Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 1420 – 1424

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

An Interactive Image Segmentation Algorithm Based on Graph Cut

Qiuhua Zheng[a]*, Wenqing Li[a], Weihua Hu[a], Guohua Wu[a]

[a]*Hangzhou Dianzi University, Hangzhou 310018, China*

Abstract

This paper introduces a new interactive image segmentation algorithm. This algorithm gets better result with fewer users interactive when segmenting single-object from images with complex foreground and background. Experiments show that this algorithm effectively solve the common over-segmentation and less-segmentation in graph cut, as well as resolve the problem of small regions error segmented in grab cut algorithm.

Keywords: graph cut; segmentation; max-flow/min-cut;

1. Introduction

Interactive foreground/background segmentation is a practical and important problem in image processing. A lot of interactive segmentation techniques have been proposed, such as snakes [1], livewire [2], level sets [3], watershed cuts [4], etc. The most popular method in recently years is graph cuts which is proposed in 2001 by Yuri Boykov and Marie Pierre Jolly [5, 6]. They globally optimized energy function with graph cut in gray scale image segmentation. Graph cuts addresses segmentation in a global optimization framework and guarantee a globally optimal solution for a wide class of energy functions.

A number of recent publications further extend the pioneer work of Boykov and Jolly and develop the use of regional cues or various object segmentation cues. Graph cut algorithms have been successfully applied to a wide range of problems in image processing. Besides used in image segmentation, graph cut

* Corresponding author. Tel.: +086-13588721967.
E-mail address: zheng_qiuhua@163.com.

doi:10.1016/j.proeng.2012.01.149

is also used in image restoration [7], stereo and motion [8], image synthesis, region merging, multi-camera scene reconstruction, 2D optical flow, interactive object delineation [9,10], etc.

Although the user input is valuable in steering the segmentation process to reduce the ambiguities, too much interaction would lead to a tedious and time-consuming work. In the segmentation algorithms based on graph cut, users should often maker many foreground and background points [1], which is usually annoying. In this paper, we proposed a novel approach by using two polygons marker. In our algorithm, FG (foreground) is constituted by points in the inner one, and BG (background) is constituted by points between the outer one and the image border. The part between two polygons is unknown region. The essence of segmenting single object from image is segmenting the unknown region. Compared with the graph cut algorithm, our method needs fewer users interactive. In addition, since our algorithm limits those points need to be labeled at the area between two polygons, it improve the accuracy of segmentation.

The rest of this paper is organized as the following. In section 2, we introduce "max flow/min cut" algorithm and the graph cut algorithm in briefly. Section 3 shows the implementation of our method. Section 4 presents experimental results of our method. Finally the conclusion is made in Section 5.

2. Image Segmentation by Graph Cut

2.1. Graph cut

Boykov and Kolmogorov regard source *S* as foreground and sink *T* as background in graph cut [5, 6]. Fig. 1 is a network flow graph. The nodes between source *S* and sink *T* are replaced by pixels which in the gray scale image (Fig. 1(a)). We calculate the min cut with "max flow/min cut" method, as Fig. 1(d) shows. The min cut is the boundary of the foreground and background, so we can get the segmentation as show in Fig. 1(b). The idea of graph cut is using image to construct network flow graph and set the flow between every two adjacent nodes, then set proper energy function. "Max flow/min cut" will make sure the min cut become the boundary of the foreground and background and minimize the energy function.

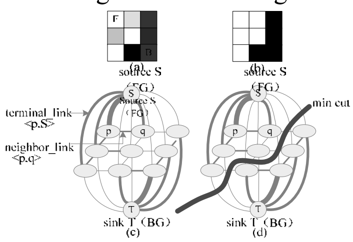

Fig.1. Graph cut based on network graph

Using *G={V, E}* denotes network flow graph, *V* contains both the image pixels set *P* and two terminal nodes (source *S* and sink *T*), and *E* includes both t-link (terminal links) which link each node to the two terminals and *n*-link (neighborhood links) which link each two adjacent non-terminal nodes. For each pixel *p* will link two terminals (*S->p* and *p->T*), for each two adjacent nodes p and q they will link each other *p->q*, *q->p*. so *E={S-p, p->T, p->q, q->p | p, q ∈P}*. Min cut *C* is the subset of *G*. When the network flow graph reach the max flow, *C* separate *V* into foreground and background two parts.

