Model-Based Multi-Object Segmentation via Distribution Matching

Daniel Freedman[1], Richard J. Radke[2], Tao Zhang[1], Yongwon Jeong[2], and George T.Y. Chen[3]

[1]Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180

[2]Electrical, Computer, and Systems Engineering Department, Rensselaer Polytechnic Institute, Troy, NY 12180

[3]Radiation Oncology Department, Massachusetts General Hospital, Boston, MA 02114

Abstract

A new algorithm for the segmentation of objects from 3D images using deformable models is presented. This algorithm relies on learned shape and appearance models for the objects of interest. The main innovation over similar approaches is that there is no need to compute a pixelwise correspondence between the model and the image; instead, probability distributions are compared. This allows for a faster, more principled algorithm. Furthermore, the algorithm is not sensitive to the form of the shape model, making it quite flexible. Results of the algorithm are shown for the segmentation of the prostate and bladder from medical images.

Keywords: deformable segmentation, prostate segmentation, shape and appearance model, medical image segmentation.

1. Introduction

The segmentation of objects from 3D images using deformable models is both an important and interesting problem in computer vision. It is important because of its natural application in the medical arena; for example, segmentation of tumors from CT or MRI images can be critical in the treatment of cancer. On the other hand, it is interesting because of the algorithmic challenges inherent in extracting deformable objects from real-world 3D images. In the context of medical imagery, the key segmentation-related challenges are the following:

- **Challenge #1**: The objects of interest are often diffuse and lack strong edges.

- **Challenge #2**: There are often many objects, both of interest and not of interest, within a small volume.

- **Challenge #3**: Many objects have fairly similar intensity profiles. Typically, this effect cannot be removed by simple pre-processing such as histogram equalization.

- **Challenge #4**: Many of the objects are of roughly the same shape. For example, the prostate and bladder are both "somewhat deformed" spheres.

The algorithm presented in this paper uses learned models for both the shape and appearance of objects to achieve segmentation; learning both types of information is the only reasonable way to deal with all 4 challenges. Our algorithm is certainly not the first algorithm to combine shape and appearance. However, existing algorithms that use both shape and appearance models (such as [5]) require a pixelwise correspondence between the model and the image; this correspondence is often very difficult to compute, and can be extremely time-consuming. Instead, our algorithm characterizes a model object by (a) its shape and (b) a probability distribution of the intensities (or colours, textures) of the pixels within its interior. As a result, comparing a particular model object to the image is as simple as comparing two probability distributions. The algorithm allows the shape to evolve until the optimal match is found. Furthermore, unlike several existing algorithms, our algorithm does not require a particular form for the shape model; any parametric shape model can be used. We used a simple PCA-based model in our experiments, as this is sufficient to achieve good results; however, a more sophisticated shape model can be used if the application demands it.

The remainder of the paper is organized as follows. Section 2 reviews the existing literature on segmentation of 3D objects using deformable models. Section 3 is the heart of the paper; it derives the equations that comprise the segmentation algorithm. Section 4 demonstrates the algorithm's effectiveness in a medical application: segmentation of the bladder and prostate from medical imagery. Finally, Section 5 concludes.

2. Prior Work

Existing algorithms for the segmentation of 3D objects using deformable models can be categorized based on the type of learned information they use.

No learned information. The main exemplar of this type of approach is the traditional active contour or "snake" [11]. More recent work (e.g. [1, 2, 13, 15]) has focused on geometric curve evolution, combined with level sets [16], to allow for both topological changes to the object and greater numerical stability. Standard active contour methods that seek edges tend to have difficulty when the objects to be segmented are blurry or not sharply delineated from the background (such as the prostate in Section 4 below). Extensions such as [4] try to segment on the basis of appearance by evolving the surface based on simple criteria related to the intensities (or colours) of the pixels in its interior; these methods achieve greater accuracy and robustness at the cost of a major reduction in speed. In general, algorithms that do not use learned information are constrained in terms of what they can segment; they will have difficulties with each of the four challenges posed above.

