International Journal of Security and Its Applications
Vol.8, No.5 (2014), pp.41-50
http://dx.doi.org/10.14257/ijsia.2014.8.5.05

Canny Edge Detection Based On Iterative Algorithm

Xumin Liu, Xiaojun Wang and Zilong Duan

College of Information Engineering, Capital Normal University, Beijing 100048, P. R. China
liuxumin@126.com

Abstract

The images of different areas require different test results on the edge. In this paper, the images of red blood cells in medical field need to be detected theirs cell size, roundness, number and other features. For the test requirements, an image edge detection algorithm is proposed based on improved Canny operator. This algorithm calculates the optimal high and low dual-threshold by using iteration arithmetic, and uses mathematical morphology to thin the detected images. The results proved that this algorithm can effectively reduce interference and noise edge and make more prominent detection characteristics, which is good ready for subsequent processing of the image.

Keywords: *Edge Detection, Red Blood Cells, Canny Operator, Mathematical Morphological Method*

1. Introduction

Medical image processing technology applies image processing technique to medical research. It is mainly used in medical image archiving and auxiliary analysis. By the computer processing, we can do all kinds of disposal on these medical images, such as enhancement, edge segmentation, conversion of image format and so on. These medical images can obtain graphical effects after these operations. With the development of computer vision and digital image processing techniques, edge detection technology has been developing rapidly, which has been used in various fields and has played an increasingly important role [1, 2]. The cell is the basic unit of life activities and all living things are composed of cells except the virus. The edge of the cell image has great value to the target boundary information, especially in the features of the cell area, roundness, the number and so on. The detection results provide an important basis for the analysis of cell morphology and disease diagnosis. The traditional edge detection operators such as Sobel, Laplace operator, accorded to the gray value changes of each pixel neighborhood and used the changes of the first order or the second order directional derivative in mathematical methods to detect the edge [3-5]. Because of their simple structures, these operators are achieved faster. However, they have much greater impact on noise. If you apply them to the edge detection of the cell image, there will be some disadvantages such as discontinuity on the edge of cell images, interference edge and losing of cell image details.

In 1986, Canny [6] proposed an edge detection operator based on the optimization algorithm. The experiments showed that the operator was superior to other traditional edge detection operators in dealing with the white Gaussian noise pollution. So it was widely used and became the standard for comparison with other experimental results. However, the traditional Canny operator [7-9] sets high and low thresholds manually with experience, which will produce false edges and along with more noise in the cell edge detection. In order

ISSN: 1738-9976 IJSIA

to making better in noise suppression, according to the detection characteristics of the cell image, this paper proposes iterative algorithm to seek the best segmentation threshold and uses mathematical morphology to refine the post-test images. This makes an effective suppression of noise interference, and gets good results in the integrity and continuity of cell edge orientation.

2. Canny Edge Detection Operator

The edge of the object is formed by the discontinuity of gray. Classical edge detection method uses differential algorithm to inspect the gray changes of each pixel value of the image within a domain [10]. It is mainly based on the principle that the first derivative on the gray level edge has the extreme value and the second derivative crosses zero. When we solve the edge of the derivative, we need to calculate the location of each pixel. In the actual, template convolution is commonly used to calculate appreciatively. When the first derivative of evaluating is higher than a certain threshold, we confirm that this point is the edge point of image. Thus, this leads to detect the redundant edge points, get the edge of the coarser. And the positioning accuracy will be not high. In order to get more detailed and more accurate edge image, we can evaluate the gradient local maximum. And then, we affirm the gradient local maximum points as the edge points. The first derivative of the local maximum points corresponds to the zero crossing of the second derivative. By looking for image grey value of zero crossing point of the second derivative, we can detect the image edge points.

2.1. The Three Criterion

Canny operator is a method based on optimization thinking, which puts forward three criteria for edge detection performance evaluation [11].

(1) High SNR. Focus on good detection results. Thus, the probability of non-edge points ruled edge points or edge points ruled non-edge points is sentenced to minimum.

(2) High positioning accuracy. It emphasizes the detected edge points and the distance to the actual edge point minimum. So that the edge achieves the highest positioning accuracy.

(3) Single-edge response. That is to single edge points, there is only a response. Most suppress the emergence of false edges.

2.2. The Process of Canny Operator Realization

The implementation of Canny operator mainly includes four parts: smooth image, calculating the magnitude and direction of the gradient, the non-maxima suppression of gradient amplitude, dual-threshold detection and the connection edge.

