Metamorphs: Deformable Shape and Appearance Models

Xiaolei Huang, *Member, IEEE,* and Dimitris Metaxas, *Senior Member, IEEE*

Abstract

This paper presents a new deformable modeling strategy aimed at integrating shape and appearance in a unified space. If we think traditional deformable models as "active contours" or "evolving curve fronts", the new deformable shape and appearance models we propose are "deforming disks or volumes". Each model has not only boundary shape but also interior appearance. The model shape is implicitly embedded in a higher dimensional space of distance transforms, thus represented by a distance map "image". In this way, both shape and appearance of the model are defined in the pixel space. A common deformation scheme, the Free Form Deformations (FFD), parameterizes warping deformations of the volumetric space in which the model is embedded in, hence deforming both model boundary and interior simultaneously. When applied to segmentation, a Metamorphs model can be initialized covering a seed region far-away from object boundary and efficiently evolve and converge to an optimal solution. The model dynamics are derived in a unified variational framework that consists of edge-based and region-based energy terms, both of which are differentiable with respect to the common set of FFD parameters. As the model deforms, its interior appearance statistics are learned adaptively, and then toward the next-step deformation, the model examines not only edge information but its exterior region statistics to ensure that it only expands to new territory with consistent appearance statistics. The Metamorphs formulation also allows natural merging and competition of multiple models. We demonstrate the robustness of Metamorphs using both natural and medical images that have high noise levels, intensity inhomogeneity, and complex texture.

Index Terms

Segmentation, deformable models, shape, appearance, region, edge, boundary, implicit shape representation, free form deformations, nonparametric intensity statistics, distance transform, level set

X. Huang is with the Computer Science and Engineering Department, Lehigh University, Bethlehem, PA, USA. D. Metaxas is with the Department of Computer Science and Department of Biomedical Engineering, Rutgers University - New Brunswick, NJ, USA. Emails: huang@cse.lehigh.edu, dnm@cs.rutgers.edu.

I. INTRODUCTION

Automated image segmentation is a fundamental problem in computer vision and medical image analysis. It remains challenging to solve the problem robustly however, due to the common presence of cluttered objects, object texture, image noise, variations in illumination, and various other conditions in natural or medical images. To address these difficulties, model-based methods have been extensively studied and widely used, with considerable success because of their ability to integrate high-level knowledge with information from low-level image processing. The models being used can be either *deformable models* [1], [2], [3], [4], [5], or *statistical shape and appearance models* [6], [7], [8]. *Deformable models* are curves or surfaces that deform under the influence of internal smoothness and external image forces to delineate object boundary. Compared to local edge-based methods, deformable models have the advantage of estimating boundary with smooth curves or surfaces that bridge over boundary gaps. When initialized far-away from object boundary, however, a model can be trapped by spurious edges and/or high noise. *Statistical shape and appearance models* are learned *a priori* from examples to capture variations in the shape and appearance of an object of interest in images. When applied to segmentation, the models deform toward object boundary but with constraints to deform only in ways characteristic of the object they represent. These statistical models encode high-level knowledge in a more specific manner and are often more robust for image interpretation, although they require training data collection, annotation, and learning processes that are often laborious and time consuming.

In this paper, we propose a new class of deformable models which does not require *a priori* off-line learning, yet enjoys the benefit of having appearance constraints by online adaptive learning of model-interior region intensity statistics. We term the new models "Metamorphs". The basic framework of applying a Metamorphs model to boundary extraction is depicted in Fig. 1. The object of interest in this example is the corpus callosum structure in an MRI image of the brain. First, a simple-shape (e.g. circular) model is initialized inside the corpus callosum (see the blue circle in Fig. 1(a)). Considering the model as a "disk", it has a shape and covers an area of the image which is the interior of the current model. The model then deforms toward edges as well as toward the boundary of a region that has similar intensity statistics as the model interior. Fig. 1(b) shows the edges detected using a canny edge detector; note that the edge detector with automatically-determined thresholds gives a result that has spurious

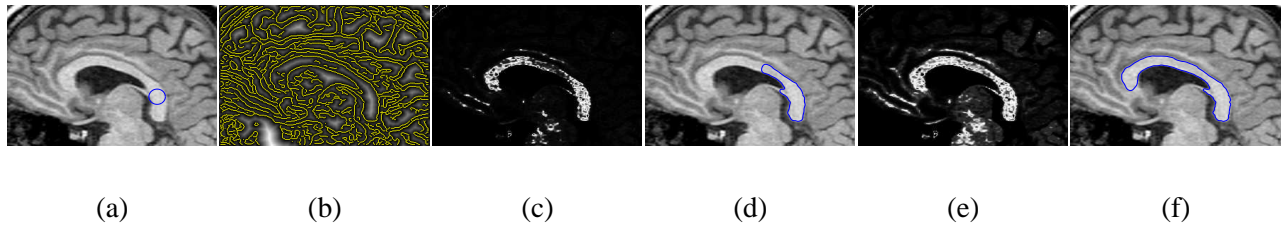| (a) | (b) | (c) | (d) | (e) | (f) |

Fig. 1. Metamorphs segmentation of a brain structure. (a) An MRI image of the brain; the initial circular model is drawn on top. (b) Edges detected using canny edge detector. (c) The intensity likelihood map computed according to the intensity probability density function of the initial model interior. (d) Intermediate evolving model after 15 iterations. (e) The intensity likelihood map according to the intermediate model's interior statistics. (f) Final converged model after 38 iterations.

edges and boundary gaps. To counter the effect of noise in edge detection, we estimate a region of interest (ROI) that has similar intensity statistics with the model interior. To find this region, we first estimate the model-interior probability density function (p.d.f.) of intensity, then a likelihood map is computed which specifies the likelihood of a pixel's intensity according to the model-interior p.d.f. Fig. 1(c) shows the likelihood map computed based on the initial model interior; and we threshold the likelihood map to get the ROI. The evolution of the model is then derived using a gradient descent method from a unified variational framework that consists of energy terms defined on edges, the ROI boundary, and the likelihood map. Fig. 1(d) shows the model after 15 iterations of deformation. As the model deforms, the model interior and its intensity statistics change, and the new statistics lead to the update of the likelihood map and the update of the ROI boundary for the model to deform toward. This online adaptive learning process empowers the model to find the boundary of objects with non-uniform appearance more robustly. Fig. 1(e) shows the updated likelihood map given the evolved model in Fig. 1(d). Finally, the model converges taking a balance between the edge and region influences, and the result is shown in Fig. 1(f).