Learned shape models. 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 [14, 19]. Cremers et al. incorporated a more sophisticated (non-PCA) shape model [6]. 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" [18, 21]. Algorithms that possess only a learned shape model will typically fall prey to challenges 3 and 4. Learned shape methods have also been extended to allow for simple (non-learned) models of appearance [22, 23]; for example, the intensities within the segmented areas may be forced to have means or variances significantly different than the background.

Learned appearance models. One version of this type of segmentation involves the non-parametric warping of a target surface to a deformable atlas [10]. Contours from the atlas can then be transferred onto the target volume. This type of method will have trouble with challenges 2, 3, and 4, and can also be slow in high dimensions. Some active contour models [17] assume that one has a probabilistic characterization of appearance that is learned beforehand. Finally, approaches related to the current paper, but for which shape models are either absent or a minor component, have been used by the authors in previous work [7, 26]. Other researchers [9] have used a similar approach.

Learned shape and appearance models. There are a variety of methods that model the shape and appearance of an object using principal component analysis (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 [8, 12, 25].

The main drawback of active shape and appearance model has already been mentioned in Section 1; they require the computation of a pixelwise correspondence between the model and the image. We will say more about this in Section 3.2.

3. The Segmentation Algorithm

In this section, we describe the heart of the algorithm: the procedure for fitting a combined shape-appearance model to an image. This optimal fitting of the model results in the segmentation of the image. The basic idea is as follows. The shape is given by a description of the surface, or multiple surfaces in the case of multi-object segmentation. The appearance is described by a probability distribution of some photometric variable inside the object of interest, or multiple distributions in the case of multi-object segmentation. A shape-appearance pair is then given by (surface, distribution), and this pair is considered sufficient to characterize an object for the sake of segmentation. The learned model is a low-dimensional manifold in the space of such pairs. To verify how well any particular shape-appearance pair matches the image, we compute the empirical distribution of the photometric variable inside the shape within the image; this distribution is then compared to the appearance model. We therefore evolve the shape of an object (or multiple objects) until the empirical distribution(s) best matches the model distribution(s). In the remainder of this section, we flesh out these ideas.

3.1. Terminology

In this section, we describe some of the notation needed to define the problem rigorously. In the case of single-object segmentation, a model-instance is described by a (surface, distribution) pair. The distribution is taken over some photometric variable; in the experiments we perform, this variable is grayscale intensity, however, it may also be colour or texture. Given that the image is discrete-valued, we will assume a probability *mass* function over the intensity. However, all of the analysis below can easily be transferred to the case of probability density functions (which might be more relevant in the case of textures). In the case of multi-object segmentation, a model-instance will be specified by J (surface, distribution) pairs, one for each object.

We assume each surface is a topological sphere, and may therefore be written $S : \mathcal{S}^2 \rightarrow \mathbb{R}^3$. For convenience, we will denote a point on the surface using a parametrization of $\mathcal{S}^2$ as $S(u)$; however, the particular

parametrization chosen is unimportant. Let us denote the image by $I : \mathbb{R}^3 \rightarrow \{1, \ldots, n\}$; the image is piecewise constant, where the "pieces" correspond to voxels (which have positive volume). We denote the probability distribution by $\mathbf{q} = (q_1, \ldots, q_n)$, where $q_i = \text{prob}(I(x) = i)$; of course $q_i \geq 0$ and $\sum_i q_i = 1$. Thus, a model-instance is given by $(S(\cdot), \mathbf{q})$. The shape-appearance model is a low-dimensional manifold in the space of such model-instances; a d-dimensional model is parametrized by $\beta \in \mathbb{R}^d$, and we will write $(S(\cdot; \beta), \mathbf{q}(\beta))$ (sometimes condensing $(S(\cdot; \beta)$ to $S(\beta)$). A particular form for the shape and appearance model $(S(\beta), \mathbf{q}(\beta))$ is discussed in Section 4; in the subsequent derivation, the particular form is unimportant.

