An Optimized Iteration Algorithm based on C-V Model and Graph Cuts

Hong Lan*[1,2], Lequan Min[1,3]
[1]Schools of Automation and Eclectronic Engineering, University of Science and Technology Beijing. Beijing,10083,China
[2]School of Information and Technology, JiangXi University of Science and Technology GanZhou,341000, China
[3] Schools of Mathematics and Physics, University of Science and Technology Beijing Beijing,10083,China
*Corresponding author, e-mail: lanhong69@163.com

Abstract

C-V model can self-adapt to the changes of curve topology but requires more iterations and needs more computing time. Graph cuts algorithm is good at getting the global optimum in a short time but not suitable for concave object extraction. To overcome the flaws of these two algorithms, an optimized iteration algorithm has been proposed. First the initial contour is deformed with an improved C-V model, which was without re-initialization during iterative process and the iteration stop condition is set by calculating changing area within the contour. Then the active contour is input to graph cuts algorithm. Dilates the contour into its neighborhood and formed an inner and an outer boundary seperatively, changes these two boundaries as source and sink, and obtains the final contour by graph cuts. Experiments show that this optimized algorithm reduces the iteration time, and has better effect and higher efficiency for image segmentation.

Keywords*: C-V model, iteration, graph cuts algorithm, image segmentation*

1. Introduction

Active contour models have gained popularity in image segmentation in recent years. They generally come in two kinds: one kind are the parametric models, such as the Snake model [1, 2], and the other are geometric models, like the Level Set method [3-5]. The first type models are the edge-based models, depending on the gradients of the images. The second kind ones are region-based, which represent contours as the zero level set of an implicit function defined in a higher dimension and calculate the evolution of this new function. Compared with the parametric models, the major advantage of geometric models is that they can self-adapt to the curve topology changes in the evolution process, and this characteristic makes up for the shortage of the parameter models which need to track the positions of the curve evolution .

C-V model [5] is a classic geometric deformation model. The model can accurately segment the object even when the initial contours are not completely in the internal or external homogeneous regions. But during the process of evolution, the contour has to be re-initialized with signed distance function in each iteration, which makes the algorithm spending more computing time and reducing the segmentation accuracy. Chunming Li [6] proposed a new variational level set model without re-initialization, which defined an external energy to be a penalty function and drived the zero level curve toward the object boundary. This model saved some iteration time,but because the penalty function was based on the gradient, the evolution results was limited to stop at the boundary nearest to the initial contour.

Graph cuts [7, 8] based on the graph theory has also been widely used in image segmentation because of their efficiency and global optimization characteristics. Min-cut/max-flow [9] algorithm is classical one of them. Ning Xu [10] proposed GCBAC (Graph Cuts Based Active Contours) algorithm, which was the combination of max-flow/min-cut graph theory and geometric deformation model. GCBAC algorithm improved computing efficiency but not suitable for cancave images. Ref [11] has presented a dual Level sets algorithm to improve GCBAC.

Received January 5, 2013; Revised May 12, 2013; Accepted May 20, 2013

The algorithm defined two initial contours in the target area and set up two level set functions. But two contours and two level set functions increased the complexity of the algorithm. And the iteration termination threshold was set as 10^{-4} to 10^{-3} times of the image size is not generality for all images.

Considering the characteristics of C-V model and graph cuts, the former is self-adapt to the changes of curve topology and the latter is fast spped. This paper proposed an optimized iteration algorithm based on the two algorithms. Contrast with the dual level sets algorithm in Ref [11], the optimized algorithm designed a single level set as initial contour to simplify the evolution process and set the iterative termination condition based on area within the contour. The proposed algorithm combines both algorithms' advantages, effectively reduced the level set iteration times and prevents the evolution falling into the local minimum. The algorithm makes the contour faster converge to the object boundary and improves the segmentation efficiency and effectiveness.

2. Research Method

The optimized algorithm is an integrated schema which includes following steps: firstly, optimize initialization with improve C-V model; secondly, provide a new approach to set iteration terminated condition; thirdly, to improve iteration efficiency with graph cuts algorithm.

2.1. Improved C-V Model to Optimize Initialization

C-V model was based on curve evolution theory and level set method. Let an image $I(x,y)$,which is divided into two homogeneous regions $in(C)$ and $out(C)$ by a contour C. Suppose that $I(x,y)$ is gray level distribution uniformity in there two regions, obtain the energy functional of the C-V model is:

$$E(C,c_1,c_2) = \mu Length(C) + \lambda_1 \iint_{in(C)} |I(x,y) - c_1|^2 \, dxdy + \lambda_2 \iint_{out(C)} |I(x,y) - c_2|^2 \, dxdy \tag{1}$$

In which, c_1 and c_2 are the average gray levels in the two homogeneous regions. $\mu \geq 0$, $\lambda_1, \lambda_2 > 0$ are weighting coefficients for each term. The first term represents the length of closed curve C and the last two are global binary fitting terms.

