Iterated Graph Cuts for Image Segmentation

Bo Peng[1], Lei Zhang[1] *, and Jian Yang[2]

[1] Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong, China.
[2] School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing , 210094, China
{csbpeng,cslzhang}@comp.polyu.edu.hk
{csjyang}@mail.njust.edu.cn

Abstract. Graph cuts based interactive segmentation has become very popular over the last decade. In standard graph cuts, the extraction of foreground object in a complex background often leads to many segmentation errors and the parameter λ in the energy function is hard to select. In this paper, we propose an iterated graph cuts algorithm, which starts from the sub-graph that comprises the user labeled foreground/background regions and works iteratively to label the surrounding un-segmented regions. In each iteration, only the local neighboring regions to the labeled regions are involved in the optimization so that much interference from the far unknown regions can be significantly reduced. To improve the segmentation efficiency and robustness, we use the mean shift method to partition the image into homogenous regions, and then implement the proposed iterated graph cuts algorithm by taking each region, instead of each pixel, as the graph node for segmentation. Extensive experiments on benchmark datasets demonstrated that our method gives much better segmentation results than the standard graph cuts and the GrabCut methods in both qualitative and quantitative evaluation. Another important advantage is that it is insensitive to the parameter λ in optimization.

Key words: Image segmentation,graph cuts, regions merging

1 Introduction

Interactive foreground/background segmentation is a practical and important problem in computer vision. Over the last decade, a number of interactive segmentation techniques have been proposed, such as snakes [1], livewire [2], level sets [3],watershed cuts [4] and random walkers [5]. Another preferable method which becomes very popular in recently years is graph cuts [6, 7]. Graph cuts addresses segmentation in a global optimization framework and guarantees a globally optimal solution for a wide class of energy functions.

* Corresponding author

A number of recent publications further extend the pioneer work of Boykov and Jolly [6] and develop the use of regional cues [8, 13] or various object segmentation cues [14, 15]. Lombaert et al. [9] studied the use of graph cuts for high-resolution data. They proposed a multilevel banded heuristic for the computation of graph cuts. The use of a smaller graph in all resolutions reduces the running time and memory consumption compared with the original graph cuts algorithm. Because the graph cuts technique can involve a wide range of visual cues, some researchers used the shape prior as an effective cue in the graph cuts framework. Freedman and Zhang [10] defined the shape prior as a single fixed template, which was specified as a distance function inspired by the idea of level sets. Das and Veksler [11] developed a graph cuts based segmentation algorithm by assuming the object is of compact shape. Further more, Veksler [12] exploited the star shape prior, which is a kind of generic shape prior, into graph cuts segmentation.

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. Usually, the extraction of foreground objects in a complex environment, from which the background can not be trivially subtracted, often requires a lot of user interaction. Moreover, the complex content of an image also makes it hard to give user guide for accurate segmentation while keeping the interaction as less as possible. Thus some algorithms allow the further user edit based on the previous segmentation result [8, 22], yet this requires additional user interaction.

In this paper, we explore the graph cuts algorithm by extending it to a region merging scheme. Specifically, we perform mean shift [16] algorithm on the original image for an initial segmentation, which partitions the image into many homogenous regions. Starting from seeds regions given by the user, we run graph cuts on a propagated sub-graph where the segmented regions by mean shift algorithm, instead of the pixels in the original image, are regarded as the nodes of the graph. An iterated conditional mode (ICM) on graph cuts is studied and, whereas it does not provide a global solution in the whole graph, global optima can be obtained on the growing subgraphs.

Our method is a novel extension of the standard graph cuts algorithm. It has many advantages and merits. First, using sub-graph can reduce significantly the complexity of background content in the image. The many unlabeled background regions in the image may have unpredictable negative effect on graph cuts optimization. This is why the global optimum obtained by graph cuts often does not lead to the most desirable result. However, by using a sub-graph and blocking those unknown regions far from the labeled regions, the background interference can be much reduced, and hence better results can be obtained under the same amount of user interaction. Second, the algorithm is run on the sub-graph that comprises foreground/background regions and their surrounding un-segmented regions, thus the computational cost is significantly less than running graph cuts on the whole graph which is based on image pixels. Third, as a graph cuts based region merging algorithm, our method obtains the optimal segmentation on each sub-graph in the iteration. Forth, the object and background color models are updated after the segmentation on each sub-graph. Thus they can provide more informative guide for the next round of segmentation.

