Image Segmentation: A Survey of Graph-cut Methods

Faliu Yi
Dept. of Computer Engineering
Chosun University
Gwangju, Republic of Korea
yifaliu@chosun.kr

Inkyu Moon*
Dept. of Computer Engineering
Chosun University
Gwangju, Republic of Korea
*Corresponding author: inkyu.moon@chosun.ac.kr

***Abstract*— as a preprocessing step, image segmentation, which can do partition of an image into different regions, plays an important role in computer vision, objects recognition, tracking and image analysis. Till today, there are a large number of methods present that can extract the required foreground from the background. However, most of these methods are solely based on boundary or regional information which has limited the segmentation result to a large extent. Since the graph cut based segmentation method was proposed, it has obtained a lot of attention because this method utilizes both boundary and regional information. Furthermore, graph cut based method is efficient and accepted world-wide since it can achieve globally optimal result for the energy function. It is not only promising to specific image with known information but also effective to the natural image without any pre-known information. For the segmentation of N-dimensional image, graph cut based methods are also applicable. Due to the advantages of graph cut, various methods have been proposed. In this paper, the main aim is to help researcher to easily understand the graph cut based segmentation approach. We also classify this method into three categories. They are speed up-based graph cut, interactive-based graph cut and shape prior-based graph cut. This paper will be helpful to those who want to apply graph cut method into their research.**

Keywords-image segmentation; N-dimensional image; energy function; graph-cut; survey;

I. INTRODUCTION

Image segmentation is partition of an image into different regions which may have similar color, intensity or texture [1-2]. Segmentation as a preprocessing step plays a significant role in computer vision, object recognition, tracking and image analysis. Conventionally, segmentation can be grouped into five categories. The first one is threshold based segmentation method [2, 3]. This method usually divides the image into two parts namely foreground and background. When the intensity of the pixels is larger/smaller than a predefined threshold, those pixels are classified as foreground. Otherwise, they will be viewed as background. Threshold based approaches are the simplest, easiest and fast ones among all of the existed segmentation methods. The difficulty is that it is not easy to find an appropriate threshold which can separate the image into two groups directly. This method also requires the foreground and background in the image have obviously different intensity values. Otsu's algorithm [1] is the most popular method for finding an appropriate threshold. Otsu's algorithm can find a threshold which may make the inter-part (between foreground and background) variance maximal and the intra-part (within foreground or background) variance minimal. The second category is edge based segmentation scheme [1-3]. This method assumes that the values of the pixels connecting foreground and background are distinct. These discontinuities are usually detected by the first or second order derivatives method like gradient and Laplace [2-4]. Sobel, Roberts and Rrewitt edge detectors [1, 4, 5], which are based on the gradient concept, are easy to be implemented and can roughly detect the contour profile but they are sensitive to the image noise. The Laplace algorithm [2-5] uses the second derivative to detect the edge which can localize the direction of the pixel along the edge but it is also sensitive to the noise. The LoG (Laplace of Gaussian) was introduced to reduce the effect of noise by smoothing the image with a Gaussian filter then utilize the Laplace operator. Canny edge detector was reported to have better detection result compared with previous ones [4-5] since this method have combined the operation of filter, enhancement and detection. Even though the boundary of the object can be detected by these proposed methods, many false edges will be included. Thus, post-processing operations are usually needed for the edge-based segmentation. The third category is region-based segmentation [2, 3, 6]. The typical algorithms are region growing and region splitting-merging [1]. For region growing scheme, a set of seeds are needed to be identified firstly. Then, the neighboring pixels are grouped to these seeds through predefined criteria such as by the similar intensity, color or texture. Hence, the skill of the selection of seed points is very important for region growing when no more prior information is known. For the region splitting-merging method, an image is first divided into a series of small regions. Then merge or split these smaller regions by a prerequisite condition. The procedure can be described as splitting of the image into many un-overlapped regions until it cannot be split anymore. Then, merge the adjacent regions that satisfy with a predefined condition. Region-based segmentation strongly relies on the intensity value of the object and background and it always produce un-smooth boundary for the extracted object. The forth category is watershed based segmentation [1, 2, 3, 7]. This method views the image as topological surface and the intensity value as height. The regional minimal values in the image are interpreted as catchment basins and the maximal values between every two neighboring catchment basins are viewed as ridge line. Watershed-based segmentation is to find the ridge line called watershed within the image. So, in order to extract the object, watershed transform algorithm is usually applied to gradient image where the object is corresponding to the catchment basins while the boundary to watershed. However, direct application of watershed algorithm will have over-segmentation problem due to the noise and other local irregularities of the gradient [1]. Marker-controlled watershed segmentation is proposed so as to reduce the over-segmentation. In this method, the regional minimal values only occur at the location of the markers. Thus, the key procedure is to identify the markers which include internal and external markers. Internal markers denote the object while the external markers represent the background and these external markers must be connected. When the markers can be identified appropriately, watershed based segmentation can obtain reasonable results. The fifth category is energy based segmentation. This method need to establish an objective (energy) function which will reach a minimal value when the image is segmented as expected result. Live wire [8], active contour [9], level sets [10-11] and graph cut [12-13] are all grouped into this category. For live wire, seeds are needed to be identified by user and these seeds have to be located at the object boundary. And then the curve position is optimized by minimizing the constructed energy function. For the active contour and level sets, the initial curve is needed to be given. When the curve is evolved with the predefined rule, extract the reasonable curve which will make the energy function have minimal value. However, active contour and level set only utilizes the boundary information and is very sensitive to the initialed curve. Furthermore, they cannot be guaranteed to get globally optimal result since they will converge at a local minimum. For the graph cut segmentation, the energy function is constructed based on regional and boundary information and it can achieve globally optimal result. Graph-cut segmentation was first proposed by Boykov and Jolly [12] in 2001. Since then, many varied methods based on graph-cut are developed and these approaches are widely used in medical image, video and natural image segmentation [12-22]. In this survey, we will first focus on the concept of graph cut segmentation. Then, we will introduce some popular methods in terms of speed up the computation of graph cut, interactive segmentation and graph cut segmentation incorporated with shape prior information.

