Interactive Image Segmentation

Fahim Mannan (260 266 294)

***Abstract*— This report presents the project work done based on Boykov and Jolly's interactive graph cuts based N-D image segmentation algorithm([1]). It is a fast and generic technique for performing binary segmentation. The approach first models the problem using Markov Random Field where an energy function representing the optimal segmentation is formulated. The different terms of the function is determined based on the user input (likelihood model) and a prior model. In this approach, the user marks certain pixels as either "background" or "object". The likelihood is then calculated using this data. Finally, the energy function is minimized using Graph Cuts based technique. The method was implemented and tested using publicly available benchmark data with ground truth ([11,12]). The results and their analysis are also provided in this report.**

I. Introduction

The problem of image segmentation has received a lot of attention since the early days of computer vision research. Automatic image segmentation is a hard problem which requires modeling the problem based on domain knowledge. And even after that, some form of human intervention is required to correct anomalies in the segmentation. Moreover, automatic segmentation methods are not generic. A slightly easier and more approachable problem – interactive image segmentation – has also received a lot of attention over the years. This report describes the work done on implementing one such interactive segmentation algorithm which is based on [1]. The method proposed there is a very general technique that works for N-D images. Using this approach, the user marks certain pixels as either "object" or "background". The segmentation problem is then modeled using Markov Random Field where an energy function encodes a prior model and the constraints which are imposed by the marked pixels. Finally, graph cut based optimization is used to find a global optimal solution. In this report, results are presented only for the case of 2D segmentation problem. Comparisons were done using benchmark dataset with ground truth data. The results, their analysis, and time taken to perform segmentation are also provided.

The report is organized as follows – in section II some of the background and related works are presented. The segmentation method is discussed in detail in section III. Implementation and results are presented in section IV. Analysis of these results and different issues are discussed in section V. And finally the project work is summarized in section VI.

II. Background and Related Works

Image segmentation methods can be broadly categorized into variational and combinatorial methods ([2]). These two categories can be further subdivided based on how the boundary is represented. Variational methods like snakes, active contours, etc. and combinatorial approaches like "path-based" graph methods uses explicit boundary representation. On the other hand, level-set method (variational method) and Interactive Graph Cuts([1]) uses implicit boundary representation. In this section, we mainly focus on methods that model the segmentation problem using Markov Random Field and perform optimization using either some stochastic approach or graph cut.

Markov Random Field (MRF) allows modeling low-level vision problems like image segmentation using a probabilistic framework. The advantage in this case is that a prior model can be used to improve the solution to the problem. In MRF formulation each pixel is considered as a site and they are considered to be nodes in an N-D lattice structure. The interaction between different sites is modeled by connecting the sites with an edge and assigning an edge weight. Each of these sites is assigned a label and a particular configuration of these labels corresponds to an energy state of the system. The objective of MRF formulation is to find the configuration that either minimizes or maximizes the energy of the system. In probabilistic formulation of computer vision problem we want to maximize the posterior (MAP) probability. This can be done using MRF by formulating the MAP estimate as an energy optimization problem. From Bayes rule, we know –

$$Posterior \propto Likelihood \times \Pr ior$$

Based on this, the energy for a particular configuration, A can be written as –

$$E(A) = E_{Likelihood}(A) + E_{\Pr ior}(A)$$

Image segmentation can be considered as a labeling problem where each pixel is assigned a label from a given set. For this case, the objective is to find a label assignment (ie. configuration A) that minimizes the energy function E(A).

Several MRF-based segmentation methods have been proposed previously. The main differences between these approaches is in, the way the prior model is defined, the likelihood term is computed and the optimization technique that is used. Geman and Geman in their seminal work ([3]), proposed Simulated Annealing method for solving the optimization problem in MRF. Based on this, Lakshmanan et al. in [4], proposed a SA based segmentation approach where the Gibbs Random Field parameters are obtained using Maximum Likelihood Estimation. Besag in [5], proposed Iterative Conditional Mode (ICM) for solving the optimization problem. Marroquin et al. in [6] proposes yet another optimization technique known as the Maximizer of Posterior Marginals (MPM). Although these works were primarily on image restoration, the approach is analogous to image segmentation.

In [7], the authors compare these different segmentation techniques that are based on SA, ICM, and MPM. In their work, they found ICM to be the most robust among the three but demonstrated that the a-posterior energy is not always minimized in the best segmentation.