The paper is organized as follows. A brief review of standard graph cuts algorithm is in Section 2. An iterated conditional mode on graph cuts is proposed in Section 3, followed by the iterated graph cuts algorithm. Section 4 presents experimental results of our method on 50 benchmark images in comparison with standard graph cuts and Grabcut. Finally the conclusion is made in Section 5.

2 Image Segmentation by Graph Cuts

Segmentation of an object from the background can be formulated as a binary labeling problem. Given a set of labels L and a set of sites S, the labeling problem is to assign a label $f_p \in L$ to each of the sites $p \in S$. The graph cuts framework proposed by Boykov and Jolly [6] addresses the segmentation of a monochrome image, which solves a labeling problem with two labels. The label set is $L = \{0, 1\}$, where 0 corresponds to the background and 1 corresponds to the object.

Let $f = \{f_p | f_p \in L\}$ stand for a labeling, i.e. label assignments to all pixels. An energy function is formulated as:

$$E(f) = \sum_{p \in S} D_p(f_p) + \lambda \sum_{\{p,q\} \in \mathcal{N}} \omega_{pq} \cdot T(f_p \neq f_q) \tag{1}$$

On the right hand side of (1), the first term is called data term, which consists of constraints from the observed data and measures how sites like the labels that f assigns to them. where D_p measures how well label f_p fits site p. A common approach, and the one we use in our work, is to build the foreground and background histograms models from the user input seeds, respectively. Then the $D_p(f_p)$ are defined as the negative log likelihoods of the constructed foreground/background models.

The second term is called the smoothness term and measures the extent to which f is not piecewise smooth. where $\mathcal{N}$ is a neighborhood system, such as a 4-connected neighborhood system or an 8-connected neighborhood system. The smoothness term typically used for image segmentation is the Potts Model [20]. Here $T(f_p \neq f_q)$ is 0 if $f_p = f_q$ and 1 otherwise. This model is a piecewise constant model because it encourages labelings consisting of several regions where sites in the same region have the same labels.

In image segmentation, we want the boundary to lie on the edges in the image. A typical choice for $\omega_{p,q}$ is : $\omega_{pq} = e^{-\frac{(I_p - I_q)^2}{2\delta^2}} \cdot \frac{1}{dist(p,q)}$, where I_p and I_q are the color values of sites p and q, and $dist(p, q)$ is the distance between sites p and q. Parameter δ is related to the level of variation between neighboring sites within the same object. The parameter λ is used to control the relative importance of the data term versus the smoothness term. Minimization of the energy function can be done using the min-cut/max-flow algorithm as described in [6].

3 The Iterated Graph Cuts

3.1 Iterated Conditional Mode

Graph cuts technique provides a globally optimal solution to image segmentation; however the complex content of an image makes it hard to precisely segment the whole image all at once. The iterated conditional mode (ICM) proposed by Besag [21] is a deterministic algorithm which maximizes local conditional probabilities sequentially. It uses the "greedy" strategy in the iterative local maximization to approximate the maximal joint probability of a Markov Random Field (MRF). Inspired by ICM, we consider the graph cuts algorithm in a "divide and conquer" style: finding the minima on the sub-graph and extending the sub-graph successively until reach the whole graph. The proposed method works iteratively, in place of the previous one-shot graph cuts algorithm [6].

Let d_i be the observed data of site i, f_i be the label of site i and $f_{S-\{i\}}$ be the set of labels which is at the sites in $S-\{i\}$, where $S-\{i\}$ is the set difference. We sequentially assign each f_i by maximizing conditional probability $P(f_i|d_i, f_{S-\{i\}})$ under the MAP-MRF framework. Here we have two assumptions in calculating $P(f_i|d_i, f_{S-\{i\}})$. First, the observed data $d_1, \ldots, d_m$ are conditionally independent given f and that each d_i depends only on f_i. Second, f depends on labels in the local neighborhood, which is Markovianity, i.e. $P(f_i|f_{S-\{i\}}) = P(f_i|f_{N_i})$, where N_i is a neighborhood system of site i. Markovianity depicts the local characteristics of labeling.

