

# Active Volume Models for Medical Image Segmentation

Tian Shen, *Student Member, IEEE*, Hongsheng Li, *Student Member, IEEE*, and Xiaolei Huang, *Member, IEEE*

**Abstract**—In this paper, we propose a novel predictive model, Active Volume Model (AVM), for object boundary extraction. It is a dynamic “object” model whose manifestation includes a deformable curve or surface representing a shape, a volumetric interior carrying appearance statistics, and an embedded classifier that separates object from background based on current feature information. The model focuses on an accurate representation of the foreground object’s attributes, and does not explicitly represent the background. As we will show, however, the model is capable of reasoning about the background statistics thus can detect when is change sufficient to invoke a boundary decision. When applied to object segmentation, the model alternates between two basic operations: (1) deforming according to current Region of Interest (ROI), which is a binary mask representing the object region predicted by the current model, and (2) predicting ROI according to current appearance statistics of the model. To further improve robustness and accuracy when segmenting multiple objects or an object with multiple parts, we also propose Multiple-Surface Active Volume Model (MSAVM), which consists of several single-surface AVM models subject to high-level geometric spatial constraints. An AVM’s deformation is derived from a linear system based on Finite Element Method (FEM). To keep the model’s surface triangulation optimized, surface remeshing is derived from another linear system based on Laplacian Mesh Optimization (LMO) [26], [27]. Thus efficient optimization and fast convergence of the model are achieved by solving two linear systems. Segmentation, validation and comparison results are presented from experiments on a variety of 2D and 3D medical images.

**Index Terms**—Segmentation, Deformable Models, Active Volume Models, Multiple Surface Models.

## I. INTRODUCTION

Image segmentation is an important task in medical image analysis. The main challenge is to retrieve high-level information from low-level image signals while minimizing the effect of noise, intensity inhomogeneity, and other factors. However, because of the variety and complexity of images, the design of robust and efficient segmentation algorithm is still a very challenging research topic. To address the challenges, model-based methods have been widely used with considerable success. Most noticeable are two types of models: deformable models [1], [2], [12], [21], [36], [39], [43], and statistical shape and appearance models [5], [6], [15]. Compared to local edge-based methods, deformable models have smooth curves or surfaces represent object boundary, which can bridge

over boundary gaps. The image region inside or outside a model surface can be obtained based on the model’s shape and position, so region analysis strategies have been proposed to assist in evolving the model. To further take advantage of prior knowledge, statistical priors can be learned from training examples to capture variations in the shape and appearance of an object of interest.

In this paper, we propose a new deformable model based approach to object boundary extraction. The new model, Active Volume Model (AVM), can integrate constraints from multiple sources, including but not limited to edges, region information, statistical priors, and geometric/spatial priors. We investigate not static priors but dynamic ones learned on-line during model deformation.

In the remainder of the Introduction section, we will review deformable models and previous works that utilize deformable models for segmentation by integrating various image-based or prior constraints. Then we discuss in more detail the novel aspects and contribution of AVM and its multiple-surface extension, MSAVM.

### A. Shape-based Deformable Models

Since the introduction of Snakes [12] by Kass *et al.* in 1988, active contours have been applied to various problems in image processing, such as segmentation, feature extraction, shape modeling, and visual tracking. Representing the model boundary parametrically,  $\mathbf{v}(s) = (x(s), y(s))$ , the snake models are splines with smoothness constraints and influenced by image forces. The original snake model was formulated to minimize the energy function

$$E(C) = \int_0^1 (\alpha |\mathbf{v}_s(s)|^2 + \beta |\mathbf{v}_{ss}(s)|^2 + \lambda P(C(s))) ds, \quad (1)$$

where  $\alpha$ ,  $\beta$  and  $\lambda$  are real positive constants to weight the smoothness constraints and image forces, and  $P(C(s))$  is a potential which is based on some image features, such as intensity, gradient and edges. Other parametric deformable models were proposed to incorporate overall shape constraints [24], [36] and to increase the attraction range of the original Snakes by *Gradient Vector Flow* (GVF) [39].

Another class of deformable models is level set based geometric models [8], [21]. This approach represents curves and surfaces implicitly as the level set of a higher-dimensional scalar function. The evolution of these implicit models is based on the theory of curve evolution, with speed function specifically designed to incorporate image gradient information. Because these models do not need explicit parameterization,

The authors are with the Department of Computer Science and Engineering at Lehigh University, Bethlehem, PA, USA.

Copyright (c) 2010 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

they can handle topology changes very naturally. Hence the level-set approach is commonly used in segmenting multiple objects [10] and achieves good results in tubular structure segmentation [37]. Coupled surface constraints and the dual-front implementation of level set active contours [17] also provide the flexibility of capturing variable degrees of localness in optimization.

One of the geometric models most closely related to the original Snake model is the *Geodesic Active Contour* model [1], whose level set function is

$$E(C) = \int_0^1 g(|\nabla I(C(s))|) |C'(s)| ds, \quad (2)$$

where  $g(|\nabla I|) = \frac{1}{1 + |\nabla I|^2}$ ,

where  $C$  represents the front (zero level set) curve of the evolving level set function. When minimizing the level set function, the front curve deforms along its normal direction  $C''$ , and this speed is controlled by the speed function  $g(|\nabla I|)$ . The speed function depends on the image gradient  $\nabla I$ , and it is positive in homogeneous regions and close to zero at edges. Hence the curve deforms at a velocity proportional to its curvature in homogeneous areas and stops at strong edges.

### B. Integrating More Constraints in Deformable Models

1) *Region information*: Both the original Snake and geodesic active contours rely on image gradient information to deform, so they are sensitive to noise and spurious edges. Both of them are prone to stop at undesirable local minima in their corresponding energy functions. Thus initialization must be chosen very carefully. Typically, the models are initialized close to the object boundary to avoid getting stuck in such local minima. In order to address the limitations and develop more robust models for boundary extraction, region-based parametric and geometric deformable models are proposed and have become increasingly popular.

The basic idea is to deform an active contour using the region statistics information from the interior and exterior of the contour. Compared to edge-based models, region-based models are promising since the region information can help the model overcome many local minima and converge at the desired object boundary. Furthermore, because region-based approaches incorporate image region statistics, which can be considered as global image information, they are less constrained by the initial position, and less sensitive to noise.

A well-known example for the region modeling cost function is the Mumford-Shah functional [25]. The segmentation problem, as formulated by Mumford and Shah, can be defined as follows: given an observed image  $u_0$ , find a decomposition  $\Omega_i$  of  $\Omega$ , where  $\Omega \subset R^2$ , such that the new ‘‘segmented’’ image  $u$  varies smoothly within each  $\Omega_i$ , and discontinuously across the boundaries of  $\Omega_i$ . The simplified case is obtained by restricting the segmented image  $u$  to be piecewise-constant (i.e.,  $u = \text{constant } c_i$  inside each component  $\Omega_i$ ) or piecewise-smooth functions. The problem is often called the ‘‘minimal partition problem’’. In the recent few years, solutions for several simplified cases of the Mumford-Shah functional have

been proposed in the level set framework. In [2], the piecewise-constant function is minimized within the level set framework by Chan and Vese:

$$E(C) = \lambda_1 \iint_{R_i} |I(p) - c_1| dp + \lambda_2 \iint_{R_o} |I(p) - c_2| dp + \mu \text{Length}(C) + \nu \text{Area}(R_i), \quad (3)$$

where  $\lambda_1$ ,  $\lambda_2$ ,  $\mu$  and  $\nu$  are positive constants to balance the contribution of each term,  $R_i$  and  $R_o$  correspond to the interior foreground and exterior background regions of the contour  $C$ ,  $c_1$  and  $c_2$  represent the mean intensities of interior and exterior regions respectively, and  $I(p)$  refers to the image intensity value at pixel  $p$ .

Another approach in [41] is able to segment images that consist of several regions, each characterizable by given statistics such as the mean intensity and variance. The approach in [30] applies a multi-phase level set representation to segmentation assuming piecewise-constant intensity within one region. It is considered as solving a classification problem because it assumes the mean intensities of all region classes are known *a priori*, and only the set of boundaries between regions is unknown. In [38], piecewise-smooth approximations of the Mumford-Shah functional are derived for multiphase cases in a variational level set framework. The optimization of the framework is based on an iterative algorithm that approximates the region mean intensities and level set shape in separate steps. *Geodesic Active Region* [28] is another method that integrates edge and region based modules in a level set framework. In summary of the above approaches, they all assume the distributions within regions to be piecewise-constant, piecewise-smooth, Gaussian, or Mixture-of-Gaussian, which may limit their effectiveness in segmenting objects whose interiors have textured appearance and/or complex multi-modal intensity distributions.

2) *Statistical priors*: Statistical modeling approaches can add constraints from prior off-line learning. Cootes *et al.* proposed methods for building *active shape models* [6] and *active appearance models* [5], by learning patterns of variability from a training set of annotated images. Integrating high-level knowledge, these models deform in ways constrained by the training data and are often more robust in image interpretation. Image interpretation based on a shape-appearance joint prior model can be conducted through image search [6], or by maximizing posterior likelihood of the model given image information in a Bayesian framework [40]. The shape prior knowledge can also be used in an active contour’s framework. The approach in [15] estimates the maximum a posteriori (MAP) position and shape of the object in the image being segmented, based on prior shape information and image information in each step of the level set evolution. Another approach in [9] utilizes kernel principal component analysis (KPCA) and encodes shape priors and image information into two energy functionals entirely described in terms of shapes. In medical imaging, shape priors particularly have been introduced to cardiac segmentation [14], [44], and to deformable models for constrained segmentation of bladder and prostate [7]. One limitation of the statistical models is in the laborious training data collection and annotation process.

3) *Spatial constraints*: Furthermore, in many medical imaging applications, we are interested in simultaneously extracting several boundary surfaces that are coupled in such a way that their relative positions are known and the distances between them are within a specific range. Clearly, integrating this high-level spatial constraint into the segmentation model will further improve accuracy and robustness. A 2D method [35] segments left ventricular Epi- and Endocardial borders using coupled active contours but needs a precise manual initialization. In 3D, Zeng *et al.* [42] incorporated spatial constraints about brain's gray matter and white matter into a level set framework which greatly improved cortex segmentation accuracy. In [18], a graph-theoretic approach detects multiple interacting surfaces by transforming the problem into computing a minimum s-t cut. Deformation of multiple surfaces in [20] has intersurface proximity constraints, which allows each surface to guide other surfaces into place. All of the three 3D methods [18], [20], [42] require manually specifying the expected thickness between surfaces as model-based constraint.

### C. Active Volume Models

In this paper, we propose a novel volumetric deformable model which we term the Active Volume Models (AVM). Compared with active contours [12] and active shape/appearance models [6], the AVM is a generative object model that does not require off-line training but generates useful appearance priors about the object. Using nonparametric statistics of its volumetric interior to represent the foreground object's appearance, the model focuses on an accurate modeling of the foreground object's attributes and embodies object feature statistics learned adaptively as the model deforms toward object boundary. With additional information about the overall image feature statistics, the model also estimates the background statistics so that a Bayes classifier can be applied to predict dynamically the object region (Section II-C). In its object-region prediction framework, the model is capable of probabilistically integrating constraints from multiple visual cues.

The AVM model's shape is represented by a parametric spline curve in 2D [34] or a simplex mesh (or finite-element triangulation) in 3D [33]. Its volumetric interior carries the various visual appearance feature statistics. An advantage of the AVM formulation is that it allows the forces generated from the predicted object region to naturally become part of a linear system (Section III-A), the solution of which gives the deformation of the model to minimize an energy function designed to deform the model toward object boundary. In 3D, to keep the quality of the AVM's surface triangulation from degenerating during deformation, we utilize a surface remeshing technique based on Laplacian Mesh Optimization (LMO) [26], [27].

We further propose the Multiple-Surface AVMs (MSAVM) [32] to segment multiple coupled objects simultaneously. Instead of setting up fixed inter-surface distance constraints during initialization, MSAVM dynamically updates the distance constraints between the interacting surfaces based on current

model surfaces' spatial interrelations. Integrating the dynamic distance constraint strategy with other energy terms based on object region information, MSAVMs are less sensitive to initial positions and yield more accurate segmentation results.

Being a parametric model fitting approach, the AVM's convergence is fast, typically taking no more than 30 iterations. Several factors contribute to this efficiency: (1) AVM focuses on modeling the foreground object and then reasons about the background, instead of modeling the background explicitly, (2) the model's deformations can be solved in a linear system, and (3) multiple external constraints are combined in a probabilistic framework and together contribute to long-range forces coming directly from the predicted object region's boundary.

Compared to traditional parametric deformable models [4], [12], [36], [39], AVM is unique in its ability to probabilistically and adaptively predict the object region, even when the model is still far-away from the object boundary. The predicted object region of interest (ROI) is used to derive effective long-range forces that make the model evolve quickly and overcome local minima caused by noise. As we will show in our experimental results (Table V), the estimated ROI is not only applicable to AVM but useful to other parametric models and level set models. Using the ROI, another parametric model deforming under *T-Snake* [22] like forces converges quickly and accurately; and a level set model, such as *Geodesic Active Contours* (GAC) [1] or *3D Active Contours Without Edge* (ACWE) [2], also gives more accurate segmentation results.