Grieg et al. in [8], however, takes a different approach to MAP estimation for binary images. They model the MAP estimation problem as a combinatorial graph problem and uses maximum flow algorithm to solve the optimization problem. They have reported their approach to work better than SA and ICM. The interactive graph cuts approach [1], presented in this report is based on that work. The authors in [1], extends it to the problem of image segmentation for N-D images.

One common problem with most of the approaches that came before [1], is that they are limited to 2D and are also unable to compute a global optimal solution. There are also several works that are based on [1]. One of these is [9], which formulates a “Gaussian Mixture Markov Random Field” (GMMRF) approach where the color mixture and the consistency parameters are learned to make interactive image segmentation much easier and accurate.

III. Interactive Image Segmentation

This section presents the details of modeling interactive image segmentation using MRF and optimization technique to solve the MAP-MRF estimation problem.

A. *MRF Modeling*

For the case of MRF modeling, the following assignments and notations are use –

- P: The site or a set of pixels (or voxels)
- N: The neighborhood which is the set of all unordered pixel pairs.
- L: The set of labels which in this case is {“obj”, “bkg”} denoting the object and the background respectively
- A: The configuration which is a particular assignment of the labels to each site
- A_p: A label assignment to pixel p.

The energy or cost of a particular configuration is defined as follows –

$$E(A) = \lambda R(A) + B(A)$$

Where,

$$R(A) = \sum_{p \in P} R_p(A_p) \,,$$

$$B(A) = \sum_{\{p,q\} \in N} B_{\{p,q\}} . \delta(A_p, A_q)$$

and,

$$\delta(A_p, A_q) = \begin{cases} 1 & if \quad A_p \neq A_q \\ 0 & otherwise \end{cases}$$

Here, R(A) and B(A), which are the likelihood and the prior terms, are known as the regional and boundary term respectively. These terms represent the penalties for labeling a pixel as either object or background. More specifically, the region term reflect how the attributes of a pixel p (eg. intensity value) is consistent with the attributes of a label. And the boundary term encodes the similarity between two neighboring pixels. The coefficient λ controls the relative importance of the region term over the boundary term. Large values of λ signifies higher confidence in the likelihood estimation – which is based on the observed data – more than the prior model. Whereas lower values of λ means that the observed data might be unreliable and hence more importance is given to the prior model.

There is no specific way of computing the regional or boundary terms. But the authors consider the negative log-likelihood for the regional term and an ad-hoc boundary term that takes the similarity and spatial relation of neighboring pixels into consideration. Therefore, the terms are computed as follows –

$$R_p(\text{"}obj\text{"}) = -\ln P(I_p \mid O)$$
$$R_p(\text{"}bkg\text{"}) = -\ln P(I_p \mid O)$$

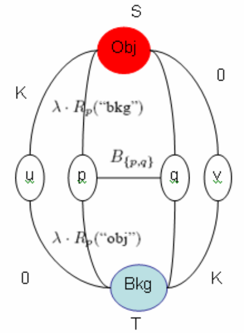
and,

$$B_{\{p,q\}} \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) . \frac{1}{dist(p,q)}$$

The I_p in the above equations can be either a pixel’s intensity value or any other feature vector. Since the regional penalty is calculated as the negative log-likelihood, if the likelihood of a pixel being of certain class is high, the penalty for assigning the pixel to that class is low. The probability in the region term is calculated using the histogram of the “object” and “background” pixels. In the boundary term, the exponential component represents the similarity between pixels and the distance component captures their spatial relationship.

B. *Graph Cut Optimization*

The next step is to formulate the energy minimization problem as a combinatorial graph problem. This is done by constructing a graph using the sites as nodes and adding two additional nodes known as the terminal nodes. The terminal nodes represent the “object” (S) and “background” (T) class. The crucial part is assigning the edge weights which represent the regional and boundary terms that were formulated in the previous section. This edge weight assignment is shown in Figure 1. The edge connecting neighboring sites is called the n-link and the edge connecting sites to terminal nodes is called t-link.



edge	weight	for
{p,q}	$B_{\{p,q\}}$	$\{p,q\} \in N$
{p,S}	$\lambda . R_p$("bkg")	$p \in P, p \notin O \cup B$
	K	$p \in P$
	0	$p \in B$
{p,T}	$\lambda . R_p$("obj")	$p \in P, p \notin O \cup B$
	0	$p \in P$
	K	$p \in B$

Where $K = 1 + \max_{p \in P} \sum_{q:\{p,q\} \in N} B_{\{p,q\}}$

Figure 1: Graph Construction and Weight Assignment

Now, the optimization problem can be formulated as a minimum cut problem where a set of edges that separates the graph into two components is selected. The constraint here is that the sum of these edges has to be minimum. This problem is solved using max-flow/min-cut algorithm.

