

# Interactive Graph Cut Based Segmentation With Shape Priors

Daniel Freedman and Tao Zhang

Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180

## Abstract

*Interactive or semi-automatic segmentation is a useful alternative to pure automatic segmentation in many applications. While automatic segmentation can be very challenging, a small amount of user input can often resolve ambiguous decisions on the part of the algorithm. In this work, we devise a graph cut algorithm for interactive segmentation which incorporates shape priors. While traditional graph cut approaches to interactive segmentation are often quite successful, they may fail in cases where there are diffuse edges, or multiple similar objects in close proximity to one another. Incorporation of shape priors within this framework mitigates these problems. Positive results on both medical and natural images are demonstrated.*

**Keywords:** segmentation; graph cuts; shape priors; level sets.

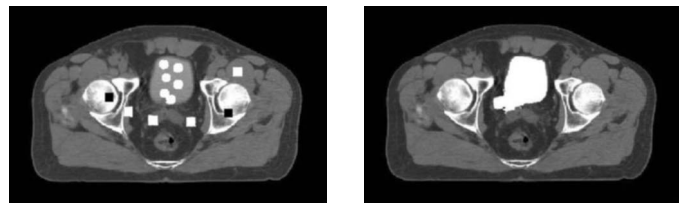
## 1 Introduction

Interactive or semi-automatic segmentation is a useful alternative to pure automatic segmentation in many applications. While automatic segmentation can be very challenging, a small amount of user input can often resolve ambiguous decisions on the part of the algorithm. One example application is the use of segmentation to improve the accuracy of prostate radiation therapy. A patient may be scanned prior to treatment; having access to the segmentation of the prostate and surrounding structures can make the therapy considerably more precise. However, the two standard options, manual segmentation and fully automatic segmentation, are both problematic. On the one hand, manual segmentation is very time-consuming – 20 minutes for the prostate alone; on the other hand, automatic segmentation is very challenging in this type of medical imagery, due to diffuse edges and the presence of multiple objects with similar intensity profiles. As a result, interactive segmentation, which relies on minimal physician input and is easily refined, is an attractive option.

Boykov and Jolly [1, 2] proposed a very effective method for interactive segmentation based on graph cuts. The user

input is minimal, consisting of a few mouse-clicks indicating some pixels which are inside the object of interest, and some which are outside. An energy function based on both boundary and region information is then minimized subject to these user-imposed constraints. The global minimum is found by using graph cut techniques. The results from this method are quite impressive: with a relatively small amount of user input, the algorithm successfully segments a variety of objects from both medical and natural images.

Unfortunately, there are some cases in both medical and natural images, where this cut-based method is insufficient. Figure 1 shows an example of this phenomenon, in the case of a medical image of a bladder, even with a significant amount of user-input. The cause of this failure is often (a)



**Figure 1. The graph cuts algorithm fails with a fair amount of user input. (a) The user input: circles indicate object points, squares indicate background point (colour is for visualization purposes only). (b) The segmentation, shown in white, leaks out of the bladder.**

the absence of strong boundaries and (b) the presence of a number of objects with similar intensity profiles. The former confounds the boundary terms in the energy, while the latter confounds the region terms.

Our solution to this problem is to include shape priors in a graph cut based formulation. A variety of segmentation methods have been designed with the idea of using shape knowledge; obviously, this knowledge can only improve the performance of the algorithm. However, to this point, it has been difficult to find a way to incorporate shape priors into a graph cuts based approach. This is highly desirable because, as was mentioned above, the graph cuts approach

is one of the few that *guarantees a global optimum*. Our solution is simple, yet effective. The idea is to use graph edge-weights which contain information about a level-set function of a template, in addition to the usual boundary and region terms. This allows the edges of the graph to convey information both about the image as well as about the prior shape knowledge. The template itself can be transformed, where the particular transformation is chosen based on the user input. Using this algorithm on a variety of images shows that it can lead to significant improvements in segmentation.

The remainder of the paper is organized as follows. Section 2 examines related work. Section 3 describes the algorithm: following a brief description of the original Boykov-Jolly algorithm, we describe how to emend the edge-weights to include shape information, and how transformations of the template can be incorporated. In Section 4, we show results on both medical and natural images, which demonstrate that shape priors can improve performance. Section 5 concludes.

## 2 Related Work

Related work falls into two categories: segmentation using shape priors, and globally optimal methods for segmentation (including graph cuts). We address each in turn.