Although the estimated ROI can be integrated in a level set framework, we choose to couple it with a parametric finite element (FE) triangulation model because of efficiency and accuracy concerns. In our experimental evaluation using various noisy medical images (Section IV-B), AVM deforms faster than level set methods such as *Geodesic Active Contours* (GAC) [1] and *Active contours without edges* [2]. Even though there are thousands of vertices (or nodes) on the surface of a 3D AVM, the model's deformation speed is still less than 30 seconds per iteration. Further, the converged 3D AVM model is directly a smooth mesh representing the segmented object surface. Therefore it does not need any post-processing step such as surface reconstruction, as required by level set; it also preserves topology thus does not produce small holes or islands inside, while level set methods often do give results with spurious holes/islands.

The remainder of the paper is organized as follows. In Section II, we introduce the model's representation, the object-region prediction module of AVM, and the dynamic spatial constraint integrated in MSAVM. In Section III, we present how to deform the model, and in 3D how to maintain the mesh's quality when deforming. In Section IV, experimental results of 2D and 3D medical image segmentation using AVM and MSAVM are presented, and the models are compared to other parametric model and non-parametric level set models. Then we conclude the paper and discuss future work in Section V.

## II. THE ACTIVE VOLUME MODELS

An AVM is a deforming solid that minimizes internal and external energies [33], [34]. The internal constraint ensures the



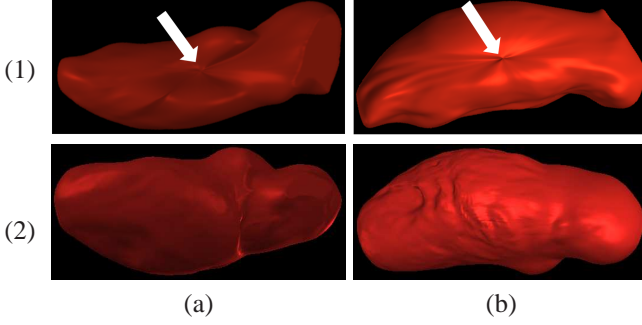


Fig. 1. Comparing heart Left Ventricle segmentation by tessellation and by triangulation models. (1) A tessellation model with  $400 \times 120$  vertices. Notice that the two poles exert very strong forces to drag other vertices which causes the incorrect result. (2) A Finite Element Triangulation model with 40,962 vertices. (a)-(b) Two different views.

model has smooth boundary surface. The external constraints come from image data, prior, and user defined features. In this section, we introduce the representations of AVM, and introduce the novel object boundary prediction module of AVM.

#### A. Explicit Shape Representation of AVM

The shape of a 2D AVM is defined similarly to active contours [12], as a spline curve that has associated elasticity and rigidity. Representing the model boundary parametrically,  $\mathbf{v}(s) = (x(s), y(s))$ , the internal energy term of 2D AVM is defined similarly to Active Contour Models.

$$E_{int} = \int_0^1 (\alpha(s)|\mathbf{v}_s(s)|^2 + \beta(s)|\mathbf{v}_{ss}(s)|^2) ds. \quad (4)$$

In [4], Cohen *et al.* used tessellation to build a 3D finite element method (FEM) surface for 3D deformable model representation. The tessellation can be either a cylinder or an ellipsoid. The model works well in the cylinder case, but fails using an ellipsoid. The problem is that the distribution of vertices on the ellipsoid is in quite an irregular manner. The vertices near the two poles exert strong internal forces to drag other vertices, which causes incorrect results. Figure 1 shows the disadvantage of using a tessellated ellipsoid—strong internal forces generated near the poles adversely affect the model’s deformation.

In order to solve the above problem in tessellated FEM meshes and enable the model to match closely object boundary, 3D AVM adopts a polyhedron mesh as the model representation which places vertices regularly on the model. More specifically, a 3D AVM is considered as an elastic solid and defined as a finite element triangulation  $\Lambda$ , which can consist of tetrahedrons, octahedrons or icosahedrons. Using the finite element method, the internal energy function can be written compactly as:

$$E_{int} = \frac{1}{2} \int_{\Lambda} (\mathbf{B}\mathbf{v})^T D(\mathbf{B}\mathbf{v}) d\Lambda, \quad (5)$$

where  $\mathbf{B}$  is the differential operator for the model vertices  $\mathbf{v}$  on the mesh and  $D$  is the stress matrix (or constitutive matrix). A more detailed explanation can be found in [19].

#### B. Implicit Shape Representation of AVM

The implicit shape representation using level set [21] makes a model’s shape an “image”, which greatly facilitates the integration of boundary and region information [11]. We compute the implicit representation of AVM model’s shape to be used in region-based external energy terms. The Euclidean distance transform is applied to embed implicitly an evolving model’s surface in a higher dimensional distance function. Let  $\Phi : \Omega \rightarrow R^+$  be a Lipschitz function that refers to the distance transform for the model shape  $\Lambda$ . By definition  $\Omega$  is bounded since it refers to the image domain. The shape defines a partition of domain: the region that is enclosed by  $\Lambda$ ,  $[\mathcal{R}_{\Lambda}]$ , the background  $[\Omega - \mathcal{R}_{\Lambda}]$ , and on the model,  $[\Lambda]$ . Given these definitions, the implicit shape representation is considered:

$$\Phi_{\Lambda}(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in \Lambda \\ +ED(\mathbf{x}, \Lambda) > 0, & \mathbf{x} \in \mathcal{R}_{\Lambda} \\ -ED(\mathbf{x}, \Lambda) < 0, & \mathbf{x} \in [\Omega - \mathcal{R}_{\Lambda}] \end{cases}, \quad (6)$$

where  $ED(\mathbf{x}, \Lambda)$  refers to the minimum Euclidean distance between the image pixel/voxel location  $\mathbf{x}$  and the model surface  $\Lambda$ .

#### C. AVM Model’s Object Boundary Prediction Module

Different from most deformable models, one of the novel features of AVM is its adaptive object boundary prediction scheme. The model alternates between two operations: deforming according to the current object boundary prediction, and predicting object boundary according to current appearance statistics of the model. Using this on-line prediction mechanism, the expected object information updates automatically while the model deforms. And long-range external forces are generated from the predicted object boundary to effectively attract the model to deform toward the boundary.

External constraints from any sources can be accounted by probabilistic integration. Let us consider that each constraint corresponds to a probabilistic boundary prediction module, and it generates a confidence-rated probability map to indicate the likelihood of a pixel being: +1 (*object* class), or -1 (*non\_object* class). Suppose we have  $n$  independent external constraints derived from image information, the feature used in the  $k$ th constraint is  $f_k$ .  $L(\mathbf{x})$  denotes the label of a pixel  $\mathbf{x}$ . Our approach of combining the multiple independent modules is applying the Bayes rule in order to evaluate the final confidence rate:

$$\begin{aligned} Pr(L(\mathbf{x})|f_1, f_2, \dots, f_n) = \\ (Pr(f_1, f_2, \dots, f_n|L(\mathbf{x}))Pr(L(\mathbf{x})))/(Pr(f_1, f_2, \dots, f_n)) \\ \propto Pr(f_1|L(\mathbf{x}))Pr(f_2|L(\mathbf{x}))\dots Pr(f_n|L(\mathbf{x}))Pr(L(\mathbf{x})). \end{aligned} \quad (7)$$

For each module, the class-dependent probability  $Pr(f_k|L(\mathbf{x}))$  is estimated based on the AVM model’s current statistics about feature  $f_k$  as well as the overall feature statistics in the image. The derivation is as follows.

$$\begin{aligned} Pr(f_k(\mathbf{x})) = Pr(f_k(\mathbf{x}), L(\mathbf{x}) = +1) + Pr(f_k(\mathbf{x}), L(\mathbf{x}) = -1) \\ = Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1)Pr(L(\mathbf{x}) = +1) \\ + Pr(f_k(\mathbf{x})|L(\mathbf{x}) = -1)Pr(L(\mathbf{x}) = -1), \end{aligned} \quad (8)$$

where  $Pr(f_k(\mathbf{x}))$  is the overall probability of observing  $f_k(\mathbf{x})$  at pixel/voxel location  $\mathbf{x}$  for the  $k$ th feature.  $Pr(f_k(\mathbf{x}))$  is known since it can be estimated using the  $k$ th feature's overall probability distribution in the entire image. Furthermore, assuming the current AVM model is either completely inside or largely overlapped with the foreground object, the model-interior appearance reflects foreground object appearance. Thus, we can approximate the probability of feature  $f_k$  in the foreground,  $Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1)$ , using the feature's distribution in the current AVM model. Both probability density functions,  $Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1)$  and  $Pr(f_k(\mathbf{x}))$ , are estimated using a nonparametric kernel-based density estimation method [11]. They are defined as:

$$Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1) = \frac{1}{|\mathcal{R}_\Lambda|} \iint_{\mathcal{R}_\Lambda} K\left(\frac{f_k(\mathbf{x}) - f_k(\mathbf{r})}{h}\right) d\mathbf{r},$$

$$Pr(f_k(\mathbf{x})) = \frac{1}{|\Omega|} \iint_{\Omega} K\left(\frac{f_k(\mathbf{x}) - f_k(\mathbf{r})}{h}\right) d\mathbf{r}, \quad (9)$$

where  $\mathcal{R}_\Lambda$  is the AVM's model-interior region,  $\Omega$  is the whole image domain,  $\mathbf{r}$  represents a pixel/voxel in the domain of integration,  $K$  is the kernel density function, and  $h$  is the kernel size. In this paper, we define  $K$  as a uniform box and the kernel size as 2. The p.d.fs thus estimated can represent complex multi-modal distributions.

For the pixel/voxel  $\mathbf{x}$ , given the known overall feature distribution,  $Pr(f_k(\mathbf{x}))$ , and the approximated foreground feature distribution,  $Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1)$ , we can now reason about the feature distribution in the background,

$$Pr(f_k(\mathbf{x})|L(\mathbf{x}) = -1) = \frac{Pr(f_k(\mathbf{x})) - Pr(f_k(\mathbf{x})|L(\mathbf{x}) = +1)Pr(L(\mathbf{x}) = +1)}{Pr(L(\mathbf{x}) = -1)} \quad (10)$$

The prior independent of image features,  $Pr(L(\mathbf{x}))$ , in (7) and (10) can be assumed uniform:  $Pr(L(\mathbf{x}) = +1) = 0.5$  and  $Pr(L(\mathbf{x}) = -1) = 0.5$ . Alternatively, spatially-varying prior is another choice. At the end of Section II-F, we define a spatially-varying prior using two distance-related Fermi functions in (II-F) to control the prior values of  $Pr(L(\mathbf{x}) = +1)$  and  $Pr(L(\mathbf{x}) = -1)$ . In Figure 4.(2) and (3), we show the segmentation results with and without the spatially-varying prior, respectively.

In this paper, we show that by considering the pixel intensity  $i(\mathbf{x})$  feature, the above framework generates reasonable estimates of background feature statistics (10) and consistently gives good estimations of the object region on a variety of medical images.

Once the posterior probabilities  $Pr(L(\mathbf{x})|f_1, f_2, \dots, f_n)$  are estimated, we apply the Bayesian decision rule to obtain a binary map  $P_B$  whose foreground represents the object region. That is,  $P_B(\mathbf{x}) = 1$  ( $\mathbf{x}$  belongs to the object) if  $Pr(L(\mathbf{x}) = +1|f_1, f_2, \dots, f_n) \geq Pr(L(\mathbf{x}) = -1|f_1, f_2, \dots, f_n)$ , and  $P_B(\mathbf{x}) = 0$ , otherwise. The probability of error for the decision at pixel  $\mathbf{x}$  is  $\min(Pr(L(\mathbf{x}) = +1|f_1, f_2, \dots, f_n), Pr(L(\mathbf{x}) = -1|f_1, f_2, \dots, f_n))$ .

After obtaining the binary map  $P_B$ , we apply a connected component analysis algorithm on  $P_B$  to retrieve the connected component that overlaps the current model. This connected

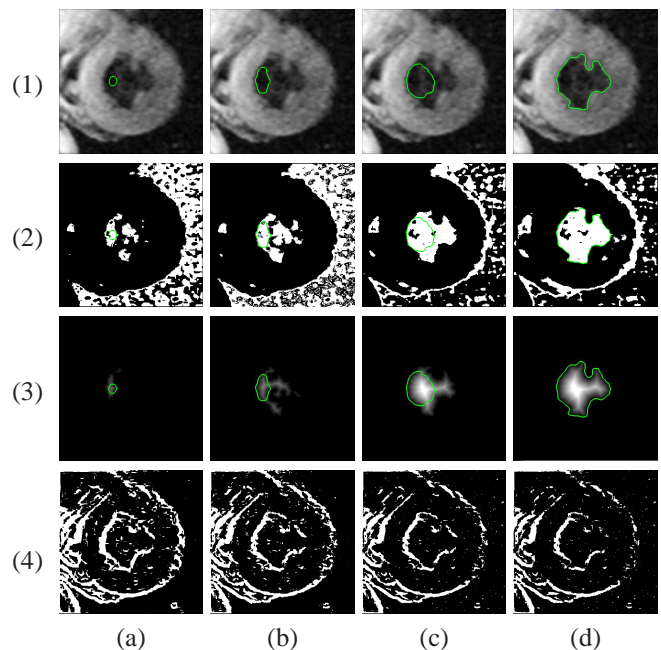


Fig. 2. Left Ventricle endocardium segmentation using a 2D active volume model. (1) The model drawn on the original heart image, (2) the binary map  $P_B$  estimated by the boundary prediction module using intensity feature, (3) distance transform of the ROI boundary,  $\Phi_R$ , (4) the binary edge map estimated by the boundary prediction module using gradient magnitude feature, (a) the initial model, (b) the model after 8 iterations, (c) the model after 18 iterations, and (d) the final converged result after 26 iterations.

region is considered as the current object ROI,  $R$ , and its boundary represents the predicted object boundary. Due to noise, there might be small holes that need to be filled before extracting the boundary of  $R$ .