The goal is to find the particular model-instance, i.e. the particular β, for which the model best matches the image. In the following section, we describe a natural criterion for scoring such matches.

3.2. Segmentation Criterion

Given a surface S, let $\mathbf{p}^S$ be the distribution (probability mass function) of intensities lying inside the surface S. This can be formally defined as follows. Let V be the volume inside of S; that is, let $S = \partial V$. In this case,

$$p_i^S = \frac{\int_{x \in V} \delta(I(x), i) dx}{\int_{x \in V} dx} \tag{1}$$

where $\delta(i, j) = 1$ if $i = j$ and 0 otherwise. We will refer to $\mathbf{p}^S$ as the *empirical* distribution corresponding the surface S.

The goal of segmentation is to find a region in the image that is most like the model. That is, we would like to find a model shape $S(\beta)$ whose empirical distribution $\mathbf{p}^{S(\beta)}$ most closely matches its model distribution $\mathbf{q}(\beta)$. In other words, the segmentation can be posed as

$$\min_\beta K(\mathbf{p}^{S(\beta)}, \mathbf{q}(\beta))$$

where K is some suitable measure of dissimilarity between probability distributions. There are several obvious candidates for K from information theory. We choose the Kullback-Leibler divergence,

$$K(\mathbf{p}, \mathbf{q}) = \sum_{i=1}^n p_i \log \frac{p_i}{q_i} \tag{2}$$

Note a key feature of this segmentation algorithm: unlike other joint shape-appearance model-based algorithms, there is no need to find a correspondence between the pixels of the model and those of the image. For example, in [5] finding the correspondence would involve repeated 3D image warpings, the most time-consuming and least rigorous part of that algorithm. In our case, by contrast, pixels are

not compared directly; instead, distributions are compared. While some information is obviously lost in performing distribution comparisons instead of pixel-wise comparisons, we show that in relevant experiments (in medical images) this loss of information does not adversely affect performance. Of course, distribution comparison measures may be computed considerably faster than pixelwise correspondences.

When we wish to segment multiple objects at once, our model is given by J object descriptors $\{(S_j(\cdot; \beta), \mathbf{q}_j(\beta))\}_{j=1}^J$, and the goal is then to solve

$$\min_\beta \sum_{j=1}^J K(\mathbf{p}^{S_j(\beta)}, \mathbf{q}_j(\beta))$$

Note that there is a single parameter vector β that controls all of the objects; this captures the notion that the objects' shapes and appearances may be interrelated. Although a more general version of this criterion might be a weighted sum of Kullback-Leibler divergences, we have found the unweighted criterion works well in practice.

3.3. Optimization of the Criterion

We wish to minimize

$$K(\beta) \equiv K(\mathbf{p}^{S(\beta)}, \mathbf{q}(\beta))$$

in the case of single object segmentation. (We will only deal with the single object case in this section; the multi-object case follows straightforwardly.) We will solve for a local minimum of the criterion via gradient descent, i.e.

$$\frac{d\beta}{dt} = -\frac{\partial K}{\partial \beta} \tag{3}$$

From (2), we have that

$$\frac{\partial K}{\partial \beta} = \sum_{i=1}^n \left[\left(1 + \log \frac{p_i}{q_i} \right) \frac{\partial p_i}{\partial \beta} - \frac{p_i}{q_i} \frac{\partial q_i}{\partial \beta} \right]$$

where we have shortened $p_i^{S(\beta)}$ to p_i and $q_i(\beta)$ to q_i. From (1),

$$p_i^{S(\beta)} = \frac{\int_{x \in V(\beta)} \delta(I(x), i) dx}{\int_{x \in V(\beta)} dx} \equiv \frac{N_i^{S(\beta)}}{V(\beta)}$$

so that

$$\frac{\partial p_i}{\partial \beta} = \frac{1}{V} \left(\frac{\partial N_i}{\partial \beta} - p_i \frac{\partial V}{\partial \beta} \right)$$