Let us begin with segmentation using shape priors. Some researchers have augmented a level-set active contour segmentation algorithm with a PCA term that biases the curve evolution towards shapes that are judged to be more likely based on the training set [13, 16]. Cremers et al. incorporated a more sophisticated (non-PCA) shape model [7]. Segmentation of 3-D medical images has also been accomplished by the coarse-to-fine deformation of a shape-based medial representation model, or “m-rep” [14, 17].

Other segmentation methods use both shape priors and an appearance model. Simple models of appearance are used in [18, 19]; for example, the intensities within the segmented areas may be forced to have means or variances significantly different than the background. There are a variety of methods that model the shape and appearance of an object using PCA. The standard-bearer for such methods is the “active shape and appearance model” of Cootes et al. [5], which has been successfully applied to the three-dimensional segmentation of medical volumes, including magnetic resonance images of the brain, heart, and articular cartilage [9, 12, 20].

Let us now examine globally optimal methods for segmentation. The most relevant papers are those by Boykov and Jolly [1, 2] which have already been mentioned. However, there are several earlier papers which are also relevant. The method of Greig et al. [8] is similar to that of [2] when there is no user input. Wu and Leahy [21] pose the problem

of segmenting an image into  $K$  components (again, without user input) in terms of graph cuts. This technique tends to have a bias towards small components. While the current algorithm, and those already mentioned, can work on two- or three-dimensional images, there are a slew of algorithms which are specialized to two-dimensional segmentation. These include snake-related method [4], ratio regions [6], deformable template methods [22], as well as the work of Jermyn and Ishikawa [11].

## 3 The Algorithm

In this section, we describe the method of incorporating shape priors into graph cut based segmentation. In Section 3.1, we focus on the method of [2] for performing graph cut segmentation without shape priors. In Section 3.2, we explain the mechanism for incorporating shape priors within this formulation for a fixed template. In Section 3.3, we show how the template itself can be adjusted.

### 3.1 Interactive Graph Cuts Segmentation

Boykov and Jolly [2] introduced a novel interactive method for segmentation. The idea is as follows: the user marks some pixels as being part of the object of interest, and some as lying outside the object i.e. within the background. The number of such points is up to the user, but in practice can be quite small (less than ten). Given these constraints, the algorithm tries to find the optimal segmentation such that these hard constraints are satisfied. In particular, a segmentation is scored according to the following criteria:

1. Each pixel inside the object is given a value according to whether its intensity matches the object’s appearance model; low values represent better matches.
2. Each pixel in the background is given a value according to whether its intensity matches the appearance model of the background; low values represent better matches.
3. A pair of adjacent pixels, where one is inside the object and the other is outside, is given a value according to whether the two pixels have similar intensities; low values correspond to contrasting intensities (i.e. to an edge).

Note that the appearance models can be learned *a priori*, or they can be learned by examining the points selected by the user as hard constraints. More will be said on this subject in Section 4.

Given these criteria for scoring a segmentation, the goal is to devise an algorithm that can find an optimal segmentation. Specifically, let  $p$  be a pixel, let  $\mathcal{P}$  be the set of all

pixels, and let  $A_p = 0$  or  $1$  if  $p$  is in the background or the object, respectively. Let  $R_p(A_p)$  be the individual pixel matching cost for pixel  $p$  (items 1 and 2 above); let  $B_{p,q}$  vary inversely with the difference of intensities of pixels  $p$  and  $q$  (see item 3 above). Then the segmentation score is given by

$$E = \mu \sum_{p \in \mathcal{P}} R_p(A_p) + \sum_{(p,q) \in \mathcal{N}: A_p \neq A_q} B_{p,q}$$

where  $\mathcal{N}$  is the set of neighboring pixels. The particular forms for  $R_p(A_p)$  and  $B_{p,q}$  are discussed in Section 4.

A fast (polynomial time) combinatorial algorithm exists for minimizing  $E$ , based on the problem of computing a minimum cut across a graph. We will defer the discussion of this until after we formulate the new energy function which incorporates shape priors.

### 3.2 Adding in Shape Priors

We begin by assuming that our shape prior is a single fixed template. Clearly, this is not a realistic assumption, as the object may, at the very least, undergo rigid transformations to move around the image. We drop this assumption later on, in showing how to accommodate such transformations. Our goal will be to emend the energy function to be

$$E = (1 - \lambda)E_i + \lambda E_s$$

where  $E_i$  is the image energy described in Section 3.1, while  $E_s$  is an energy based on the shape prior.