With the two assumptions we have:

$$P(f_i|d_i, f_{S-\{i\}}) = \frac{P(d_i|f_i) \cdot P(f_i|f_{N_i})}{P(d)} \tag{2}$$

where $P(d)$ is a normalizing constant when d is given. There is:

$$P(f_i|d_i, f_{S-\{i\}}) \propto P(d_i|f_i) \cdot P(f_i|f_{N_i}) \tag{3}$$

where $\propto$ denotes the relation of direct proportion. The posterior probability satisfies:

$$P(f_i|d_i, f_{S-\{i\}}) \propto e^{-U(f_i|d_i, f_{N_i})} \tag{4}$$

where $U(f_i|d_i, f_{N_i})$ is the posterior energy and satisfies:

$$\begin{aligned} U(f_i|d_i, f_{N_i}) &= U(d_i|f_i) + U(f_i|f_{N_i}) \\ &= U(d_i|f_i) + \sum_{i' \in N_i} U(f_i|f_{i'}) \end{aligned} \tag{5}$$

$U(d_i|f_i)$ is the data term corresponding to function (1), and $\sum_{i' \in N_i} U(f_i|f_{i'})$ is the smoothness term which relates to the number of neighboring sites whose labels $f_{i'}$ differ from f_i. The MAP estimate is equivalently found by minimizing the posterior energy:

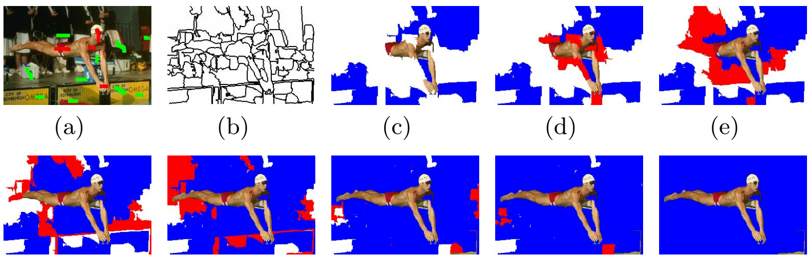
$$f^{k+1} = \arg\min_f U(f|d, f_N^k) \tag{6}$$

where f_N^k is the optimal labeling of graph nodes obtained in previous k iterations. The labeling result in each iteration is reserved for later segmentation. This process is done until the whole image is labeled.

3.2 The Iterated Graph Cuts Algorithm

In the original graph cuts algorithm, the segmentation is directly performed on the image pixels. There are two problems for such a processing. First, each pixel will be a node in the graph so that the computational cost will be very high; second, the segmentation result may not be smooth, especially along the edges. These problems can be solved by introducing some low level image segmentation techniques, such as watershed [17] and mean shift [16], to graph cuts. In [22], Li et al. used watershed for initial segmentation to speed up the graph cuts optimization process in video segmentation. In this paper, we choose to use mean shift for initial segmentation because it produces less over-segmentation and has better edge preservation than watershed. Fig.1(b) shows the mean shift initial segmentation of the image in Fig.1(a).

The initial labeling f^0 of graph cuts is given by a group of foreground/background seeds from the user. Regions which have pixels marked as foreground are called foreground seed regions, while the regions with background seeds are thus called background seed regions. The initial sub-graph contains only seed regions. Start from the initial sub-graph, in the iteration only the adjacent regions to the previously labeled regions are added into the updated sub-graph. Running graph cuts algorithm on the updated sub-graph, an updated optimal segmentation is obtained. The iteration stops when all the region nodes are labeled as either foreground (i.e. object) or background.