The rest of the paper is organized as follows. In section 2, we describe the concept of graph-cut based segmentation. In section 3, we present the classification of graph cut based algorithms. In section 4, we conclude this paper.

II. GRAPH-CUT SEGMENTATION

In this section, we will introduce the concept of graph cut and how to establish the graph with the given image which will be segmented by the graph cut.

A. *graph cut*

Let an undirected graph be denoted as ***G=< V, E >*** where ***V*** is a series of vertices and ***E*** is the graph edge which connect every two neighbor vertices. The vertex ***V*** is composed of two different kinds of nodes (vertices). The first kind of vertices is neighborhood nodes which correspond to the pixels and the other kind of vertices are called terminal nodes which consist of ***s*** (source) and ***t*** (sink). This kind of graph is also called ***s-t*** graph where, in the image ***s*** node usually represent the object while ***t*** node denote the background. In this kind of graph, there are also two types of edges. The first type of edges is called ***n-links*** which connect the neighboring pixels within the image (Here we adopt 4-connected system in the 2D image). And the second type of edge is called ***t-links*** which connect the terminal nodes with the neighborhood nodes. In this kind of graph, each edge is assigned with a non-negative weight denoted as w_e which is also named as cost. A cut is a subset of edges ***E*** which can be denoted as ***C*** and expressed as $C \subset E$. The cost of the cut $|C|$ is the sum of the weights on edges **C** which is expressed as follows.

$$|C| = \sum_{e \in C} w_e \tag{1}$$

A minimum cut is the cut that have the minimum cost called min-cut and it can be achieved by finding the maximum flow which is verified in [12, 13, 23] that the min-cut is equivalent to max-flow. The max-flow/min-cut algorithm developed by Boykov and Kolmogorov [23] can be used to get the minimum cut for the s-t graph. Thus, the graph is divided by this cut and the nodes are separated into two disjoint subsets S and T where $s \in S$, $t \in T$ and $S \cup T = V$. The two subsets correspond to the foreground and background in the image segmentation. This kind of graph can be depicted in figure 1.