Take Heaviside and Dirac function into Equation (1) and get the level set evolution equation:

$$E(\varphi, c_1, c_2) = \mu \iint_\Omega \delta_\varepsilon(\varphi) |\nabla(\varphi)| dxdy + \lambda_1 \iint_\Omega |I(x,y) - c_1|^2 H_\varepsilon(\varphi) dxdy + \lambda_2 \iint_\Omega |I(x,y) - c_2|^2 (1 - H_\varepsilon(\varphi)) dxdy \tag{2}$$

Where $H_\varepsilon = \frac{1}{2}\left(1 + \frac{2}{\pi} \arctan\left(\frac{\varphi}{\varepsilon}\right)\right)$ and $\delta_\varepsilon = H_\varepsilon'(\varphi) = \frac{1}{\pi} \frac{\varepsilon}{\varepsilon^2 + \varphi^2}$.

Minimizing the energy functional $E(\varphi, c_1, c_2)$ gets the fomulations of c_1 and c_2 to be:

$$c_1(\varphi) = \frac{\iint_\Omega I(x,y) H(\varphi(x,y)) dxdy}{\iint_\Omega H(\varphi(x,y)) dxdy}, \quad c_2(\varphi) = \frac{\iint_\Omega I(x,y)(1 - H(\varphi(x,y))) dxdy}{\iint_\Omega 1 - H(\varphi(x,y)) dxdy} \tag{3}$$

Substitute the representations (3) of c_1 and c_2 into equation (2). And then according to the variational principle and gradient descent flow method, solution of the equation (2) is:

$$\frac{\partial \varphi}{\partial t} = \delta(\varphi)\left[\mu\, div\left(\frac{\nabla \varphi}{|\nabla \varphi|}\right) - \lambda_1 (I(x,y)-c_1)^2 + \lambda_2 (I(x,y)-c_2)^2\right] \quad (4)$$

In the original C-V model, the solution of Equation (4) depends on its initial signed distence function φ_0. During the iteration process, level set fucntion φ has to be re-initialized in each iteration with φ_0, because the level set function is easy to deviate far from the signed distance function. Its solution is sensitive to initial contour and spends much more computing time.

In order to solve the re-initialization problem and improve the speed and efficiency for original C-V model, we now reference the method which proposed by Chunming Li [6]. Introduced a penalizing energy into the signed distance function, the fomulation of the penalized item is :

$$P(\varphi) = \iint_{\Omega} \frac{(|\nabla \varphi| - 1)^2}{2} dxdy \quad (5)$$

Calculate its gradient descent flow:

$$\frac{\partial \varphi}{\partial t} = \Delta \varphi - div\left(\frac{\nabla \varphi}{|\nabla \varphi|}\right) = div\left[\left(1 - \frac{1}{|\nabla \varphi|}\right)\nabla \varphi\right] \quad (6)$$

Substuituite the Equation (6) into Equation (4) and get the evolution formulation for C-V model without re-initialization:

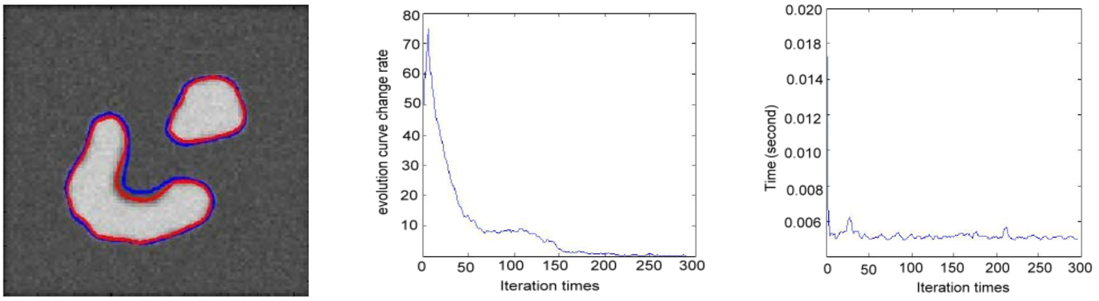
$$\frac{\partial \varphi}{\partial t} = \delta(\varphi)[\mu\, div\left(\frac{\nabla \varphi}{|\nabla \varphi|}\right) - \lambda_1 (I-c_1)^2 + \lambda_2 (I-c_2)]^2 + \nu\left(\Delta \varphi - \nabla \cdot \left(\frac{\nabla \varphi}{|\nabla \varphi|}\right)\right) \quad (7)$$

Where ν is the weighted coefficient for the last term.

