classified based on the goal of the experiment as previously described in the experimental section (to simplify the description, we designated our method TCD).

5.1. First goal: examine the TCD algorithm under different settings

- *Results based on the detection operator*

The goal was to measure the validity of the edge detection process by comparing this algorithm using two edge detector operators, i.e., the Sobel and Canny edge detectors. Fig. 2 shows how the different detection operators can produce differences in the precision of segmentation based on the measured accuracy rate of the detected edges. We found that the Sobel operator mainly exhibited incorrect edge position and loss of some cell edges. Moreover, the Canny edge detection produced greater segmentation accuracy than the method using a Sobel edge detector as shown in Table 1.

5.2. Second goal: compare the TCD with other methods

We evaluated the accuracy by comparing our algorithm with two other methods, i.e., the DCUIS algorithm [8] and the method proposed by Feng-Ming in [10] which is designated CellSeg.

As shown in Table 2, the accuracy of the TCD algorithm is greater than that of the other methods; the DCUIS [10] method produced the worst segmentation results according to the F-score. This result may be due to the type of boundary-based technique that was used in the DCUIS algorithm [8], and the CellSeg [10] method did not perform a preprocessing step to enhance the internal details of an image by creating a 2D filter.

Furthermore, Table 2 shows that the segmentation of the DCUIS method results in slightly lower values than those for the TCD algorithm, but the CellSeg method obtains a better segmentation result than the TCD method according to the Dice similarity metric. This finding may have been due to the amount of noise and degree of cell overlap in this dataset. With a low amount of noise and a large degree of cell overlap, our method exhibits poorer performance than the CellSeg algorithm because the watershed algorithm can efficiently manage cell overlapping using its mechanism, which operates in the top-to-bottom and left-to-right directions until it locates a closed object. Therefore, the segmentation performance of the TCD algorithm is limited in the case of overlapping cells, but this method can efficiently manage noise.

We found that the AUCs shown in Fig. 3 for the TCD, CellSeg, and DCUIS methods were 0.06355, 0.04092 and 0.00474, respectively. Thus, regarding image segmentation, the TCD algorithm produced a greater AUC than the other two methods. In contrast, the visual result is more important than the statistical result in some cases of image processing, and this has been a proven limitation of the TCD method as shown in Fig. 4. Regarding the visual results, we observed an unclosed cell shape, which will require the use of an active-shape technique to compensate for this limitation.

Table 1. Comparison of the TCD algorithm with the Sobel and Canny operators regarding the F-score and Dice similarity.

Edge detector	Measures	Img#1	Img#2	Img#3	Img#4	Img#5
Canny operator	F-score	**0.0051**	**0.0042**	0.0021	**0.0055**	**0.0047**
	Dice similarity	**0.9197**	**0.9045**	0.7888	**0.9042**	**0.888**
Sobel operator	F-score	0.0041	0.0035	**0.0029**	0.0034	0.0033
	Dice similarity	0.8477	0.8493	**0.8819**	0.8711	0.8388

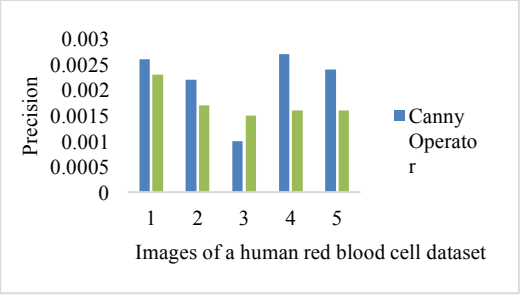

Fig. 2. Comparison of the precision of edge detection by the Canny and Sobel operators.

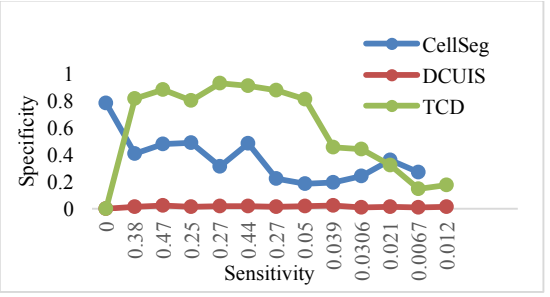

Fig. 3. Comparison of the area under the ROC curve among three methods: TCD, CellSeg [10], and DCUIS [8]

Table 2. Comparison of the TCD method with the other two methods according to the F-score and Dice similarity.

Algorithm	Metric	Img#1	Img#2	Img#3	Img#4	Img#5
DCUIS [8]	F-score	NaN	NaN	NaN	NaN	NaN
	Dice similarity	0.0045	0.0022	0.0028	0.0047	0.0014
CellSeg [10]	F-score	0.0769	0.0606	NaN	NaN	NaN
	Dice similarity	**0.9397**	0.5873	**0.7898**	**0.9142**	0.4834
TCD	F-score	**0.0051**	**0.0042**	**0.0021**	**0.0055**	**0.0047**
	Dice similarity	0.9197	**0.9045**	0.7888	0.9042	**0.8880**

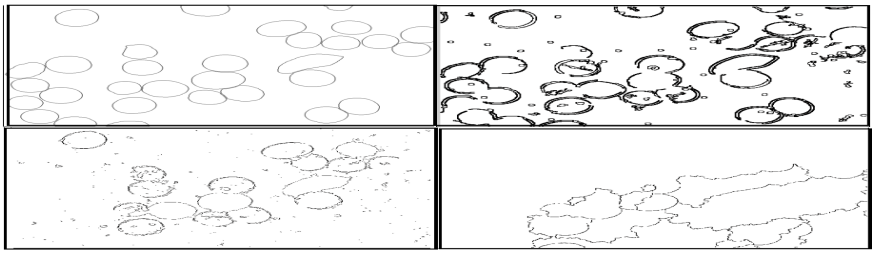

Fig. 4. Comparison of the visual segmentation results: first row: GT image (left) and TCD method (right); second row: DCUIS [8] method (left) and CellSeg [10] method (right).