Our first attempt at defining a shape energy might run as follows. Suppose our fixed template is the curve specified parametrically as  $\bar{c}(s)$ . Our segmentation is given by the variables  $A_p$ ; the curve surrounding the segmented object may be written as

$$c = bd(\{p \in \mathcal{P} : A_p = 1\})$$

where  $bd(S)$  is the boundary of the set  $S$ . (In this case, the boundary might be extracted by standard set operations on images.) We could then parametrize this curve to get  $c(s)$ ; if the parameter  $s$  is defined on  $[0, 1]$ , a natural energy function is

$$E_s[c(\cdot)] = \int_0^1 [c(s) - \bar{c}(s)]^2 ds$$

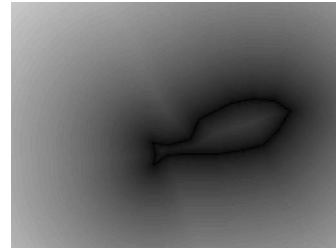
Such an energy will indeed achieve its global minimum when  $c(s) = \bar{c}(s)$  for all values of  $s$ . Unfortunately, however, there are two main problems with this simple energy. First, it is dependent upon a parametric specification of both  $c$  and the template  $\bar{c}$ . This implies that there must be a reasonable correspondence made between the parameters of  $c$  and  $\bar{c}$ ; otherwise, the functional  $E_s$  can give meaningless

values. For example, suppose that  $c(s) = \bar{c}(1 - s)$ ; geometrically, then,  $c$  and  $\bar{c}$  represent the same curves. However, they are parametrically distinct, and therefore the value of  $E_s$  will be positive in comparing these curves (and could, potentially, be quite large). Furthermore, combining this shape-based criterion with the image-based criterion of the previous section may lead to even more unpredictable results. The second problem is perhaps even more important: we cannot use graph cut techniques to minimize such a function. As a result, we cannot guarantee the global optimality of the solution, which is critical to our approach.

Thus, we must find a functional which allows us to match the segmented curve  $c$  with a template curve  $\bar{c}$ , and yet does not rely on a parametric specification of either the segmented curve or the template. In order to achieve this goal, let us specify the template as a distance function whose zero level set corresponds to the template. That is, let  $\bar{\phi} : \mathbb{R}^2 \rightarrow \mathbb{R}$  be such that

$$\bar{c} = \{x \in \mathbb{R}^2 : \bar{\phi}(x) = 0\}$$

where  $\bar{c}$  so specified is given as a collection of points. Note that unlike much of the work in the active contours literature [15],  $\bar{\phi}$  is *not* a signed distance function; it is a regular, *unsigned* distance function. An example of such a function for a curve which is the contour of a fish is given in Figure 2.



**Figure 2. Distance function  $\bar{\phi}(x)$  for the contour of a fish.**

Using this idea of a level-set template, the shape energy can be written in the following form.

$$E_s = \sum_{(p,q) \in \mathcal{N}: A_p \neq A_q} \bar{\phi}\left(\frac{p+q}{2}\right)$$

where  $\mathcal{N}$  is the set of neighboring pixels. Let us explain the meaning of this function. The energy will be low if

$$\bar{\phi}\left(\frac{p+q}{2}\right) \approx 0$$

for all neighbouring pixels  $p$  and  $q$  where one of the pixels belongs to the object and the other to the background (i.e.

$A_p \neq A_q$ ). But recall that if a point  $x$  lies near the shape template, then it will satisfy  $\bar{\phi}(x) \approx 0$ . Since  $(p+q)/2$  is roughly a point on the boundary of the segmented object, the condition for  $E_s$  to be small is the same as the condition that the boundary of the segmented object lies near the shape template. Thus, this particular form for  $E_s$  neatly captures the idea of shape priors.

Finally, we have that the energy function is

$$E = \sum_{p \in \mathcal{P}} (1 - \lambda) \mu R_p(A_p) \quad (1)$$

$$+ \sum_{(p,q) \in \mathcal{N}: A_p \neq A_q} \left[ (1 - \lambda) B_{p,q} + \lambda \bar{\phi} \left( \frac{p+q}{2} \right) \right]$$

### 3.3 Transforming the Template

The foregoing discussion centered on using a fixed template. In reality, we would like to deal with a deformable template, which can undergo a variety of transformations, i.e. with a *family* of learned curves. In what follows, we will focus on the group of euclidean similarity transformations of a particular fixed template; we will treat the rigid part of the transformation and the scale part separately. It is important to note that we are not dealing with any other types of non-rigid deformations, which may be specific to the object at hand. At the same time, the algorithm seems to be experimentally quite robust to the situation in which the template is not exact, i.e. in which the true object and the template are related by some non-rigid deformation. More will be said on this topic in Section 4.