Using Equation (7) gives the improved C-V model without re-initialization. Equation (7) is better to avoide the re-initialization than Equation (4) and improves C-V model's speed in some degree.

C-V model is based on level set functions, which increases the space dimensions and computation complexity. Actually, the key point of the algorithm implementation for C-V model is numerical iteration. When to end iteration depends on the iterative stop conditions. Although the more times the algorithm iterates, the closer the contour is to the object boundary, it spends much more time either. Especially when the contour is quite close to the object boundary, in each iteration the contour is only changed a little or sometimes no change.

Figure 1 shows an example about the relations of the iteration times, cost time and segmentation results with C-V model.



(a) (b) (c)

Figure 1. Relation Chart of Evolution Curve Change Rate and Iteration Times

Figure 1(a) is a testing image, bule and red curve respectively represents the place after iterated 200 and 300 times. Figure 1(b) is the relation chart between iteration times and curve change rate, here we define ΔS denotes the area surrounded by the curve on the ith iteration, t_i means the iteration cost time. curve change rate $r_i = \Delta S / t_i$. Figure1(c) is the relation between the iteration times and cost time.

From figure 1 we can see that when the algorithm iterated from 200 times to 300 times, there’re no significant changes for the evolution curve. And during the whole iteration, the cost time is almost the same. Usually the suitable iteration times get through tests or experiments. The best one to be choosed should consider segmentation accuracy and cost time. In this paper we introduce a new method to set the end condition of iterations so that improve the efficiency of C-V model.

2.2. Iteration Terminated Condition based on Area

In the original C-V model, the threshold of termination iterated condition α value is usually estimated through experiments. In our optimized algorithm, we introduced a changing variable P to be the iteration terminated condition. P denotes the inner area surrounded by evolution contour. The formulation of calculting P is shown as below :

$$P = \frac{\sum\limits_{|\varphi_{i,j}^{n}|<h} \left|\varphi_{i,j}^{n+1} - \varphi_{i,j}^{n}\right|}{num\left\{(i,j)\,\middle|\,|\varphi_{i,j}^{n}| < h\right\}} \le h^2 \tau \qquad (8)$$

Where (i,j) denotes the coordinate of each pixel in the image, $\varphi_{i,j}^{n}$ and $\varphi_{i,j}^{n+1}$ are functions respectively to represent the nth and $(n+1)th$ level set function. Actually P is a finite difference formulation, in which h is the time interval and τ is the space interval. The values of h and τ are set depending on different images, usually $h \in [0.1,1]$ and $\tau \in [0.01, 0.1]$. $num\left\{(i,j)\,\middle|\,|\varphi_{i,j}^{n}| < h\right\}$ means the number of grids which satisfy the conditional inequality $|\varphi_{i,j}^{n}| < h$. Corresponding to the term threshold of termination iterated condition, define $\alpha = h^2 \tau$.

In Equation (8), when P satisfies the criterion, the algorithm will exit the iteration process of level set function. Otherwise the iteration will continue. Using formulation (8) to create termination iterated condition for Equation (7) speeds up the iteration efficency of abrove improved C-V model. In our proposed algorithm, improved C-V model is used to pr-process the inital contour.

2.3. Graph Cuts Algorithm for Image Segmentation

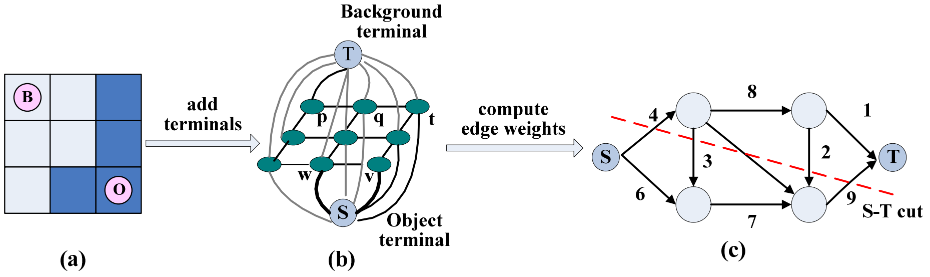


Figure 2. $S - T$ cut Schematic Diagram

The basic idea of graph cuts algorithm is to change the image into a graph and then uses graph theory to realize segmentation [7]. Given an image with the seeds of object and background respectively, as shown in Figure 2(a), O denotes object and B denotes background. Construct its weighted graph $G=(V,E)$, shown in Figure 2(b) and 2(c). V is the vertex set of the graph and consists of image pixels, additional including two terminals S and T , which respectively denotes the terminal of object and background. E is the edge set of the graph representing the connection relations of adjacent pixels in the image, also including those edges from each pixel to terminal S and T . Compute edge weights based on each vertex and its neighbourhood [12].

