Glomerulus Extraction by Using Genetic Algorithm for Edge Patching

Jiaxin Ma, Jun Zhang, and Jinglu Hu

Abstract—**Glomerulus is the filtering unit of the kidney. In the computer aided diagnosis system designed for kidney disease, glomerulus extraction is an important step for analyzing kidney-tissue image. Against the disadvantages of traditional methods, this paper proposes a glomerulus extraction method using genetic algorithm for edge patching. Firstly, Canny edge detector is applied to get discontinuous edges of glomerulus. After labeling to remove the noises, genetic algorithm is used to search for optimal patching segments to join those edges together. Lastly, the edges and the patching segments with high fitness would be able to form the whole edge of the glomerulus. Experiments and comparisons indicate the proposed method can extract the glomerulus from kidney-tissue image both fast and accurately.**

I. Introduction

As the computer technology develops, it plays a more and more vital role in medical diagnostics. Many traditional diagnostic methods could benefit from the automation brought by computer technology. One example is Nephric biopsy [1], which is the primary method in clinical diagnosis for kidney disease. In this method diagnosis is made by observing the changes of glomerulus' size and shape in the kidney-tissue image. Aided by computer image processing, it is possible to automatically extract the glomerulus, which means great help for kidney-tissue analyzing.

Fig. 1 shows a standard microscopic slice image of kidney-tissue, which contains both the glomerulus (the triangular tissue in the center of the figure) and other irrelevant tissues considered as noises. To successfully extract the glomerulus from the image, there are three major obstacles for traditional methods. First, unlike the round or elliptic cells, the glomerulus has irregular shapes. Ellipse fitting [2] which is commonly used in medical image processing cannot deal with all the situations. Second, there is not only glomerulus but also numerous noises in the image. And edge detection method [3] cannot exclude the effect of noises, as well as active contour method [4]. Third, because of the coloring uniformity of slice images, extraction by color feature [5] is not possible. One available method to solve this problem is using genetic algorithm (GA) to do spline function fitting [6], [7]. The heuristic nature of GA makes it capable to search for an optimal B-spline curve [8] to represent the shape of the glomerulus in high accuracy. However, this method mainly flaws on that the searching space of GA is too large, so its time efficiency needs to be improved.

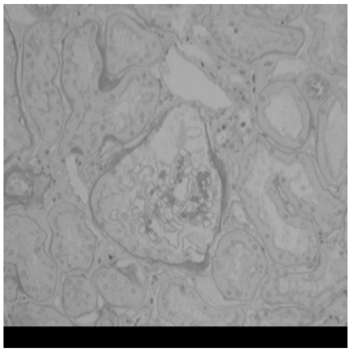


Fig. 1. Kidney-tissue image

Taking account the disadvantages stated above, this paper proposes an edge patching method which combines edge detection and genetic algorithm to solve this problem. First, edge detection will extract the edges of all the tissues in the image. Then the labeling method is used to denoise and it leaves mainly the edges of glomerulus. According to the extracted edge information, various connections between endpoints are established to try to patch discrete edges into a whole, in which GA runs to search for the optimal way of patching.

II. Edge Detection

Although an ordinary edge detection method cannot extract the whole edge of glomerulus, as stated above, it could still retrieve useful information of edges from image. If we could take full advantage of this information, the searching efficiency of GA can be greatly improved. Thus, the first important step is to detect the edges of the glomerulus. At present, Canny edge detector [9] is one of the best-performing edge detection method which could mark as many real edges in the image as possible, so we adopt the Canny to extract enough edge information for the following step of GA.

Canny edge detector has two kinds of parameters. First is σ, the size of the Gaussian filter which directly affects the results of the Canny algorithm. Smaller σ causes less blurring, and allow detection of small, sharp lines. On the contrary, a larger

Jiaxin Ma is with the Graduate School of Information, Production and Systems, WASEDA University, Kitakyshu, Fukuoka, Japan (090-4352-1894; e-mail: sirwh@ fuji.waseda.jp).

Jun Zhang is with the Graduate School of Information, Production and Systems, WASEDA University, Kitakyshu, Fukuoka, Japan (e-mail: zhangjun_lucky@163.com).