The progressive ROI updating can be clearly seen from a 2D AVM example in Figure 2. In the example, the ROI (Figure 2.(2)) evolves according to the changing object appearance statistics (estimated by current model's volumetric interior statistics). And the image forces generated by the ROI region energy term (12) deform the model to converge to the object boundary.

Besides intensity feature  $i(\mathbf{x})$ , other features such as image gradient and texture can also be used in our framework. Take image gradient as an example. Since in most applications both object and background regions have similar gradient distributions but object boundaries have a different gradient distribution, following the estimation method above, we can obtain a probability (and binary) map of the object boundary. In Figure 2.(4), we show the predicted object boundary map using gradient magnitude as feature. In our previous work [33], [34], we did allow the integration of the boundary map estimated using image gradient in our model's energy function. However, due to image noise, small spurious edges or gaps exist in the predicted boundary (e.g., Figure 2.(4)). And we found that in many medical image segmentation experiments, having the gradient-based boundary information in our framework did not improve performance. Thus, in this paper, we use only the image intensity feature and its predicted object ROI to derive image forces that deform AVM.

The initialization of an AVM model is very flexible. In 2D,

initialization is done by asking a user to manually place a simple shape, such as a circle with specified center and radius or a polygon defined by a set of user-input points. And in 3D, a user can either initialize the model with a simple shape, for instance an ellipsoid with user-specified radius and orientation, or initialize it with a standard atlas that has similar structure as the object to be segmented. In order for the initial model's appearance to be a good approximation of the foreground object's appearance, the initial model should be placed either completely inside or largely overlapped with the object.

#### D. Definition of Energy Functions

In order to fit to the boundary of an object, the AVM model is driven by the internal smoothness term, and the region data term which is derived from image information (*i.e.*, the estimated ROI). The overall energy function is defined as:

$$E = E_{int} + E_{ext} = E_{int} + E_R, \quad (11)$$

where  $E_{int}$  is defined in (4) for 2D and (5) for 3D. The weight factor between  $E_{int}$  and  $E_{ext}$  is implicitly embedded in  $E_{int}$  ( $\alpha(s)$  and  $\beta(s)$  in (4) for 2D and  $\gamma$  in (20) for 3D).

Given the current model, a Region of Interest (ROI)  $R$  representing the predicted object region can be computed by the prediction module introduced in Section II-C. Let us denote the signed distance transform of the ROI boundary shape as  $\Phi_R$ . Combining  $\Phi_R$  and the current model's implicit shape representation  $\Phi_\Lambda$  in (6), the region-based external energy term is defined as:

$$E_R = \int_{\Lambda} \Phi_\Lambda(\mathbf{v}) \Phi_R(\mathbf{v}) d\Lambda. \quad (12)$$

The multiplicative term provides two-way balloon forces that deform the model toward the predicted ROI boundary. This allows flexible model initializations either overlapping the object or inside the object.

#### E. Topology

Being a parametric model that deforms toward the estimated object region boundary, AVM preserves topology during deformation. However, as we will show in the next Section II-F and in experiments, segmenting an object of arbitrary topology can be handled by initializing multiple AVMs coupled together, one for each structure in the object. Therefore, in this framework, we assume the correct topology of the object is known and then initialize AVMs with that topology to segment the object. We argue that, when segmenting organs with known topology in medical images, this strategy can be more reliable than using models that can freely change topology.

#### F. Multiple-Surface Active Volume Models

In some medical images, there may not be enough information (*e.g.*, contrast) that can be derived from the images to clearly distinguish the object boundaries of interest. This could be due to neighboring objects having very similar tissue types, or due to limitations in medical imaging technology. Therefore, a single surface based deformable model may stop

at local minima or leak out to incorrectly converge at a nearby object's boundary. Often such mistakes can be avoided by considering spatial constraints between multiple objects – for instance, by integrating the spatial constraints in a multiple-surface based deformable model framework [7], [32], [42] and deforming all interacting surfaces simultaneously to extract the object boundaries with better accuracy.

A MSAVM is initialized as several AVMs inside an outer AVM<sup>1</sup>. And each AVM has its own predicted ROI. We introduce a novel adaptive spatial constraint to constrain the multiple model surfaces' deformation. To do that, we add into the energy function a new energy term,  $E_{dist}$ , which is derived from the spatial distance constraint. Then the energy function for the  $i$ th surface of MSAVM is defined as:

$$E = E_{int} + E_R + E_{dist}, \quad (13)$$

where  $E_{int}$  is the same as the internal energy in (11).  $E_R$  is the external energy term derived from the predicted object ROI, which we will define in (17).

We construct two distance-related Gaussian Mixtures functions,  $g_R(dist)$  and  $g_D(dist)$ , to control the weights of  $E_R$  and  $E_{dist}$ , respectively. The functions are defined based on the distance value between surfaces. Let  $i, j$  be surface indices, the mean distance value of the  $i$ th surface to other surfaces is defined as:

$$\overline{dist}_i = \frac{\int_{\Lambda_i} dist(\mathbf{v}) d\Lambda_i}{\int_{\Lambda_i} d\Lambda_i}, \text{ where} \\ dist(\mathbf{v}) = \min_{\forall j, j \neq i} (|\Phi_{\Lambda_j}(\mathbf{v})|), \quad (14)$$

where  $\mathbf{v}$  is a vertex on the  $i$ th surface  $\Lambda_i$ , and  $\Phi_{\Lambda_j}$  is the implicit representation (*i.e.*, signed distance transform) of the  $j$ th surface  $\Lambda_j$ .

The two distance-related Gaussian Mixtures functions of the  $i$ th surface are defined in (15) and illustrated in Figure 3.(1).

$$g_R(dist) = \frac{(e^{-(dist - \overline{dist}_i)^2 / 2\sigma_1^2} + \alpha e^{-(dist - \overline{dist}_i)^2 / 2\sigma_2^2})}{(1 + \alpha)}, \\ g_D(dist) = \kappa \cdot (1 - g_R(dist)). \quad (15)$$

In the above definitions, a greater  $\alpha$  means that  $g_R(dist)$  has a higher lower bound and  $g_D(dist)$  has a lower upper bound.  $\kappa$  is a constant used to control the relative weight of the distance constraint term. In all our experiments,  $\alpha$  and  $\kappa$  are set as 0.5 and 1.0, respectively.  $\sigma_1$  and  $\sigma_2$  ( $\sigma_1 < \sigma_2$ ) are standard deviations of the two Gaussians, which are empirically set as  $0.15 \times \overline{dist}_i$  and  $1.5 \times \overline{dist}_i$  for all the experiments.

Using the distance-constrained weight functions,  $g_R(dist)$  and  $g_D(dist)$ , the energy terms  $E_R$  and  $E_{dist}$  in (13) are defined as:

$$E_R = \int_{\Lambda} g_R(dist(\mathbf{v})) \Phi_\Lambda(\mathbf{v}) \Phi_R(\mathbf{v}) d\Lambda, \quad (16)$$

$$E_{dist} = \int_{\Lambda} g_D(dist(\mathbf{v})) (dist(\mathbf{v}) - \overline{dist}_i)^2 d\Lambda. \quad (17)$$

<sup>1</sup>The outer AVM is required because of an assumption in our spatial distance constraint—that the distances of a surface's points to other surfaces follow a unimodal distribution with mode at the mean distance.



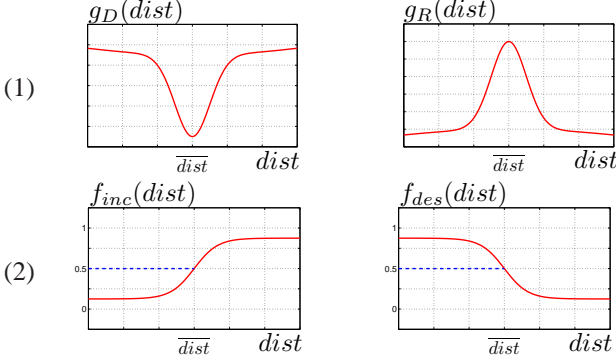


Fig. 3. Functions to control the spatial constraints. (1) Distance-related Gaussian Mixtures functions to balance the contributions of the region term and the spatial constraint term, and (2) distance-related Fermi functions to embed spatial constraints into the ROI boundary prediction modules.

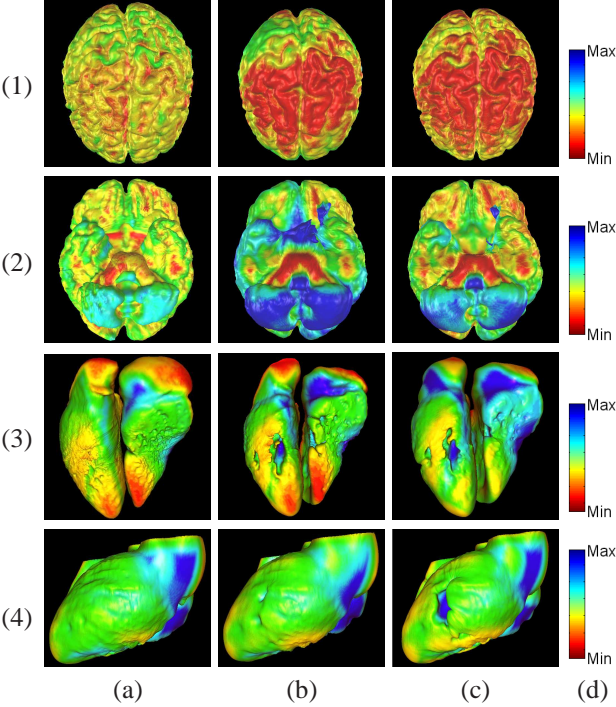


Fig. 4. (a) Results of distance-related Gaussian Mixtures functions, (b) results of distance-related Fermi functions, (c) results without spatial constraint, (d) the color bar used to map the distance information, (1)-(2) Distance-Color (DC) mapping of Gray Matter segmentation, the range for distance between Gray Matter and White Matter is 2~15 voxels, and (3)-(4) DC mapping of heart segmentation, the distance range is 2~25 voxels.

Given a vertex  $\mathbf{v}$  on the  $i$ th surface, its minimum distance value to all the other surfaces  $dist(\mathbf{v})$  can be calculated based on (14). According to  $g_R(dist(\mathbf{v}))$  and  $g_D(dist(\mathbf{v}))$  (Figure 3.(1)), if  $dist(\mathbf{v})$  is close to the  $i$ th surface's mean distance (to other surfaces),  $\bar{dist}_i$ , then  $g_R(dist(\mathbf{v}))$  is large and the region term  $E_R$  makes more contribution toward the surface's local deformation near  $\mathbf{v}$ ; conversely, if  $dist(\mathbf{v})$  is far away from  $\bar{dist}_i$ , which means the local surface near the vertex may be stuck at local minima or have a leakage, the energy term for distance constraint  $E_{dist}$  is given more power to deform the local surface to satisfy the distance constraint and guide it into place.

MSAVM maintains the fast convergence and flexible initialization properties of AVM. Instead of setting a static spatial constraint manually or empirically, after each iteration, MSAVM updates each surface's mean distance value  $\bar{dist}$  based on the spatial relationship among its current model surfaces. The  $g_R(dist)$  and  $g_D(dist)$  functions for each surface are then shifted accordingly to make sure the new  $\bar{dist}$  still corresponds to the centerline of these functions. This unsupervised strategy for online learning of spatial distance constraints between MSAVM's multiple surfaces, coupled with its AVMs' online learning of region appearance statistics, make MSAVM possess both adaptive spatial constraints and adaptive region-based constraints. These properties allow MSAVM to often have even more flexible initialization and faster convergence than the original AVM, as we will show in our experiments (Table VI and Figure 22).

In MSAVM, by designing the weight functions,  $g_R(dist)$  and  $g_D(dist)$ , and constraining a vertex's deformation so that its smallest distance to other surfaces is close to its surface's mean distance, we implicitly make the assumption that the distances (to other surfaces) of a surface's vertices follow a unimodal distribution whose mode is at the mean distance. However, instead of a static unimodal distance distribution, MSAVM adaptively modifies its distance constraint as the mean distances between surfaces change along with model deformation. Thus, the mode (*i.e.*, mean distance) of a surface's distribution changes as the model deforms, and the modes of different surfaces' distributions are separately kept and are usually different from each other. Compared with the distance constraint function in [42], which assumes constant distance between surfaces thus only works well in the case of brain segmentation because of a nearly constant thickness of the cortical layer, the MSAVM's adaptive distance constraint is more general. It can not only be used to segment brain gray matter and white matter, but also has very good performance in extracting ventricles from heart and lungs in the thorax, even though distances between these coupled ventricular surfaces vary greatly. In a heart (or lung) segmentation case, an MSAVM is always initialized with the correct topology, with the outer AVM representing the epicardial (or thorax) surface and the inner AVMs representing the endocardial (or left and right lung) surfaces. Figure 4 shows two segmentation results by Distance-Color (DC) mapping the spatial distance information into color space.