Let us now turn to the treatment of rigid transformations of the template, i.e. to rotations and translations. The key idea is to realize that we already have a pretty good idea of where the segmented object is, based on the user input. For example, the centroid of the user input gives us a reasonable idea of the centroid of the true object. More formally, we can treat this input as landmark data, and use the Procrustes Method [10] to match the template curve  $\bar{c}$  to the data, yielding the transformed template curve  $\bar{c}_{trans}$ . (In fact, we use a variant of the Procrustes Method, see [23].) Given this transformed curve  $\bar{c}_{trans}$ , we can then compute its distance function  $\bar{\phi}_{trans}$  as our input to the scale-based algorithm described in the next paragraph. Note that the rigid transformation we will compute via the Procrustes Method will not be extremely accurate, due to the paucity of user input. However, this method turns out to be sufficient for the reasons mentioned in the previous paragraph, namely that the algorithm is robust to the situation in which the template is not exact. This is born out in the experiments of Section 4.

It remains to deal with scale. Once we have computed the optimal rigid transformation, our approach to scale is based on brute force. We compute a gaussian pyramid of the image, and simply minimize the energy function given

in (1) – via the graph cut techniques described in the next section – for each level of the pyramid. The key is that for each level we use the level-set template  $\bar{\phi}_{trans}$  at the *same, fixed scale*.<sup>1</sup> By keeping the scale of the template fixed, but shrinking the scale of the image, we are effectively looking for a larger object.

The reason for proceeding in this way, rather than by expanding the template, has to do with the complexity of the operation. Rather than dealing with several operations on an image of a fixed size, we are dealing with operations on multiple smaller images. In fact, despite the brute force nature of the operation, it is not very expensive. The total number of pixels in a pyramid are less than 4/3 times the number of pixels in the original image, as each level is 1/4 the size of the previous one. Since the graph cut techniques used are, in practice, linear in the number of pixels [3], the time to run the algorithm should only increase by 1/3 at most.

To decide on the best segmentation amongst all of the scales, we may simply compare the scale-normalized values of the optimal energies at each level. That is, for pyramid level  $k$  (where level 1 is the finest), we must multiply the optimal energy by  $4^k$ , as there are  $4^k$  more pixels in level 1 as there are in level  $k$ . In practice, using four levels of the pyramid gives good results.

### 3.4 Minimizing the Energy

In order to minimize the energy function given in (1), we can use graph cut techniques. Our undirected graph  $G = (V, E)$  is as follows. The set of vertices is just the set of pixels augmented by two special vertices:  $V = \mathcal{P} \cup \{S, T\}$  where  $S$  is the source and  $T$  is the sink. The set of edges consists of all neighbouring pairs of pixels, along with an edge between each pixel and the source and sink:

$$E = \mathcal{N} \cup \{(p, S), (p, T) : p \in \mathcal{P}\}$$

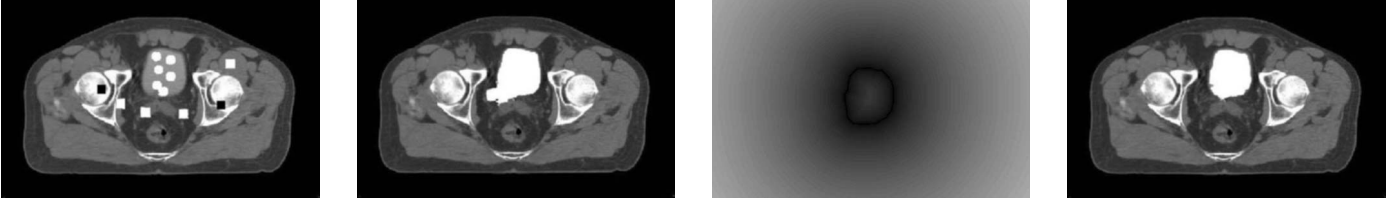
In terms of the weights on the edges, there are three cases to consider. If  $(p, q) \in \mathcal{N}$ , then

$$w(p, q) = (1 - \lambda) B_{p,q} + \lambda \bar{\phi}_{trans} \left( \frac{p+q}{2} \right)$$