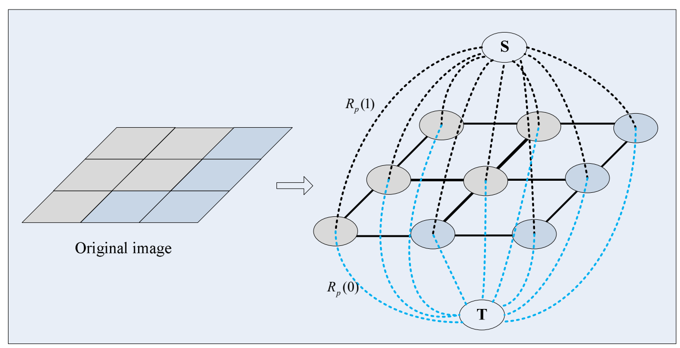

Figure 1. Illustration of s-t graph. The image pixels correspond to the neighbor nodes in the graph(except s and t nodes). The solid lines in the graph are n-links and the dotted lines are t-links.

B. *Graph cut segmentation*

Image segmentation can be regarded as pixel labeling problems. The label of the object (s-node) is set to be 1 while that of the background (t-node) is given to be 0 and this process can be achieved by minimizing the energy-function through minimum graph cut. In order to make the segmentation reasonable, the cut should be occurred at the boundary between object and the background. Namely, at the object boundary, the energy (cut) should be minimized. Let L= $\{l_1, l_2, l_3, \cdots, l_i, \cdots l_p\}$ where p is the number of the pixels in the image and $l_i \in \{0,1\}$. Thus, the set L is divided into 2 parts and the pixels labeled with 1 belong to object while others are grouped into background. The energy function is defined as following equation which can be minimized by the min-cut in the s-t graph [12-13].

$$E(L) = \alpha R(L) + B(L) \tag{2}$$

Identify applicable sponsor/s here. *(sponsors)*

Where, $R(L)$ is called regional term which incorporates the regional information into the segmentation and $B(L)$ is called boundary term which incorporates the boundary constraint into segmentation, α is the relative importance factor between regional and boundary term. When α is set to be 0, it means that the regional information is ignored and only considering the boundary information. In the energy function in eq. (2), the regional term is defined as following equation.

$$R(L) = \sum_{p \in P} R_p(l_p) \tag{3}$$

Where, $R_p(l_p)$ is the penalty for assigning the label l_p to pixel p. The weight of $R_p(l_p)$ can be obtained by comparing the intensity of pixel p with the given histogram (intensity model) of the object and background. The weight of the t-links is defined as following equations [12-13].

$$R_p(1) = -\ln \Pr(I_p \mid 'obj') \tag{4}$$

$$R_p(0) = -\ln \Pr(I_p \mid 'bkg') \tag{5}$$

From eq. (4) and (5), we can see that when $\Pr(I_p \mid 'obj')$ is larger than $\Pr(I_p \mid 'bkg')$, $R_p(1)$ will be smaller than $R_p(0)$. This means when the pixel is more likely to be the object, the penalty for grouping that pixel into object should be smaller which can reduce the energy in eq. (2). Thus, when all of the pixels have been correctly separated into two subsets, the regional term would be minimized. $B(L)$ In eq. (2) is the boundary term which is defined as following equation [12-13].

$$B(L) = \sum_{\{p,q\} \in N} B_{<p,q>} \cdot \delta(l_p, l_q) \tag{6}$$

Where p, q is neighboring pixels and $\delta(l_p, l_q)$ is defined as:

$$\delta(l_p, l_q) = \begin{cases} 1 & if\ l_p = l_q \\ 0 & if\ l_p \neq l_q \end{cases} \tag{7}$$