The key property of Metamorphs is that these new models have both shape and appearance, and they naturally integrate edge information with region statistics when applied to segmentation. By doing so, these new models generalize two major classes of deformable models in the literature: the *parametric* models and the *geometric* models, which are traditionally shape-based, and take into account only edge or image gradient information. In the remainder of the Introduction section, we will review *parametric* and *geometric* deformable models, as well as previous efforts to incorporate region statistics into these models. We then discuss in more detail the novel aspects and contributions of Metamorphs.

A. Shape-based Deformable Models

Various deformable models proposed in the literature largely fall under two categories. The first class is the parametric (explicit) models that represent deformable curves and surfaces in their parametric form during segmentation. Examples are "Snakes" (or Active Contour Models) [1] and their extensions in both 2D and 3D [2], [3], [9], [10], [11]. The other class of deformable models is the geometric (implicit) models [4], [5], [12], [13] that represent 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 information. Comparing the two types of deformable models, the parametric models have a compact representation and allow fast convergence, while the geometric models can handle naturally topological changes.

Although the parametric and geometric deformable models differ in formulation and in implementation, both traditionally use primarily edge (or image gradient) information to derive external image forces that drive a shape-based model. In parametric models, a typical formulation [1] for the energy term deriving the external image forces is as follows:

$$E_{ext}(\mathcal{C}) = -\int_0^1 \left|\nabla \hat{I}\big(\mathcal{C}(s)\big)\right|^2 ds \tag{1}$$

Here $\mathcal{C}$ represents the parametric curve model parameterized by curve length s, $\hat{I} = G_\sigma * I$ is the image I after smoothing with a Gaussian kernel of standard deviation σ, and $\nabla \hat{I}(C)$ is the image gradient along the curve. Basically by minimizing this energy term, the accumulative image gradient along the curve is maximized, which means that the parametric model is attracted by strong edges that correspond to pixels with local-maxima image gradient values.

In geometric models, a typical objective function [4] that drives the front propagation of a level set function is:

$$E(\mathcal{C}) = \int_0^1 g\big(\left|\nabla \hat{I}(\mathcal{C}(s))\right|\big)\left|\mathcal{C}'(s)\right|ds, \quad \text{where} \quad g\big(\left|\nabla \hat{I}\right|\big) = \frac{1}{1 + \left|\nabla \hat{I}\right|^2} \tag{2}$$

Here $\mathcal{C}$ represents the front (i.e. zero level set) curve of the evolving level set function. To minimize the objective function, the front curve deforms along its normal direction $\mathcal{C}''(s)$, and its speed is controlled by the speed function $g\big(\left|\nabla \hat{I}\right|\big)$. The speed function definition, $g\big(\left|\nabla \hat{I}\right|\big)$, depends on image gradient $\nabla \hat{I}$, and it is positive in homogeneous

areas and zero at ideal edges. Hence the curve moves at a velocity proportional to its curvature in homogeneous regions and stops at strong edges.

The reliance on image gradient information in both parametric and geometric deformable models, however, makes them sensitive to noise and spurious edges so that the models often need to be initialized close to the boundary to avoid getting stuck in local minima. Geometric models, in particular, may leak through boundary gaps or generate small holes/islands. In order to address the limitations in these shape-only deformable models, and develop more robust models for boundary extraction, there have been significant efforts to integrate region information into both parametric and geometric deformable models.

B. Integrating Region Statistics Constraints

Along the line of parametric models, region analysis strategies have been proposed [14], [15], [16], [17], [18] to augment the "snake" (active contour) models. Assuming the partition of an image into an object region and a background region, a region-based energy criterion for active contours is introduced in [14], which includes photometric energy terms defined on the two regions. In [15], a generalized energy function that combines aspects of snakes/balloons and region growing is proposed and the minimization of the criterion is guaranteed to converge to a local minimum. This formulation still does not address the problem of unifying shape and appearance however, because of the large difference in representation for shape and appearance. While the model shape is represented using a parametric spline curve, the region intensity statistics are captured by parameters of a Gaussian distribution. This representation difference prevents the use of gradient descent methods to update both region parameters and shape parameters in a unified optimization process, so that the two sets of parameters are estimated in separate steps in [15] and the overall energy function is minimized in an iterative way. In other hybrid segmentation frameworks [17], [16], a region based module is used to get a rough binary mask of the object of interest. Then this rough boundary estimation serves as initialization for a deformable model, which will deform to fit edge features in the image using gradient information. In [18], Markov Random Fields are coupled with deformable models through graphical models interaction. And registration-like energy criteria for active contours are proposed in [19], [20] for region tracking. In all these frameworks, the region-based and edge-based modules are separate energy minimization processes, so that the integration is still imperfect and errors from one module can hardly be corrected by the other.