Define a $S-T$ cut to be a subset C of the edge set E , $C \subseteq E$. $S-T$ cut seprates the graph into two connected sub-graphs. The vertices connecting with S and T are respectivelly corresponding to the object and background pixels. The capacity of a $S-T$ cut is defined as the summation of the edges weights acrossing the cut , i.e. $c(S,T)=\sum_{u\in S,v\in T,(u,v)\in E} c(u,v)$. According to the Theorem of Ford-Fulkerson [8] , in the $S-T$ network, the maximum flow from a vertex S to vertex T is equal to the capacity $c(S,T)$ of the minimum cut separating S and T . The $S-T$ minimum cut problem can be solved by using max-flow algorithms.

Based on max-flow/min-cut algorithm, Ning Xu et.al. [10] proposed GCBAC algorithm in 2007. The basic idea of the algorithm is to represent the image as an adjacency graph. Define an initial contour and dilate the contour into its neighborhood with the inner and outer boundary, eventrally form a narrow band. Then treat the vertices in the inner boundary as the source S and the vertices in the outer boundary as the sink T . Calculate the edge weights and use $S-T$ minimum cut method to obtain a new boundary.

GCBAC algorithm overcomes the defects that the traditional deformation models are easy to fall into local optimal, and because its calculating is within a narrow band, the algorithm is high efficiency and low time complexity. But GCBAC algoritm is not good at segmentating the concave images, and because the initial contour in GCBAC is set randomly, if the contour is not close to the object boundary, it will affect the narrow band to form and result in the inaccurate segmentation results. Figure 3 shows an liver image segmentation with GCBAC.

(a) Initial contour (b) Segmentation result

Figure 3 Liver Image Segmentation with GCBAC Algorithm

In Figure 3, image pixels of the foregrand object and the background are similar. there's a cancave inside the liver, from Figure 3(b) we can see it's not good for GCBAC's algorithm to get the object boundary.

Aim at the shortage of GCBAC algorithm depending on the initial contour, we use our improved C-V model to pre-process the initial contour. Because C-V model can adapt to the changes of curve topology, it can help GCBAC algorithm to realize cancave images segmentation. On the other hand, GCBAC algorithm can help C-V model to improve the segmenting speeds and accuracy.

2.4. Main Process of the Optimized Algorithm

The optimized algorithm of our proposed includes two parts: initialization optimization and iteration process optimization. The main process of the optimized algorithm is shown as Figure 4.

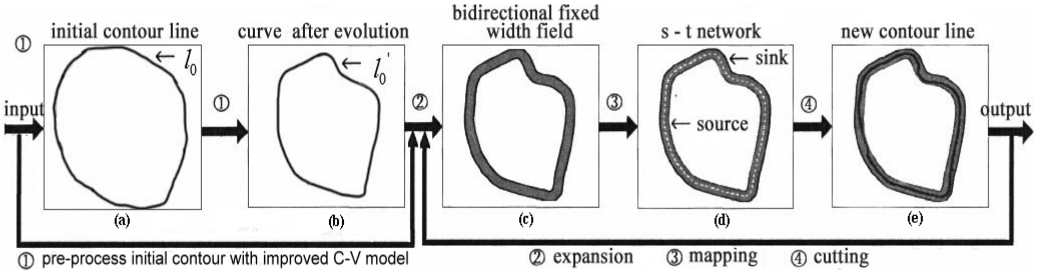


Figure 4. The Process of the Optimized Algorithm

In Figure 4, the first two steps (a) and (b) indicate the pre-process initialization with improved C-V model and the last three steps (c) to (e) show the evolution process with GCBAC algorithm.

(1) Initialization optimization. First set a single level set contour as an initial contour. As shown in Figure 4(a), set an initial contour $\boldsymbol{l_0}$ surrounded the object area, choose our improved C-V model, iterate the evolution contour by level set without re-initialization. When iteration stops , get the curve of after evolution $\boldsymbol{l_0'}$ and output to the next step, shown as Figure 4(b). Because $\boldsymbol{l_0'}$ has been pre-processed with the level set evolution , much closer to the object boundary than random initial contour, it is more efficiency for next step's further evolution.