(1) Smooth image. Select a one-dimensional Gaussian function *G(x)* to construct filters. Make the convolution operations of the original image *f(x,y)* by row and column to obtain a smoothed image *I(x,y)*.

$$G(x) = \frac{\exp(-x^2/2\sigma^2)}{2\pi\sigma^2} \tag{1}$$

$$I(x,y) = [G(x)G(y)] * f(x,y) \tag{2}$$

σ Is the standard deviation of the Gaussian function and used to control the degree of smoothing. The one-dimensional Gaussian function is selected as smooth function. On the basis of the above three criteria, we use functional derivative method to export the expression of a product by the edge location accuracy and signal-to-noise ratio. This expression approximates a first derivative of the Gaussian function.

(2) Calculate the gradient magnitude and direction. The calculation of the gradient uses commonly the method of the partial derivative. Canny operator uses the finite difference of the first order partial derivatives of the 2×2 neighborhood to calculate the gradient magnitude *M(x,y)* of the image *I(x,y)* after smoothing and the gradient direction *H(x,y)*:

$$M(x,y) = \sqrt{k_x^2(x,y) + k_y^2(x,y)} \tag{3}$$

$$H(x,y) = \arctan[\ k_x(x,y), k_y(x,y)] \tag{4}$$

$$f_x = \begin{bmatrix} -0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix}, \quad f_y = \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & -0.5 \end{bmatrix} \tag{5}$$

k_x and k_y are the role of the results of the original image *I(x,y)* by the filter f_x and f_y along the row and column.

(3) The non-maxima suppression of gradient amplitude. Make the interpolation of the gradient magnitude along the gradient direction of the *M(i,j)* of all the elements with the the window of 3×3 in the field of eight directions. For each point, we use the neighborhood center element *M(i,j)* to compare with the two gradient magnitude along the gradient direction interpolation result. If the value of *M(i,j)* is less than two interpolation results in the gradient direction, it will make *M(i,j)* corresponding to the edge flag bit 0 value.

(4) Dual-threshold detection and connection edge. Use the high threshold value Hth and the low threshold value Lth to process the gradient magnitude of non-maxima suppression. We assign the value of pixel gray scale of the gradient less than the threshold as 0. They are split to obtain two threshold edge images *H(i,j)* and *L(i,j)*.Connect the edge contour in image *H(i,j)*. When the endpoint is connected, find the weak edges point in *L(i,j)* to fill image *H(i,j)* of the edge voids. Thus, the confirmation of high and low threshold concern to the substantive issues of the edge detection. So the key of Canny operator is selecting the appropriate level of threshold.

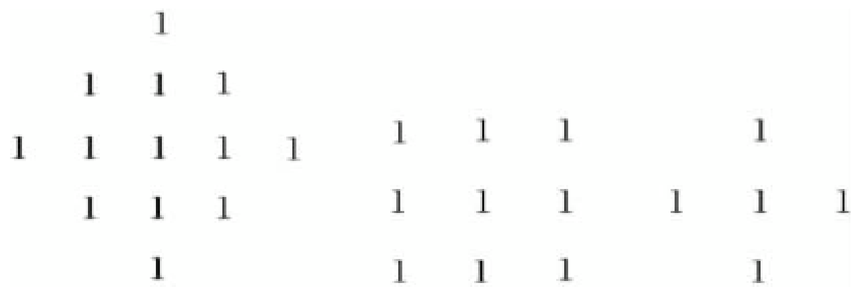
3. Mathematical Morphology

The basic idea of mathematical morphology [12, 13] is that we use the structural elements with a certain shape to measure and the extract the corresponding shape of the image in order to achieve the purpose of image analysis and recognition. Mathematical morphology is a mathematical method, which is composed by a group of morphological algebraic operations to analyze the geometrical shape and structure. Mathematical morphology is used to process the binary image firstly. The basic theory is the binary morphology. Later, the gray scale morphological operators of dealing with gray-scale images is promoted by the theory of binary morphology.

Dilation and erosion is the basic operations of mathematical morphology. Opening operation and closing operation are combination composite arithmetic of dilation and erosion. Based on these basic operations, we can assemble and obtain a variety of mathematical morphology practical algorithm, such as the hit / miss in transformation. These operations are image elements and the interaction between structural elements. They are used to study the nonlinear theory of overall shape features. They use collection view to describe and analyze the geometrical structure of the image. Thus, we can obtain the size, shape, connectivity, concave and convex, smoothness, directional and so on.