Along the line of geometric models, the integration of region and edge information [21], [22], [23], [24] has been mostly based on solving reduced cases of the minimal partition problem in the Mumford and Shah model for segmentation [25]. In the Mumford-Shah model, an optimal piecewise smooth function is pursued to approximate an observed image, such that the function varies smoothly within each region, and rapidly or discontinuously across the boundaries of different regions. Although the framework nicely incorporates gradient and region criteria into a single energy function, no practical globally-optimal solution for the function is available, most notably because of the mathematical difficulties documented e.g. in [25]. In the recent few years, progress has been made and solutions for several reduced cases of the Mumford-Shah functional have been proposed in the level set framework. One approach in [21] is able to segment images that consist of several regions, each characterizable by a given statistics such as the mean intensity and variance. Nevertheless the algorithm requires known *a priori* the number of segments in the image and its performance depends upon the discriminating power of the chosen set of statistics. Another approach in [22] applies a multi-phase level set representation to segmentation assuming piece-wise 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 the works presented by [13], [23], piece-wise constant and piece-wise smooth approximations of the Mumford-Shah functional are derived for two-phase (i.e. two regions) [13] or multiphase (i.e. multiple regions) [23] 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* [24] is another method that integrates edge and region based modules in a level set framework. The algorithm consists of two stages: a modeling stage that constructs a likelihood map of edge pixels and approximates region/class statistics using Mixture-of-Gaussian components, and a segmentation stage that uses level set techniques to solve for a set of smooth curves that are attracted to edge pixels and partition regions that have the expected properties of the associated classes. In summary of the above approaches, they all solve the frame partition problem, which can be computationally expensive when dealing with busy images that contain many objects and clutter. Their assumptions of piece-wise constant, piece-wise smooth, Gaussian, or Mixture-of-Gaussian intensity distributions within regions can also limit their effectiveness in segmenting objects whose interiors have textured appearance and/or complex multi-modal intensity distributions.

C. The Metamorphs Model

In this paper, we propose a new class of deformable models, "Metamorphs", which combine the best features of *parametric* and *geometric* models, and introduce novel modeling strategies that unify the representation and deformation schemes for shape and intensity. The shape of a Metamorphs model is implicitly embedded in a higher-dimensional space of distance transforms, thus represented by a distance map "image". In this way, no explicit parameters are needed for specifying model geometry, and the representations of shape and intensity are unified in the pixel space. To capture model-interior intensity statistics, the nonparametric kernel-based estimation of intensity distributions [26], [27] is used instead of Gaussian or Mixture-of-Gaussian parameters. The nonparametric statistics also does not have explicit parameters and it gets updated automatically as the model interior changes due to model deformation. The only set of parameters for Metamorphs is for specifying model deformations, which are parameterized by the cubic B-spline based Free Form Deformations (FFD) [28], [29]. When a Metamorphs model is used for boundary finding in images, we formulate, in a variational framework, both edge- and region-based energy terms that are differentiable with respect to the common set of model-deformation parameters. The overall energy function is then optimized by gradient descent to solve for the optimal deformations that deform the model to fit object boundary. During model evolution, the online-learning aspect of a Metamorphs model will constrain the model deformations such that the interior statistics of the model after each deformation is consistent with the statistics learned from the past history of the model interiors. The edge and region energy terms will have complementary effects and they will aid the model to overcome local minima due to small spurious edges inside the object, to prevent the model from leaking at boundary gaps, and to enable the segmentation of objects with intensity inhomogeneity and multi-modal interior statistics.

The remainder of the paper is organized as follows: section II introduces in detail the model representations for Metamorphs; section III presents the energy functional and optimization scheme when the model is applied to boundary finding in images; and section IV shows experimental results. We discuss interesting future directions in section V, and conclude in section VI.

II. SHAPE, APPEARANCE AND DEFORMATION REPRESENTATIONS IN METAMORPHS

In this section, we present the shape representation, model deformations, and nonparametric intensity statistics of the Metamorphs deformable models.

A. *The Model's Shape Representation*

The model's shape is embedded implicitly in a higher dimensional space of distance transforms. The Euclidean distance transform is used to embed the boundary of an evolving model as the zero level set of a higher dimensional distance function [30]. In order to facilitate notation, we consider the 2D case. Let $\Phi : \Omega \to R^+$ be a Lipschitz function that refers to the distance transform for the model shape $\mathcal{M}$. By definition Ω is bounded since it refers to the image domain. The shape defines a partition of the domain: the region that is enclosed by $\mathcal{M}$, $[\mathcal{R}_{\mathcal{M}}]$, the background $[\Omega - \mathcal{R}_{\mathcal{M}}]$, and on the model, $[\partial \mathcal{R}_{\mathcal{M}}]$ [1]. Given these definitions the following implicit shape representation for $\mathcal{M}$ is considered:

$$\Phi_{\mathcal{M}}(\mathbf{x}) = \begin{cases} 0, & \mathbf{x} \in \partial \mathcal{R}_{\mathcal{M}} \\ +\mathcal{D}(\mathbf{x}, \mathcal{M}), & \mathbf{x} \in \mathcal{R}_{\mathcal{M}} \\ -\mathcal{D}(\mathbf{x}, \mathcal{M}), & \mathbf{x} \in [\Omega - \mathcal{R}_{\mathcal{M}}] \end{cases} \tag{3}$$

where $\mathcal{D}(\mathbf{x}, \mathcal{M})$ refers to the minimum Euclidean distance between the image pixel location $\mathbf{x} = (x, y)$ and the model $\mathcal{M}$.