In order to compute $\partial N_i / \partial \beta$ and $\partial V / \partial \beta$, we need to be able to determine derivatives of the form $\partial \psi / \partial \beta$, where $\psi = \int_{x \in V(\beta)} \nu(x) dx$. The variational derivative of ψ with respect to the surface S (where $S = \partial V$) is given by $\frac{\delta \psi}{\delta S} = \nu(u) n(u)$, where $n(u)$ is the normal to the surface

at the point $S(u)$ (see, for example, [3]). It can be shown by a sort of generalized chain rule that

$$\frac{\partial \psi}{\partial \beta} = \int_{u \in \mathcal{S}^2} \nu(u) \frac{\partial S}{\partial \beta}(u; \beta) n(u; \beta) du$$

where $\partial S / \partial \beta$ is a $d \times 3$ matrix ($d = \dim(\beta)$). To simplify future computations, we introduce the notation

$$\Gamma(u; \beta) = \frac{1}{V(\beta)} \frac{\partial S}{\partial \beta}(u; \beta) n(u; \beta)$$

so that

$$\frac{\partial \psi}{\partial \beta} = \int_{u \in \mathcal{S}^2} V(\beta) \Gamma(u; \beta) \nu(u) du$$

We have, therefore, that

$$\frac{1}{V} \frac{\partial N_i}{\partial \beta} = \int_{u \in \mathcal{S}^2} \Gamma(u; \beta) \delta(I(S(u; \beta)), i) du$$

$$\frac{1}{V} \frac{\partial V}{\partial \beta} = \int_{u \in \mathcal{S}^2} \Gamma(u; \beta) du$$

Some simplification gives

$$\frac{\partial K}{\partial \beta} = \int_{\mathcal{S}^2} \Gamma(u; \beta) \left[\sum_{i=1}^{n} \delta(I(S(u; \beta)), i) \left(1 + \log \frac{p_i}{q_i} \right) \right] du$$
$$- \left(\int_{\mathcal{S}^2} \Gamma(u; \beta) du \right) \left[\sum_{i=1}^{n} p_i + p_i \log \frac{p_i}{q_i} \right] - \sum_{i=1}^{n} \frac{p_i}{q_i} \frac{\partial q_i}{\partial \beta}$$
$$= \int_{\mathcal{S}^2} \Gamma(u; \beta) du + \int_{\mathcal{S}^2} \Gamma(u; \beta) \log \frac{p_{I(S(u;\beta))}}{q_{I(S(u;\beta))}} du$$
$$- \int_{\mathcal{S}^2} \Gamma(u; \beta) du - K(\beta) \int_{\mathcal{S}^2} \Gamma(u; \beta) du - \sum_{i=1}^{n} \frac{p_i}{q_i} \frac{\partial q_i}{\partial \beta}$$

which finally yields

$$\frac{\partial K}{\partial \beta} = \int_{\mathcal{S}^2} \Gamma(u; \beta) \left(\log \frac{p_{I(S(u;\beta))}}{q_{I(S(u;\beta))}} - K(\beta) \right) du - \sum_{i=1}^{n} \frac{p_i}{q_i} \frac{\partial q_i}{\partial \beta} \tag{4}$$

We will use (4) to find a minimum of the Kullback-Leibler divergence with respect to the model parameters, via gradient descent. In general, we cannot compute the integral in (4) analytically; we must resort to a finite element method. This is relatively straightforward, given that the surface representation we use is that of a mesh (simplicial complex). For any triangle of the mesh, the normal is fixed; furthermore, the triangles are chosen to be small enough so that neither $\partial S / \partial \beta$ nor I varies much over the triangle. As a result, we can approximate the integral in equation (4) by

$$\sum_{t \in T} \Gamma(u_t; \beta) \left(\log \frac{p_{I(S(u_t;\beta))}}{q_{I(S(u_t;\beta))}} - K(\beta) \right) a_t$$

where T is the set of triangles in the mesh, u_t is a representative point on the triangle t (typically the centroid), and a_t is the area of the triangle t.