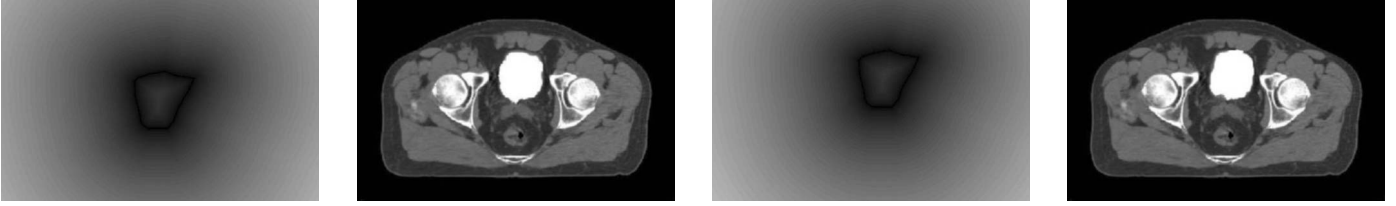
On the other hand, if the edge contains the source  $S$  as one of its vertices, then

$$w(p, S) = \begin{cases} \infty & p \in \mathcal{O} \\ 0 & p \in \mathcal{B} \\ (1 - \lambda) \mu R_p(A_p = 0) & \text{otherwise} \end{cases}$$

<sup>1</sup>Of course, for bookkeeping purposes, we must shift the template according to the decimation entailed by the gaussian pyramid. That is, the pixel  $(2i, 2j)$  at level  $n$  of the pyramid corresponds to the pixel  $(i, j)$  at level  $n + 1$ ; thus, we must shift the centroid of the template accordingly.



**Figure 3. Segmenting a bladder. (a) The user input: circles indicate the object, squares indicate the background (colour is for visualization purposes only). (b) The result without shape priors – segmentation is shown in white. (c) The level-set of the shape template after transformation,  $\bar{\phi}_{trans}$ . (d) The result with shape priors.**



**Figure 4. Changing the template. (a) The level-set of the deformed template (compare with Figure 3). (b) The segmentation result is largely unchanged. (c) The level-set of the deformed and shifted template. (d) The segmentation result is largely unchanged.**

where  $\mathcal{O}$  is the set of pixels selected by the user to belong to the object, and  $\mathcal{B}$  is the set of pixels selected by the user to belong to the background. Finally, if the edge contains the sink  $T$  as one of its vertices, then

$$w(p, T) = \begin{cases} 0 & p \in \mathcal{O} \\ \infty & p \in \mathcal{B} \\ (1 - \lambda)\mu R_p(A_p = 1) & \text{otherwise} \end{cases}$$

It is relatively straightforward to show that the minimum cut on the graph  $G$  corresponds to the minimum of the energy function in (1). We refer the interested to [2] for a formal proof of this fact. The actual maximum flow algorithm which is used for solving minimum cut problem is that described in [3].

## 4 Results

We have run the algorithm on a number of examples, both of natural and medical images. Before describing the results, let us discuss some aspects of the implementation. As in [2], we use the following form for  $B_{p,q}$ :

$$B_{p,q} \propto \frac{e^{-(I(p)-I(q))^2/2\sigma^2}}{\|p - q\|}$$

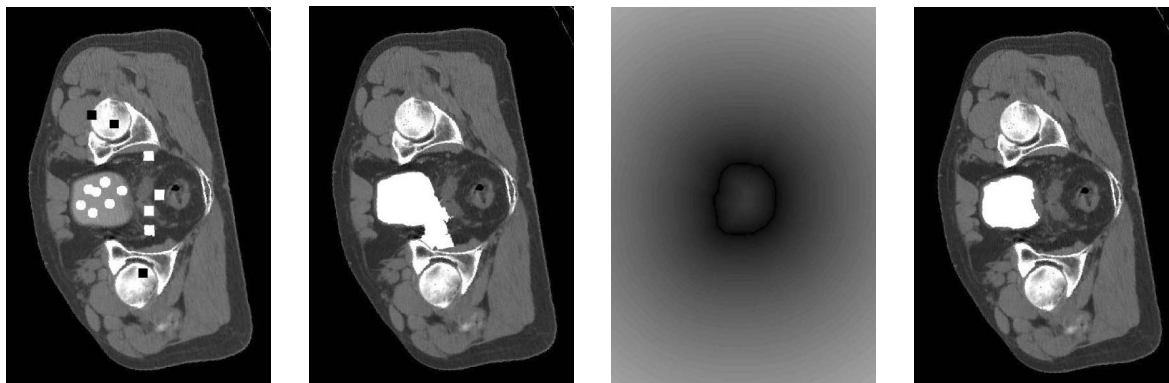
The idea is to make the edge-weight large when pixels have similar intensities, and small when they are dissimilar; in