For the regional constraint, it can be interpreted as assigning labels l_p, l_q to neighboring pixels. When the neighboring pixels have the same labels, the penalty is 0 which means the regional term would only sum the penalty at the segmented boundary. For the term $B_{<p,q>}$, it is defined to be a non-increasing function of $|I_p - I_q|$ as follows [24]:

$$B_{<p,q>} \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \tag{8}$$

where σ can be viewed as camera noise. When the intensity of two neighboring pixel is very similar, the penalty is very high. Otherwise, it is low. Thus, when the energy function obtains minimum value, it is more likely occurred at the object boundary. In [12, 13], Boykov and Jolly have showed that the minimized energy can be computed by the min-cut through max-flow. Thus, the minimum energy problem is converted into the graph cut problem. In order to get a reasonable segmentation result, the assignment of the weight in the s-t graph is very important. In Boykov and Jolly's method, the weight of the s-t graph is given as following.

$$weight = \begin{cases} B_{<p,q>} & \{p,q\} \in Neighboring\ pixel \\ \alpha \cdot R_p(0) & for\ edge\ \{p,S\} \\ \alpha \cdot R_p(1) & for\ edge\ \{p,T\} \end{cases} \tag{9}$$

Eq.(9) can also be explained as that, in the s-t graph, when the intensity of the pixel is inclined to be the object, the weight between this pixel and s-node will be larger than that between pixel and t-node which means the cut is more likely occurred at the edge with smaller weight. For the neighboring pixels, when their intensity is very similar, the weight is very big which is not likely to be separated by the cut. Thus, when the minimum cut is achieved from the s-t graph, the location of the cut is close to the object boundary. The implementation of the graph cut can be fulfilled by the max-flow/min-cut as described in [12, 13, 23]. In figure 2, we illustrate the graph cut for a 3×3 image segmentation. The thickness of the edge denotes the magnitude of the weight.

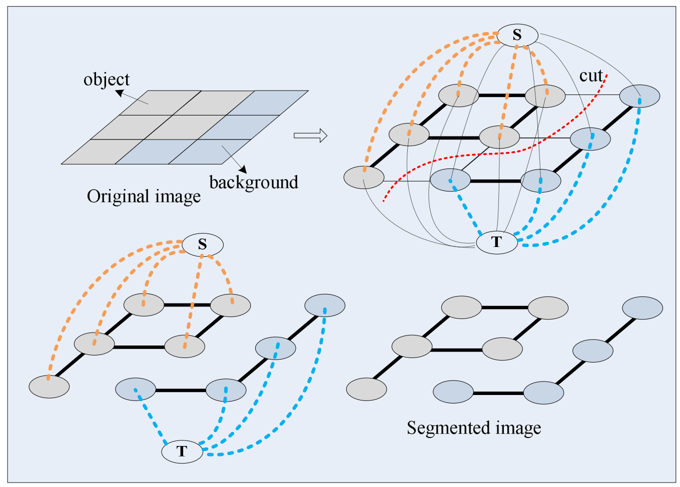

Figure 2. Illustration of graph cut for image segmentation. The cut is corresponding to the minimal energy of eq.(2)

III. CLASSIFICATION OF GRAPH CUT BASED ALGORITHMS

In this section, we classify graph cut based techniques into three classes. The first one is speed-up based graph cut which aims at improving the speed of graph cut based method. The second one is interactive-based graph up which incorporate user's interaction into the segmentation so as to improve the segmentation result. The last one is shape prior-based graph cut which take the shape prior of object into consideration to make the segmentation more reasonable. Next, we describe these techniques one by one.