4. Application: Prostate Radiotherapy

In this section, we present the results of our segmentation algorithm applied to an important real-world problem in medical imaging: automatic contouring of organs from volumetric computed tomography (CT) imagery.

A key advance in cancer treatment with radiation has been the introduction of a new technology known as intensity modulated radiotherapy (IMRT). This is a computer-controlled method of delivering radiation using several beams from different angles that can precisely irradiate a 3D target of interest (e.g. a tumor) while simultaneously avoiding nearby radiation-sensitive organs.

The bottleneck in 3D IMRT systems is the significant amount of time and human intervention required to delineate the tumor and nearby structures on each scan. A radiation oncologist can often take 30-45 minutes to outline all of the structures of interest in every axial CT slice. Performing this operation each of the 30-40 times a patient is treated is tedious. Using an uncompiled MATLAB implementation on a modest machine (1.67 GHz AMD machine with 448 MB RAM), our algorithm performs the same procedure in five minutes.

Here, our focus is on prostate cancer, the second leading cause of cancer death for American men. The state of the art in computer vision algorithms applied to prostate segmentation is considerably less advanced than for other sites. In the best hospitals across the country, the status quo is still manual contouring. Several vision approaches have been presented for prostate segmentation from ultrasound; one of the most effective is an active-contour-based method proposed by Shen, Zhan and Davatzikos [20]. The work most comparable to the algorithm described here was recently proposed by Tsai et al. [22, 23] and applied to 3D MRI images of the prostate. These algorithms do not use learned appearance models; the disadvantage of using only learned shape models has already been discussed in Section 2.

4.1. Learning the Shape Model

In order to implement (4), we must have a shape-appearance model, $(S(\beta), \mathbf{q}(\beta))$, ideally learned from training data. Our training set consisted of 17 sets of 512x512x90 CT images of the male pelvis from different patients, in each of which a radiation physicist had outlined the prostate and bladder. Our testing set consisted of 3 different image sets not in the training database. There is a substantial amount of shape variability in this inter-patient training data set. To capture this variability, we used a linear shape model obtained from principal component analysis, a common technique in the model-based segmentation literature; see for example [5]. The gen-

COMPUTER
SOCIETY

eral scheme is as follows: the surface of each object is represented as a mesh, and specified by a vector whose elements are the x-, y-, and z-coordinates of each of the vertices. The vectors representing the organs are then stacked into a combined vector. There is one such combined vector for each training image, and PCA is performed on these combined vectors.

In our case, points from homologous structures were automatically extracted and put into correspondence using a variational-implicit-surface-based algorithm [24] that resamples every set of contours to have the same number of slices and the same number of points equally spaced around each contour. We resampled each object to have 20 slices with 20 points on each contour, and constructed a 10-mode PCA model using the 800 points from each set in the training database. We emphasize that if necessary, this simple model could be replaced with a more sophisticated shape model like those discussed in Section 2, but the segmentation equations would remain the same.

4.2. Learning the Appearance Model

To form an appearance model $\mathbf{q}(\beta)$ from the training data, we could perform PCA on training histograms. In fact, the vectors representing the histograms could be appended to those representing the shapes, which would yield the desired joint model. However, there are two major problems with this approach. First, PCA on histograms does not preserve the property that histograms are positive and sum to 1. Second, a linear combination of training histograms often produces new histograms very unlike any training example.

Instead, we employ a different approach, based on the idea that there will be some overlap (perhaps small) between the initial guess of the object's position and its true position. Our goal should be to extract that section of the initial object volume which overlaps with the true object, and to form our model density solely based on this. Of course, it is not obvious how to extract this overlapping volume. The following heuristic is extremely successful in practice.