3.1. The Basic Operations of Mathematical Morphology

The structural element is the basic part of the mathematical morphology operation, which is used to detect the input image. It is usually much smaller than the image to be processed. The structural element is a probe with the design of collection of image information in the study of image analyzes. The observer moves the probe constantly to inspect relationship between the various parts of the images and to extract useful features. The selection of the structural elements affects the effect of morphological operations directly. So which should be selected depends on the image feature. Selecting structural elements needs to consider two principles. Firstly, structural elements must be simpler than the original image in geometry. Secondly, the morphology of the structural elements must be preferably convex, such as round, square and cross-shaped. They are shown in Figure1.



(a) Round (b) Square (c) Cross-shaped

Figure 1. Structure Elements

(1) Dilation and erosion operations

Set *S* to represent an image. *B* is a structural element. The operation of *S* dilated by *B* is defined as follows:

$$S \oplus B = \left\{ x \mid \left[(\overline{B}) x \cap X \right] \subseteq X \right\} \tag{6}$$

In the formula, $\overline{B}$ represents the image *B*.

Erosion and dilation operations follow duality principle. The operation of *S* corroded by *B* is defined as follows

$$S \Theta B = \left\{ x \mid (B) x \subseteq S \right\} \tag{7}$$

The geometric meaning of the definition of dilation operation is described below. Firstly, set *B* as the mapping on the origin of $\overline{B}$. Secondly, do transformation of *x* to form a set of $(\overline{B})_x$. Finally, calculate the collection of structural elements reference of set $(\overline{B})_x$ and set *S* as nonempty set. The geometric meaning of the definition of erosion operation is described below. The structure elements reference a series of operations that set *B* to do transformation of *x* which is still in the set *S*.

(2) Opening operation and closing operation

Set *S* represents an image and *B* is a structural element. The opening operation of *S* with respect to *B* is defined as follow:

$$S \circ B = (S \Theta B) \oplus B \tag{8}$$

The closing operation of *S* with respect to *B* is defined as follow:

$$S \bullet B = (S \oplus B) \Theta B \tag{9}$$

(3) Hit / miss in transformation

Set *S* represents an image and *B* is a structural element. *B* consists of two disjoint portions B_1 and B_2 . *S* hit by *B* is defined as follow:

$$S \otimes B = (S \Theta B_1) \cap (S^c \Theta B_2) \tag{10}$$

In the formula, S^c is the complementary set of *S*.

3.2. The Thinning Algorithm of Mathematical Morphology

Generally, Refining refers to a region having a certain area represented with a curve. The refining process of mathematical morphology from the hit / miss in transformation is given a series of structural elements with a certain shape.

(1) The set S refined by the structural element B is defined as follow:

$$S \leftarrow B = S - (S \otimes B) = S \cap (S \otimes B)^c \tag{11}$$

(2) Let $\{B\} = \{B_1, B_2, \ldots, B_n\}$ be a structural element sequence, then S refined by a sequence of structure elements is defined as follow:

$$S \leftarrow \{B\} = (\ldots((S \leftarrow B_1)B_2)\ldots)B_n \tag{12}$$

From the formula 7, we can see as follows. Firstly, refine the image with the structure element B_1 once. Then make iterative refinement of the results. Do the process back and forth until it refined with B_n once. And the whole process is repeated until no change update. Refining includes direction. In order to refine more symmetrically, this article selects the structural element sequence with eight directions (as shown in Figure 2).

1	*	0
1	o	0
1	*	0

(a) B_1

0	*	1
0	o	1
0	*	1

(b) B_2

1	1	1
*	o	*
0	0	0

(c) B_3

0	0	0
*	o	*
1	1	1

(d) B_4

1	1	*
1	o	0
*	0	0

(e) B_5

0	0	*
0	o	1
*	1	1

(f) B_6

*	0	0
1	o	0
1	1	*

(g) B_7

*	1	1
0	o	1
0	0	*

(h) B_8

Figure 2. The Structure Element Sequence

In Figure 2, the structural elements $B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8$ are used to remove the eight direction points on the east, west, south, north, southeast, northwest, northeast and southwest symmetrically. "O" indicates the reference center point; "1" indicates a point on the target image; "0" indicates a point on the background image; "*" represents either a point on the target image or a point on the background image.