A. Speed-up based graph cut

In order to speed up the computation of graph cut algorithm, implementation based on GPU with CUDA code is proposed in [25]. This kind of method is to fulfill the speed up by the parallel computing which have good performance compared with sequentially computing. However, the most used method for reducing the computational time for the graph cut related algorithms is based on the reduction of the graph nodes during the reconstruction of graph [14-16, 26-28]. Conventionally, each pixel in the image will be viewed as one node in the graph. Thus, with the increase of image resolution, the graph will be very big and make the computation of graph cut slowly. Since most of the case, the object just occupy a small region in the whole image, the object can be segmented by only considering a relative smaller part which cover the target [26-28]. In this case, the graph size is reduced since the image size is decreased to some extent. The smaller area can be selected by user's input or by some matching algorithm [26-28]. Moreover, for the image, when the intensity values of the two neighboring pixels are very similar, the corresponding weight in the graph will be very heavy which means they will not be separated by the graph cut. In other words, when the intensity values of two neighboring pixels are very similar, they can be regarded as one pixel. This idea produces the application of clustering method in graph cut. The most popular approach used in graph cut is watershed algorithm [14, 16, 26]. Usually, the watershed algorithm is applied to the gradient image. In watershed algorithm, the gradient image is viewed as a topological surface while the gradient values are regarded as the height. When drill a hole at each regional minimal value in the topological surface and put it into the water, the water will be erupted from each hole. The places where water can be reserved are called the catchment basin. When the water in the two neighboring catchment basins tends to merge, a dam is built between them. After the entire surface is immersed in the water, the left dams are the watershed line. Thus, the smaller region which has similar intensity value can be grouped as one cluster by watershed and these clusters are taken as points during the reconstruction of graph for graph cut. After the pre-segmentation of watershed, the average intensity of each cluster is viewed as the pixel value of the new point (super point) which consists of many similar pixels in the original image. Thus, the weight of the t-links can be set by comparing the value of the super point with the established histogram of the object and background similar with eq. (4) and (5). In some papers [26], the weight of the n-links is provided by the following equation.

$$W(i,j) = (\exp(\frac{-grad_{ij}(i)}{\max(grad_{ij}(i), grad_{ij}(j))}) + \exp(\frac{-grad_{ji}(j)}{\max(grad_{ij}(i), grad_{ij}(j))}))^6 \quad (9)$$

Where, $grad_{ij}(i)$ is the average gradient magnitudes of cluster i while j is its neighboring cluster. Since the watershed algorithm can produce many small clusters and the boundaries among these clusters can always preserve the original edge of the object in the original image, watershed based graph cut can segment the object appropriately which is similar with the result when the graph is established by the individual pixel in the original image. Furthermore, the watershed doesn't need to be executed in the whole image but just on some part of the image like the method in [26]. In [26], only the regions around the boundary are considered and segmented by watershed based graph cut. Speed-up based graph cut especially watershed-based graph cuts are widely used in different kinds of image such as medical image, industrial image and natural image [14-16] due to its accuracy and fast speed.

B. *Interactive-based graph cut*

To most of the images, it is difficult for the application of pure automatic segmentation. Especially for the natural images and images which the accuracy requirements of target segmentation are very high, interactive segmentation is inevitable. Interactive based graph cut varies from easily choosing the interest object region or simple seed points to iteratively seed point selection. In [26, 28, 29], they use the bounding box to select the interested object area. The center area in the bounding box is taken as object and the histogram of the object can be derived from these pixels while the area outside of the bounding box is viewed as background and the background histogram can also be obtained by them. In [18-19], they need to choose both object seeds and background seeds at one time so as to establish the graph with more reasonable weight. In [12-13], the iteratively interactive graph cut is applied. Thus, every time when the result is not perfect, more seeds will be added and the segmentation result can be revised until got the satisfying interest object. Iteratively interactive graph cut is also robust to the object with week boundary since it can ideal choose the object and background seeds along the week edge. All of the interactive graph cut will make the seed points being perfectly segmented to the object or background. Conventionally, the weight for the graph in the interactive graph cut segmentation can be given as following table [12-13] when the graph is denoted as $\boldsymbol{G}$=< $\boldsymbol{V}$, $\boldsymbol{E}$ > where **V=P** $\cup$ **{S,T}**.