Next, we discuss an alternative way of integrating the spatial constraint in MSAVM, by modifying the pixel label prior  $Pr(L(\mathbf{x}) = +1)$  and  $Pr(L(\mathbf{x}) = -1)$  in the object ROI estimation module (Section II-C, (7 and 10)). Instead of assuming a uniform prior (*i.e.*,  $Pr(L(\mathbf{x}) = +1) = Pr(L(\mathbf{x}) = -1) = 0.5$ ), we define two distance-related Fermi functions (sigmoidal),  $f_{inc}(dist)$  and  $f_{des}(dist)$ , to be the spatially-varying prior. The two functions of the  $i$ th surface are defined in (II-F) and illustrated in Figure 3.(2).

$$\begin{aligned} f_{inc}(dist) &= 1/(1 + e^{-s(dist - \bar{dist}_i)}), \\ f_{des}(dist) &= 1/(1 + e^{s(dist - \bar{dist}_i)}), \end{aligned} \quad (18)$$

where  $s$  is a positive constant to control the steepness of

the step near  $\overline{dist}_i$ .  $s$  is set as 0.2 for the experiments shown in Figure 4.(b). The Fermi functions are used to modulate the label (+1 or -1) prior at every voxel when estimating the  $i$ th surface's object ROI. For a voxel  $\mathbf{x}$ , let its minimal distance value to all surfaces other than the  $i$ th one be  $dist(\mathbf{x})$ . Then, if the  $i$ th surface is an inner surface, we set  $Pr(L(\mathbf{x}) = +1) = f_{inc}(dist(\mathbf{x}))$  and  $Pr(L(\mathbf{x}) = -1) = f_{des}(dist(\mathbf{x}))$ ; otherwise, if the  $i$ th surface is the outer surface of MSAVM, we set  $Pr(L(\mathbf{x}) = +1) = f_{des}(dist(\mathbf{x}))$  and  $Pr(L(\mathbf{x}) = -1) = f_{inc}(dist(\mathbf{x}))$ . When the model surfaces deform and the mean distances between them change, the  $f_{inc}(dist)$  and  $f_{des}(dist)$  functions for each surface are then shifted accordingly to make sure the new  $\overline{dist}$  still corresponds to the centerline of these functions.

Comparing the above spatial prior using Fermi functions with the one using the Gaussians Mixture functions  $g_R(dist)$  and  $g_D(dist)$ , we found that integrating the spatial prior using the Gaussian Mixtures functions and the additional energy term  $E_{dist}$  is more effective than using the Fermi functions. We reason that this is because the Fermi functions are used to weight the prior probability of voxels being inside the *object* or *non\_object*, and if in some region, the conditional probability derived from image information is not good enough, the effect of the spatial prior in the combined posterior probability map can be greatly reduced. For instance, in Figure 4.(b)(3)-(4), the holes on the Fermi-based MSAVM model developed is due to the model being stuck at a local minima caused by noise in the posterior probability map and the estimated object ROI. The results obtained using the Fermi-based spatially varying prior (Figure 4.(b)) do not show marked improvement over the results obtained using the uniform prior (Figure 4.(c)). Therefore, in all our experiments, we use the uniform prior for object ROI prediction (Section II-C), and we use the GM functions  $g_R(dist)$  and  $g_D(dist)$  and the additional energy term  $E_{dist}$  for integrating the adaptive spatial distance constraint in MSAVM.

### III. MODEL DYNAMIC DEFORMATION

#### A. The Model's Deformation

Minimization of the AVM's energy function can be achieved by solving the following linear system

$$A \cdot V = L_V, \quad (19)$$

where matrix  $A$  is symmetric and positive definite, and its size equals the number of control vertices.  $V$  is the vector of vertices on the curve (2D) or surface (3D) of AVM.  $L_V$  is the external force vector of the control vertices.

For 2D AVM,  $A$  is the pentadiagonal banded matrix derived from the internal energy term, (4). The detailed derivation and the full form of  $A$  can be found in the Appendix of [12]. For 3D AVM, let us denote the FE triangulation mesh  $\Lambda$  as a graph  $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ , with vertices  $\mathbf{V}$  and edges  $\mathbf{E}$ . Then  $A$  is the stiffness matrix derived from (5) by using a continuous piecewise linear basis function. More specifically, given the basis function  $\phi_i$  at the  $i$ th vertex  $\mathbf{v}_i$  and the number of vertices  $|\mathbf{V}|$ , the model's FE mesh is represented by  $\Lambda(\mathbf{x}) = \sum_{i=1}^{|\mathbf{V}|} \mathbf{v}_i \phi_i(\mathbf{x})$ . The continuous piecewise linear

basis function  $\phi_i$  is defined like a "tent", such that it has a positive value at  $\mathbf{v}_i$  and zero value at all other vertices:

$$\phi_i(\mathbf{v}_j) = \delta_{ij} \equiv \begin{cases} \gamma & i = j \\ 0 & i \neq j \end{cases}, \quad (20)$$

where  $\gamma$  is the positive value at  $\mathbf{v}_i$ . (In practice,  $\gamma$  controls the smoothness of the model mesh; larger  $\gamma$  leads to higher internal energy and smoother mesh.) And, the  $i$ th row and  $j$ th column element  $a_{ij}$  of the matrix  $A$  is defined as:

$$a_{ij} = \begin{cases} \int_{\Lambda} \nabla \phi_i(\mathbf{x}) \nabla \phi_j(\mathbf{x}) d\Lambda & i = j \text{ or } (\mathbf{v}_i, \mathbf{v}_j) \in \mathbf{E} \\ 0 & \text{otherwise} \end{cases}. \quad (21)$$

For the external force vector  $L_V$ , the  $i$ th element  $l_{v_i}$  corresponds to the external force on the  $i$ th vertex  $\mathbf{v}_i$  based on (12), which is defined as:

$$l_{v_i} = -\nabla(\Phi_{\Lambda}(\mathbf{v}_i)\Phi_R(\mathbf{v}_i)) = -\Phi_R(\mathbf{v}_i) \cdot \nabla\Phi_{\Lambda}(\mathbf{v}_i). \quad (22)$$

If a model vertex  $\mathbf{v}$  is far away from the ROI's boundary,  $|\Phi_R(\mathbf{v})|$  has a greater value. Thus, according to (22), the external force on this vertex is stronger, which can deform the vertex toward object boundary quicker.

The linear system in (19) can be solved by using finite differences [4]. After initializing the 3D AVM, the final converged result can be obtained iteratively based on the following equation:

$$(V^t - V^{t-1})/\tau + A \cdot V^t = L_{V^{t-1}}, \quad (23)$$

where  $V^{t-1}$  is the current AVM's vertex vector and  $\tau$  is the time step size. (23) can be written in a finite differences formulation, which yields

$$\begin{aligned} M \cdot V^t &= V^{t-1} + \tau L_{V^{t-1}}, \\ M &= (I + \tau A). \end{aligned} \quad (24)$$

For MSAVM, each surface is treated with an independent linear system. For the  $i$ th surface,

$$A_i \cdot V_i = L_{V_i}, \quad (25)$$

where  $A_i$  is the stiffness matrix defined the same way as 3D AVM,  $V_i$  is the vector of vertices of the  $i$ th surface, and  $L_{V_i}$  is the corresponding external force vector. The difference from AVM is that the spatial constraint is a part of the external force vector (17) in MSAVM. An element  $l_{\mathbf{v}}$  of  $L_{V_i}$  is defined as:

$$\begin{aligned} l_{\mathbf{v}} &= -g_R(dist(\mathbf{v})) \cdot \nabla(\Phi_{\Lambda}(\mathbf{v})\Phi_R(\mathbf{v})) \\ &\quad - g_D(dist(\mathbf{v})) \cdot \nabla((dist(\mathbf{v}) - \overline{dist}_i)^2). \end{aligned} \quad (26)$$

Thus deforming MSAVM can be achieved by solving several independent linear systems.

#### B. Model Shape Optimization During Evolution

One of the advantages of AVM is the flexible initialization. The final converged results can be very different from the initial models. However, the flexible initialization also has negative effects. If the initialization is too far away from the real object, the model will deform rapidly in each iteration, which causes the quality of the simplex mesh to degrade sharply. In 2D, this problem can be easily solved by reparameterizing the



curve, which is commonly used in 2D deformable models, such as [39] and [34]. However, such reparameterization in 3D is more difficult since the 3D meshes' structure and connectivity are much more complicated than 2D curves. The straightforward solution in 3D is adopting a stricter smoothness control and a smaller step size. This strategy can partially solve the problem but will lead to some other problems, such as delaying the convergence time and causing the model to get stuck at local minima. To overcome this problem, we use Laplacian Mesh Optimization (LMO) [26], [27] to maintain the surface mesh quality after each iteration. LMO is a remeshing technique that optimizes triangle shapes and smoothes a triangular mesh while preserving geometric features on the mesh. It is guided by vertex Laplacians and relocates vertices so that they approximate prescribed Laplacians and positions in a weighted least-squares sense. Its non-iterative solution is well-defined and can be computed efficiently using optimized sparse linear solvers. Using LMO, we remesh the AVM model after each iteration of deformation. In this way, the model's mesh quality is maintained at a high level. Thus we can use a relatively weak smoothness control and larger step size to deform the model. Based on our experiments, AVM with LMO remeshing converges faster than the AVM without remeshing.

### C. Steps to deform AVM

In summary, we adopt the following steps to deform the AVM toward matching the desired object boundary.

- 1) Initialize the AVM, set up matrix  $A$  in (19) and step size  $\tau$  in (24).
- 2) Compute  $\Phi_\Lambda$  in (6) based on the current model; predict the object ROI  $R$  by applying the Bayesian Decision rule to binarize the current estimated *object* probability map (Section II-C), and compute the signed distance transform of the ROI's boundary,  $\Phi_R$ . For each surface in MSAVM, update  $\overline{dist}$  based on (14) and shift distance-related control functions according to  $\overline{dist}$ .
- 3) Deform the model according to (24). Apply LMO mesh optimization to the model.
- 4) For 3D AVM and MSAVM, adaptively reduce the smoothness control  $\gamma$  in (20) and decrease the step size  $\tau$  in (24). For 3D AVM, update its stiffness matrix  $A$ . For MSAVM, update the stiffness matrices for all the surfaces.
- 5) Repeat steps 2-4 until convergence. The convergence criterion is that the maximum movement of the vertices is less than 3 pixels/voxels.

In Step 4, using a relatively strong internal smoothness constraint at the very beginning helps reduce the effect of image noise on model deformation and enables the model to deform quickly toward object boundary. Then, by gradually decreasing the values of smoothness control parameters ( $\alpha(s)$  and  $\beta(s)$  in 2D and  $\gamma$  in 3D) and the step size  $\tau$ , the model can extract more details on the object boundary as it gets close to convergence. This scheme to adaptively change parameters is particularly useful in 3D, see Table II. Given the initial parameter values  $\gamma_{beg}$  and  $\tau_{beg}$ , and the lower bound

TABLE I  
2D AVM MODEL'S PARAMETER SETTINGS.

	$\alpha(s)$ in (4)	$\beta(s)$ in (4)	$\tau$ in (24)
Figure 5	30	30	0.1
Figure 6	20	20	0.05
Figure 18.2(a), (c)-(d)	30	30	0.1
Figure 18.2(b)	25	25	0.03

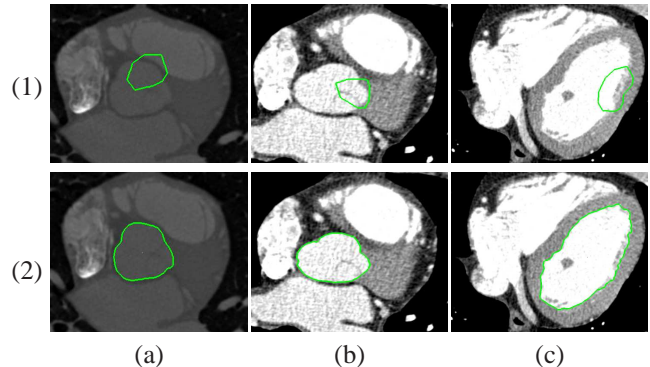


Fig. 5. The segmentation results on cardiac CT images. (1) The initial models, (2) the final converged results after (a) 7, (b) 5 and (c) 6 iterations.

values  $\gamma_{end}$  and  $\tau_{end}$ , we decrease  $\gamma$  and  $\tau$  by  $\frac{\gamma_{beg}-\gamma_{end}}{25}$  and  $\frac{\tau_{beg}-\tau_{end}}{25}$  per iteration until the iteration number reaches 25 or the model converges. If the model has not converged after 25 iterations, the lower bound values,  $\gamma_{end}$  and  $\tau_{end}$ , are used until convergence.

## IV. EXPERIMENTAL RESULTS

In this section, we present experiments of using AVM and MSAVM for segmenting organs in images of the human body, evaluated the model's performance and compared it with several other segmentation models. All the experiments were tested on a PC workstation with an Intel Duo Core 3GHz E6850 processor.

### A. Results of AVMs

1) *Results of 2D AVMs:* We have applied 2D AVM to extract boundaries in various medical images. We first tested the model by using a set of cardiac CT images. Considering that the CT images give relatively clear contrasts, we selected a large step size. Table I gives 2D parameters  $\alpha(s)$  and  $\beta(s)$  for smoothness, and  $\tau$  for step size.

We also used a set of ultrasound images to test the robustness of the model to speckles and noise. Because of the nature of ultrasound images, there is no clear contrast edges to indicate the object boundary. In this case, the region-based properties of the AVM become very important. Figure 6 shows 2D AVM segmentation results for several ultrasound images, in which there are noisy gradients and spurious edges inside the objects of interest. In this case, the object prediction represented by the ROI is the only reliable information that enabled the finding of object boundary. Figure 5 and 6 also show that model initialization can either partially overlap the object or inside the object. The model is able to expand or shrink to converge to the boundary of the object that dominates the initial model appearance.