Jinglu Hu is with the Graduate School of Information, Production and Systems, WASEDA University, Kitakyshu, Fukuoka, Japan (e-mail: jinglu@waseda.jp).

978-1-4244-2959-2/09/$25.00 © 2009 IEEE

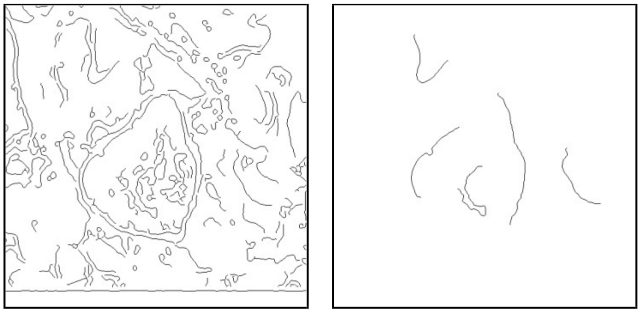


Fig. 2. Results of Canny edge detection and denoise

σ causes more blurring, smearing out the value of a given pixel over a larger area of the image. The second kind of parameters is two thresholds: the use of two thresholds with hysteresis allows more flexibility than in a single-threshold approach, but general problems of thresholding approaches still apply. A threshold set too high can miss important information. On the other hand, a threshold set too low will falsely identify irrelevant information (such as noise) as important.

In this paper, we set the size of the Gaussian filter relatively large, let $\sigma = 2$, in order to offer better continuity of edges. And other two thresholds can be automatically selected by the Canny method in MATLAB. Fig. 2 (a) shows the processing result of Canny edge detector for Fig. 1. And then, labeling method is used to preserve some longest edges in the image, and to delete those shorter ones. Thus most of the noises will be erased, like in Fig. 2 (b).

Afterwards, GA will search for connections to patch the discontinuous edges in Fig. 2 (b), and those connections are evaluated by the overlap ratio with the edge information in Fig. 2 (a).

III. Edge Patching Using Genetic Algorithm

In the method of spline function fitting, the main idea is using GA to search and adjust a spline curve which could best fit the edge information in a denoised graph. Fig. 3 shows a crude fitting result of this method, in which the red line is the spline curve.

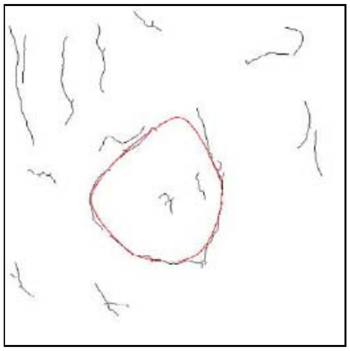


Fig. 3. Spline function fitting

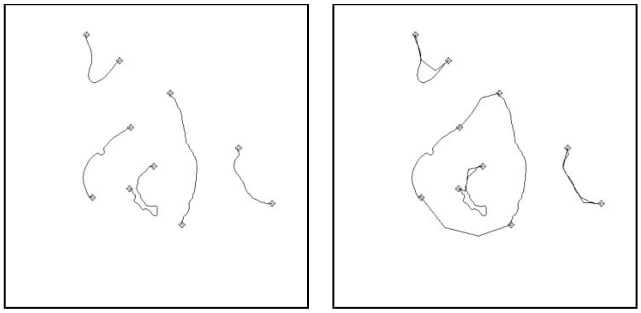


Fig. 4. Concepts of edge patching

As previously mentioned, the method is mainly trapped in the time cost because the searching for the spline curve runs in a domain of the whole image space. When searching space is too large, the speed and accuracy of GA would be discounted. Additionally, constructing a spline function to describe a smooth closed curve is not fast, but it must be executed every time GA calculates fitness, totally about thousands of times in one complete running. Against those existing weaknesses, the proposed method makes essential modifications in two aspects. One is using edge patching instead of whole edge fitting, which will reduce the size of searching space considerably. And the other is using n-fold segment to substitute the spline curve, which accelerates the fitness calculation.