TABLE I. WEIGHT OF EDGES

edge	**weight**	**condition**
<p, q>	$B_{<p,q>}$	{p, q} $\in$ N
{p, S}	$\alpha \cdot R_p(0)$	p $\in$ P (unknown)
	K	p $\in$ object
	0	p $\in$ background
{p, T}	$\alpha \cdot R_p(1)$	p $\in$ P (unknown)
	0	p $\in$ object
	K	p $\in$ background

In the above table, $k = 1 + \max_{p \in P} \sum_{q:\{p,q\} \in N} B_{<p,q>}$. Thus, when one point is classified as object, the weight between this point and **S** node in the established graph will be high while that will be 0 with **T** node. Similarly, when the point is selected as background, the weight between this node and T terminal will be high and the weight will be 0 for the edge between this node and **S** terminal. This strategy can make sure that the selected object points will be classified with the **S** node and the selected background points will be grouped into **T** node. However, for the iteratively interactive graph cut, the weight of the edge will be changed dynamically. After a round of graph cut, when some new points are chosen as object or background, the weight would be updated as the way given in table 1. However, when the weight is reset as the method in table 1, the graph cut has to be run again from the 0 flow by the maximum flow algorithm which will limit the computation time to a large extent. In order to utilize the previous graph and continue to run the mini-cut (maximum flow) algorithm based on the previous flow, the weight of the new selected points is given as table II [12-13].

TABLE II. WEIGHT OF NEW SELECTED NODES

edge	weight	condition
{p, S}	K+ $\alpha \cdot R_p(1)$	p ∈ object
	$\alpha \cdot R_p(1)$	p ∈ background
{p, T}	$\alpha \cdot R_p(0)$	p ∈ object
	K+ $\alpha \cdot R_p(0)$	p ∈ background

Therefore, when new points are selected as object or background, the new weight does not change the previous optical cut, which means the next optimal cut can be obtained based on the previous maximum flow. This kind of method is applied by almost all of the iteratively interactive algorithms.

C. Shape prior-based graph cut

Due to noise, diffuse edge or occluded objects, conventional graph cut which only incorporate regional and edge information cannot get ideal segmented object. Even though the interactive-based graph cut can reduce these problems to some extent, many rounds of interaction will be needed and this will affect the segmentation efficiency. Thus, the shape prior-based graph cut algorithms are widely researched which will incorporate the shape information of the object into the energy function so as to improve the segmentation result [18-22, 30-34]. Especially for the images with known prior information, shape prior-based graph cut can work well when the shape is described appropriately. To some shape prior-based graph cut method, they use the same energy function as described in section II [32, 33]. In [33], they first segment the object within a selected area, then, they use an ellipse shape to fit the obtained object and segment the area around the fitted ellipse boundary. However, most of the cases, the energy of the shape prior is combined into the energy function. The energy function for the shape prior-based graph cut is usually defined as following equation [18-22, 31, 34].

$$E(L) = R(L) + B(L) + E_{shape} \tag{10}$$

Where, $R(L)$ is the regional term, $B(L)$ is the boundary term and E_{shape} is the prior shape term. The energy for the regional and boundary term can be described as that in section 2. For the energy in the shape term, the aim is to make it smaller for the pixels around the object boundary and bigger for that far away from the boundary of object. There are two kinds of energy representation for the shape prior. The first one is to choose the approximating center pixel of the target as reference point and set the values of the pixel around the center points to be high while other values to be smaller according to the distance between the pixel and the center pixel or use the pre-obtained object as mask where the values within the object are high and the values in the background are low [17, 31-32]. Then, the new image can be regarded as shape template of the object and be incorporated into the energy function. Since the values of the center part in the object is relative bigger than other location in the template, the values in this temple is combined with the regional term. So, the energy function in eq.(10) is changed as following equation.