Such implicit embedding makes the model shape representation a distance map "image", which greatly facilitates the integration of shape and appearance information. It also provides a feature space in which objective functions that are optimized using a gradient descent method can be conveniently used. A sufficient condition for convergence of gradient descent methods requires continuous first derivatives, and the considered implicit representation satisfies this condition. In fact, one can prove that the gradient of the distance function is a unit vector in the normal direction of the shape. This property will make our model evolution fast. Examples of the implicit representation can be found in Fig. 2(2).
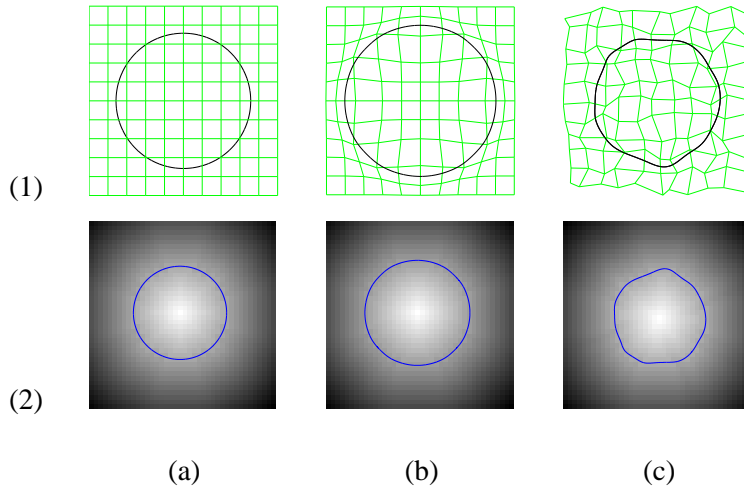
Fig. 2.   Shape representation and deformations of Metamorphs models. (1) The model shape. (2) The implicit distance map "image" representation of the model shape. (a) Initial model. (b) Example FFD control lattice deformation to expand the model. (c) Another example of the free-form model deformation given the control lattice deformation.

B. The Model's Deformations

The deformations that a Metamorphs model can undergo are defined using a space warping technique, the Free Form Deformations (FFD) [31], [29], which is a popular approach in graphics and animation. The essence of FFD is to deform the shape of an object by manipulating a regular control lattice F overlaid on its volumetric embedding space. The deformation of the control lattice consists of displacements of all the control points in the lattice, and from these sparse displacements, a dense deformation field for every pixel in the embedding space can be acquired through interpolation using interpolating basis functions such as the Cubic B-spline functions. One illustrative example is shown in Fig. 2. A circular shape [Fig. 2(1.a)] is implicitly embedded as the zero level set of a distance function [Fig. 2(1.b)]. A regular control lattice is overlaid on this embedding space. When the embedding space deforms due to the deformation of the FFD control lattice as shown in Fig. 2(b), the shape undergoes an expansion. Fig. 2(c) shows another example of free-form shape deformation given a particular FFD control lattice deformation. In this paper, we consider an Incremental Free Form Deformations (IFFD) formulation using the cubic B-spline basis functions for interpolation.

Let us consider a lattice of control points

$$F = \{F_{m,n}\} = \{(F_{m,n}^x, F_{m,n}^y)\}; \; m = 1, ..., M, \quad n = 1, ..., N \tag{4}$$

[1]In practice, we consider a very narrow band around the model shape $\mathcal{M}$ in the image domain as $\partial \mathcal{R}_{\mathcal{M}}$.

overlaid on a region $\Gamma = \{\mathbf{x}\} = \{(x, y)|l_x \le x \le h_x, l_y \le y \le h_y\}$ in the embedding space that encloses the model. Let us denote its initial regular configuration with no deformation as F^0 (e.g., as in Fig. 2(a)), and the deforming configuration as $F = F^0 + \delta F$. Then the IFFD parameters $\mathbf{q}$ are the deformation improvements of the control points in both x and y directions:

$$\mathbf{q} = \delta F = \{(\delta F_{m,n}^x, \delta F_{m,n}^y)\}; \ (m, n) \in [1, M] \times [1, N] \tag{5}$$

The deformed position of a pixel $\mathbf{x} = (x, y)$ given the deformation of the control lattice from F^0 to F, is defined in terms of a tensor product of Cubic B-spline polynomials:

$$D(\mathbf{x}) = \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) F_{i+k,j+l} \tag{6}$$

where $i = \lfloor \frac{x-l_x}{h_x-l_x} \cdot (M-1) \rfloor$, $j = \lfloor \frac{y-l_y}{h_y-l_y} \cdot (N-1) \rfloor$. This is the familiar definition for cubic B-spline based interpolation, and the terms in the formula refer to:

1) $F_{i+k,j+l}$, $(k, l) \in [0, 3] \times [0, 3]$ are the coordinates of the closest sixteen control points to pixel $\mathbf{x}$.

2) $B_k(u)$ represents the kth basis function of cubic B-spline:

$$B_0(u) = (1-u)^3/6, \qquad\qquad B_1(u) = (3u^3 - 6u^2 + 4)/6$$

$$B_2(u) = (-3u^3 + 3u^2 + 3u + 1)/6, \quad B_3(u) = u^3/6$$

with $u = \frac{x-l_x}{h_x-l_x} \cdot (M-1) - \lfloor \frac{x-l_x}{h_x-l_x} \cdot (M-1) \rfloor$.

$B_l(v)$ is similarly defined, with $v = \frac{y-l_y}{h_y-l_y} \cdot (N-1) - \lfloor \frac{y-l_y}{h_y-l_y} \cdot (N-1) \rfloor$.