IV. Implementation and Results

The algorithm was implemented using C++ and OpenCV library. The max-flow/min-cut algorithm implementation developed in [10] was used to solve the minimum cut problem. Results were obtained on a machine with 1.79GHz processor and 768MB RAM running Windows XP. Different parameter values and scene of different complexity were considered during experimentation. The overall time taken to perform segmentation was also recorded. In the implementation, the n-link assignments are done either at the beginning or whenever the value of σ is modified. And if new seeds are added then the t-links are updated before executing max-flow/min-cut algorithm.

Figure 2 shows the results for performing segmentation on a set of benchmark dataset images with ground truth [11,12]. The error was calculated as the percentage of misclassified pixels.

Timing measurements were taken for both small and large images. For small images (481x381) time taken to assign weights to n-links was between 1.5-2 secs. The t-link assignment and max-flow/min-cut algorithm in total took about 0.3-0.6s. For larger images (800x600) n-link construction took 5-6s and t-link took 3-5s.

V. Discussion

In this section, the different effects of changing parameter values λ and σ are discussed. Figures 3 (a) and (b) show the effect of changing the parameter value λ. When λ is high (Figure 3a) it means that we are giving more importance to the observed data. Therefore, high value of λ makes it similar to thresholding based segmentation. This makes segmentation difficult if the object and background have similar color. For this case, setting $\lambda = 0$ yields a

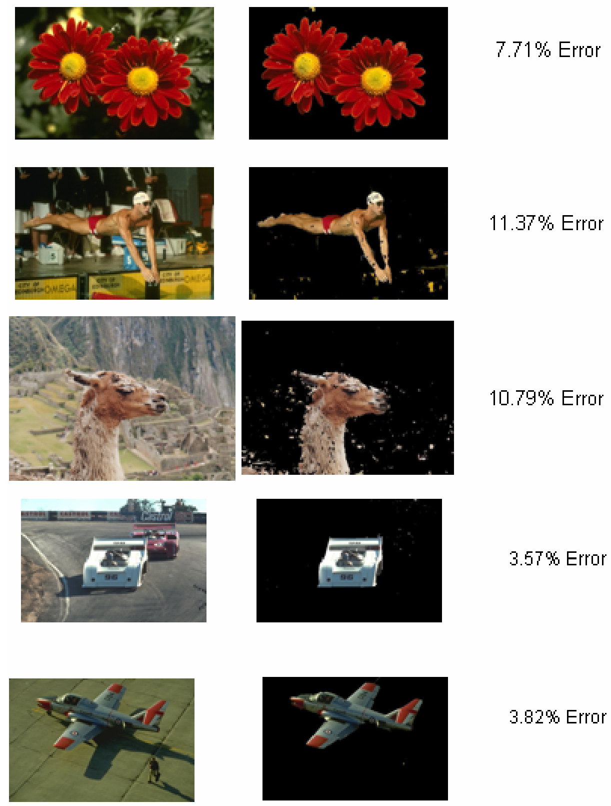
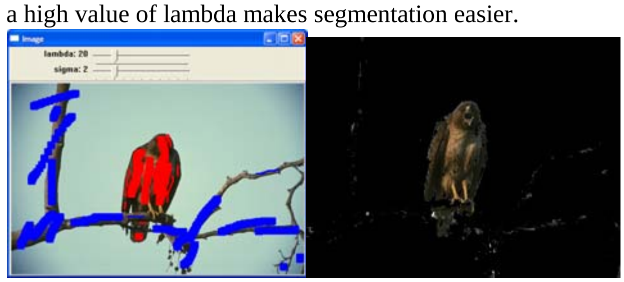
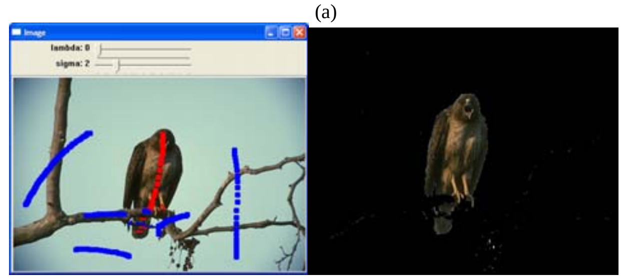


Figure 2: Segmentation results for benchmark dataset with ground truth

much better result (Figure 3b). However, if we wanted to consider both the bird and the tree as the object, then having a high value of lambda makes segmentation easier.