In order to diminish the searching space, we need to utilize the information got from edge detection as much as possible. As a result, connecting the extracted edges to form the complete glomerulus edge becomes the solution. Fig. 4 shows the conceptual image of the idea of edge patching, in which the endpoints of edges selectively connects with each other by n-fold segment.

In the proposed method, for each endpoint, it will have several neighbor endpoints as connection candidates. An optimal connection is found between those endpoint pairs. And finally, all the connections are evaluated by their fitness. Those with high fitness will constitute the complete glomerulus edge in the final step.

In order to establish the searching of connections, we define a searching space named admission area for each two

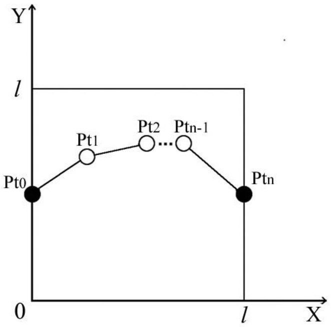


Fig. 5. Admission area for two endpoints

endpoints. This admission area is a square with length equal to l, as shown in Fig. 5, where Pt_0 and Pt_n are two endpoints, and Pt_1 to Pt_{n-1} are intermediate points. The connection which is realized by n-fold segment cannot go beyond the admission area. The reason is that due to the approximately spherical shape of the glomerulus, it is hardly for desired connection segment to break out of the admission area. Thus for any n-fold segment, it conforms to these rules:

$$x(Pt_0) < x(Pt_1) < x(Pt_2) < \ldots < x(Pt_n) \qquad (1)$$

$$0 \le x(Pt_k) \le l,\ \ 0 \le y(Pt_k) \le l \qquad (2)$$

where $x(Pt_k)$ is the x-axis value of kth point in the virtual coordinate. And Pt_0 is assigned as $(0, l/2)$, Pt_n is $(l, l/2)$. Note that this virtual coordinate does not need to be parallel with the real coordinate.

A. Chromosome Design

According to the ideas mentioned above, the task has become a searching problem which is suitable for GA to solve. GA will run once for every two endpoints which are likely to connect. And the object of each single GA is to determine the coordinate values of intermediate points Pt_1 to Pt_{n-1}, so we design the chromosome of GA like this:

$x(Pt_1)$	$y(Pt_1)$	…	$x(Pt_{n-1})$	$y(Pt_{n-1})$

where each square is a 9-digit binary number representing a number that is no larger than 512 in decimal since our image source is 512×512 pixels. As the different value of n is selected, the precision of the result will be different. It is reasonable to set n as 3, 4, or 5 in our problem, which means using 3-fold, 4-fold, or 5-fold segment respectively to approximate edge curves. Also, we can let GA dynamically adopt larger n for larger l since the requirement of precision will differ depending on different l. In our simulation program, we just simply set n equal to 3.

It is well known that GA is a robust and fast method for complex searching problem because its time complexity is about $O(t_1t_2+t_3)$, where t_1 for generation, t_2 for population and t_3 for gene length (does not consider the complexity of fitness calculation, the same below). Obviously, GA would get more advantage than common algorithms as exhaustive method when we use large n, since its time complexity is about $O(s^{n-1})$, where s for admission area. However, it can be proved that even if n is small (≥ 3), GA still runs faster than exhaustive method. For both methods in this case, Table I shows the time cost for n-fold segment connection.

TABLE I
GA VS. EXHAUSTIVE METHOD IN TIME COST

Symbol	Exhaustive Method[a]		Genetic Algorithm[b]	
	T_a	T_b	T_a	T_b
$n = 2$	900	0	5000	1
$n = 3$	900^2	0	5000	2
…	…		…	
$n = i$	900^{i-1}	0	5000	i-1

[a] admission area = 30×30 pixels

[b] population = 50, max generation = 100

In Table I, T_a represents the time for fitness calculation, and T_b represents the time for genetic operations (include selection, crossover, and mutation). Here we suppose the admission area is 30×30 pixels, but note that in many situations it will exceed this value. When $n = 2$, the total time cost of GA will be $T = 5000T_a + T_b$, which is larger than that of exhaustive method. However, when $n = 3$, the time cost of GA is only added by $1T_b$, whereas the time cost of exhaustive method becomes $81000T_a$. In our case, $81000T_a$ is much larger than $5000T_a + 2T_b$.