Since $F = F^0 + \delta F$, we can re-write Eq. 6 in terms of the IFFD parameters $\mathbf{q} = \delta F$:

$$
\begin{aligned}
D(\mathbf{q}; \mathbf{x}) &= \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) (F_{i+k,j+l}^0 + \delta F_{i+k,j+l}) \\
&= \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) F_{i+k,j+l}^0 + \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) \delta F_{i+k,j+l}
\end{aligned}
\tag{7}
$$

Based on the linear precision property of B-splines, a B-spline curve through collinear control points is itself linear, hence the initial regular configuration of control lattice F^0 generates the undeformed shape and embedding space, i.e., for any pixel $\mathbf{x}$ in the sampling domain, we have:

$$\mathbf{x} = \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) F_{i+k,j+l}^0 \tag{8}$$

where i, j are derived the same way as in Eq. 6. Now combining Eq. 7 and Eq. 8, we have:

$$
\begin{aligned}
D(\mathbf{q}; \mathbf{x}) &= \mathbf{x} + \delta D(\mathbf{q}; \mathbf{x}), \quad \text{and} \\
\delta D(\mathbf{q}; \mathbf{x}) &= \sum_{k=0}^{3} \sum_{l=0}^{3} B_k(u) B_l(v) \delta F_{i+k,j+l}
\end{aligned}
\tag{9}
$$

where δD is the incremental deformation for pixel $\mathbf{x}$, given the FFD control lattice's incremental deformation δF.

Compared with optical-flow type of deformation representation (i.e. pixel-wise displacements in x and y directions) commonly used in the literature, the IFFD parameterization we use allows faster model evolution and convergence, because it has significantly fewer parameters. A hierarchical multi-level implementation of IFFD [32], which uses multi-resolution control lattices according to a coarse-to-fine strategy, can account for deformations of both large and small scales. The advantages of coarse-to-fine optimization have been demonstrated in deformable contour frameworks in the literature [33]. Another property of IFFD is that it imposes implicit smoothness constraints, since it guarantees C^1 continuity at control points and C^2 continuity everywhere else. Therefore there is no need to introduce computationally-expensive regularization components on the deformed shapes. As a space warping technique, IFFD also integrates naturally with the implicit shape representation which embeds the model shape in a higher dimensional space.

C. The Model's Appearance Representation

Rather than using Gaussian or Mixture-of-Gaussian parameters (such as mean and variance) to approximate the intensity distribution of the model interior, we model the distribution using a nonparametric kernel-based density estimation method. Kernel-based density estimation (also known as the Parzen windows technique [34]) is a popular nonparametric statistical method [35]. It represents a generalization of the Mixture-of-Gaussian model, where it does not make assumptions about the number of modes in a distribution, rather it treats every single sample as a Gaussian distribution and integrates over these small Gaussian kernels to derive the overall nonparametric estimation of the probability density function (p.d.f.). Recently this nonparametric technique has been applied to imaging and computer vision, most notably in modeling the varying background in video sequences [26], and in approximating multi-modal intensity density functions of color images [27]. In this paper, we use this representation to approximate the intensity p.d.f. of the model interior.
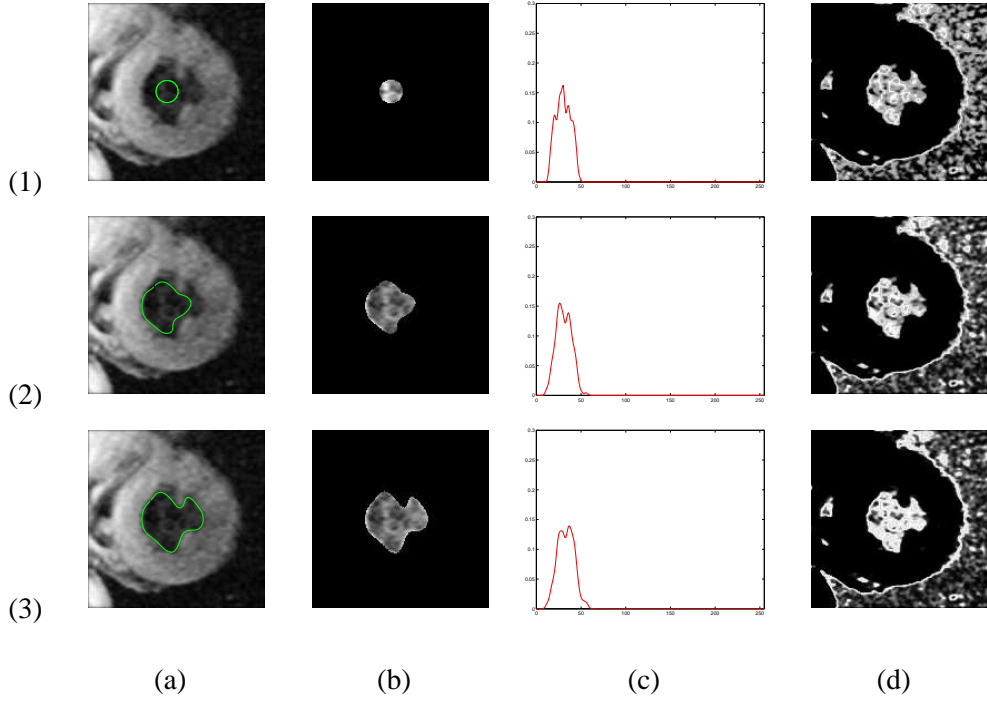
Fig. 3. Left Ventricle endocardium segmentation, demonstrating Metamorphs appearance representation. (1) Initial model. (2) Intermediate result after 4 iterations. (3) Final converged result after 10 iterations. (a) The evolving model drawn on original image. (b) Interior region of the evolving model. (c) The intensity p.d.f. of the model interior. (d) The image intensity probability map according to the p.d.f. of the model interior.