$$E(L) = (R(L) + E_{shape}) + B(L) \tag{11}$$

Thus, in the constructed graph, the shape prior information will be reflected in the regional term (t-links) while the boundary term (n-links) can be obtained by the way in section 2. The other representation for the shape prior energy also use the distance transform. But this type of distance transform is different from the previous one because the distance values in the boundary of the reference object is zero while other valves are the shortest distance between pixel to the boundary (zero level set) [18-20]. So, the pixel value around the object boundary will be small while other values are bigger in this shape template. For this kind of energy expression, the shape prior information will be incorporated into the boundary term in eq.(10) and can be denoted as follows.

$$E(L) = R(L) + (B(L) + E_{shape}) \tag{12}$$

Therefore, the weights of the t-links are obtained as the traditionally way while the weights of the n-links are computed with the gradient and shape prior information. For the shape prior-based graph cut, the establishment of the shape template is very important. Many pre-segmented object will be adopted to be the training set to build the shape template. Furthermore, the alignment of the training set is also inevitable which will make the shape template fit the segmented image better [31-32].

IV. CONCLUSION

In this paper, we have briefly described the existed segmentation method and the advantage of graph cut in the introduction section. We also present the graph-cut concept in detail which will be helpful for the new researchers. Due to a lot of graph cut-based segmentation method which will confuse the research direction, we have classified these methods into three categories. They are speed up-based graph cut, interactive-based graph cut and shape prior-based graph cut. After this classification, researcher can put weight to different aspect as their requirement. However, it is not necessary for the three kinds of graph cut methods to be executed independently. Most of the time, they can be combined so as to improve the segmentation result.

ACKNOWLEDGMENT

This research was supported by Mid-career Researcher Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0010391).

References

[1] Gonzalez, R.C.; Woods, R.E. *Digital Imaging Processing*; Prentice Hall: New York, NY, USA, 2002.

[2] S. Raut, M. Raghuvanshi, R. Dharaskar, A. Raut,"Image Segmentation-A State-of-Art Survey for Prediction," ICACC, pp.420-424,2009.

[3] S. Naz, H. Majeed, H. Irshad,"Image segmentation using fuzzy clustering: A survey," International conference on emerging technologies, pp.181-186,2010.

[4] L. Wang,"Comparison for Edge Detection of Colony Image," IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.9A,2006.

[5] Z. Musoromy, R. Soodamani, B. Nico,"Edge Detection Comparison for License Plate Detection,"Int.Control,Automation, Robotics and Vision,2010.

[6] I. Karoui, J. Boucher, J. Augustin,"Variational Region-Based Segmentation Using Multiple Texture Statistics," IEEE trans on Image processing, vol.19,no.12,2010..

[7] C. Rambabu, I. Chakrabarti, A. Mahanta,"Flooding-based watershed algorithm and its prototype hardware architecture," IEE Proceedings-vision image and signal processing, vol.151,no.3,pp.224-234,2004.

[8] A. X. Falcao, J. K. Udupa, F. K. Miyazawa,"An ultra-fast user-steered image segmentation paradigm: live wire on the fly,"IEEE trans on Medical Imaging,vol.19,no.1,pp.55-62,2002.

[9] G. Sundaramoorthi, A. Yezzi, A. C. Mennucci,"Coarse-to-Fine Segmentation and Tracking Using Sobolev Active Contours,"IEEE trans on Pattern Analysis and Machine Intelligence,vol.30,no.5,pp.851-864,2008.

[10] R. M. Willett, R. D. Nowak,"Minimax Optimal Level-Set Estimation," IEEE trans on Image Processing, vol.16,no.12,pp.2965-2979,2007.

[11] H. Yu, D. Wang, Z. Tan,"Level Set methods and image segmentation,"International workshop on medical imaging and augmented reality, pp.204-208,2002.

[12] Y. B. Yuri, G. F. Lea, "Graph Cuts and Efficient N-D Image Segmentation," International Journal of Computer Vision,vol.70,no.2,pp.109-131,2006.

[13] Y. B. Yuri, M. P. Jolly,"Interactive Graph Cuts for optimal Boundary & Region Segmentation of Objects in N-D Images," Proceedings of Internation Conference on Computer Vision,vol.I,pp.105-112,2001.