Simulations show that when $n = 3$, exhaustive method cannot obtain the results within several minutes, but GA could finish its work in 2 seconds. So it is obvious that GA runs faster than exhaustive method if $n \ge 3$.

B. Fitness Definition

Fitness is very important in GA because it decides how to evaluate the quality of solutions then the evolution could proceed. Since the result of Canny edge detection holds almost the complete information of glomerulus' edges, it could be used to calculate the fitness of any possible connection for each of two endpoints. If one connection segment could correctly patch a corresponding edge, it will be evaluated for a high fitness. The fitness function established below indicates that for each two endpoints i and j, the connection method C_{ij} has a fitness of:

$$Fitness(C_{ij}) = \frac{1}{n}\sum_k^n gray_k - \alpha L_{ij} - pe \qquad (3)$$

where, for kth point (or pixel) on the whole connection patch, $gray_k$ is the corresponding grayscale value (ranges from 0 to 255) at the same position of the result image of Canny edge detection. That means the first item of (3) becomes the overlap degree between the connection line and the detected edge. L_{ij} is the total length of connection segment between endpoint i and j; α is a coefficient; and pe is other penalties of fitness.

1) Average Grayscale: Average grayscale is a main index of the fitness function which reflects how much a connection segment overlaps with a detected edge. If a connection segment perfectly overlaps with an edge, its value of average grayscale should be 255. (Note that figures in this paper are black and white reversed, so 255 becomes the grayscale of black.) In addition, the edge information should be enhanced by dilation operation in order to accelerate the convergence speed of GA. Fig. 6 shows the result of dilation operation for the image of Fig. 2 (a).

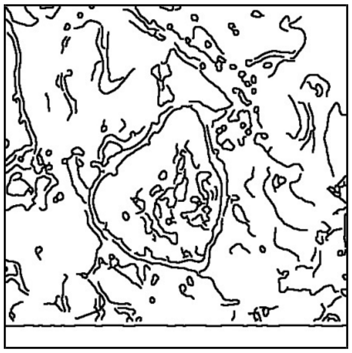


Fig. 6. Dilation result

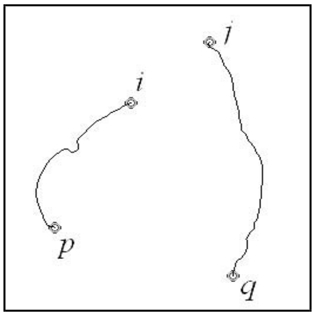


Fig. 7. Example of penalty

2) Length Factor: For one endpoint, there is not only one correspondence to build the connection. Since there are several candidates, usually, the nearest neighbor has the highest initiate to be connected. On the other hand, a lot of noise information could influence the connection. In most situations, if there are more than one connection path which has high average grayscale value, we would like to select the shortest one. As a result, the length factor is introduced to cut down the fitness according to the length of connections. In our simulation, the coefficient α is empirically set as 0.5.

3) Other Penalties: Due to the flexibility of GA, any fitness penalty (or bonus) which is suitable for the problem could be added in to optimize the solutions. In this problem, a rule is set as below:

$$(\forall C_{ij})\left(d_{iq} < d_{ij} \cup d_{jp} < d_{ij}\right) \rightarrow Fitness\left(C_{ij}\right) \text{ is penalized by } 50 \qquad (4)$$

where d_{ij} is the distance between i and j; p is another endpoint which is on the same curve of edge as i, and q is another endpoint on the same curve of edge as j. This rule reduces the possibility that the edge connections would intersect with each other. Take Fig. 7 as an example, where C_{ij}, C_{pq} will not be penalized whereas C_{iq}, C_{jp} will. This is meaningful because C_{ij} and C_{pq} can make the patched edge be approximately circular but C_{iq} and C_{jp} cannot.