(2) Iteration process optimization. Set $\boldsymbol{l_0'}$ as a basic bounday, expand to bidirection based on the pixel and its neighbood information until reach the fixed width field and form the inner boundary and outer boundary, shown as Figure 4(c). Then map these vertiex to S - T network, Identify all the vertices corresponding to the inner boundary as a single source S and the vertices corresponding to the outer boundary as a single sink T , shown as Figure 4(d). Finally, use min-cut/max flow algorithm to get the real object boundary extraction, shown as Figure 4(e). From Figure 4(c) to 4(e), the algorithm runs from step ②—expantion, step ③—mapping, and step ④—cutting. This process is a cycle, until the contour reach the desired boundary.

2.5. Realization Steps of the Optimized Algorithm

According to the above introduction of the algorithm's baisc idea, we can get the specific steps of the optimized iterative termination algorithm based on the improved C-V model and graph cut as follows:

Step 1: Initialize the threshold $a = h^2 t$ and the contour $\boldsymbol{l_0}$;

Step 2: Create the corresponding level set function j_0;

Step 3: Use PDE method , ie. Equation (7) to iterated evolution with level set function;

Step 4: Compute the area P in Equation (8) , surrounded by contour after evolution, Judge if the area is less than a , if yes, stop iteration and go to Step 5, otherwise, go back step 3.

Step 5: Output the final contour $\boldsymbol{l_0'}$ as a new initial contour , then expand the contour bidirectionally to form a narrow band area.

Step 6: Construct the S - T network including terminals of a single source and a single sink with the node's identity algrithm.

Step 7: Use the max-flow/min-cut algorithm to relazise the segmentation ,and get the final contour $\boldsymbol{l}$.

Summarize the steps of the algorithm, the flow chart of the optimized iterative termination algorithm is shown in Figure 5.

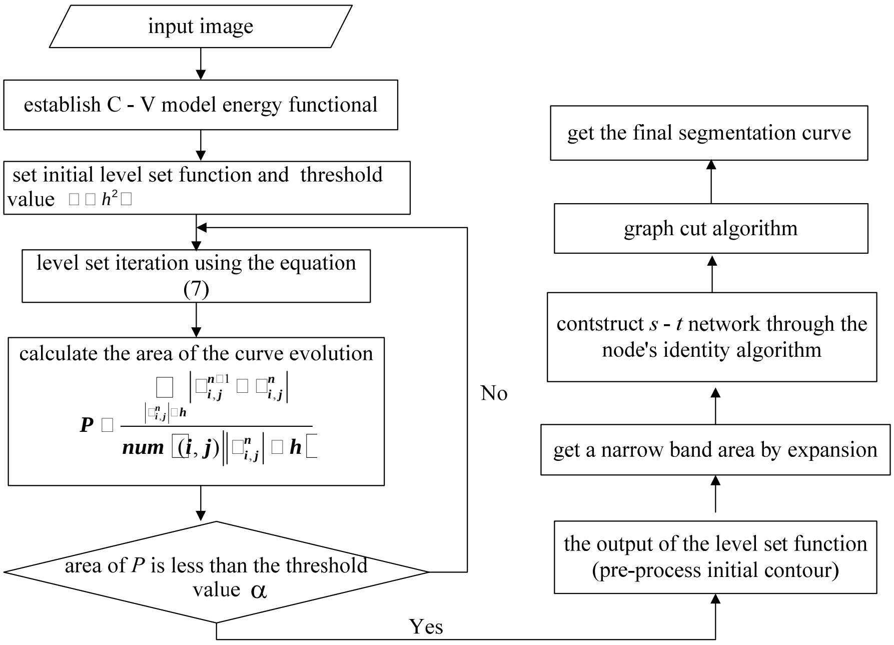

input image
establish C - V model energy functional
set initial level set function and threshold value h^2
level set iteration using the equation (7)
calculate the area of the curve evolution
P
num
h
area of P is less than the threshold value α
No
Yes
the output of the level set function (pre-process initial contour)
get a narrow band area by expansion
contstruct s - t network through the node's identity algorithm
graph cut algorithm
get the final segmentation curve