this case, we will prefer to cut through edges where the pixels are quite different, i.e. along contours in the image. The denominator is the distance between pixels  $p$  and  $q$ ; this term is relevant because we use 8-neighbourhoods, rather than 4-neighbourhoods, so that not all neighbours are equally close. Note the single parameter  $\sigma$ ; in all of the experiments except for the corpus callosum, we set  $\sigma = 3$ , which gave the best performance for the algorithm without shape priors. In the case of the corpus callosum,  $\sigma$  was varied over a wide range in an effort to improve the results of the algorithm without shape priors; please see the discussion below.

The form for  $R_p(A_p)$  given in [2] requires knowledge of some information about the object and the background. In particular, suppose that we know the probability distributions over intensity for both the object and the background, i.e.  $Pr(I|obj)$  and  $Pr(I|back)$ . In this case,

$$\begin{aligned} R_p(A_p = 0) &= -\log Pr(I|back) \\ R_p(A_p = 1) &= -\log Pr(I|obj) \end{aligned}$$

(Recall that  $A_p = 0$  corresponds to the background and  $A_p = 1$  corresponds to the object.) These distributions can either be learned beforehand, or they can be learned based on the user input: two histograms can be built up of the intensity information, one based on the object seeds, and one on the background seeds. In the experiments we have



**Figure 5. Segmenting the transformed bladder image. (a) The user input: circles indicate the object, squares indicate the background (colour is for visualization purposes only). (b) The result without shape priors – segmentation is shown in white. (c) The level-set of the shape template after transformation,  $\bar{\phi}_{trans}$ . (d) The result with shape priors.**

performed, we have found that this region-based term has been helpful sometimes, but not others; for example, in the bladder image, it is not particularly useful, as much of the surrounding tissue is roughly similar to the bladder in intensity. As a result, we have tried running the algorithm with and without this region-based term; we show the result that is best for the original Boykov-Jolly algorithm.

We show the results of experiments on several images; in all cases, the template is captured by hand from an image of a similar object. Let us first examine a case from medical imagery: segmentation of the bladder from a slice of a CT scan. This is a challenging task, as the bladder and the surrounding tissue have similar intensity profiles. In the left two panels of Figure 3, one can see the effect of running the Boykov-Jolly algorithm, even with a relatively large amount of user input: the bladder segmentation bleeds out into the surrounding tissue. The right two panels show the shape template, and the effect when it is used: the segmentation is much more accurate. A natural question is whether the algorithm without shape priors would proceed successfully by simply adding a little more user input. In the experiments in this paper, the answer is generally no; one needs to add significantly more input, typically 2-3 as many points, to find the correct segmentation.

It is a natural question to wonder how accurate the template needs to be for this method to be successful.<sup>2</sup> This is answered in two separate ways. First, we deform the bladder template somewhat, and run the algorithm; this is illustrated in the left two panels of Figure 4. Second, we translate this deformed template by 10 pixels in both verti-

<sup>2</sup>In many of the images, the template appears very well aligned with the object. This is not always the case – as already noted, the Procrustes Method is not perfect. However, in examining the images side-by-side, the eye tends to line up the objects.

cal and horizontal directions (for comparison purposes, note that the bladder has dimensions of about 40 pixels); the results are shown in the right two panels of Figure 4. In both cases, the segmentation remains successful, and is only altered slightly.

Figure 5 shows the success of the algorithm in dealing with transformations. The image in Figure 5 is gotten by rotating the image of Figure 3 by  $100^\circ$  and scaling it down by a factor of 2. The resulting segmentation, shown in the rightmost panel, is once again correct.

Figures 6, 7, and 8 show the segmentations of a maple leaf, a fish, and a corpus callosum, respectively. In all cases, the algorithm without shape priors cannot deal with the complex imagery, despite quite a lot of user input; the shape priors are critical to computing the correct segmentation. In particular, the case of the corpus callosum is interesting. Distinguishing the middle part from the surrounding brain tissue is quite complicated for the non-shape based algorithm; indeed, we tried varying  $\sigma$  over the range 0.1–6 without significant improvement in the final segmentation. ( $\sigma = 1$ , which gives the best performance without using shape priors, is shown.) By contrast, the shape-based method is successful.