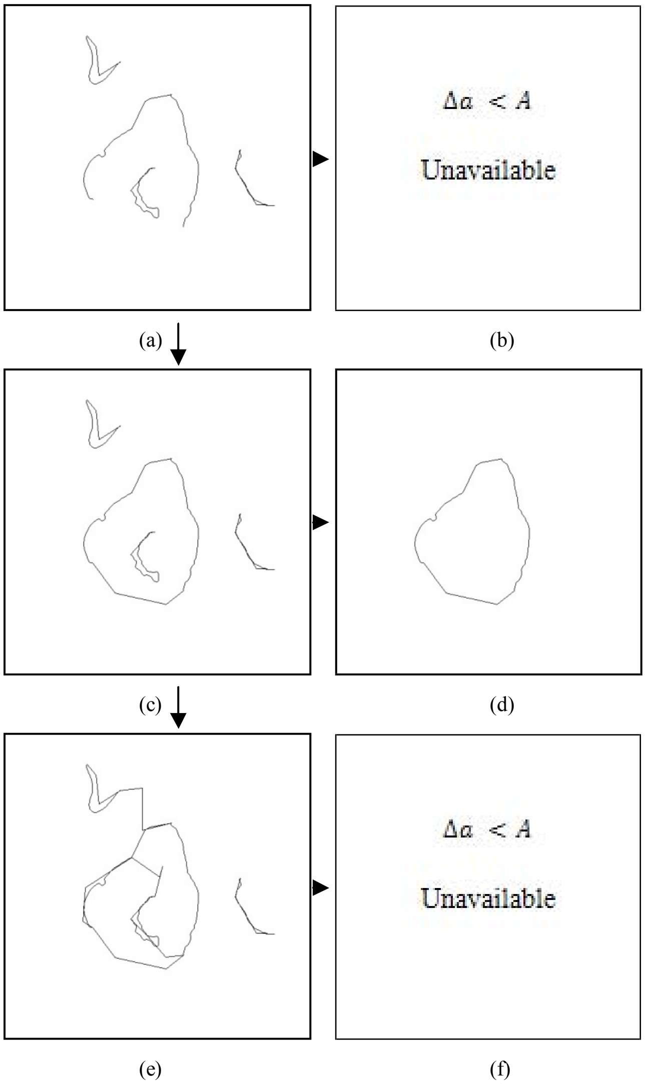
4) Heuristic Rules: There are some rules cannot be represented as penalties but also useful to improve the quality of solution. We adopt two rules as below:

a. For any endpoint, it could connect with one and only one endpoint per edge.

b. If there is more than one connection between two endpoints, only the one with largest fitness remains.

C. Genetic Operation

In our simulation, the population P = 50, maximal generation G = 100. For each generation, one elite is automatically selected into the next generation, and other individuals are selected by tournaments. One-point crossover is used, probability of crossover P_c = 0.2, probability of mutation P_m = 0.2. Each endpoint has three candidates for connection. And for all the connections, their final fitness after evolutionary are recorded for obtaining the result in the next step.



(a) (b)

(c) (d)

(e) (f)

Fig. 8. Procedure of obtaining result

IV. Obtaining The Result Based On Fitness

After the searching of GA, each of the two endpoints has an optimal connection with its fitness recorded. Arrange those connections according to fitness from large to small, and then patch them into the denoised image will lead to the final result.

In our designation, connection segments with large fitness will be the desired edges of original image, and those with small fitness should be discarded, but there is not a fixed fitness point to divide them. As a result, we introduce Δa, the area size increment to indentify when the edge patching has found its result. When each connection segment is placed in the image, trace a, the maximal area of all the inner contours of the graph, and compare it with the previous area size a_{prev}. If the area size increment $\Delta a = a - a_{prev}$ is larger than a predefined value A ($A = 10000$ in our program), it means the edge patching work is done. See Fig. 8 for the procedure description. The left side of Fig. 8 is the edge patching procedure by sequence of fitness, whereas the right side shows the output of result only if the program detects that Δa of left side image is larger than A.