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Fig. 1. The iterated segmentation process. (a) Original image with user input seeds. (b) Initial mean shift segmentation. (c) The user input seed regions. The background is shown in blue color. (d)-(i) show the intermediate segmentation results in the iteration. The newly added regions in the sub-graphs are shown in red color and the background is shown in blue color. In (j), the target objects are well segmented from the background within 6 iterations.

Fig.1 illustrates the iterated segmentation process. In the first iteration, regions chosen to be labeled are those which are only adjacent to the foreground regions, as shown in Fig.1(d). In the following iterations (Figs.2c-2h), new regions which are only in the neighborhood of previous foreground regions are added into the sub-graph for further labeling. In practice we have found that adding regions which are adjacent to either the foreground or the background or both of them does not make much difference for the segmentation results. The desired objects are extracted as shown in Fig.1(j). The iterated graph cuts algorithm is summarized in Algorithm 1. We assume that the foreground regions are connected unless separated parts of the foreground are initially marked by the user. Therefore, the regions which can not be involved in the iterations will be labeled as the background regions.

Algorithm 1 Iterated GraphCuts.

The input are mean shift initial segmentation of the given image and a graph G whose nodes consist of the user input foreground/background seed regions R. The output is the segmentation result.

1. Add adjacent regions of foreground regions into G.
2. Construct foreground and background data models from seed regions R.
3. Use graph cuts algorithm to solve $\arg\min_f U(f|d, f_N^k)$.
4. Add foreground and background regions resulting from step 3 into R.
5. Add adjacent regions of the foreground seeds into G.
6. Go back to step 2, until no adjacent regions can be found.
7. Set labels of the remaining regions to be the background.
8. Return the segmentation result.

4 Experimental Results

In this section, we validate the segmentation performance of our method in comparison with the standard graph cuts algorithm [6] and GrabCut [8]. Since the proposed iterated graph cuts algorithm uses mean shift for initial segmentation, for a fair comparison we also extended the standard graph cut to a region based scheme, i.e. use the mean shift segmented small regions, instead of the pixels, as the nodes in the graph. Usually this yields better results than the original graph cuts. The GrabCut algorithm is an interactive segmentation technique based on graph cuts and has the advantage of reducing user's interaction under complex background. It allows the user to drag a rectangle around the desired object. Then the color models of the background and foreground are constructed according to this rectangle. Similarly to our method, an iterative estimation scheme of color models is used in GrabCut to segment the object.

We use the mean shift segmentation software- the EDISON System[3] -to obtain the initial segmentation. Experiments are performed on a database which contains 50 benchmark test images selected from online resources [4] [5], where 10 of them contain objects with simple background and 40 are natural images with relatively complex background. Every image in our database has a figure-ground assignment labeled by human subjects.

[3] http://www.caip.rutgers.edu/ riul/research/code/EDISON/doc/segm.html
[4] http://www.research.microsoft.com/vision/cambridge/segmentation/
[5] http://www.cs.berkeley.edu/projects/vision/grouping/segbench/

4.1 Comparison with Standard Graph Cuts

We first compare the proposed iterated graph cuts with the standard graph cuts. In this subsection we use several example images to evaluate them qualitatively. The quantitative evaluation will be given in subsection 4.3. Fig.2 includes some images with simple background (Fig.2(a)-2(b)) and some with complex background (Fig.2(c)-2(e)). In the later ones, camouflage makes the objects containing weak boundaries due to poor contrast and noise, and the colors of some background regions are very close to those of the objects. Given the same amount of user input, the iterated graph cuts method achieves much better segmentation than standard graph cuts.

(a) (b) (c) (d) (e)

Fig. 2. Segmentation results of images with simple or complex background. The first row shows the original images with seeds. The second row shows the segmentation obtained by standard graph cuts. The third row shows the segmentation of iterated graph cuts.

4.2 Comparison with GrabCut

The ways of user input are different for GrabCut and Graph cuts. Graph cuts requires user to indicate some background and foreground regions, while GrabCut only needs the user to drag a rectangle around the object. In experiments, we choose the user inputs that lead to the best results for GrabCut.