## 5 Conclusions and Directions for Future Research

We have devised an algorithm for incorporating shape priors into a graph cuts based interactive segmentation. The shape priors are embedded into the weights on the edges in the graph, by using a level-set formulation. Transformations of the shape template are also taken into account. The results of the algorithm on medical and natural images show

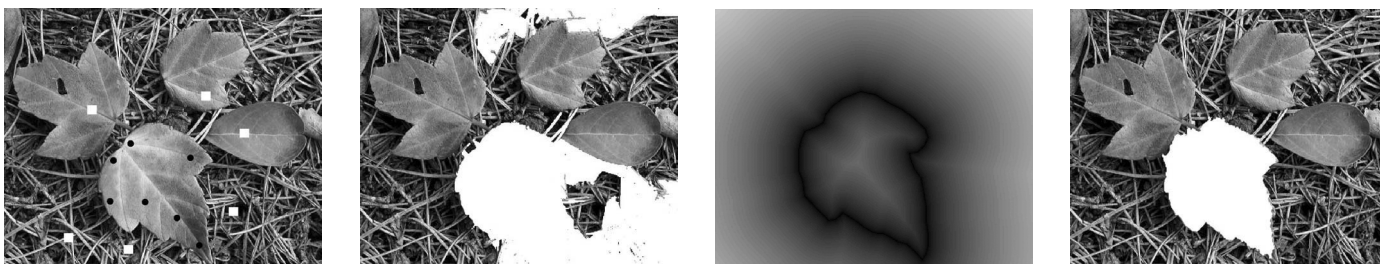


Figure 6. Segmenting a maple leaf. (a) The user input: circles indicate the object, squares indicate the background (colour is for visualization purposes only). (b) The result without shape priors – segmentation is shown in white. (c) The level-set of the shape template after transformation,  $\bar{\phi}_{trans}$ . (d) The result with shape priors.

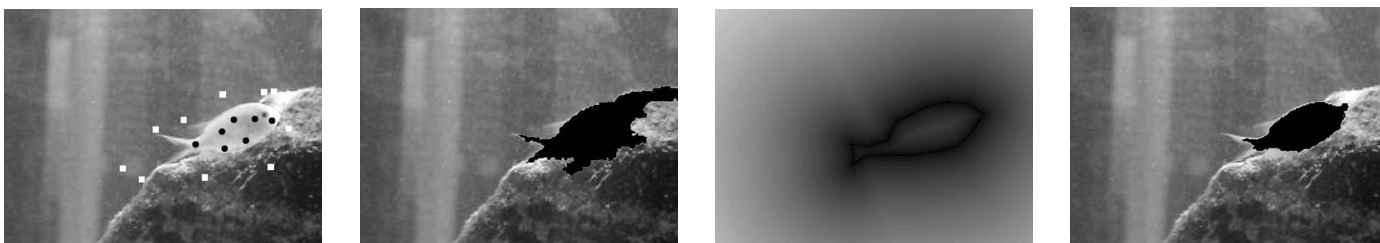


Figure 7. Segmenting a fish. (a) The user input: circles indicate the object, squares indicate the background (colour is for visualization purposes only). (b) The result without shape priors – segmentation is shown in black. (c) The level-set of the shape template after transformation,  $\bar{\phi}_{trans}$ . (d) The result with shape priors.

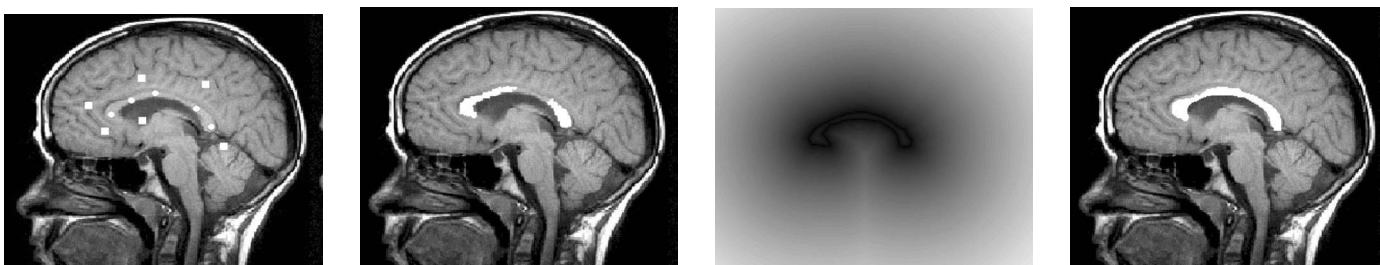


Figure 8. Segmenting a corpus callosum. (a) The user input: circles indicate the object, squares indicate the background (colour is for visualization purposes only). (b) The result without shape priors – segmentation is shown in white. (c) The level-set of the shape template after transformation,  $\bar{\phi}_{trans}$ . (d) The result with shape priors.

the new method shows some promise in making graph cut segmentation more precise.