The proposed method could also trace more than one area increment in the same time. During the procedure, if there is a second area size whose growth Δa_2 becomes larger than A, it indicates a second glomerulus exists in the image. For spline function fitting method, to find a second glomerulus in an image requires twice time cost as usual. (Remove the first glomerulus away from the original image, and then redo the whole procedure.) But this method could directly extract multiple glomeruli at a time.

V. Experiment And Comparison

We randomly took 13 glomerulus images as experiment samples, all of which could at least hardly be identified by human eyes. The results can be evaluated by overlap ratio of extracted glomerulus and real ones. And we divided the results into three levels:

good – overlap ratio larger than 95%
normal – overlap ratio larger than 80%
failed – overlap ratio lower than 80%

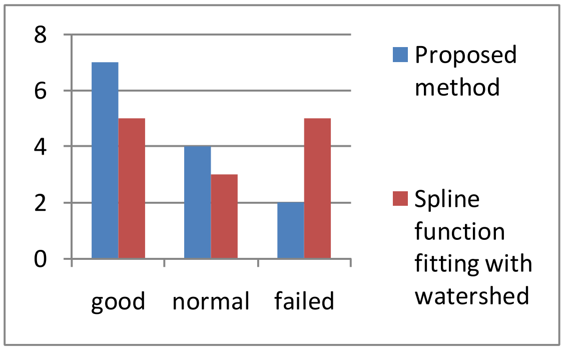


Fig. 9. Comparison results (both with default parameters)

At first, we tested the traditional method for cell extraction problems, which was ellipse fitting without any evolutionary computation. And the result was 3 of 13 good and 10 of 13 failed. The outcome indicated that the traditional method did not work well in this problem due to the various shape of glomerulus.

Secondly, the proposed method was compared with spline function fitting method combined with watershed transform proposed in [7]. Under the default parameters, the results of