In this case, the refinement iteration factor of the image can be defined as follow:

$$S \leftarrow \{B\}_m = \{(\ldots((S \leftarrow B_1)B_2)\ldots)B_8\}_m \tag{13}$$

In this formula, *m* represents the total number of cycles when thinning convergence. Its size depends on the size of the target image and the edge thickness.

4. Basic Principles of the Improved Canny Edge Detection Operator

There are two key points [14] in the image edge detection. One point is reducing the impact of noise effectively. The other point is selecting the threshold objectively and correctly. If the threshold is selected properly, it can not only reduce the noise, but also have a good effect on the suppression of the pseudo-edge. If the threshold is set too high, it may lead to the rupture of the edge. If the threshold is set too low, it may result in excessive false edges and even make noise for detection. The traditional Canny operator sets the high and low thresholds artificially with prior experience, which may need to repeat several experiments in order to finding a suitable threshold. The ratio between the high and low thresholds is a fixed value. However, the actual image is vulnerable to the uncertainties of the light, scene and so on. The ratio of the best high and low thresholds is not a fixed value necessarily in different images. So the traditional Canny operator is non-adaptive on the determination of the threshold. Based on the above mentioned shortcomings of the Canny operator, this paper presents iterative algorithm to calculate the optimal dual-threshold and refine the detected edge image with mathematical morphology. Use analysis and calculation of each pixel's gray value of the image itself to select the high and low thresholds. We also use them to determine the edge point, which can reduce the noise interference effectively.

5. Iterative Algorithm for Segmentation Threshold

Threshold segmentation divides the image into target and background with threshold. The threshold satisfies the number of the error pixels of image segmentation to the minimum. In fact, it's difficult to obtain suitable threshold because of the influence of noise. In this paper, we can obtain the optimal threshold by using iterative algorithm to reduce the impact of noise effectively.

Iterative algorithm [15] can be described as follows. Firstly, make the average of the minimum and maximum gray values through the histogram as the initial threshold. Secondly, use the initial threshold to divide all the gray values into two parts. One part is higher than the initial threshold value, while the other part is lower. Thirdly, average the two parts separately and then calculate the two means, which obtains the threshold after the first iteration. Fourthly, compare this threshold with the initial threshold value. If the two thresholds are equal or the difference meets a certain relation, end the iterative, and this threshold is the optimal one. Otherwise, use this threshold to split all the gray values and repeat the above steps. After several iterations, the final threshold is much better than the initial threshold, which makes the error image pixels to the least. At the same time, it's more suitable for image segmentation than the traditional Canny operator that determined artificially and has a certain percentage of the high and low thresholds.

The specific steps of the algorithm are as follows.

Step 1: Get the initial threshold T0 by making a statistics of the gray histogram.

$$T_0 = \{T_K \mid K = 0\} \tag{14}$$

$$T_0 = \frac{Z_{\max} + Z_{\min}}{2} \tag{15}$$

K is the number of iterations. $Z_{\min}$ and $Z_{\max}$ represent the minimum and maximum gray values.

Step 2: Divide the image into H_1 and H_2 two parts by the threshold.

$$H_1 = \left\{ f(x, y) \middle| f(x, y) \geq T_K \right\} \tag{16}$$

$$H_2 = \left\{ f(x, y) \middle| f(x, y) < T_K \right\} \tag{17}$$

Step 3: Calculate M_1 and M_2 of the average gray values of H_1 and H_2 separately.

$$M_1 = \frac{\sum_{f(i,j)\geq T_K} f(i, j)}{\sum_{f(i,j)\geq T_K} N_H(i, j)} \tag{18}$$

$$M_2 = \frac{\sum_{f(i,j)< T_K} f(i, j)}{\sum_{f(i,j)< T_K} N_L(i, j)} \tag{19}$$

f (i, j) is the gray value of point *(i, j)*, and $N_H(i,j)$, $N_L(i,j)$ separately satisfy the following conditions.

$$N_H(i, j) = \begin{cases} 1, f(i, j) \geq T_K; \\ 0, otherwise \quad . \end{cases} \tag{20}$$

$$N_L(i, j) = \begin{cases} 1, f(i, j) < T_K; \\ 0, otherwise \quad . \end{cases} \tag{21}$$

Step 4: Calculate the new threshold.

$$T_{k+1} = \frac{M_1 + M_2}{2} \tag{22}$$

Step 5: If T_K=T_{K+1} meets the specified requirements in the end. Otherwise $K = K+1$and turn to step 2.