For a given test image, the volume corresponding to the initial object position is divided into blocks; denote the set of blocks $B = \{b_j\}$. For each block, we compute the histogram, $\mathbf{h}(b_j)$; we then determine how similar a particular block is to the model by measuring its closeness to each of the training histograms, $\{\mathbf{q}_i^{train}\}$ (there is one such training histogram for each training image we receive). In particular, we compute

$$K_j = \min_i K(\mathbf{h}(b_j), \mathbf{q}_i^{train})$$

If such a value is low, then we know that the Kullback-Leibler distance between the block's histograms and at least one of the training histograms is small; as a result, the block

is likely to belong to the true object. We can then rank order the blocks by their K_j values, and choose only the fraction α of the blocks with the lowest K_j values. These "good" blocks are then deemed to be part of the true object, and the model density $\mathbf{q}$ can be computed as the histogram of the union of these blocks.

Note hat α must be chosen to be less than the fraction of the initial volume that overlaps the true volume; while this fraction is not known *a priori*, $\alpha = 0.25$ produced good results in practice. In fact, the model density as computed using this algorithm is often almost indistinguishable from the density corresponding to the true position of the object.

4.3. Results

We initialized the model using $\beta = 0$ in (3), that is, at the position of the mean shape, and allowed it to converge. Note that this is in contrast to other algorithms where rough manual placement is required to guarantee convergence to the correct result. The algorithm ran successfully on three test images; however, for space reasons, the result for only one of the test patients is pictured in Figure 1. The model and segmentation algorithms are fully three-dimensional, but the results in Figure 1 are represented as 2-D slices to ease visualization. One can appreciate the difficulty of the segmentation problem in this context: the CT images have relatively low contrast in the area of interest, and the prostate, and bladder have gray-scale intensities that are very similar to each other as well as to structures in many other regions.

15 iterations of a discrete version of equation (3) were required; the corresponding running time was just over 4 minutes on a 1.67 GHz AMD machine with 448 MB RAM. Therefore, we conclude that our segmentation algorithm has substantial promise for the problem of rapid, automatic contouring.

5. Conclusions and Future Work

We have demonstrated a segmentation algorithm that matches a learned model of shape and appearance to an image by comparing empirical and model probability distributions. The algorithm produces good results for difficult 3D medical images. Future work will focus on coarse-to-fine methods of sampling the model space, to ensure that the algorithm is not trapped in an incorrect optimum. We also plan to investigate more sophisticated shape models that better fit our data.

We will shortly obtain a corpus of training data that includes CT volumes and contours that vary along both inter-patient and intra-patient axes. That is, for each of many patients we will have many examples of that patient's bodily state. Using this data, we plan to learn a model of inter-
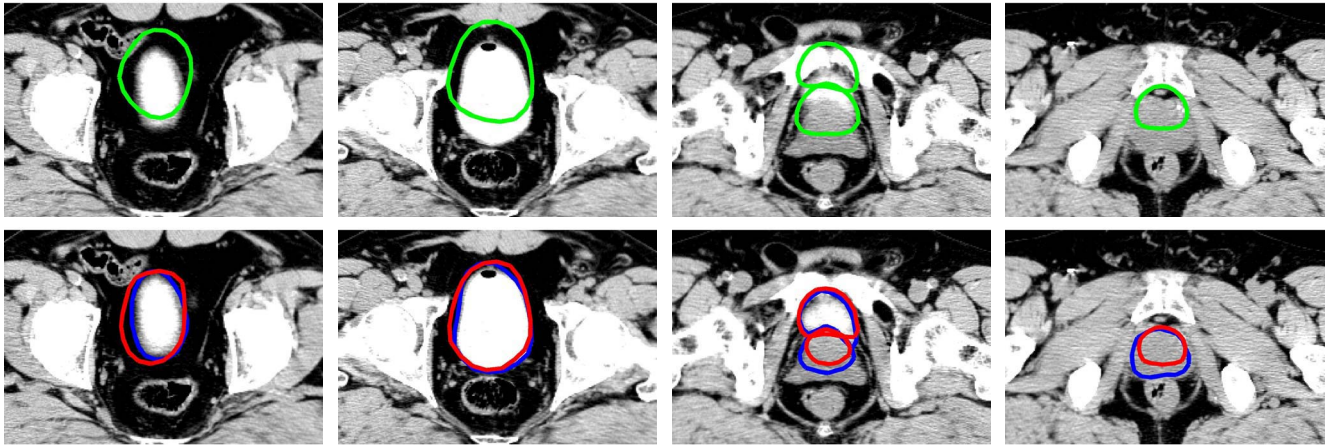
Figure 1. Segmentation results for slices 28, 34, 42, and 48 of patient 2662. The top row shows the initial boundaries for the segmentation (corresponding to the mean shape). The bottom row shows the segmentation result at convergence (red) versus the hand-drawn ground-truth contours supplied by a radiation physicist (blue). The top contour is the bladder and the bottom contour is the prostate. Note that both organs are not visible in every slice.