2.2. Energy function

Many algorithms based on graph cuts have been developed to solve energy minimization problems. Each of these techniques constructs a graph such that the minimum cut on the graph also minimizes the energy. The graph construction is complex and highly specific to a particular energy function. Vladimir

Kolmogorov and Ramin Zabih [10] give an energy function, which is defined as following.

$$E(A) = \lambda \bullet R(A) + B(A) \tag{1}$$

$$R(A) = p \in P \tag{2}$$

$$B(A) = \{p, q\} \in N \tag{3}$$

The energy function contains the region properties *R(A)* and the boundary properties *B(A)*. The coefficient $\lambda \geq 0$ specifies a relative importance of *R(A)* versus *B(A)*. *P* denotes the non-terminal nodes in the network flow graph. $|P|$ is elements number in set *P*. *R(A)* and *B(A)* are relate to t-link and n-link respectively, *R(p)* reflects on how the intensity of pixel *p* fits into a known intensity model of the foreground and background. If node *p* is similar to node *q, B{p, q}* will be zero otherwise be large.

3. Our proposed algorithm

The framework of our algorithm is demonstrated by Fig.3. First, marks the foreground and background with two polygons. Then perform pre-segmentation on the original image. After pre-segmentation, the network flow graph is constructed and standard graph cut is performed.

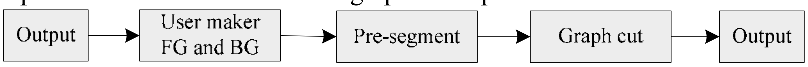

Fig.2. Framework of the image segmentation algorithm

3.1. User Marker and image pre-segment

As show in the left of Fig. 3, we use two polygons to maker "foreground boundary" and "background boundary". One polygon that maker "foreground boundary" along the inner side of the boundary of the object with red points, the part that foreground boundary contains named *FG*. Similarly, maker "background boundary" along the outer side of the boundary of the object with blue points, the part between the background boundary and the boundary of the image named *BG*. Red translucent zone and blue zone in the right of Fig.3 is *FG* and *BG*; both of them are the input parameters in the graph cut.

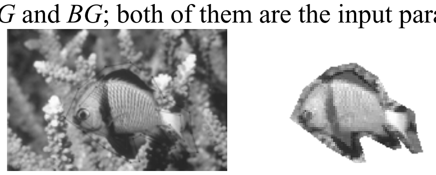

Fig.3. Images marked the FG and BG along the foreground boundary

For speeding up the process of segmentation, we use watershed algorithm to pre-segment the original image after marker step. Then we merge the small regions, and construct network flow graph by the regions segmented by the previous step. Small regions transform into nodes in network flow graph. Each node *p*={R, G, B} is one 3 dimension vector. The node's RGB value is the mean value of all the pixels in corresponding small region. Because foreground and background is made of not only one color, so *FG* and *BG* can be divided into *K* class, $FG=\{FGC_1, FGC_2,\ldots, FGC_K\}$ and $BG=\{BGC_1, BGC_2,\ldots, BGC_K\}$. Each class is a three dimension vector {R, G, B} where each element is the mean value of all the nodes in this class. Through pre-segment we can separate foreground from background better and keep foreground or background to be the whole. Experiments show that the segmentation results are best when *K* is 5 to 8.

3.2. Constructing network flow graph

- T-link

The nodes in the network flow graph are made from the small regions in the image after pre-segmentation. The flow between node p and terminals is defined as follows:

$$R_p(foreground) = min(||p\text{-}FGC_k||^2) / [(min(||p\text{-} FGC_k ||^2)+ min(||p\text{-} BGC_k ||^2)],\ FGC_k \in FG \tag{4}$$

$$R_p(background) = min(||p\text{-}BGC_k||^2) / [(min(||p\text{-} FGC_k ||^2)+ min(||p\text{-} BGC_k ||^2)],\ BGC_k \in BG \tag{5}$$

where p，FGC_k （foreground class k）and BGC_k（background class k） are three dimension vectors {R, G, B}. ||*|| denotes euclidean distance. $min(\|p\text{-}FGC_k\|^2)$ denotes the minimum of square the euclidean distance between vector p and every element in the FG; $min(\|p\text{-}BGC_k\|^2)$ denotes the minimum of square the euclidean distance between vector p and each element in the BG. If one image has a lot of colors in foreground and background, dividing the FG and BG to more classes will get better segmentation result.

- N-link

The flow is between every two adjacent node p and node q.

$$B\{p,q\} =1/(1+||p\text{-}q||^2) \tag{6}$$

where ||*|| denotes euclidean distance. Node p and q are three dimension vectors {R, G, B}.