Step 6: Finish the iteration and take the final values M_1 and M_2 as the best high and low thresholds for image segmentation.

6. Specific Steps for the Improved Algorithm

The specific steps for the improved algorithm are as follows:

Step 1: Smooth the images and use the Gaussian filter to suppress the noise;
Step 2: Calculate gradient magnitudes and directions of the smoothed image;
Step 3: Make non-maxima suppression for the gradient;
Step 4: Get the best high and low thresholds by iterative algorithm;
Step 5: Detect and connect the edge with dual-threshold algorithm;
Step 6: Refine the edge by using mathematical morphology method.

7. Experiments and Analysis

We used the red blood cells of medical image (256 * 256) and E. coli image (256 * 256) as the image to be detected. The three images are simulating in *MATLAB2009b (7.9)*. The simulation results are as follows:

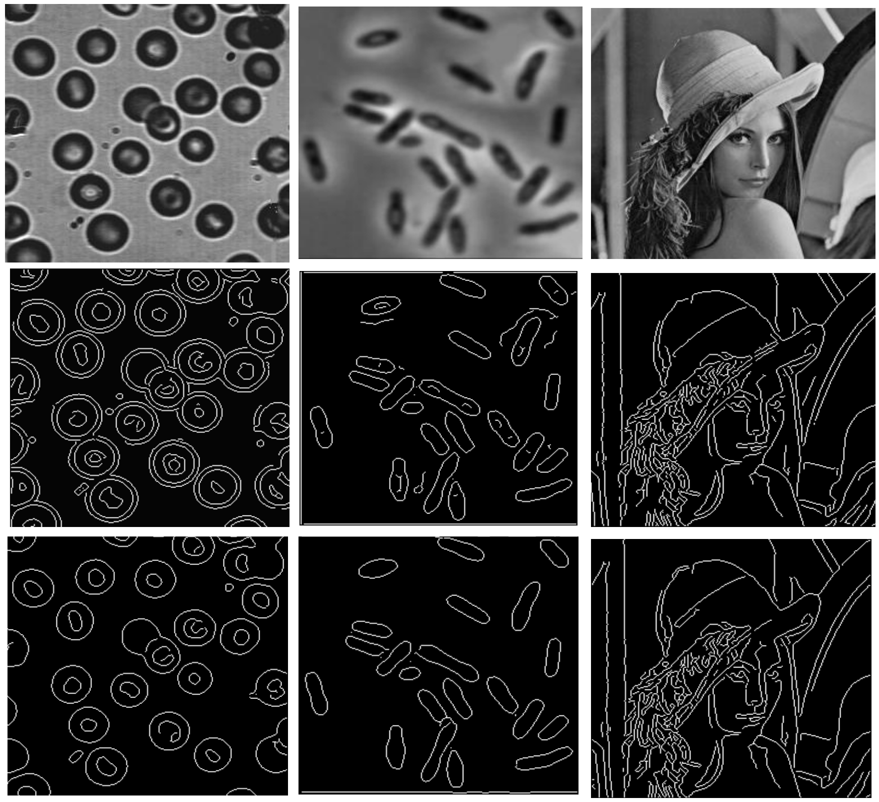


Figure 3. Different Edge Images Obtained by Using Different Operators

Row 1 is original images of red blood cells. E. Row 2 is obtained by using the Canny detector and Row 3 is improved by Canny detector.

Two methods are used in each experiment. The threshold values are in the below Table.

Table 1. Different Detection Thresholds

Original image	Red blood cells		E.coli		Lena	
Threshold value	low	high	low	high	low	high
Canny	0.0750	0.1875	0.0563	0.1406	0.0688	0.1719
Improved Canny	0.0672	0.3468	0.0454	0.1609	0.0655	0.2671

As shown in the above Table, different images use different algorithms with different threshold values. As seen from Figure 1, the improved Canny algorithm is better than the traditional Canny detector obviously. The specific analysis results are as follows.