Suppose the model is placed on an image I, the image region bounded by current model $\Phi_{\mathcal{M}}$ is $\mathcal{R}_{\mathcal{M}}$, then the intensity p.d.f. of the model interior region can be represented using a Gaussian kernel-based density estimation:

$$\mathbf{P}(i|\Phi_{\mathcal{M}}) = \frac{1}{V(\mathcal{R}_{\mathcal{M}})} \iint_{\mathcal{R}_{\mathcal{M}}} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(i-I(\mathbf{y}))^2}{2\sigma^2}} d\mathbf{y} \tag{10}$$

where $i = 0, ..., 255$ denotes the pixel intensity values, $V(\mathcal{R}_{\mathcal{M}})$ denotes the volume of $\mathcal{R}_{\mathcal{M}}$, $\mathbf{y}$ represents pixels in the region $\mathcal{R}_{\mathcal{M}}$, and σ is a constant specifying the width of the Gaussian kernel [36]. One example of this nonparametric density estimation can be seen in Fig. 3. The zero level set of the evolving models $\Phi_{\mathcal{M}}$ are drawn on top of the original image in Fig. 3(a). The model interior regions $\mathcal{R}_{\mathcal{M}}$ are cropped and shown in Fig. 3(b). Given the model interiors, their nonparametric intensity p.d.f.s $\mathbf{P}(i|\Phi_{\mathcal{M}})$ are shown in Fig. 3(c), where the horizontal axis denotes the intensity values $i = 0, ..., 255$, and the vertical axis denotes the probability values $P \in [0, 1]$. Finally, over the entire image I, we evaluate the probability of every pixel's intensity, according to the model interior intensity p.d.f., and the resulting probability (or likelihood) map is shown in Fig. 3(d).
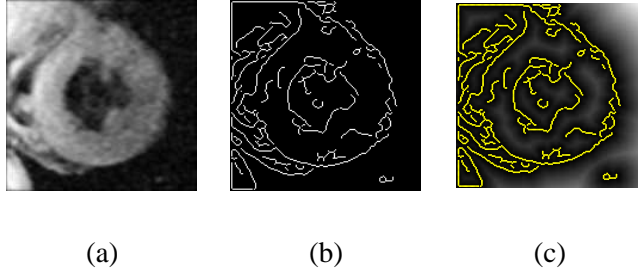
(a)   (b)   (c)

Fig. 4. Effects of small spurious edges on the "shape image". (a) An MRI image of the heart; the interested object boundary is the endocardium of the left ventricle. (b) Edge map of the image. (c) The derived "shape image" (distance transform of the edge map), with edges drawn on top. Note how the small spurious edges affect the "shape image" values inside the object.

Using this nonparametric estimation, the intensity distribution of the model interior gets updated automatically while the model deforms to cover a new set of interior pixels; and it avoids having to estimate and keep a separate set of intensity parameters such as the mean and variance if a Gaussian or Mixture-of-Gaussian model was used. Moreover, this kernel-based estimation in Eq. 10 is a continuous function, which facilitates the computation of derivatives in a gradient-descent based optimization framework.

III. VARIATIONAL FORMULATION FOR BOUNDARY FINDING WITH METAMORPHS MODEL

In this section, we present the variational framework in which a Metamorphs model can be used to find the boundary of an object of interest. In the framework, the motion of the model is driven by two types of energy terms derived from the image: the edge data terms E_E, and the region data terms E_R. So the overall energy functional E is defined by:

$$E = E_E + kE_R \qquad (11)$$

where k is a constant balancing the contributions from the two types of terms. In our formulation, we are able to omit the model smoothness term, since this smoothness is implicit by using Free Form Deformations. Next, we derive the edge and region data terms respectively.

A. The Edge Data Term

A Metamorphs model is attracted to edge features with high image gradient. We encode the edge information using a "shape image" Φ, which is the un-signed distance transform of the edge map of the image. The edge map
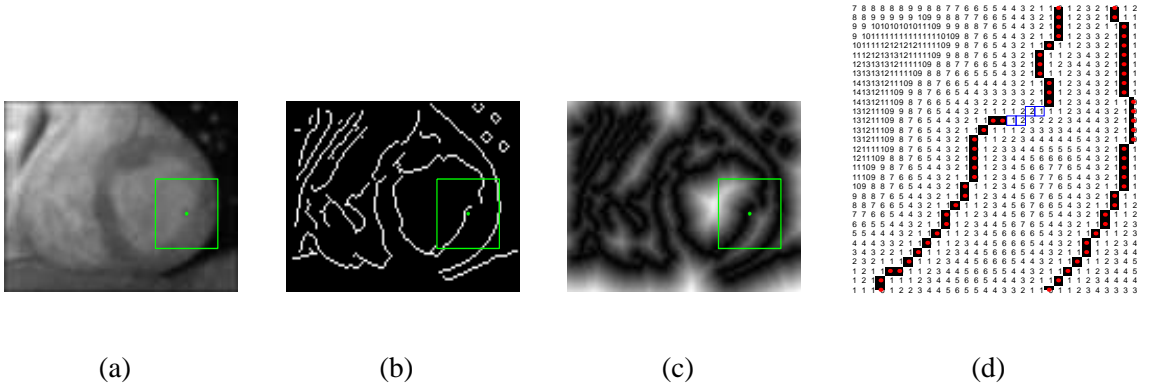
Fig. 5. At a small gap in the edges, the edge data term constraints the model to go along a path that coincides with the smooth shortest path connecting the two open ends of the gap. (a) Original Image. (b) The edge map, note the small gap inside the square. (c) The "shape image". (d) Zoom-in view of the region inside the square. The numbers are the "shape image" values at each pixel location. The red dots are edge points, the small blue squares indicate a path favored by the edge term for a Metamorphs model.

is computed using Canny Edge Detector with default parameter settings. In Fig. 4(c), we can see the "shape image" of an example MR heart image.

To evolve a Metamorphs model toward edges, we define an edge-based data term E_E on pixels in a narrow band around the model:

$$E_E = \frac{1}{S(\partial \mathcal{R}_\mathcal{M})} \iint_{\partial \mathcal{R}_\mathcal{M}} \left(\Phi(D(\mathbf{q};\mathbf{x}))\right)^2 d\mathbf{x} \tag{12}$$

where $\partial \mathcal{R}_\mathcal{M}$ refers to the model boundary affinity (i.e. a narrow band around the zero level set of the distance map $\Phi_\mathcal{M}$), $S(\partial \mathcal{R}_\mathcal{M})$ represents the surface area of the narrow band, and $\mathbf{x}$ denotes pixels in the narrow band.