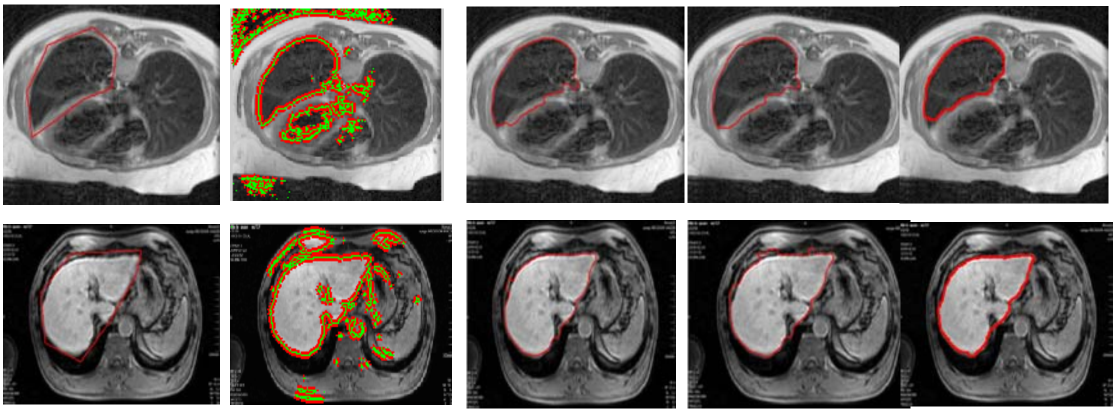
Figure 5. FlowChat of the Optimized Iterative Termination Algorithm

3. Results and Analysis

In order to verify the validity of optimized iterative termination algorithm, the algorithm has been realized with Matlab and C++ on Window XP operating system platform. Comprehensive considering the factors of the images, including image types, concave and noises etc. In this paper, we choose three types images to test. Using traditional C-V model algorithm, Reference [10]'s algorithm and our optimized algorithm to segment these three images respectively. Compared their segmentation results as shown in Figure 6 to Figure 8.

3.1. Experiments

Figure 6 showed two medical images segmentation results. One is lung CT image with the size of 211×225 pixels and the other is liver MR image with the size of 633×1031 pixels. Take the same operations.



(a)Initial Contour

(b)segmentation result of the C-V model algorithm

(c) segmentation result of Ref [10]'s algorithm

(d) pre-process initial contour of our algorithm

(e) segmentation result of our algorithm

Figure 6. Comparison Chart of Medical Images Segmentation Results by using Three Algorithms

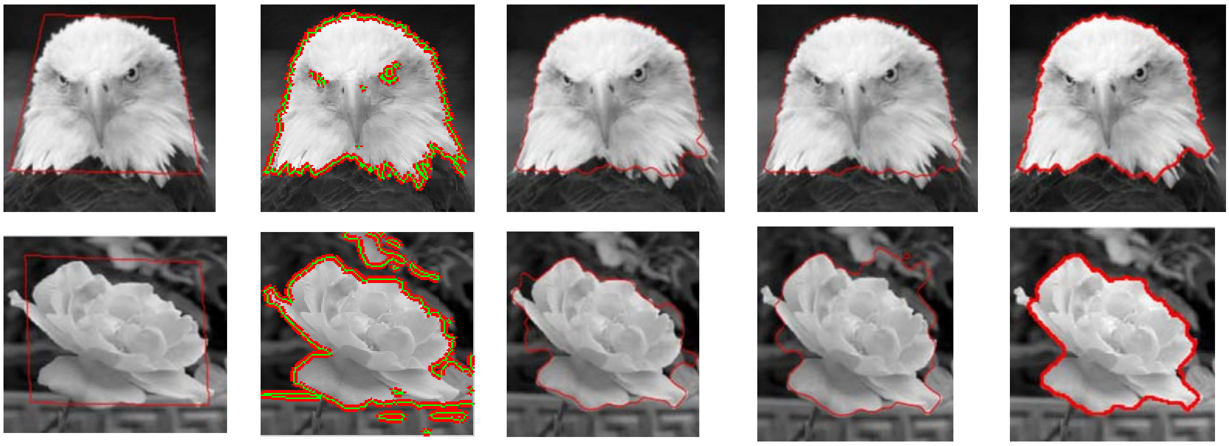
First Set an initial contour shown as Figure 6(a). Figure 6(b) showed the segmentation results of traditional C-V model algorithm, which iterated 300 times, respectively cost time 36.985 seconds and 34.372 seconds. From Figure 6(b) we can see, C-V model algorithm couldn't get good segmentation results for those images with unhomogeneous gray levels and there would cause many local minimums during the iteration process.

Figure 6(c) showed the segmentation results of the algorithm of Reference [10], which iterated the same times with Figure 6(b), respectively cost time 8.354 seconds and 10.385 seconds. Using this algorithm the iterative contours are almost closed to the real target edges, better avoid the local optimums.

Figure 6(d) showed the first-part results, ie. pre-process initial contours with our optimized algorithm. Set the same threhold a =0.025 , lung image iterated 106 times and liver image iterated 100 times. When iteration stoped, get these contours as new initial contours and called graphs cut algorithm for segmentation, get the results as shown in Figure 6(e).