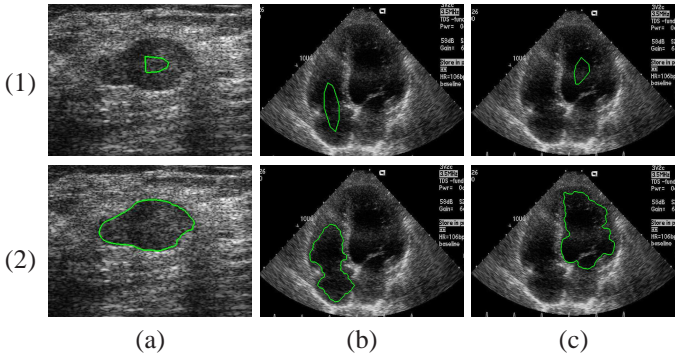


Fig. 6. Segmentation results on ultrasound images. (1) The initial model, (2) the final converged result after (a) 21, (b) 35 and (c) 23 iterations.

TABLE II  
3D AVM AND MSAVM MODELS' PARAMETER SETTINGS.

	$\gamma$ in (20)	$\tau$ in (24)
3D AVM for Lung	40 ~ 15	0.07 ~ 0.015
3D AVM for Left Ventricle	40	0.07 ~ 0.015
3D AVM for Brain	60 ~ 15	0.06 ~ 0.01
MSAVM for lung, heart and brain	35 ~ 20	0.07 ~ 0.01

2) *Results of 3D AVM*: We tested 3D AVM to extract boundary surfaces of the lung, left ventricle, Brain Gray Matter and White Matter from 3D volumetric image stacks. Since the model is robust to noise, all the image data used in the testing are the original datasets without any preprocessing, e.g., smoothing and morphological operations.

First, we used a 3D AVM model to segment the right lung using chest CT data provided by McLaughlin *et al.* [23]. The model was initialized as an a simple Finite-Element-Triangulated ellipsoid whose long axis is perpendicular to the axial image plane. Notice that the model only partially overlaps the predicted ROI. External forces from the adaptively changing ROI's boundary deform the model to finally converge on the object boundary. Figure 7 shows the initial model and the final converged result.

Then we tested 3D AVM model in segmenting the left ventricle in a CT stack. The model was again initialized as an ellipsoid and was placed in the stack after a set of rotation operations so that the ellipsoid's long axis roughly aligned with the left ventricle's long axis. Some boundary condition was also specified so that the model did not deform beyond the  $z$ -range defined by the top and bottom slices of the stack. Figure 8 shows the initial model and the final converged result.

We ran an experiment to examine the effects of parameter setting ( $\gamma$  in (20) and  $\tau$  in (24)) and model initialization on the

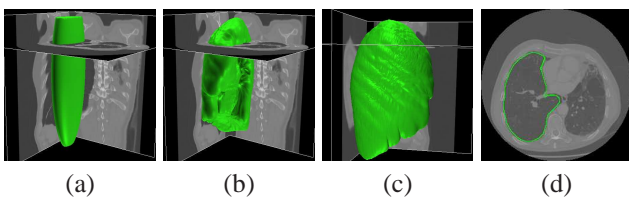


Fig. 7. Right Lung segmentation using a 3D AVM with an FE triangulation mesh consisting of 32,770 control vertices. (a) The initial model, (b) the final converged result after 27 iterations, (c) a different view of the final result, and (d) the final converged result in a 2D slice.

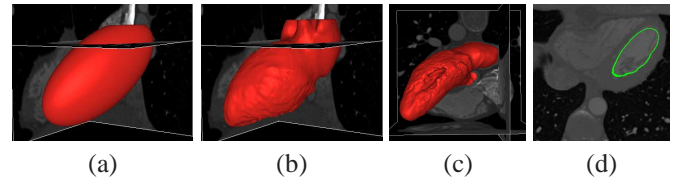


Fig. 8. Left Ventricle segmentation using a 3D AVM with 65,538 control vertices. (a) The initial model, (b) the final converged result after 21 iterations, (c) a different view of the final result, and (d) the final converged result in a 2D slice.

results of 3D AVM. In Figure 9.(2)-(4), we show the model's segmentation results using three different parameter settings, and in Figure 9.(a)-(c) the results using three different model initializations. There exist only slight differences among the results. The average running time of the 9 segmentation cases is 514 seconds. The average *Dice Similarity Coefficient* (DSC) [29] accuracy value is 92.1%, and the variance of the 9 DSC values is 0.88%. Since the results are not very sensitive to parameter setting and model initialization, we use the same parameter setting for all the lung segmentation examples, another parameter setting for all the heart segmentation examples, and so on. The actual parameter settings for various 3D AVM and MSAVM experiments are listed in Table II; the adaptive parameter change scheme outlined in Section III-C is adopted in all cases.

Next, we tested 3D AVMs using more datasets: 10 volumetric cardiac CT datasets, and 15 human lung CT datasets. In the heart datasets, 3D AVMs are used to segment the left ventricle, and in the lung datasets, 3D AVMs are used to segment the left and right lungs. Figure 10 and Figure 11 show the initial models and final converged results on sample heart and lung segmentation cases. The average DSC value of heart segmentations is 90.8%, and the average DSC value of lung segmentations is 95.5%.

Since there are thousands of vertices on the FE triangulation surface, an AVM model can extract very detailed information on object surfaces. This advantage can be seen from the human brain Gray Matter and White Matter segmentation example in Figure 12. The 3D simulated MRI brain images are provided by BrainWeb [3]. The MRI stack is of  $T_1$  modality, has 1mm slice thickness, 3% noise level and 20% intensity non-uniformity (INU). Both models (one for Gray Matter and one for White Matter) were initialized as ellipsoids. As the models were getting closer to the approximated object boundary, the models decreased the smoothness constraint automatically based on the deformation strategy. Then a lot of details on the object surfaces appeared on the models.

In terms of efficiency, the running time of 3D AVM depends on the number of vertices on the model and the size of the 3D image volume. For the experiments shown in Figure 9.(2)-(4), using a model with 40,962 vertices, the average running time of one iteration is 22.3s. Finding the region of model interior according to the mesh position and computing the model's distance transform  $\Phi_\Lambda$  takes 7.3s (32.7%); generating the object ROI  $R$  and its distance transform  $\Phi_R$  takes 12.2s (54.7%); and the remaining steps, including solving the linear system and remeshing of the model surface, take 2.8s (12.6%).



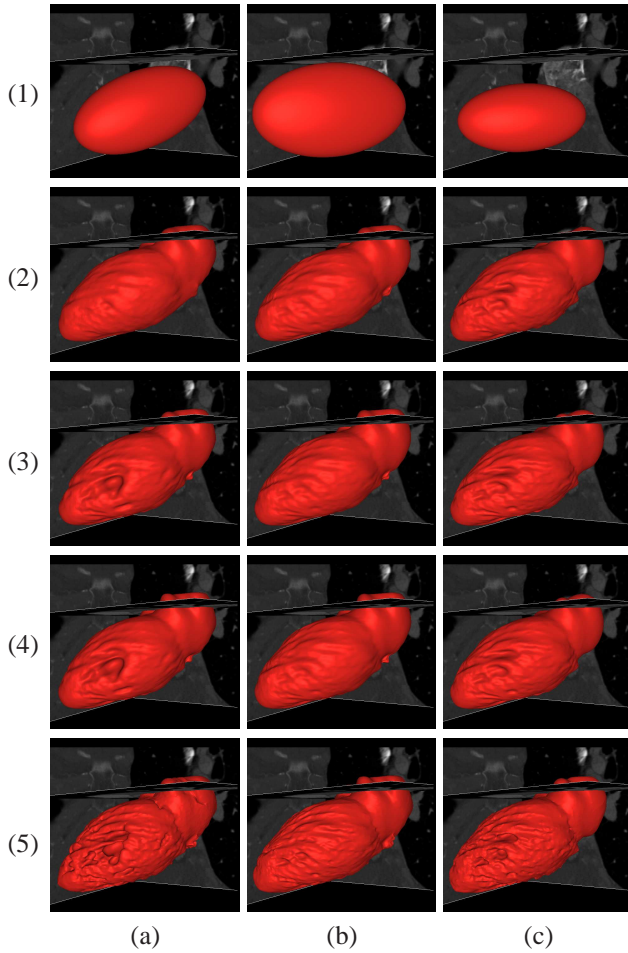


Fig. 9. Segmentation results using three different parameter settings (2, 3 and 4), three different model initializations (a, b and c), and using a mesh with more control vertices (5). (1) Initial models with 40,962 vertices, (2) parameters  $\gamma$  in Equation (20) and  $\tau$  in Equation (24) are set as:  $\gamma = 30$  and  $\tau = 0.05$ , (3)  $\gamma = 40$  and  $\tau = 0.09$ , and (4)  $\gamma = 50$  and  $\tau = 0.2$ . (2)-(4) Average running time is 514s, and average DSC value is 92.1%. (5) The model with 163,842 vertices,  $\gamma = 30$  and  $\tau = 0.05$ , the average running time is 1124s, and average DSC value is 92.8%.

In Figure 9.(5), using a model with 163,842 vertices, the average running time of one iteration is 48.4s. Generating the object ROI and computing  $\Phi_R$  takes roughly the same amount of time as before (12.0s, 24.8%); finding the model-interior region and computing  $\Phi_\Lambda$  take longer (25.6s, 52.9%); and solving the linear system and remeshing also take longer time (10.8s, 22.3%). Since the models in Figure 9.(5) have more control vertices, they can recover more boundary details. And the average DSC value of Figure 9.(5) is 92.8%, slightly higher than that of Figure 9.(2)-(4).

The memory requirement of 3D AVM depends mainly on the image volume size, since the original image volume, the ROI binary map and its distance transform, and the model-interior binary map and its distance transform need to be stored in the memory. The size of the model mesh and the matrices in the linear system affect little the memory usage. More specifically, given the volume sizes (Table IV), brain segmentation takes around 280MB, heart and lung segmentations take around 660MB. Currently all experiments use the original whole image volumes without cropping. We expect

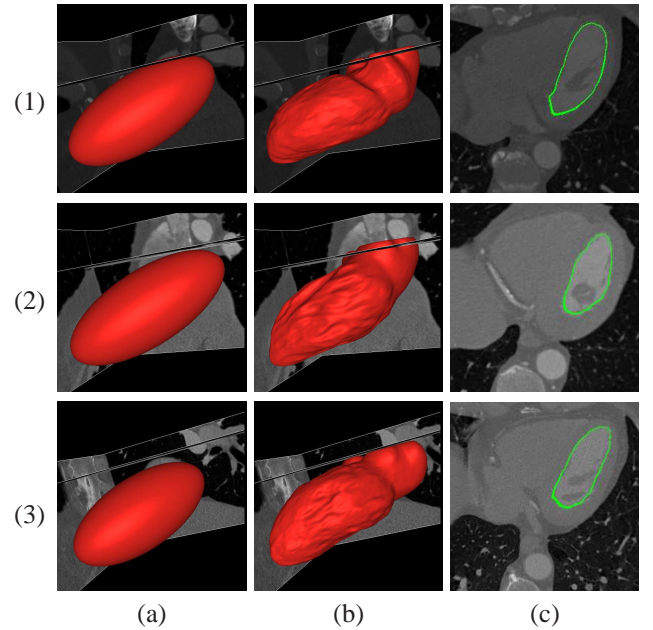


Fig. 10. Left ventricle segmentation results using 3D AVM (40,962 vertices) on 3 different 3D cardiac CT datasets. (a) The initial models, (b) the final converged results, and (c) the final converged results in 2D slices.

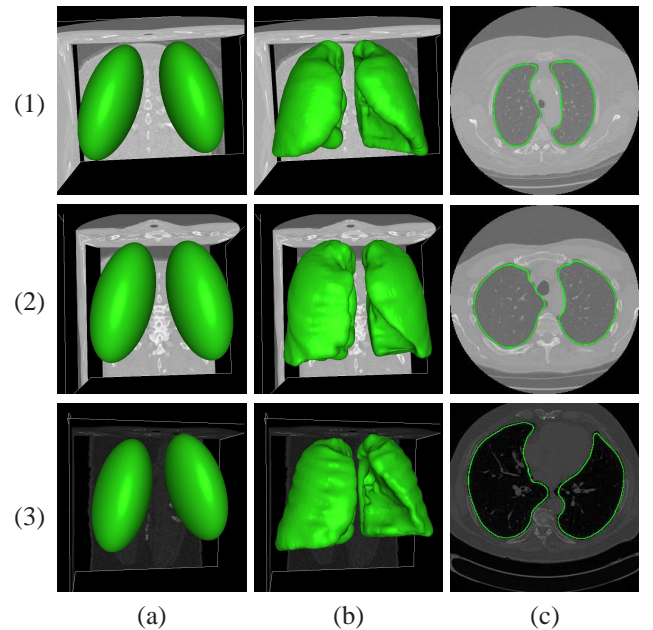


Fig. 11. Lung segmentation results using 3D AVM (40,962 vertices) on 3 different 3D lung CT datasets. (a) The initial models, (b) the final converged results, and (c) the final converged results in 2D slices.

that the memory usage and running time will be lower if we use a smaller subvolume containing the object of interest for segmentation, instead of using the whole volumes.