Intuitively, this edge term encourages deformations D that map the model boundary pixels $\mathbf{x}$ to image locations $D(\mathbf{q};\mathbf{x})$ closer to edges so that the underlying "shape image" values $\Phi(D(\mathbf{q};\mathbf{x}))$ are as small (or as close to zero) as possible. During optimization, this term will deform the model along the gradient direction of the underlying "shape image" toward edges. Thus it will expand or shrink the model accordingly, serving as a two-way balloon force implicitly and making the attraction range of the model large.

One additional advantage of this edge term is that, at an edge with small gaps, this term will constrain the model to go along the "geodesic" path on the "shape image", which coincides with the smooth shortest path connecting the two open ends of a gap. This behavior can be seen from Fig. 5. Note that at a small gap on an edge, the edge

term favors a path with the smallest accumulative distance values to the edge points.

B. The Region Data Terms

An attractive aspect of the Metamorphs deformable models is that their interior intensity statistics are learned dynamically, and their deformations are influenced by forces derived from this dynamically-changing region information. This region information is very important in helping the models out of local minima, and converge to the true object boundary. In Fig. 4, the spurious edges both inside and around the object boundary degrade the reliability of the "shape image" and the edge data term. Yet the intensity probability maps computed based on model-interior intensity statistics, as shown in Fig. 3(d), give a consistent and clear indication on where the rough boundary of the object is. In another MR heart image shown in Fig. 6(1.a), a large portion of the object boundary (LV endocardium) is missing during computation of the edge map using default canny edge detector settings [Fig. 6(1.b)]. Relying solely on the "shape image" [Fig. 6(1.c)] and the edge data term, a model would have leaked through the large gap and mistakenly converged to the outer epicardium boundary. In this situation, the probability maps [Fig. 6(2-4.d)] computed based on model-interior intensity statistics become the key to optimal model convergence.

In our framework, we define two region data terms – a "Region Of Interest" (ROI) based balloon term E_{R_l} and a Maximum Likelihood term E_{R_m}, so the overall region-based energy function E_R is:

$$E_R = E_{R_l} + b E_{R_m} \tag{13}$$

1) The ROI based Balloon Term: We determine the "Region Of Interest" (ROI) as the largest possible region in the image that overlaps the model and has a consistent intensity distribution as the current model interior. The ROI-based balloon term is designed to efficiently evolve the model toward the boundary of the ROI.

Given a model $\mathcal{M}$ on image I [Fig. 7(a)], we first compute the image intensity probability map P_I [Fig. 7(b)], based on the model interior intensity statistics (see Eq. 10 in section II-C). A threshold (typically the mean probability over the entire image domain) is applied on P_I to produce a binary image P_B. More specifically, those pixels that have probabilities higher than the threshold in P_I are given the value 1 in P_B, and all other pixels are set to the value 0 in P_B. We then apply a connected component analysis algorithm based on run-length encoding and scanning [37] on P_B to extract the connected component that overlaps the model. Considering this connected
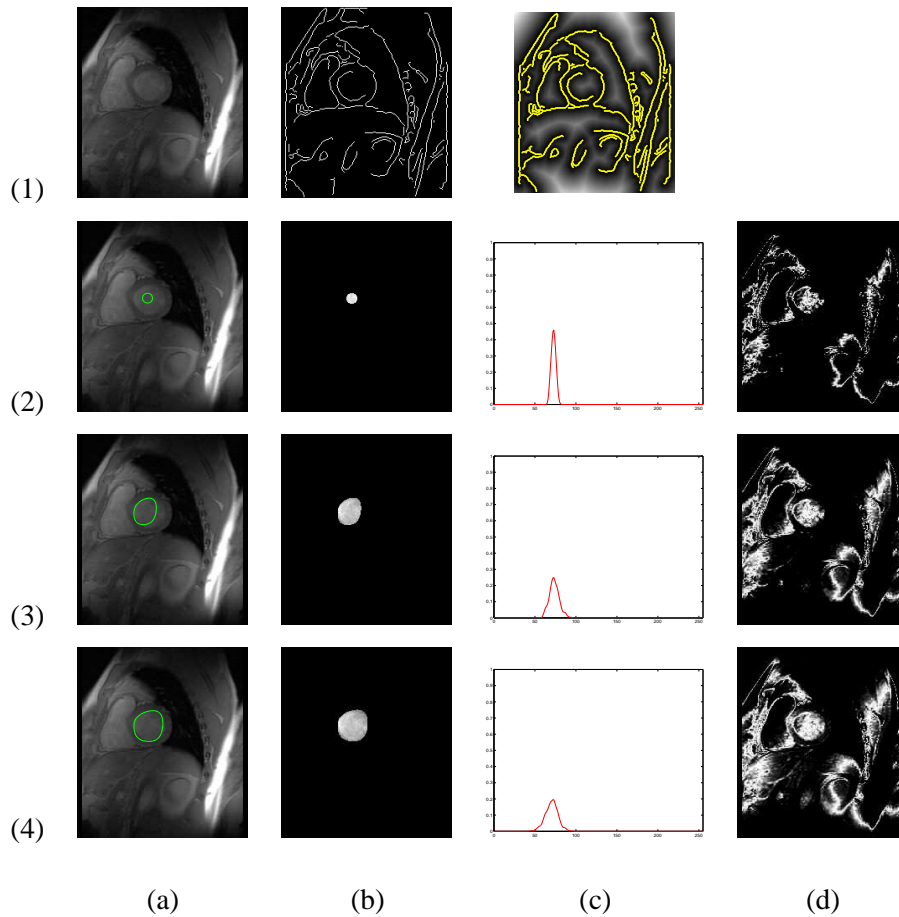
Fig. 6. Segmentation of the Left Ventricle endocardium in an MRI image. (1.a) the original image. (1.b) the edge map; note that a large portion of the object boundary is missing in the edge map. (1.c) the "shape image". (2) initial model. (3) intermediate model. (4) converged model. (a) zero level set of the current model drawn on the image. (b) model interiors. (c) the interior intensity p.d.f.s. (d) intensity probability maps.