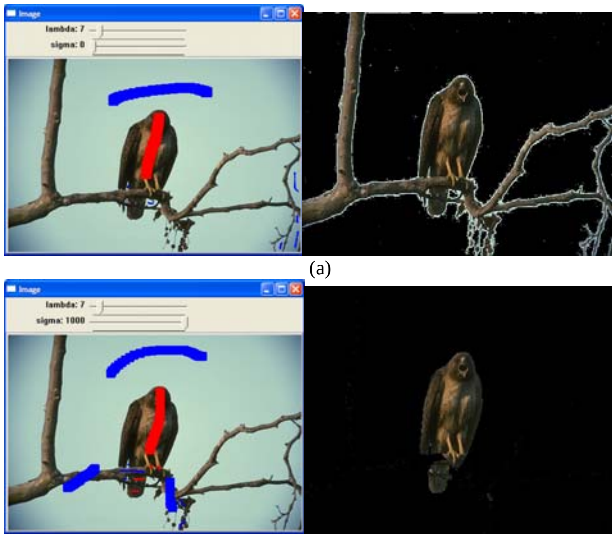
(a)



(b)

Figure 3: Effect of different values for λ (a) $\lambda = 20$ (b) $\lambda = 0$

The effect of changing σ can be seen in Figures 4 (a) and (b). Figure 4(a), shows the result of segmentation using $\sigma = 0.1$. Due to the small value of σ the boundary penalty term between the neighboring pixels is very high and as a result the effect shown in the figure is seen. Figure 4(b), shows the result for $\sigma = 100$. In this case the misclassification near the boundary is less visible.

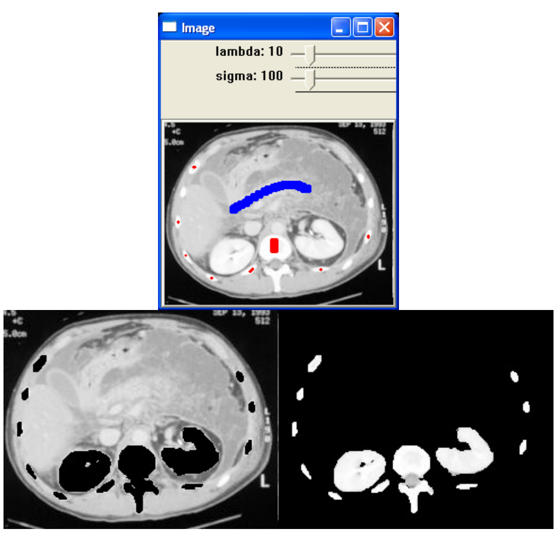


(a)

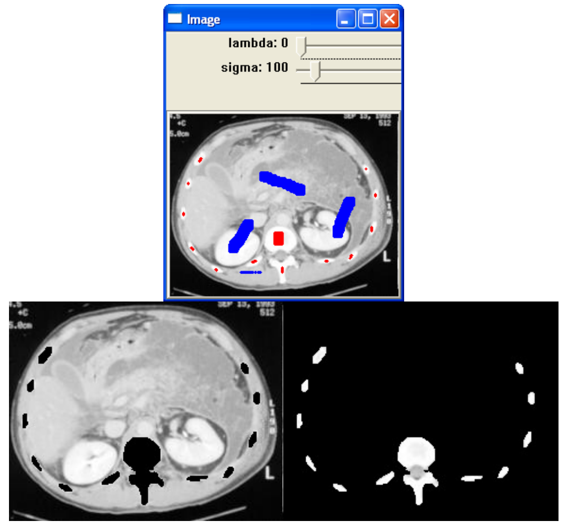
(b)

Figure 4: Effect of different values of σ (a) $\sigma = 0.1$ (b) $\sigma = 100$

The flexibility of interactive image segmentation can be seen from the following figure. In the first case we are interested in extracting both bone and kidneys. This is done by setting λ to a large value (eg. 10) and marking the regions with a few brushstrokes. For the case where we want to extract only the bones, λ is set to 0. However, in this case, considerably more brushstrokes is required to segment the image.