3) *Results of MSAVM*: We applied MSAVM to segment various coupled organ surfaces in volumetric medical images. An MSAVM is initialized as several inner AVMs in an outer AVM. The AVMs will deform simultaneously and in a coordinated way to fit the object boundaries. First, we put an MSAVM into a thorax CT stack to segment the lungs of a lung cancer patient. The model was initialized as one outer



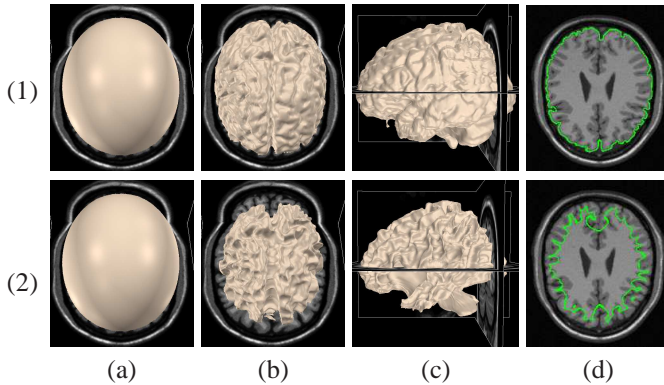


Fig. 12. Brain Gray Matter and White Matter segmentation using 3D AVM and each model has 131,074 control vertices. (1)(a) The initial model of Gray Matter, (1)(b)-(1)(c) the final converged result after 24 iterations, (2)(a) the initial model of White Matter, (2)(b)-(2)(c) the final converged result after 21 iterations, and the final converged results of (1)(d) Gray Matter and (2)(d) White Matter in 2D slices.

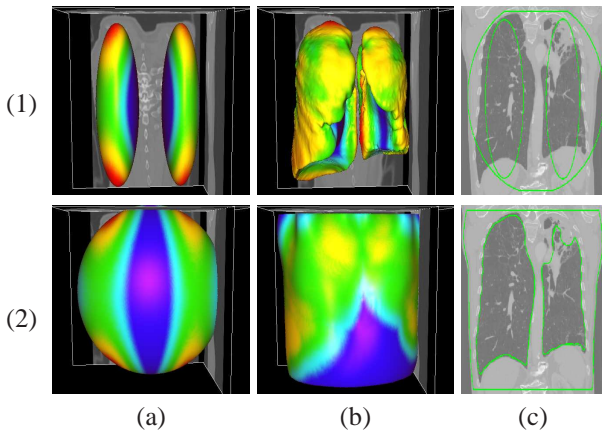


Fig. 13. Distance-Color (DC) mapping of lung surfaces segmentation using MSAVM, the distance range is 3-45 voxels, and each model has 32,770 vertices. (1) The inner surfaces, (2) the outer surface, (a) the initial model, (b) the final converged result after 21 iterations, (1)(c) the initial model in a 2D slice, and (2)(c) the converged result in the 2D slice.

ellipsoid around the thorax and two inner ellipsoids whose long axes are perpendicular to the axial image plane. Figure 13 shows the 3D DC mapping images during deformation. A 2D coronal projection view is also included in Figure 13 to show a cross section of the initial model and converged result.

Then we experimented with the model on segmenting heart surfaces in a cardiac CT stack. The MSAVM model was initialized as three ellipsoids: one for the epicardial surface of the myocardium, one for the endocardial surface of the left ventricle, and a third one for the endocardial surface of the right ventricle. To keep the model from deforming into connected structures with similar intensities (e.g., atria), we specified some boundary condition for each of the three model surfaces so that the surface did not deform beyond its top- and bottom-slice thresholds. Figure 14 and Figure 15 show the deformation steps of the heart MSAVM from two 3D viewpoints. 2D sagittal and coronal projection views are also provided in Figure 14.(c) and Figure 15.(c). Due to the structures such as papillary muscles inside the left ventricle, it would be difficult for a single surface deformable model to reach the desired boundary without supervised learning priors.

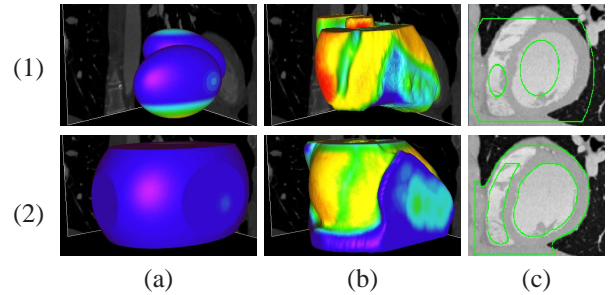


Fig. 14. Distance-Color mapping of heart segmentation using MSAVM viewed from the right, the distance range is 2-25 voxels, and each model has 65,538 vertices. (1) The inner surfaces, (2) the outer surface, (a) the initial model, (b) the final converged result after 24 iterations, (1)(c) the initial model in a 2D slice, and (2)(c) the converged result in the 2D slice.

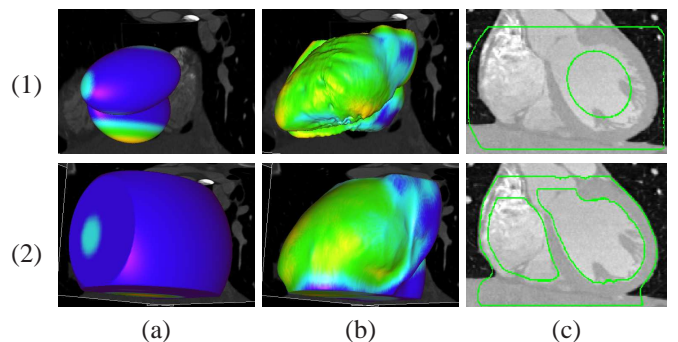


Fig. 15. Distance-Color mapping of heart segmentation using MSAVM viewed from the left, the distance range is 2-25 voxels, and each model has 65,538 vertices. (1) The inner surfaces, (2) the outer surface, (a) the initial model, (b) the final converged result after 24 iterations, (1)(c) the initial model in a 2D slice, and (2)(c) the converged result in the 2D slice.

However, deforming according to the on-line predicted object boundary and spatial constraints, MSAVM can overcome the local minima and extract accurately the multiple cardiac surfaces.

MSAVM can also be used to segment Brain Gray Matter and White Matter simultaneously. In Figure 16, the model was initialized as one outer and one inner ellipsoids. The 3D simulated MRI brain images provided by BrainWeb was again used. Figure 16 shows the DC mapping of the model deformation progress.

Figure 17 also demonstrates some results using MSAVM for lung and heart segmentations. Notice that even though the initializations were far away from the desired boundary, MSAVM still reliably yielded accurate results.

## B. Comparison and Evaluation

1) *Comparison between AVM and other deformable models*: To evaluate the performance of 2D AVMs, we tested the model using MRI and CTA images. Figure 18 shows segmentation results for a variety of medical images. For all the images shown in Figure 18, we presented comparison between 2D AVM, the *Gradient Vector Flow* (GVF) model [39], and the level-set based *Active Contours without Edges* (ACWE) [2]. Based on the running time (to converge) in Table III and the segmentation results in Figure 18, 2D AVM is more efficient than ACWE, and achieves more accurate results than GVF and ACWE. 2D AVM and GVF produce smooth

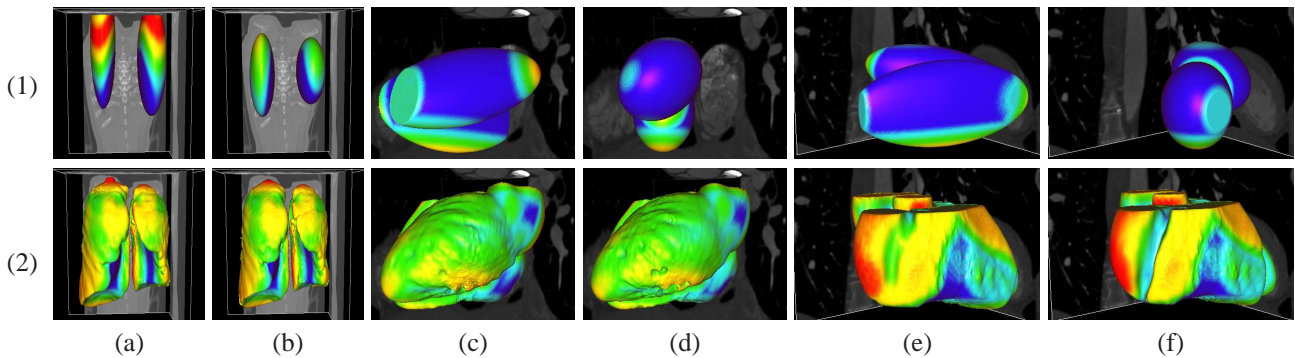


Fig. 17. Different initializations of MSAVM in lung and heart segmentation examples. (a)-(b) Each model has 32,770 vertices, (c)-(f) each model has 65,538 vertices, (1) initial models and (2) final converged results.

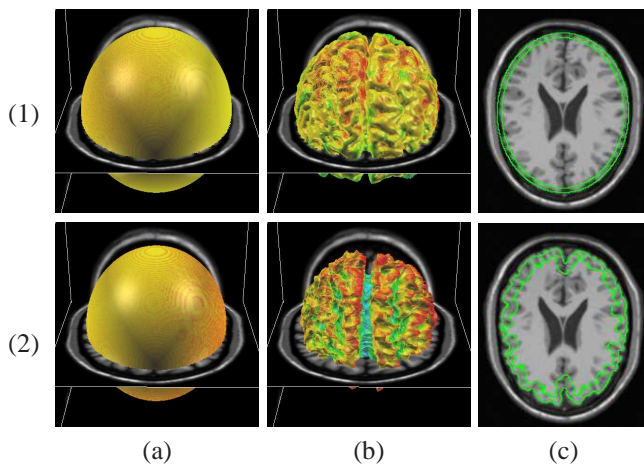


Fig. 16. Distance-Color mapping of brain segmentation using MSAVM, the distance range is 2-15 voxels, and each model has 131,074 vertices. (1) The Gray Matter, (2) the White Matter, (a) the initial model, (b) the final converged result after 24 iterations, (1)(c) the initial model in a 2D slice, and (2)(c) the converged result in the 2D slice.

TABLE III  
RUNNING TIME AND NUMBER OF ITERATIONS FOR FIGURE 18

Model	Case	1	2	3	4	5
2D AVM	Time	8.56s	29.19s	37.88s	6.60s	5.33s
	Iterations	10	26	31	8	9
GVF [39]	Time	7.84s	18.41s	5.32s	9.17s	6.49s
	Iterations	80	40	30	70	15
ACWE [2]	Time	253.5s	185.8s	15.1s	16.5s	12.9s
	Iterations	1600	800	200	100	100

boundaries directly while the ACWE result contains small holes and islands.

Using the same initializations, we compared 3D AVM with level set based 3D *Geodesic Active Contours* (GAC) [1], 3D *Active Contours Without Edge* (ACWE) [2] and 3D *Level Set Evolution Without Reinitialization* (LSEWR) [16] by measuring the running times and validating the segmentation results using expert ground truth markings. Figure 19 shows the final triangulation surface of 3D AVM, compared with the final converged results of GAC, ACWE and LSEWR after surface reconstruction. Table IV presents DSC values and running times for various experiments. One can see that 3D AVM is particularly good for segmenting heart and lungs; its running time is lower than all three level set methods while its

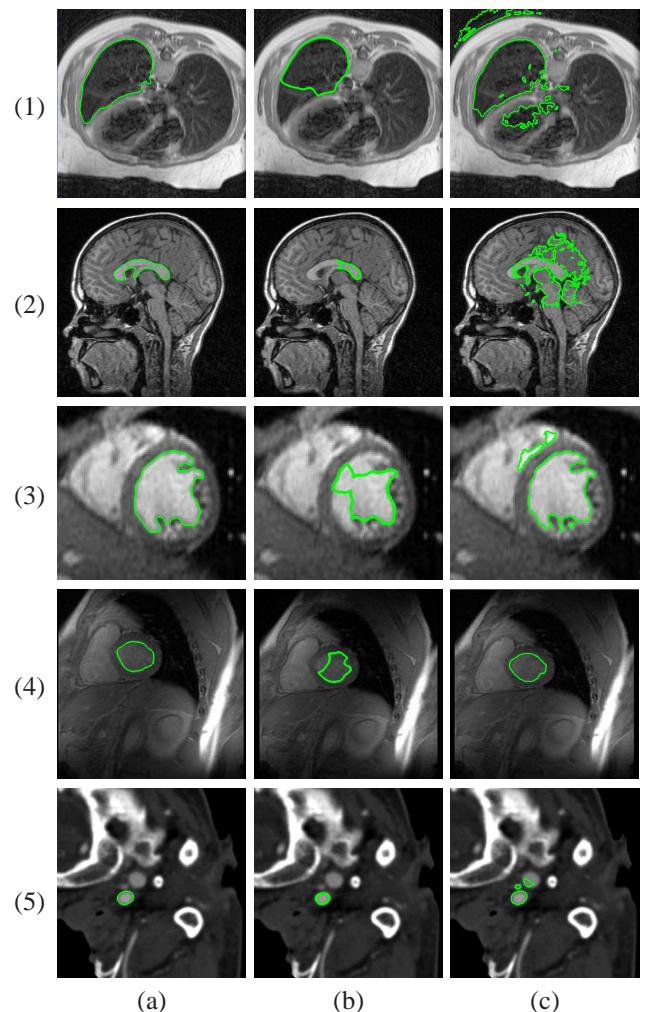


Fig. 18. 2D Segmentation results on MRI and CTA images. (a) Final converged results obtained by 2D AVM after (1) 10, (2) 26, (3) 31, (4) 8, (5) 9 iterations, (b) results from Gradient Vector Flow model after (1) 80, (2) 40, (3) 30, (4) 70, (5) 15 iterations, (c) results from Active Contours Without Edges after (1) 1600, (2) 800, (3) 200, (4) 100 and (5) 100 iterations.

accuracy (*i.e.*, DSC value) is higher. For the brain case, AVM takes longer time than ACWE and LSEWR, mainly because of the need for more iterations and decreased smoothness parameter values in order to capture details on the Gray Matter and White Matter surfaces.