The optimized algorithm total cost time 5.825 seconds and 6.978 seconds respectively. From Figure 6(e) we can see that the segmentation results are quite closed to the desired boundaries and the object regions have been recognized well.

Figure 7 showed two gray images segmentation results. One is owl image with the size of 168×167 pixels and the other is flower image with the size of 143×140 pixels. Take the same operations.



(a)Initial Contour

(b)segmentation result of the C-V model algorithm

(c) segmentation result of Ref [10]'s algorithm

(d) pre-process initial contour of our algorithm

(e) segmentation result of our algorithm

Figure 7. Comparison Chart of Gray Images Segmentation Results by using Three Algorithms

First Set an initial contour shown as figure 7(a). Figure 7(b) showed the segmentation results of traditional C-V model algorithm, which iterated 300 times , respectively cost time 22.558 seconds and 28.885 seconds. From the result we can see that the C-V model is good at segmenting gray images, but because its speed in the curve evolution is slow, it needs much more time.

Figure 7(c) showed the segmentation results of the algorithm of Ref.[10] ,which only cost time 6.882 seconds and 7.97 seconds.The algorithm saved time and the results are closed to the real target edges.

Figure 7(d) showed the first-part results, ie. pre-process initial contours with our optimized algorithm. Set the same threhold a = 0.02, owl image iterated 77 times and flower image iterated 50 times . When iteration stoped, get these contours as new initial contours and called graphs cut algorithm for segmentation, get the results as shown in figure 7(e).

The optimized algorithm total cost time 5.476 seconds and 5.867 seconds respectively. From Figure 7(e) we can see that the segmentation result are quite good.

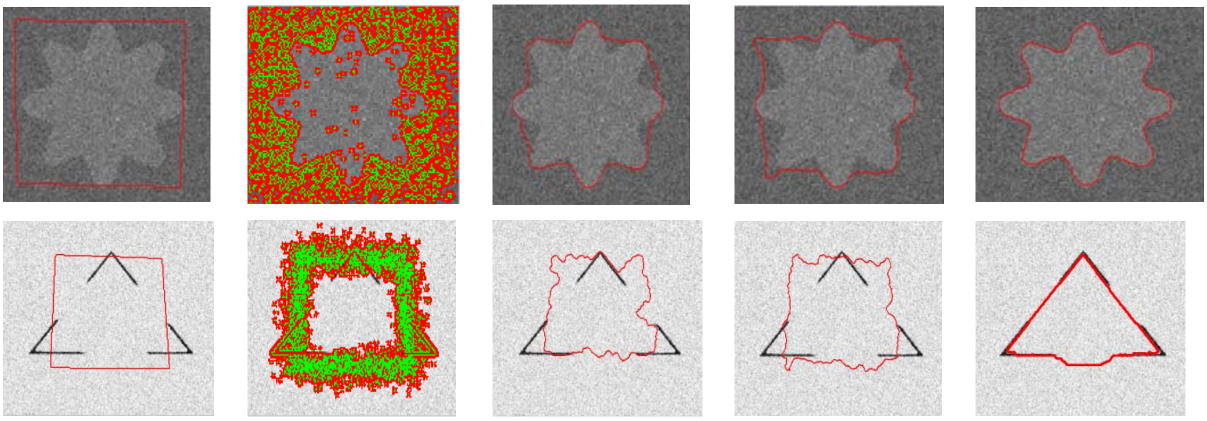
Figure 8 showed two noise images. One is star with the size of 283×387 pixels and the other is triangle with the size of 90×98 pixels. Take the same operations.

First Set an initial contour shown as figure 8(a). Figure 8(b) showed the segmentation results of traditional C-V model algorithm, which iterated 300 times, respectively cost time 53.909 seconds and 74.061seconds.. From the results we can see that the C-V model is weak to segment the noise images.

Figure 8(c) showed the segmentation results of the algorithm of Reference [10], which iterated 300 times, respectively cost 13.774 seconds and 15.487 seconds. The results showed the algorithm is robust but the segmenting contours are far from the desired boundaries.

Figure 8(d) showed the first-part results, ie. pre-process initial contours with our optimized algorithm. Set the same threhold α =0.4 star image iterated 85 times and flower image iterated 79 times. When iteration stoped, get these contours as new initial contours and called graphs cut algorithm for segmentation, get the results as shown in Figure 8(e).

The optimized algorithm total cost time 6.204 seconds and 8.372 seconds respectively. From Figure 8(e) we can see that the segmentation results are quite close to the object boundaries, which proved that the optimized algoritm is better denoising performance than the first two algorithms.