6. Conclusion and future work

Red blood cell segmentation is a vital step that allows doctors to recognize, track and count cells. The threshold value is an important factor in a successful cell segmentation process. In this paper, we have proposed a new method and described two contributions of this process: automatic tuning of the threshold value and proof of the capability of the Canny edge detection method to perform cell segmentation with an appropriate prefiltering process. All experiments with the TCD algorithm were performed using a public human red blood cell dataset with different settings, and the results of this method were compared with those of two other methods. We demonstrated a higher average accuracy, based on the AUC, for our method than for the other methods, and the average similarity of segmentation compared to manual segmentation was 87.928%.

Obtaining an unclosed cell shape in the segmentation results was a drawback of this method. This result was obtained because the method missed some edges in cases involving increased overlapping among cells. Integration of an active-shape technique will be performed to improve the segmentation results for red blood cells.

References

[1] Meijering, Erik. (2012) “Cell segmentation: 50 years down the road [life sciences].” *EEE Signal Processing Magazine* **29 (5)**: 140-145.
[2] Bajcsy, P., Cardone, A., Chalfoun, J., Halter, M., Juba, D., Kociolek, M., Majurski, M., Peskin, A., Simon, C., Simon, M. and Vandecreme, A. (2015) “Survey statistics of automated segmentations applied to optical imaging of mammalian cells.” *BMC bioinformatics* **16 (1)**: 330.
[3] Kale, Asli, and Selim Aksoy (2010) " Segmentation of cervical cell images. " *Pattern Recognition (ICPR)*. 20th International Conference on IEEE.
[4] Canny, J. (1986) “A Computational Approach to Edge Detection.” *Pattern Analysis and Machine Intelligence IEEE Trans.* **8**: 679-714.
[5] Sankur, M. Sezgin and B. Jan. (2004) "Survey over image thresholding technique and quantitative performance evaluation." *Journal of Electronic Imaging,* **13**: 146-168.
[6] HORGAN, C A GLASBEY and G W. (1994) “CH4, SEGMENTATION.” *IMAGE ANALYSIS FOR THE BIOLOGICAL SCIENCES*.
[7] Sharif JM, Miswan MF, Ngadi MA, Salam MS, bin Abdul Jamil MM. (2012) “Red blood cell segmentation using masking and watershed algorithm: A preliminary study” *In Biomedical Engineering (ICoBE),* International Conference IEEE:258-262.
[8] Community of Math Work. (2009) “Detecting a cell using image segmentation.” *Image Processing Tools Examples.*
[9] Maitra, Mausumi, Rahul Kumar Gupta, and Manali Mukherjee. (2012) "Detection and counting of red blood cells in blood cell images using hough transform." *International journal of computer applications* **53(16)**.
[10] Li, Feng-Ming. (2016) “Cells Segmentation Using the Hybrid of Image Morphology and Edge Detector Algorithm and Cell Counting.” *irjet*
[11] Wang R, Fang B. (2012) “A combined approach on RBC image segmentation through shape feature extraction.” *Mathematical Problems in Engineering*.
[12] Chaoxin Zheng, and Khurshid Ahmad. (2007) “The segmentation of images of biological cells -A survey of methods and systems.” *School of Computer Science & Statistics (SCSS), Faculty of Engineering, Mathematics and Science.* Trinity College, Dublin, Ireland.
[13] Sadeghian, F., Seman, Z., Ramli, A.R., Kahar, B.H.A. and Saripan, M.I. (2009) “A framework for white blood cell segmentation in microscopic blood images using digital image processing.” *Biological procedures online* **11 (1)**:196.
[14] Sadeghian, F., Seman, Z., Ramli, A.R., Kahar, B.H.A. and Saripan, M.I. (2009) “A framework for white blood cell segmentation in microscopic blood images using digital image processing.” *Biological procedures online* **11 (1)**:196.
[15] Aly, Ashraf A., Safaai Bin Deris, and Nazar Zaki. (2014) “Intelligent algorithms for cell tracking and image segmentation.” *International Journal of Computer Science & Information Technology* **6 (5)**.

[16] Eddins, Steve. (2010) “Cell Segmentation method.” *http://blogs.mathworks.com/steve/2006/06/02/cell-segmentation.*
[17] Sahoo, Tamanna, and Sandipan Pine. (2016) " Design and Simulation of SOBEL Edge Detection using MATLAB Simulink." *Irjet* **3(5).**
[18] Raju, CH Nooka, G. S. N. Raju, and VK Varma Gottumukkala. (2016) “Comparative Studies on Cell Segmentation by Fuzzy Logic and Canny Edge”. *ijarece* **5 (5)**:1555-1559.
[19] Ljosa, Vebjorn, Katherine L. Sokolnicki, and Anne E. Carpenter. (2012) "Annotated high-throughput microscopy image sets for validation." *Nat Methods* **9(7)**: 637.
[20] T. Liu, M. Seyedhosseini and T. Tasdizen. (2016) “Image Segmentation Using Hierarchical Merge Tree.” *IEEE Transactions on Image Processing* **25 (10)**: 4596-4607.
[21] Peskin, Adele P., Alden A. Dima, Joe Chalfoun, and John T. (2010) “Predicting segmentation accuracy for biological cell images.” *International Symposium on Visual Computing.* Springer Berlin Heidelberg.
[22] Akram, Saad Ullah, Juho Kannala, Lauri Eklund, and Janne Heikkilä. (2016) “Cell segmentation proposal network for microscopy image analysis.” *Springer.* Cham. **21(29)**.