The three edge images in Row 2 are obtained by using the Canny detector. Its high and low thresholds are calculated by the formula according to the gray values of the image. The ratio between them is 0.4. If you use this algorithm to detect image, you will get clear edge detection. But affected by external conditions such as light, after the detection the image will appear double-edge phenomenon, which can be seen from the first two images in the second line. In addition, the edge detection image contains more noise by the Canny operator. The images in the third line are detected by the new algorithm. It is based on the Canny operator and uses iterative algorithm to calculate the optimal threshold. As is shown in Figure 1, the edge in Row 3 is significantly better than that in the second line. This algorithm can not only be able to extract the edge accurately, but also remove the interference and noise points. But if you use this algorithm to detect the image such as Lena, the detection effect won't be much better than the Canny operator and the amount of computation is larger than it. So in contrast, this algorithm is more suitable for the detection of the medical cell image that detects features as their roundness, area and number.

8. Conclusion

This paper introduces an improved Canny operator for edge detection, which uses iterative algorithm to calculate the best segmentation threshold, and then refines the detected image with mathematical morphology. This method can suppress noise effectively, and be able to obtain the best segmentation threshold. It is applicable to the edge detection for the medical cell image, such as the detection of the characteristics of roundness, size and number of cells. But, for the image edge detection like Lena which needs to extract the image details, this algorithm has the shortcomings of large calculation and we will improve it next step.

Acknowledgments

This research was supported by the National Natural Science Foundation of China General Projects Grant No. 61272029.

References

[1] R. Biswas and J. Sil, "An Improved Canny Edge Detection Algorithm based on Type-2 Fuzzy Sets", Procedia Technology, vol. 4, **(2012),** pp. 820-824.
[2] H. Yu, D. Qi and S. Guan, "Edge Detection of Frothy Wood-Polymer Composites CT Image based on Log Operator", Journal of Convergence Information Technology, vol. 8, no. 2, **(2013),** pp. 469-478.
[3] S. Gao, R. Kang and Z. Dong, "Edge Chipping of Silicon Wafers in Diamond Grinding", International Journal of Machine Tools and Manufacture, vol. 64, **(2013),** pp. 31-37.
[4] H. Mu, D. Qi and M. Zhang, "Edge Detection of Wood Image with Rot Based on BP Neural Network", Journal of Convergence Information Technology, vol. 8, no. 2, **(2013),** pp. 506-513.
[5] H. Xia, "Study on the Video Image Tracking Technology based on the Particle Filter Algorithm", Journal of Convergence Information Technology, vol. 7, no. 11, **(2012),** pp. 384-391.
[6] J. Canny, "A Computational Approach to Edge Detection", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 6, no. 8, **(1986),** pp. 679-698.
[7] A. Orzan, A. Bousseau and P. Barla, "Diffusion Curves: a Vector Representation for Smooth-Shaded Images", Communications of the ACM, vol. 56, no. 7, **(2013),** pp. 101-108.
[8] D. Zhang and S. Zhao, "An Improved Edge Detection Algorithm Based on Canny Operator", Applied Mechanics and Materials, vol. 347, **(2013),** pp. 3541-3545.

[9] M. Li, J. Yan and G. Li, "Self-adaptive Canny Operator Edge Detection Technique", Journal of Harbin Engineering University, vol. 28, no. 9, **(2007),** pp. 1002-1007.
[10] X. Ma, B. Li and Y. Zhang, "The Canny Edge Detection and Its Improvement, Artificial Intelligence and Computational Intelligence", Springer Berlin Heidelberg, **(2012),** pp. 50-58.
[11] B. Ramamurthy and K. R. Chandran, "Content based Image Retrieval for Medical Images using Canny Edge Detection Algorithm", International Journal of Computer Applications, vol. 17, no. 6, **(2011),** pp. 32-37.
[12] J. Serra, "Introduction to Mathematical Morphology", Computer Vision, Graphics, and Image Processing, vol. 35, no. 3, **(1986),** pp. 283-305.
[13] B. A. Suleimanov and O. A. Dyshin, "Application of Discrete Wavelet Transform to the Solution of Boundary Value Problems for Quasi-linear Parabolic Equations", Applied Mathematics and Computation, vol. 219, no. 12, **(2013),** pp. 7036 -7047.
[14] M. E. Ambühl, C. Brepsant and J. J. Meister, "High - resolution Cell Outline Segmentation and Tracking from Phase - contrast Microscopy Images", Journal of Microscopy, vol. 245, no. 2, **(2012),** pp. 161-170.
[15] D. Li, J. Qian and N. Lu, "Improving of Image Edge Detection Technique", Computer Engineering and Applications, vol. 46, no. 18, **(2010),** pp. 164-166.