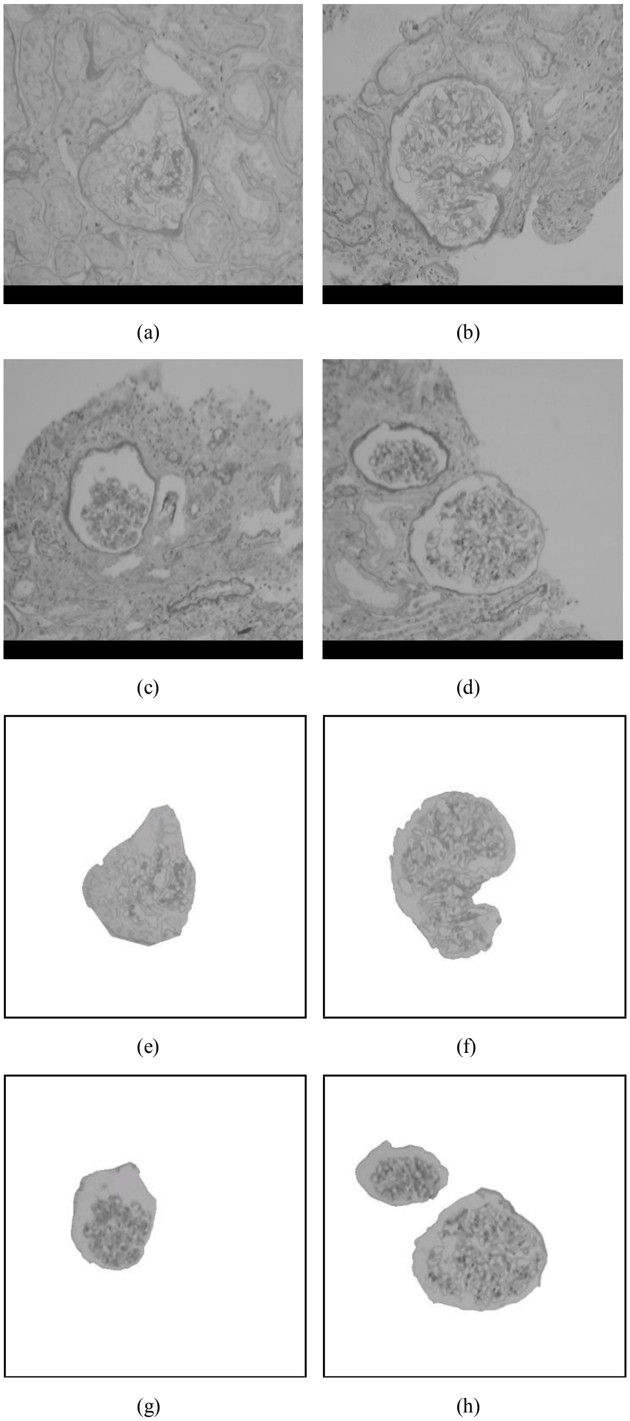

(a) (b)
(c) (d)
(e) (f)
(g) (h)

Fig. 10. Examples of experiment results

two methods are depicted in Fig. 9.

Although our test samples are not sufficient, the performance advantage of the proposed method is clear. Only two samples were failed and many got good results. Additionally, the average time cost of proposed method was about 3 seconds per image (running on T5500 1.66 GHz dual CPU and 1GB memory), whereas the spline function fitting needed about 1 minute. Fig. 10 (a) to (d) show the original kidney-tissue images, and Fig. 10 (e) to (h) are the corresponding results processed by the proposed method.

VI. Conclusion And Future Work

Glomerulus extraction is an important technique for realizing the diagnosis automation of kidney disease. This paper proposes a glomerulus extraction method by using genetic algorithm for edge patching. First of all, Canny edge detection followed by labeling is used to preprocess the kidney-tissue image. Then the method centers on GA to conduct a searching, in which we introduce in the a priori knowledge, like admission area and fitness penalties, as much as possible to diminish the searching space, improve the searching quality and raise the efficiency. Finally through tracing the area size increment, the edge patching procedure automatically detects the desired edges of glomerulus. Experiment and comparison results indicate that the proposed method has a significant advantage in processing speed without sacrificing the accuracy.

However, to use the proposed method in practice, we still need to increase the success rate of this method. Thus our future work mainly involves three aspects as follows. First, we want to add a step of fine tuning in the end to raise the accuracy further. Second, it is very helpful if we can find a more robust or suitable edge detection and denoising method. And third, as the proposed method is designed for glomerulus extraction, we also want to seek for its flexibility on other applications.

Acknowledgment

Jiaxin Ma thanks Wa Si for supporting to accomplish this paper.

References

[1] Zhou WZ, Chen YP, Zhang YK et al. *Illustrated Handbook of Nephric Biopsy Pathological Diagnosis*, Beijing: People's Medical Publishing House, 2000.

[2] N. Kharma, H. Moghnieh, J. Yao et al. "Automatic segmentation of cells from microscopic imagery using ellipse detection," IET2007, vol. 1, pp. 39-47, March 2007.

[3] Carlo Bruni, Alberto De Santis, Daniela Iacoviello, and Giorgio Koch. "Modeling for edge detection problems in blurred noisy images," *IEEE transactions on image processing*, vol. 10, no. 10, pp. 1447-1453, October 2001.

[4] Chan, T.F.; Vese, L.A.; "Active contours without edges," *IEEE transactions on image processing*, vol. 10, pp. 266-277, Feb. 2001

[5] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and W. G. Aref. "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE transactions on image processing*, vol. 10, no. 10, pp. 1454-1466, October 2001.

[6] Jun Zhang, Hong Zhu, and XueMing Qian et al. "Genetic algorithm for edge extraction of glomerulus area," ICIA2004, pp. 335-338, 2004.

[7] Jun Zhang, Jiulun Fan. "Glomerulus Extraction Based on Genetic Algorithm and Watershed Transform," IROS2006, pp4863-4866, 2006.

[8] Xie M, Ma Z. "Research of Edge Detection for B-Spline Wavelet," ACTA ELECTRONICA SINICA, vol. 27, pp. 106-108, 1999.

[9] Sharifi. M, Fathy. M, Mahmoudi. M. T. "A classified and comparative study of edge detection algorithms," Information Technology: Coding and Computing, 2002, Proceedings, pp. 117-120, 2002.