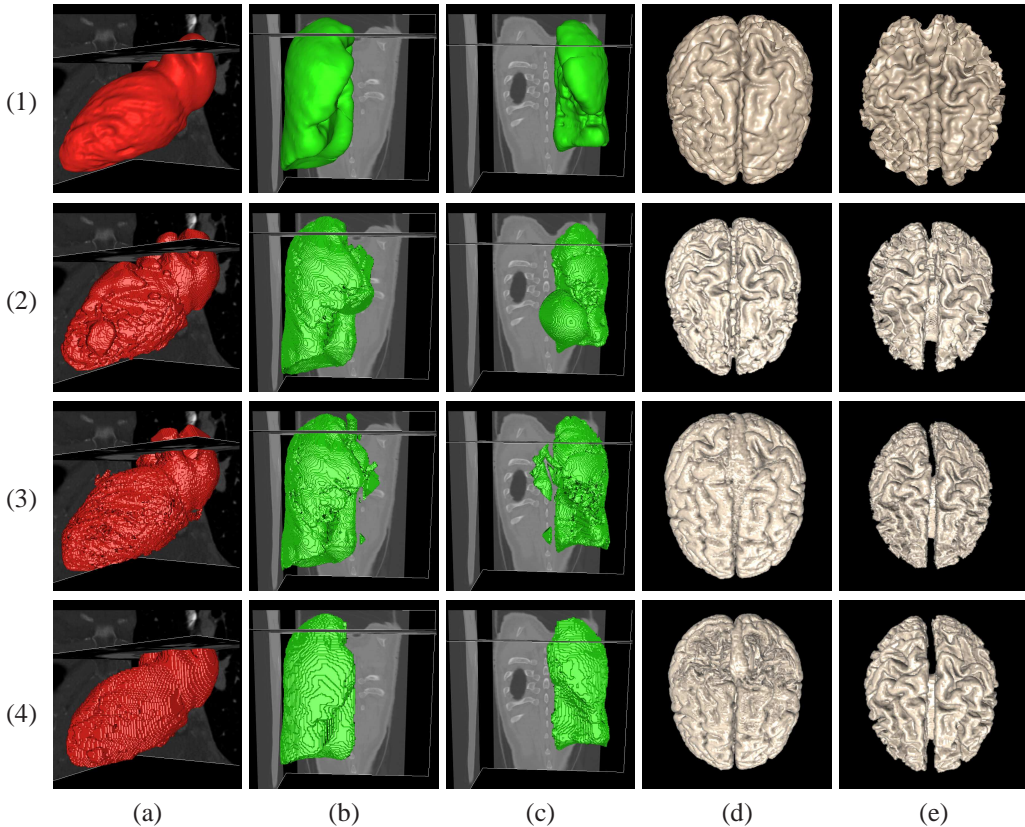


Fig. 19. Comparing 3D AVM with Geodesic Active Contours (GAC), Active Contour without edges (ACWE) and Level Set Evolution Without Reinitialization (LSEWR). Results of (1) 3D AVM, (2) GAC, (3) ACWE and (4) LSEWR. (a) Heart Left Ventricle segmentations, (b) right lung segmentations, (c) left lung segmentations, (d) Gray Matter segmentations, and (e) White Matter segmentations. In the heart and lung cases (a)-(c), 3D AVM has 40,962 vertices. In the brain case (d)-(e), 3D AVM has 131,074 vertices.

TABLE IV  
SEGMENTATION ACCURACY AND RUNNING TIME COMPARISON AMONG 3D AVM, GAC, ACWE AND LSEWR

Volume Size	Heart		Right Lung		Left Lung		Brain GM		Brain WM	
	DSC	Time	DSC	Time	DSC	Time	DSC	Time	DSC	Time
3D AVM	92.93	487s	95.62	512s	94.41	508s	91.5	1023s	78.2	1068s
3D GAC	88.82	633s	87.16	899s	82.53	832s	85.0	2332s	72.5	1752s
3D ACWE	91.81	1099s	86.07	1845s	86.15	1487s	90.4	984s	91.1	864s
3D LSEWR	91.14	1370s	92.77	1655s	93.01	1573s	89.4	621s	89.0	643s

Next, instead of directly using image intensity or gradient information to deform GAC, ACWE and LSEWR, we designed an experiment to deform these models based on the predicted ROIs so that we can compare 3D AVM with these level set models deforming under similar forces. First, according to the GAC, ACWE, or LSEWR's current model position, the ROI prediction steps are followed (Section II-C) to estimate the binary object ROI. Second, the model deforms for several iterations based on the current ROI. For GAC and LSEWR, the external forces are derived from the gradient map of the ROI (*i.e.*, non-zero gradient around ROI boundary and zero gradient everywhere else). For ACWE, the intensity values of the ROI are used to compute the mean intensities; thus, mean foreground intensity is 1 and mean background intensity is 0. By repeating the above two steps—ROI prediction and model deformation, the ROI based GAC, ACWE or LSEWR model can deform according to the adaptively changing ROIs and reach the final converged result. Figure 20 compares the

segmentation results by these ROI-based level set models with the results by 3D AVM. The running time and quantitative accuracy of these models are also compared with those of the original models without ROIs; the comparison is shown in Table V.

We also designed another ROI-based parametric deformable model and compared 3D AVM to it. After the model obtains its predicted object ROI, its external forces are set in a way similar to *T-Snakes* [22], according to the vertices' positions and normals. For a model vertex  $\mathbf{v}$ , the external force is

$$f_{\mathbf{v}}^{ext} = \begin{cases} \eta \cdot \hat{\mathbf{n}}_{\mathbf{v}} & \mathbf{v} \in R \\ -\eta \cdot \hat{\mathbf{n}}_{\mathbf{v}} & \text{otherwise} \end{cases}, \quad (27)$$

where  $\hat{\mathbf{n}}_{\mathbf{v}}$  is the surface normal at this vertex and  $\mathbf{v} \in R$  means the vertex is inside the predicted ROI  $R$ . Therefore, the model expands or shrinks to match the ROI boundary. The ROI in this case is also dynamically updated according to the model's changing position. The segmentation results of this ROI-based



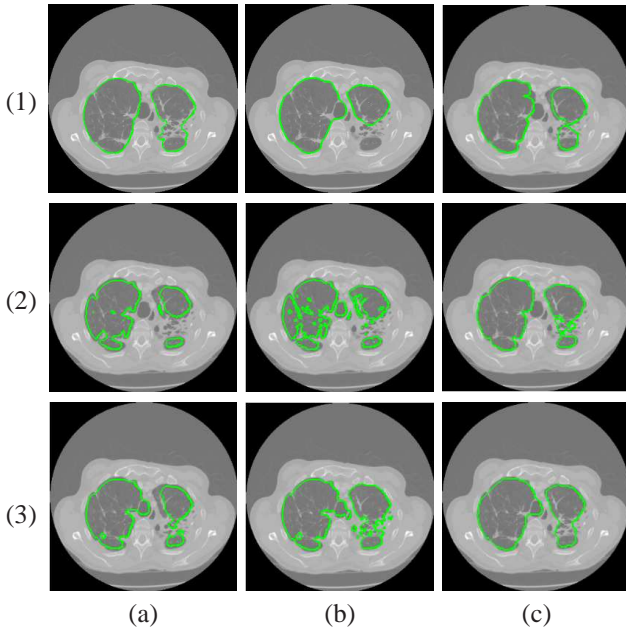


Fig. 21. Segmentation results on a representative 2D cross-sectional image. The result of (1)(a) 3D AVM, (1)(b) 2D AVM, (1)(c) another ROI-based parametric deformable model with T-snake like external forces. Original level set models of (2)(a) GAC, (2)(b) ACWE, (2)(c) LSEWR, and modified level set models of (3)(a) GAC, (3)(b) ACWE, (3)(c) LSEWR using predicted ROIs.

parametric model are compared to 3D AVM in Figure 20, and the running time and quantitative accuracy are reported in Table V. One can see that 3D AVM is more efficient than this other parametric model with T-snake like external forces; this is because AVM's external forces are derived from (12) using information from the distance transform of the ROI and are long-range adaptive forces which are stronger on model points far-away from ROI boundary, while the other parametric model deforms under constant force (27).

In Figure 21, we show the lung segmentation results of Figures 19 and 20 on a representative cross-sectional image of the 3D thorax CT volume. Note that the patient has lung cancer which causes part of the lung interior region to have similar texture as the lung exterior. Comparing the original GAC, ACWE and LSEWR models (Figure 21.(2)) with the modified ones that deform based on the predicted ROIs (Figure 21.(3)), the ROI-based models give results with simplified topology since the ROI estimation procedure (Section II-C) cleans small holes and islands inside and surrounding the models. Also, comparing 3D AVM with 2D AVM, 2D AVM failed to reach the lung boundary but stopped at a local minimum (Figure 21.(1)(b)), while 3D AVM reached the true boundary of the lung despite the abnormality (Figure 21.(1)(a)).

Compared with GAC, ACWE and LSEWR level set models, 3D AVM is represented by Finite Element triangulation, thus smooth mesh surfaces can be obtained directly by AVM without any post-processing such as morphological operations and surface reconstruction. 3D AVM preserves topology during deformation. And using probabilistically estimated object ROI, 3D AVM is also good at overcoming local minima and avoiding leakage (Figures 19, 20 and 21). Thus 3D AVM is very suitable for extracting organ boundaries from volumetric

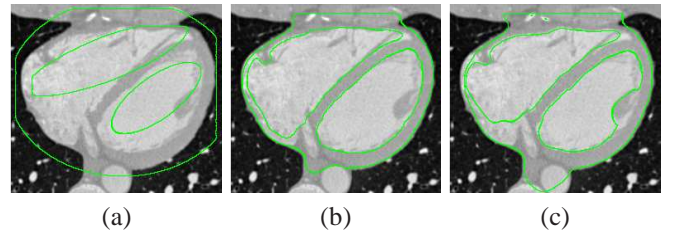


Fig. 22. Heart segmentation in a 2D slice projection. (a) The initial model, (b) the final converged result after 24 iterations and (c) the converged result of three independently-evolving AVMs after 27 iterations.

medical images.

2) *Comparison Between 3D AVM and MSAVM*: Since MSAVM integrates high level spatial information, it is more robust and allows more flexible initialization and parameter setting when compared to the 3D AVM. As shown in Table II, the same parameter setting was used for all the segmentation examples of MSAVM.

Table VI provides the quantitative evaluation and running times. MSAVM took similar numbers of iterations to converge compared with 3D AVM, but achieved improved segmentation accuracy. In each iteration, MSAVM needs extra time to calculate the spatial distance between surfaces; this overhead is very small, however, since the distance transform of each surface is readily available through the implicit model representation  $\Phi_{\Lambda}$ .

TABLE VI  
COMPARISON BETWEEN MSAVM AND 3D AVM

	MSAVM		3D AVM	
	DSC	Time	DSC	Time
Lung in Figure 13	96.2	1913s	94.6	1847s
Heart in Figure 14	92.3	1785s	91.2	1743s
GM in Figure 16	93.2	1094s	91.5	1023s
WM in Figure 16	82.3	1126s	78.2	1068s

To demonstrate the advantage of MSAVM more clearly, we show 2D axial projection slices from a case of 3D heart segmentation using MSAVM in Figure 22.(a)-(b), and compared them with the converged result of using several independent 3D AVMs with the same initialization in Figure 22.(c). Due to structures like papillary muscles inside the inner surfaces and obscure boundary of the outer surface, the 3D AVMs without spatial constraints either leaked to the outer-most (*e.g.*, outer surface) or stopped at local minima (*e.g.*, papillary muscle boundary). In contrast, deforming under the spatial constraints, MSAVM avoided such leakage and overcame the local minima to find the desired object boundary.

## V. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed the novel active volume model and multiple-surface active volume model, which are the natural extensions of parametric deformable models to integrate object appearance and region information. The main contributions include: (1) a clean formulation integrates on-line learning and region statistics into active contours and surfaces, which provides flexible initialization and rapid convergence, (2) the finite differences optimization framework

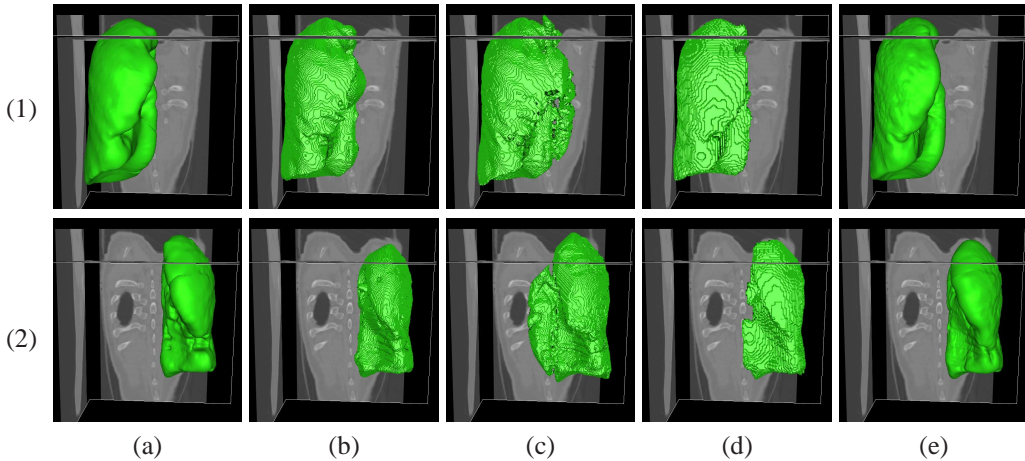


Fig. 20. Segmentation results of 3D AVM, and GAC, ACWE, LSEWR and an intensity-based parametric deformable model that also deform under ROI-based forces. (1) Right lung segmentation results, (2) left lung segmentation results. The results using (a) 3D AVM with 40,962 vertices, (b) ROI-based GAC, (c) ROI-based ACWE, (d) ROI-based LSEWR and (e) another ROI-based parametric deformable model with 40,962 vertices which deforms under T-snake like external forces.