An comparison with GrabCut is shown in Fig. 3. The first row shows the original images with the user inputs. The red and green seeds are for the proposed iterated graph cuts, while the blue rectangles are for the GrabCut. The second row shows the segmentation results of GrabCut. Implementation of GrabCut uses 5 GMMs to the model RGB color data and parameter λ is fixed to be 50. The third rows are results of iterated graph cuts. When the objects to be segmented contain similar colors with the background, GrabCut might fail to correctly segment them. Although overall graph cuts may use more user interaction than GrabCut, it can produce more precise segmentation results.

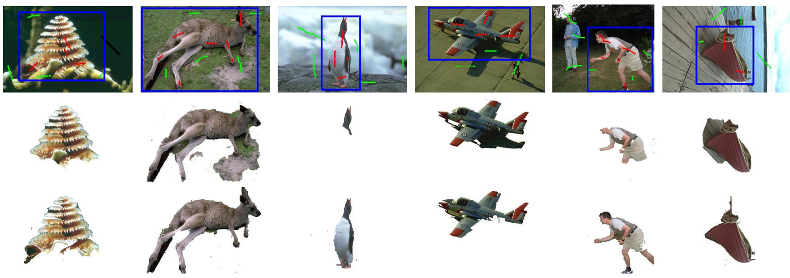


Fig. 3. Segmentation results of GrabCut and proposed iterated graph cuts. The first row shows the original images with seeds. Red and green strokes represent the object and background seeds for graph cuts. User inputs for GrabCut are denoted by blue rectangles. The second row shows the results of GrabCut . The third row shows the results of iterated graph cuts. The proposed method can segment more accurately the desired objects than GrabCut.

4.3 Quantitative Evaluation

Quantitative evaluation of the segmentations is given by comparing with ground truth labelings. The qualities of segmentation are calculated by using four measures: the true-positive fraction (TPF), false-positive fraction (FPF), true-negative fraction (TNF) and false-negative fraction (FNF), which are defined as follows:

$$TPF = \frac{|A_A \cap A_G|}{|A_G|}, FPF = \frac{|A_A - A_G|}{|\overline{A_G}|}, TNF = \frac{|\overline{A_A \cup A_G}|}{|\overline{A_G}|}, FNF = \frac{|A_G - A_A|}{|\overline{A_G}|}$$

where A_G represents the area of the ground truth of foreground and its complement is $\overline{A_G}$; A_A represents the area of segmented foreground by the tested segmentation method.

Table 1 lists the results of TPF, FPF, FNF and TNF by the three methods over the 50 test images. We see the iterated graph cuts method achieves the best FPF, TNF and FNF results. The GrabCut method has higher TPF index than iterated graph cuts because it usually leads to a bigger segmentation area, which includes both foreground and background. Thus it also has much higher FPF rate.

Algorithms	TPF(%)	FPF(%)	TNF(%)	FNF(%)
GrabCut	93.88	16.35	96.59	16.35
Graph cuts	84.23	4.65	95.87	9.26
Ours	90.90	2.97	97.59	6.42

Table 1. The TNF, TPF, FNF and FPF results by different methods.

4.4 Discussion

In graph cuts based segmentation, parameter λ has great effect on segmentation results. It is used to tune the balance between different terms in the energy function. When given different images, a fixed value of λ can not give satisfactory segmentation. Since the appropriate λ values would vary largely among different images, the user may have to spend a significant amount of time searching for it. In the recent works [18, 19], much effort has been made to study the selection of λ. From our experiments, parameter λ was easier to set up for our method and thus brings much benefit for users in real applications.

5 Conclusion and Future Work

An iterated graph cuts algorithm was developed in this paper. It performs segmentation on the sub-graph which is updated in each iteration. The proposed iterated graph cuts can reduce the interference of unknown background regions far from the labeled regions so that more robust segmentation can be obtained. Qualitative and quantitative comparisons with standard graph cuts and GrabCut show the efficiency of the proposed method. With the same amount of user input, the proposed method can achieve better segmentation results than the standard graph cuts, especially when extracting the foreground from complex background. Moreover, the search space of parameter λ can also be reduced by our method.