- Energy function

In our algorithm, we use the energy function defined by Vladimir Kolmogorov and Ramin Zabih.

4. Experiments

In Our experiments, λ is set as 1.1. FG and BG are divided into 5 classes by kmeans algorithm. Experiments' results are compared with Grabcut [9].

Fig.4(a) shows the fish in the ocean on the ground. This figure's foreground is very similar to background. Fig.4(b) shows the result segmented by our proposed algorithm while Fig.4(c) shows the segmentation result calculated by Grabcut. In Fig.4(b), the foreground in the segmentation result as a whole appear is without error discrete small regions.

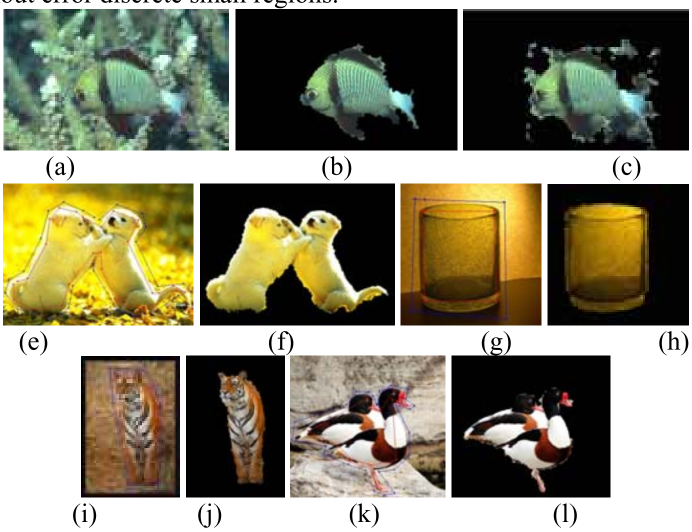

Fig.4. Results of segmentation

More experimental results are shown in Fig.4(f)-(l). Better segmentation result is reached when segmenting single-object from complex foreground and background with less users' markers. Because the red translucent zone and the blue zone are belong to foreground and background, so our algorithm can solve the over-segmentation and less-segmentation and get segmentation result more effectively.

5. Conclusion

This paper proposed a novel interactive image segmentation algorithm with fewer users interactive. It first markers foreground and background with two polygons. Then it performs a pre-segmentation processing by watershed algorithm. At last, the segmentation task is transform into Max flow/min cut problem and be solved by graph cut algorithm. Experiments show that our proposed algorithm can effectively solve the common over-segmentation and less-segmentation in graph cut, as well as resolve the problem of small regions error segmented in grab cut algorithm.

Acknowledgements

This research is supported by Chinese National Natural Science Foundation 61100100 and Natural Science Foundation of Zhejiang Province Y1110232.

References:

[1] Kass,M., Witkin,A.,Terzopoulos,D. Snakes: Active contour models. *International Journal of Computer Vision*, vol.2, pp: 321-331, 1988.

[2] Falaco,A. X., Udupa,J.K.,Samarasekara,S.,Sharma,S. User-steered image segmen-tation paradigms: Live wire and live lane. *Graphical Models and Image Processing*, vol.60, pp: 233-260, 1998.

[3] Osher,S., Sethian,J.A. Fronts propagating with curvature dependent speed: Algorithm based on hamilton jacobi formulations. *Journal of Computational Physics*, Vol.79, pp: 12-49, 1988.

[4] Grady,L. Random Walks for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.28, no.11, pp: 1768-1783, 2006.

[5] Boykov, Y. and M. Jolly, Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. *Internation Conference on Computer Vision*, 2001.

[6] Boykov, Y. and M. Jolly, Interactive Organ Segmentation Using Graph Cuts. *Proc.Medical Image Computing and Computer Assisted Intervention*, 2000: p. 276-286.

[7] Boykov, Y., O. Veksler and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts. *Proc.IEEE Trans.Pattern Analysis and Machine Intelligence*, 2001. 23: p. 1222-1239.

[8] Birchfield, S. and C. Tomasi, Multiway Cut for Stereo and Motion with Slanted Surfaces. *IEEE International Conference on Computer Vision*, 1999: p. 489-495.

[9] Rother, C., V. Kolmogorov and A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. in SIGGRAPH '04. 2004. New York, NY, USA: ACM.

[10] Kolmogorov, V. and R. Zabih, What Energy Functions Can Be Minimized via Graph Cuts. IEEE transactins on Pattern Analysis and Machine Intelligence, 2004. 26.