[14] Z. Yu, M. Xu, Z. Gao,"Biomedical image segmentation via constrained graph cuts and presegmentation,"International conference of the IEEE on EMBC, pp.5714-5717,2011.

[15] X. Wu, W. Xu, L. Li, G. Shao, J. Zhang,"An Interactive Segmentation Method Using Graph Cuts for Mamographic Masses,"International conference on iCBBE, pp.1-4,2011.

[16] X. Wu, Y. Wang,"Interactive Forreground/Background Segmentation Based on Graph Cut,"Congress on Image and Signal processing,pp.692-696,2008.

[17] V. Jelen, J. Janacek, Z. Tomori,"Mobility tracking by interactive graph-cut segmentation with Bi-elliptical shape prior,"IEEE inter symposium on SAMI, pp.2258-230,2010.

[18] D. Freeedman, T. Zhang,"Interactive graph cut based segmentation with shape priors,"IEEE computer society conference on CVPR,vol.1,pp.755-762,2005.

[19] V. Nhat, B. S. Manjunath,"Shape prior segmentation of multiple objects with graph cuts,"IEEE conference on CVPR,pp.1-8,2008.

[20] H. Wang, H. Zhang,"Adaptive shape prior in graph cut segmentation,"IEEE international conference on ICIP, pp.2029-3032,2010.

[21] P. Furnstahl, A. Schweizer, L. Nagy, G. Szekely, M. Harders,"Automatic and robust forearm segmentation using graph cuts," IEEE inter symposium on ISBI, pp.77-80,2008.

[22] J. Zhou, M. Ye, X. Zhang,"Graph cut segmentation with automatic editing for Industrial images,"Inter conference on ICICIP,pp.633-637,2010.

[23] Y. Boykov, V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorrithms for energy minimization in vision,"IEEE trans on PAMI,vol.26,no.9.pp.1124-1137,2004.

[24] P. Das,O. Veksler,"Semiautomatic Segmentation with Compact Shape Prior,"Canadian Conference on Computer and Robot Vision,pp.28-41,2006.

[25] V. Vineet, P.J.Narayanan,"CUDA cuts: Fast graph cuts on the GPU,"IEEE conference on CVPRW,pp.1-8,2008.

[26] Y. Peng, R. Liu,"Object segmentation based on watershed and graph cut," International congress on CISP, pp.1431-1435,2010.

[27] Y. Hou, B. L. Guo,J. S. Pan,"The Application and Study of Graph Cut in Motion Segmentation,"Inter conference on IAS,pp.265-268,2009.

[28] V. Lempitsky, P. Kohli, C. Rother, T. Sharp,"Image segmentation with a bounding box prior,"IEEE conference on computer vision,pp.277-284,2010.

[29] L. Chang, W.H. Hsu,"Foreground segmentation for static video via multi-core and multi-modal graph cut,"IEEE international conference on ICME,pp.1362-1365,2009.

[30] X. Lang, F. Zhu, Y. Hao, Q. Wu,"Automatic image segmentation incorporating shape priors via graph cuts,"International conference on ICIA,pp.192-195,2009.

[31] A. M. Ali, A. A. Farag, A. S. El-Baz,"Graph Cuts Framework for Kidney Segmentation with Prior Shape Constraints,"MICCAI,vol.4791,pp.384-392,2007.

[32] J. Malcolm, Y. Rathi, A. Tannenbaum,"Graph Cut Segmentation with Nonlinear Shape Priors,"IEEE international conference on ICIP, pp.365-368,2007.

[33] G. Slabaugh, G. Unal,"Graph cuts segmentation using an elliptial shape prior,"IEEE international conference on ICIP,vol.2,pp.1222-1225,2005.

[34] O. Veksler,"Star Shape Prior for Graph-Cut Image Segmentation,"Proceeding of the 10th European Conference on Computer Vision,2008.