and intra-patient variation that is more suitable for image-guided therapy applications than the inter-patient model described here. We expect intra-patient data to have much less inherent variability than inter-patient data, and hence for our segmentation algorithms to perform more accurately.

Acknowledgments

References

[1] V. Caselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numer. Math.*, 66:1–31, 1993.

[2] V. Caselles, R. Kimmel, and G. Sapiro. On geodesic active contours. *Int. J. Comput. Vis.*, 22(1):61–79, 1997.

[3] A. Chakraborty, L. Staib, and J. Duncan. Deformable boundary finding in medical images by integrating gradient and region information. *IEEE Trans. Medical Imaging*, 15(6):859–870, 1996.

[4] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. Image Proc.*, 10(2):266–277, 2001.

[5] T. Cootes and C. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Proc. SPIE Medical Imaging*, 2001.

[6] 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.

[7] D. Freedman and T. Zhang. Active contours for tracking distributions. *IEEE Trans. Image Processing*, 2004. Accepted, to appear.

[8] 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.

[9] S. Jehan-Besson, M. Barlaud, G. Aubert, and O. Faugeras. Shape gradients for histogram segmentation using active contours. In *Proc. Int. Conf. Computer Vision*, pages 408–415, Nice, 2003.

[10] S. Joshi. *Large Deformation Diffeomorphisms and Gaussian Random Fields for Statistical Characterization of Brain Submanifolds*. PhD thesis, Washington University, 1997.

[11] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: active contour models. In *Proc. Int. Conf. Computer Vision*, London, June 1987.

[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] S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum. Gradient flows and geometric active contour models. In *Proc. Int. Conf. Computer Vision*, pages 810–815, Cambridge, 1995.

[14] M. Leventon, E. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proceedings of CVPR 2000*, 2000.

[15] R. Malladi, J. Sethian, and B. Vemuri. Shape modelling with front propagation: A level set approach. *IEEE Trans. Pattern Anal. Machine Intell.*, 17:158–175, 1995.

[16] S. Osher and J. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulation. *J. Comput. Phys.*, 79:12–49, 1988.

[17] N. Paragios and R. Deriche. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(3):266–280, 2000.

[18] 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.

[19] 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.

[20] D. Shen, Y. Zhan, and C. Davatzikos. Segmentation of prostate boundaries from ultrasound images using statistical shape model. *IEEE Transactions on Medical Imaging*, 22(4), April 2003.

[21] 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.

[22] A. Tsai, W. Wells, C. Tempany, E. Grimson, and A. Willsky. Coupled multi-shape model and mutual information for medical image segmentation. In *IPMI 2003*, pages 185–197, 2003.

[23] A. Tsai, A. Yezzi, W. Wells, C. Tempany, 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.

[24] G. Turk and J. F. O'Brien. Shape transformation using variational implicit functions. In *Computer Graphics (Proc. SIGGRAPH '99)*, pages 335–342, 1999.

[25] 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.

[26] T. Zhang and D. Freedman. Tracking objects using density matching and shape priors. In *Proc. Int. Conf. Computer Vision*, pages 1056–1062, Nice, 2003.