Standard graph cuts can be viewed as a special case of the proposed iterated graph cuts when there is only one iteration in segmentation and all regions are involved in the optimization. Future work will be focused on how to reduce its dependency on the initial segmentation result and how to reduce the user interaction while preserving the segmentation accuracy.

Aknowledgement

This work is supported by the Hong Kong RGC General Research Fund (PolyU 5351/08E) and partially supported by the Program for New Century Excellent Talents in University of China, the NUST Outstanding Scholar Supporting Program, and the National Science Foundation of China under Grants No. 60632050.

References

1. Kass,M., Witkin,A.,Terzopoulos,D.: Snakes: Active contour models. International Journal of Computer Vision, vol.2,321–331 (1988)
2. Falaco,A. X., Udupa,J.K.,Samarasekara,S.,Sharma,S.: User-steered image segmentation 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. Cousty,J.,Bertrand,G.,Najman,L.,Couprie,M.: Watershed Cuts: Minimum Spanning Forests and the Drop of Water Principle. IEEE Transactions on Pattern Analysis and Machine Intelligence, to appear
5. Grady,L.: Random Walks for Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.28, no.11, pp.1768–1783, Nov. (2006)
6. Boykov,Y., Jolly,M. P.: Interactive graph cuts for optimal boundary and region segmentation. International Conference on Computer Vision, vol. I, pp 105–112 (2001)
7. Boykov,Y., Funka Lea,G.: Graph cuts and efficient n-d image segmentation. International Journal of Computer Vision. 69(2),109–131, (2006)
8. Rother,C., Kolmogorov,V.,Blake,A.: Grabcut-interactive foreground extraction using iterated graph cuts. ACM Transactions on Graphics. vol. 23, No.3, pp. 309-314 (2004)
9. Lombaert,H.,Sun,Y.,Grady,L.,Xu,C: A multilevel banded graph cuts method for fast image segmentation. International Conference on Computer Vision,vol.1, pp.259–265 (2005)
10. Freedman,D., Zhang, T.: Interactive graph cut based segmentation with shape prior. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 755-762 (2005)
11. Das,P., Veksler,O.,Zavadsky, V., Boykov,Y.: Semiautomatic Segmentation with Compact Shape Prior. Image and Vision Computing. vol. 27, issues 1-2, 1 January,pp.206-219 (2009)
12. Veksler,O.: Star Shape Prior for Graph-Cut Image Segmentation. European Conference on Computer Vision. pp,454–467 (2008)
13. Kolmogorov,V., Criminisi,A., Blake,A.,Cross,G., Rother,C.: Bi-layer segmentation of binocular stereo video. IEEE Conference of Computer Vision and Pattern Recognition. pp. 407-414 (2005)
14. Boykov,Y.,Kolmogorov,V.: Computing geodesics and minimal surfaces via graph cuts. International Conference on Computer Vision. vol. I, pp. 26–33 (2003)
15. Kolmogorov,V., Boykov,Y.: What metrics can be approximated by geo-cuts, or global optimization of length/area and flux. International Conference on Computer Vision. vol. I, pp. 564–571 (2005)
16. Comaniciu,D., Meer,P.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence. 24,603-619 (2002)
17. Vincent,L., Soille,P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13,6, 583-598 (1991)
18. V. Kolmogorov, Y. Boykov, C. Rother. Applications of Parametric Maxflow in Computer Vision. International Conference on Computer Vision, pp. 1-8 (2007)
19. Peng,B., Veksler,O.: Parameter Selection for Graph Cut Based Image Segmentation. British Machine Vision conference (2008)
20. Boykov,Y., Veksler,O.,Zabih,R.: Markov random fields with effcient approximations. IEEE Conference on Computer Vision and Pattern Recognition, pp.648-655 (1998)
21. Besag,J.: On the statistical analysis of dirty pictures. Journal of the Royal Statistical Society, series B 48,259-302 (1986)
22. Li,Y.,Sun,J.,Shum,H.: Video Object Cut and Paste. ACM Transactions on Graphics vol.24,issue 3, pp. 595-600, (2005)