(a)Initial Contour

(b)segmentation result of the C-V model algorithm

(c) segmentation result of Ref [10]'s algorithm

(d) pre-process initial contour of our algorithm

(e) segmentation result of our algorithm

Figure 8. Comparison Chart of Two Noised Images Segmentation Results by using Three Algorithms

3.2. Results Anylasis

In order to evaluate the Experiments' results, this paper we choose FOM(Figure of Merit) method to anylasis the performance of the algorithms. The definition of the FOM [12] is:

$$FOM = \frac{1}{max\{N_i, N_t\}} \sum_{i=1}^{N_t} \frac{1}{1+\alpha d_i^2} \tag{9}$$

Where N_i denotes the qualities of the edge's pixels from the manual segmentations, which is thought as a standard result. N_t denotes the qualities of the edge's pixels from the experiment's segmentation. α is compensation coefficient (usually set 1/9), d_i is the nearest distance between the points on the testing edge and real edge. The value of FOM is between 0 and 1. The value is larger, the effects of the segmentation is better. Use FOM on our above images testing, get Table 1. From Table 1 we can see that most of the FOM values of our optimized algorithm are larger than FOM of C-V model and Reference [10]'s algorithm, which proves that our optimized algorithm have good performance.

Table 1. FOM Values Comparison with Three Algorithms

	FOM(lung)	FOM(liver)	FOM(owl)	FOM(flower)	FOM(star)	FOM(triangle)
C-V model algorithm	0.7203	0.6856	0.8801	0.8423	0.3501	0.3834
Ref [10]'s algorithm	0.7611	0.8345	0.7612	0.6921	0.5023	0.4256
optimized algorithm	0.8203	0.8342	0.8501	0.8706	0.8801	0.8234

4. Conclusion

Based on the improved C-V model and graph cut algorithm, this paper presented an optimized iteration algorithm for image segmentation. This algorithm comibines the advantages of these two algorithms and improves the iteration speeds and segmentation accuracy. The algorithm may solve some problems that the traditional C-V model algorithm need more computing time on numerical iterations and the problem that iteration termination condition is difficult to determine. Simulation results indicated that the proposed algorithm is robust and high efficiency.

Acknowledgments

This work is jointly supported by the National Nature Science Foundation of China (No.61074192), the Research Fund of the Jiangxi Province Education Office (Grant No.GJJ11465) and the Funds of the USTB (Nos.YJ2010-019, 06108104).

References

[1] M Kass, A Witkin, D Terzopoulous. Snakes: Active contour models. *Intemational Journal of Computer Vision.* 1988; 1(4): 321-331.

[2] V Caselles, R Kimmel, G Sapiro. Geodesic active contours. *International Journal of Computer Vision*. 1997; 22(1): 61–79.

[3] S Osher, R Fedkiw. Level Set Methods and Dynamic Implicit Surfaces. *Springer-Verlag*. New York, 2002.

[4] Malladi R, Sethian JA, Vemuri BC. Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1995; 17(2): 158-175.

[5] Chan TF, Vese LA. Active contours without edges. *IEEE Transactions on Image Processing*. 2001; 10(2): 266-277.

[6] Li C, Xu C, Gui C, et al. *Level set evolution without re-initialization: a new variational formulation*. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego: IEEE Press. 2005; I: 1-7.

[7] Y Boykov, G Funka Lea. Graph cuts and efficient N-D image segmentation. *International Journal of Computer Vision*. 2006; 70(2): 109-131.

[8] Boykov Y, Kolmogorov V. An Experimental Comparison of Min-Cut /Max-Flow Algorithms for Energy Minimization in Vision. *IEEE transactions on pattern analysis and machine intelligence*. 2004; 26(9).

[9] Ford LR, Fulkerson DR. Maximal flow through a network. *Canadian Journal of Mathematics*. 1956; 8(3): 399-404.

[10] Ning Xu, Ahuja N, Bansal R. Object segmentation using graph cuts based active contours. *Computer Vision and Image Understanding*. 2007; 107(3): 210-224.

[11] YANG Jiangong, WANG Xili. Image segmentation approach based on graph cuts and dual level set method. *Computer Engineering and Applications*. 2012; 48(3): 195-197.

[12] Lan Hong, Liu Xiantao. Energy minimization model for image segmentation via graph cut optimization. *Application Research of Computers*. 2012; 29(11): 4381-4384.

[13] Yuanhui Yu, Chinchen Chang. A new edge detection approach based on image context analysis. *Image and Vision Computing*. 2006; 24: 1090-1102.