The main direction for future research is to examine whether more complex transformations of the template can be easily incorporated into the scheme. Many objects can bend and shear in ways that are not taken account of through euclidean similarity transformations. While we have seen that the current algorithm is robust to modest modifications of the shape, it will be interesting to see whether this robustness holds in the case of greater modifications; and if this is not the case, how the algorithm may be modified to account for these changes.

## Acknowledgments

This work was supported by the U.S. National Science Foundation, under Award IIS-0133144.

## References

- [1] Y. Boykov and M.-P. Jolly. Interactive organ segmentation using graph cuts. In *Proc. MICCAI*, page 276286, 2000.
- [2] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proc. ICCV*, volume 1, pages 105–112, 2001.
- [3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.
- [4] L. D. Cohen. On active contour models and ballons. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 53(2):211–218, 1991.
- [5] T. Cootes and C. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Proc. SPIE Medical Imaging*, 2001.
- [6] I. J. Cox, S. B. Rao, and Y. Zhong. “Ratio regions”: a technique for image segmentation. In *Proc. Int. Conf. Pattern Recognition*, volume 2, pages 557–564, 1996.
- [7] D. Cremers, T. Kohlberger, and C. Schnorr. Nonlinear shape statistics in mumford-shah based segmentation. In *Proc. European Conf. Computer Vision*, pages 93–108, 2002.
- [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):271279, 1989.
- [9] A. Hill, A. Thornham, and C. Taylor. Model-based interpretation of 3-D medical images. In *Proceedings of 4th British Machine Vision Conference*, pages 339–348, September 1993.
- [10] I.L.Dryden and K.V.Mardia. *Statistical shape analysis*. Chichester New York, 1998.
- [11] I. H. Jermyn and H. Ishikawa. Globally optimal regions and boundaries. In *Proc. ICCV*, volume 2, pages 904–910, 1999.
- [12] A. Kelemen, G. Szekely, and G. Gerig. Elastic model-based segmentation of 3-D neuroradiological data sets. *IEEE Transactions on Medical Imaging*, 18(10):828–839, October 1999.
- [13] M. Leventon, E. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proceedings of CVPR 2000*, 2000.
- [14] S. Pizer, G. Gerig, S. Joshi, and S. Aylward. Multiscale medial shape-based analysis of image objects. *Proceedings of the IEEE*, 91(10):1670–1679, 2003.
- [15] J. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, 1999.
- [16] D. Shen and C. Davatzikos. An adaptive-focus deformable model using statistical and geometric information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):906–913, August 2000.
- [17] M. Styner, G. Gerig, S. Pizer, and S. Joshi. Automatic and robust computation of 3D medial models incorporating object variability. *International Journal of Computer Vision*, 2002. To appear.
- [18] A. Tsai, W. Wells, C. Tempny, E. Grimson, and A. Willsky. Coupled multi-shape model and mutual information for medical image segmentation. In *IPMI 2003*, pages 185–197, 2003.
- [19] A. Tsai, A. Yezzi, W. Wells, C. Tempny, D. Tucker, A. Fan, E. Grimson, and A. Willsky. A shape based approach to curve evolution for segmentation of medical imagery. *IEEE Trans. Medical Imaging*, 22(2), February 2003.
- [20] Y. Wang and L. H. Staib. Integrated approaches to non-rigid registration in medical images. In *Proceedings of IEEE WACV 1998*, pages 102–108, October 1998.
- [21] Z. Wu and R. Leahy. An optimal graph theoretic approach to data clustering: theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1101–1113, 1993.
- [22] A. Yuille and P. Hallinan. Deformable templates. In A. Blake and A. Yuille, editors, *Active Vision*, page 2038. MIT Press, 1992.
- [23] T. Zhang and D. Freedman. Tracking objects using density matching and shape priors. In *Proc. Int. Conf. Computer Vision*, pages 1056–1062, Nice, 2003.