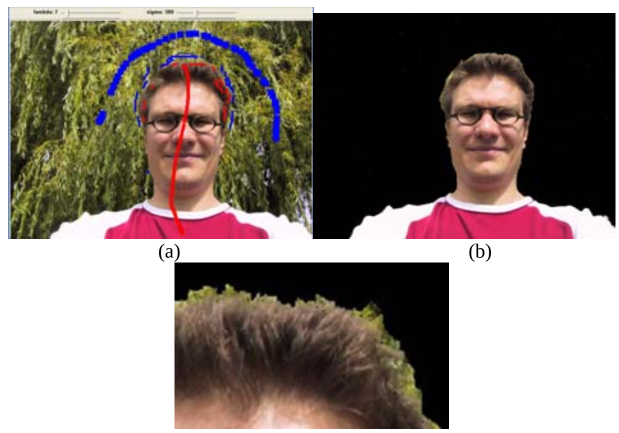


(a)



(b)

Figure 5: (a) Segmenting bones and kidneys (b) Segmenting only bones



(a) (b)

(c)

Figure 6: (a) Input image (b) Foreground (c) Error near the boundary

One of the problems with the implementation is that pixels on the boundary of the object are harder to classify, especially in the case of hair. Figure 6, shows an example where two brushstrokes are sufficient to separate the person from the background. However, it was not possible to do fine segmentation near the person's hair. Figure 6(c) shows the misclassified pixels.

Interactive image segmentation has disadvantage in cases where manually assigning certain pixels as seed is difficult. One example of this would be in medical images where pixels of interest may be very hard to detect manually. Also automatic segmentation methods that take multiple images into consideration (eg. same images of different modalities) will have clear advantage over interactive segmentation. Also for the method presented here, the parameter values λ and σ has to be modified depending on the problem we are trying to solve and the complexity of the image that we are trying to segment. In some cases, this may be a difficult thing to do.

VI. CONCLUSION

. The main contribution of [1] is in generalizing the algorithm in [8] by taking user defined constraints into consideration. The resulting method provides a fast and easy way of doing interactive image segmentation of N-D images.

This report presented an overview of the work done in [1] as well as the implementation details. The results obtained for the benchmark dataset with ground truth ([11,12]) were also presented and the effects of changing different parameters were discussed. The results that were obtained are similar to that reported in the paper. In the report, timing measurement for each steps were also provided to give an estimate of how fast the algorithm works. The current implementation can be further improved to take different features - like local intensity gradient, edge information or gradient direction, into consideration. It can also be extended to handle general N-D image segmentation.

REFERENCES

[1] Y. Boykov and M.P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images". *In Proceedings of International Conference on Computer Vision*, Vol. 1, pp. 105-112, July 2001.

[2] Y. Boykov and G. Funka-Lea, "Graph Cuts and Efficient N-D Image Segmentation". *International Journal of Computer Vision*, 70(2), 109-131, 2006.

[3] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distribution, and the Bayesian restoration of images," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 721--741, 1984.

[4] S. Lakshmanan and H. Derin, "Simultaneous Parameter Estimation and Segmentation of Gibbs Random Fields Using Simulated Annealing," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 11, pp.799—813, 1989.

[5] J. Besag, "On the Statistical Analysis of Dirty Pictures," Journal of the Royal Statistical Society, Series B, Vol. 48, pp.259—302, 1986

[6] J. Marroquin, S. Mitter, and T. Poggio, "Probabilistic Solution of Ill-Posed Problems in Computational Vision," J. American Statistical Association, Vol.82, pp.76—89, 1987.

[7] R.C.Dubes, A.K. Jain, S.G. Nadabar, and C.C. Chen, "MRF model-based algorithms for image segmentation," In Proceedings of 10th International Conference on Pattern Recognition, 1990., Vol.I, no., pp.808-814 vol.1, 16-21 Jun 1990.

[8] D. Greig, B. Porteous, and A. Seheult, "Exact Maximum A Posteriori Estimation for Binary Images," Journal of the Royal Statistical Society, Series B, 51(2):271—279,1989.

[9] A. Blake, C. Rother, P. Perez, and P. Torr, "Interactive Image Segmentation using an adaptive GMMRF model," In Proceedings of European Conference on Computer Vision, pp. 428—441, 2004.

[10] Y. Boykov, and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," In IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), September 2004.

[11] Microsoft Research. Microsoft Foreground Extraction Benchmark Dataset. http://research.microsoft.com/vision/cambridge/i3l/segmentation/GrabCut.htm

[12] The Berkeley Segmentation Dataset and Benchmark. http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/