TABLE V  
COMPARISON BETWEEN ORIGINAL MODELS AND MODELS THAT DEFORM ACCORDING TO THE PREDICTED OBJECT ROIS

	3D AVM		GAC		ACWE		LSEWR		Parametric Model	
	DSC	Time	DSC	Time	DSC	Time	DSC	Time	DSC	Time
Original models on right lung	95.62	512s	87.16	899s	86.07	1845s	92.77	1655s	n/a	n/a
Models using ROIs on right lung	95.62	512s	92.08	882s	90.87	2246s	93.31	1182s	95.30	632s
Original models on left lung	94.41	508s	82.53	832s	86.15	1487s	93.01	1573s	n/a	n/a
Models using ROIs on left lung	94.41	508s	93.40	851s	91.22	1504s	94.60	1356s	94.35	574s

enables very fast gradient- and appearance-statistics based model deformations, (3) the combination of multiple sources of information in a unified framework for predicting object region and boundary makes the model easily extensible, (4) integrating high-level geometric spatial constraints for simultaneously segmenting multiple interacting objects' boundary surfaces further improves the flexibility of initialization and speed of convergence. Using various experiments on 2D and 3D medical images, we demonstrate that the AVM model can perform segmentation efficiently and reliably on CT, MRI and Ultrasound images with flexible initialization and rapid convergence. Taking advantage of parametric deformable models' topology preserving property, AVM and MSAVM are very suitable for extracting boundaries of organs, such as lung, heart and brain from medical images.

One of the novelties of the AVM is that it can predict background appearance statistics from the current model-interior foreground appearance. The prediction procedure is embedded in the segmentation process. In the first few iterations when the model surface was still far away from the true object boundary, the foreground appearance (inside the current model) was only an approximation of the final foreground appearance. Therefore, in order to avoid bias in appearance approximation for cases where the initial model covers both object and partial background, we require an AVM to be initialized either completely inside or largely overlapped with the foreground object. The initialization of a 2D model is by asking the user to manually place a circle (or a polygon) inside or largely overlapped with the object; the initialization of a 3D model is by asking the user to manually place an ellipsoid (with user-

specified radius and orientation) inside or mostly overlapped with the object. In this way, we make sure that even in the first few iterations, the current model-interior appearance is dominated by the true foreground object's appearance. Examples where the initial models are completely inside the objects can be seen in Figure 6. And examples where the initial models are not completely inside but are largely overlapped with the foreground objects can be seen in Figure 5.

As we discussed in Section III-B, flexible initialization and fast convergence could also have negative effects. Even though Laplacian Mesh Optimization maintains the mesh quality, triangles on the mesh can still have different area sizes especially when the model deforms rapidly. Because there are not sufficient vertices in some regions of the simplex-mesh, such as near high-curvature boundary regions or regions that have expanded greatly from the initial model, it is still hard for the model to reach very fine details of such regions (*e.g.*, White Matter in Figure 12 and top-right tip of the right ventricle in Figure 22). Unfortunately, based on our study of this problem, there is no suitable area-based isotropic remeshing technique that can be used directly. The typical remeshing techniques are also too time-consuming. In the future, we plan to address this problem by adaptively reparameterizing the model and adding vertices near branches and high-curvature structures since vertices in such areas are sparser than those on the main body. The reparameterization procedure has another advantage since using it we can choose a model with fewer vertices to start the deformation, and then adaptively add more vertices based on the model vertices' local sparseness. In this way we can potentially improve the efficiency of the model.

One advantage of the MSAVM spatial constraint, compared to more specific constraints such as using active shape models [6], is that it does not require off-line training. One can initialize an MSAVM with the correct topology using multiple simple ellipsoidal surfaces, and then the model can deform to segment heart, lung, or brain surfaces, depending on the image data. And if necessary, incorporating statistical shape priors in our AVM framework is straightforward: The prior shape model can be treated similarly to the boundary of the ROI, and an additional energy term can be added to generate forces that attract the AVM or MSAVM to match the prior shape model.

Another future direction to improve AVMs is to use a *soft* probabilistic formulation in the ROI prediction step (Section II-C). In this paper, we make a hard decision based on the posterior probability  $Pr(L(x)|f_1, f_2, \dots, f_k, \dots, f_n)$ ; the ROI is represented by a binary map and its distance transform  $\Phi_R$  is used in the energy function. Instead of this *hard* formulation, we plan to investigate representing the ROI softly using a probability map, and then apply gradient-weighted distance transform [13] to compute  $\Phi_R$ . We expect that the  $\Phi_R$  computed by weighted distance transform of a soft probabilistic ROI will likely produce forces that attract the model to grow faster in high-probability regions therefore add to the adaptive appearance model more information sooner, which in turn helps resolve ambiguities in low-probability regions.

Since our model can be easily extended, we will explore more application areas of AVMs. For example, texture and tensor-based information can be used to predict object ROIs, which implies the models' potential use in segmenting DTI images. 3D AVM and MSAVM can also be extended to segment and track organs' movements by integrating temporal information in 4D volumetric medical images. Taking advantage of recent developments in Graphic Processing Unit (GPU) technology and parallel architectures, we plan to have parallelized implementations of the models for real-time 3D and 4D medical image segmentation.

#### ACKNOWLEDGMENTS

The cardiac CT datasets and ground truth used in this work are provided by Dr. Leon Axel (NYU) and also from the Rotterdam Coronary Artery Algorithm Evaluation Framework [31]. The lung CT datasets are obtained from the National Cancer Institute's Lung Image Database Consortium (LIDC). The authors would also like to acknowledge stimulating discussions with Dr. Dimitris Metaxas (Rutgers), Junzhou Huang (Rutgers) and Yaoyao Zhu (Lehigh). This work is supported by NSF grant IIS-0812120.

#### REFERENCES

- [1] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, vol. 22, pp. 61–79, 1997.
- [2] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. Image Processing*, vol. 10, pp. 266–277, 2001.
- [3] C. Cocosco, V. Kollokian, R. Kwan, and A. Evans, "Brainweb: Online interface to a 3D MRI simulated brain database," *NeuroImage*, vol. 5, no. 4, 1997.
- [4] L. Cohen and I. Cohen, "Finite-element methods for active contour models and balloons for 2-D and 3-D images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1131–1147, 1993.
- [5] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *European Conference on Computer Vision*, vol. 2, pp. 484–498, 1998.
- [6] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape model - their training and application," *Computer Vision and Image Understanding*, vol. 61, pp. 38–59, 1995.
- [7] M. Costa, H. Delingette, S. Novellas, and N. Ayache, "Automatic segmentation of bladder and prostate using coupled 3D deformable models," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 252–260, 2007.
- [8] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2007.
- [9] S. Dambreville, Y. Rathi, and A. Tannenbaum, "A framework for image segmentation using shape models and kernel space shape priors," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1385–1399, 2008.
- [10] X. Fan, P.-L. Bazin, and J. Prince, "A multi-compartment segmentation framework with homeomorphic level sets," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2008.
- [11] X. Huang and D. Metaxas, "Metamorphs: Deformable shape and appearance models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1444–1459, 2008.
- [12] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, pp. 321–331, 1988.
- [13] R. Kimmel, N. Kiryati, and A. M. Bruckstein, "Sub-pixel distance maps and weighted distance transforms," *Journal of Mathematical Imaging and Vision*, vol. 6, pp. 223–233, 1999.
- [14] T. Kohlberger, D. Cremers, M. Rousson, R. Ramaraj, and G. Funka-Lea, "4D shape priors for a level set segmentation of the left myocardium in SPECT sequences," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 92–100, 2006.
- [15] M. E. Leventon, W. Eric, W. E. L. Grimson, and O. Faugeras, "Statistical shape influence in geodesic active contours," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1316–1324, 2000.
- [16] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level set evolution without re-initialization: A new variational formulation," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 430–436, 2005.
- [17] H. Li and A. Yezzi, "Local or global minima: Flexible dual-front active contours," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 1–14, 2007.
- [18] K. Li, X. Wu, D. Chen, and M. Sonka, "Optimal surface segmentation in volumetric images—a graph-theoretic approach," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 119–134, 2006.
- [19] D. L. Logan, *A First Course in the Finite Element Method*, 4th ed. CL Engineering, 2006.
- [20] D. MacDonald, N. Kabani, D. Avis, and A. C. Evans, "Automated 3D extraction of inner and outer surfaces of cerebral cortex from MRI," *NeuroImage*, vol. 12, no. 3, pp. 340–356, 2000.
- [21] R. Malladi, J. Sethian, and B. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.
- [22] T. McInerney and D. Terzopoulos, "T-snake: Topology adaptive snakes," *Medical Image Analysis*, vol. 4, pp. 73–91, 2000.
- [23] R. A. McLaughlin, J. P. Williamson, M. J. Phillips, J. J. Armstrong, S. Becker, D. R. Hillman, P. R. Eastwood, and D. D. Sampson, "Applying anatomical optical coherence tomography to quantitative 3D imaging of the lower airway," *Optics Express*, vol. 16, pp. 17521–17529, 2008.
- [24] D. N. Metaxas and D. Terzopoulos, "Shape and nonrigid motion estimation through physics-based synthesis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 580–591, 1993.
- [25] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.
- [26] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, "Laplacian mesh optimization," *GRAPHITE '06: Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, pp. 381–389, 2006.
- [27] A. Nealen, O. Sorkine, M. Alexa, and D. Cohen-Or, "A sketch-based interface for detail-preserving mesh editing," *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 1142–1147, 2005.
- [28] N. Paragios and R. Deriche, "Geodesic active regions and level set methods for supervised texture segmentation," *International Journal of Computer Vision*, vol. 46, no. 3, pp. 223–247, 2002.



- [29] A. Popovic, M. de la Fuente, M. Engelhardt, and K. Radermacher, "Statistical validation metric for accuracy assessment in medical image segmentation," *International Journal of Computer Assisted Radiology and Surgery*, vol. 2, pp. 169–181, 2007.
- [30] C. Samson, L. Blanc-Féraud, G. Aubert, and J. Zerubia, "A level set model for image classification," *International Journal of Computer Vision*, vol. 40, no. 3, pp. 187–197, 2000.
- [31] M. Schaap, C. Metz, and et al., "Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms," *Medical Image Analysis*, vol. 13, no. 5, pp. 701–714, 2009.
- [32] T. Shen and X. Huang, "3D medical image segmentation by multiple-surface active volume models," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 1059–1066, 2009.
- [33] T. Shen, H. Li, Z. Qian, and X. Huang, "Active volume models for 3D medical image segmentation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 707–714, 2009.
- [34] T. Shen, Y. Zhu, X. Huang, J. Huang, D. Metaxas, and L. Axel, "Active volume models with probabilistic object boundary prediction module," *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 331–341, 2008.
- [35] L. Spreeuwiers and M. Breeuwer, "Detection of left ventricular Epi- and Endocardial borders using coupled active contours," *Computer Assisted Radiology and Surgery*, pp. 1147–1152, 2003.
- [36] L. Staib and J. Duncan, "Boundary finding with parametrically deformable models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1061–1075, 1992.
- [37] O. Subakan and B. Vemuri, "Image segmentation via convolution of a level-set function with a Rigaut kernel," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2008.
- [38] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the mumford and shah model," *International Journal of Computer Vision*, vol. 50, pp. 271–293, 2001.
- [39] C. Xu and J. Prince, "Snakes, shapes and gradient vector flow," *IEEE Trans. Image Processing*, vol. 7, pp. 359–369, 1998.
- [40] J. Yang and J. Duncan, "3D image segmentation of deformable objects with joint shape-intensity prior models using level sets," *Medical Image Analysis*, vol. 8, no. 3, pp. 285–294, 2004.
- [41] A. Yezzi Jr, A. Tsai, and A. Willsky, "A statistical approach to Snakes for bimodal and trimodal imagery," *IEEE International Conference on Computer Vision*, vol. 2, pp. 898–903, 1999.
- [42] X. Zeng, L. Staib, R. Schultz, and J. Duncan, "Segmentation and measurement of the cortex from 3D MR images using coupled-surfaces propagation," *IEEE Trans. Medical Imaging*, vol. 18, pp. 927–937, 1999.
- [43] S. Zhu and A. Yuille, "Region Competition: Unifying snakes, region growing, and Bayes/MDL for multi-band image segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, 1996.
- [44] Y. Zhu, X. Papademetris, A. Sinusas, and J. Duncan, "Segmentation of the left ventricle from cardiac MR images using a subject-specific dynamical model," *IEEE Trans. Medical Imaging*, vol. 29, pp. 669–687, 2010.