component as a "disk" that we want the Metamorphs model to match, it is likely that this disk has small holes due to noise and intensity inhomogeneity, as well as large holes that correspond to real "holes" inside the object. How to deal with compound objects that potentially have holes using Metamorphs is an interesting question that we will discuss briefly in Section V. In this paper, we assume the objects of interest that we apply Metamorphs to segment are without interior holes. Under this assumption, the desired behavior of the model is to evolve toward the ROI border regardless of small holes in the ROI connected component. Hence we take the outer-most border of the selected connected component as the current ROI boundary. We encode this ROI boundary information by computing its "shape image", which is its un-signed distance transform [Fig. 7(c)]. Denote this "shape image" as
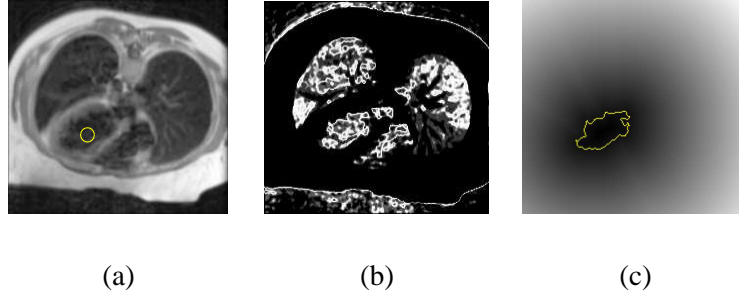
Fig. 7. Deriving the ROI based region data term. (a) The model shown on the original image. (b) The intensity probability map computed based on the model interior statistics. (c) The "shape image" encoding boundary information of the ROI.

Φ_r, the ROI-based balloon term is defined as follows:

$$E_{R_l} = \frac{1}{S(\partial \mathcal{R}_{\mathcal{M}})} \iint_{\partial \mathcal{R}_{\mathcal{M}}} \Phi_r(\mathbf{x})\big(\Phi_{\mathcal{M}}(D(\mathbf{q};\mathbf{x}))\big)d\mathbf{x} \tag{14}$$

Note that although this term is defined on pixels in the model boundary affinity $\partial \mathcal{R}_{\mathcal{M}}$, it is a region-based term because Φ_r is derived from the intensity probability map (see Eq. 10), which is computed based on the intensity statistics of the entire model-interior region.

There are two components in the above ROI term, and they are combined multiplicatively. The key to understand the first component, $\Phi_{\mathcal{M}}(D(\mathbf{q};\mathbf{x}))$, is to take note that the model shape representation, $\Phi_{\mathcal{M}}$, has negative values outside the model, zero value on the model, and positive values inside the model (see Eq. 3). By this component alone, the model boundary-affinity pixels $\mathbf{x}$ will be mapped outward to locations $D(\mathbf{q};\mathbf{x})$, where the model shape representation values $\Phi_{\mathcal{M}}(D(\mathbf{q};\mathbf{x}))$ are smaller. Hence the model would expand and grow like a balloon so as to minimize the value of the energy term. The second component in the energy term, Φ_r, is the ROI "shape image" and encodes the distance value of each pixel from the ROI region boundary. It serves as a weighting (or modulation) factor for the first component so that the speed of model evolution is proportional to the distance of the model from the ROI boundary. That is, the model moves fast when it is far away from the ROI boundary and the underlying $\Phi_r(\mathbf{x})$ values are large; it slows down as it approaches the boundary, and stops at the boundary. This property of adaptively changing speed leads to improved model evolution behavior.

Within the overall energy minimization framework, the ROI-based balloon term is the most effective in countering the effect of un-regularized or inhomogeneous region intensities such as that caused by speckle noise and spurious edges inside the object of interest (e.g. in Fig. 4 and Fig. 9). This is because the ROI term deforms the model

toward the outer-most boundary of the identified ROI, disregarding all small holes inside. Although this makes the assumption that the object to be segmented has no holes, it is a very effective measure to discard incoherent pixels and make noise and intensity inhomogeneity not to influence model convergence. Moreover, the ROI term generates adaptively changing balloon forces that expedite model convergence and improve convergence accuracy, especially when the object shape is elongated, or has salient protrusions or concavities.

2) The Maximum Likelihood Term: The previous ROI term is efficient in deforming the model toward object boundary when the model is still far away. When the model gets close to the boundary, however, the ROI may become less reliable due to intensity changes in the boundary areas. To achieve better convergence, we design another Maximum Likelihood (ML) region-based data term that constrains the model to deform toward areas where the pixel intensity probabilities of belonging to the model-interior intensity distribution are high. This ML term is formulated by maximizing the log-likelihood of pixel intensities in a narrow band around the model:

$$
\begin{aligned}
E_{R_m} &= -\frac{1}{S(\partial\mathcal{R}_\mathcal{M})}\iint_{\partial\mathcal{R}_\mathcal{M}} log\mathbf{P}(I(D(\mathbf{q};\mathbf{x}))\big|\Phi_\mathcal{M})d\mathbf{x} \\
&= -\frac{1}{S(\partial\mathcal{R}_\mathcal{M})}\iint_{\partial\mathcal{R}_\mathcal{M}}\Big[log\frac{1}{V(\mathcal{R}_\mathcal{M})}+log\frac{1}{\sqrt{2\pi}\sigma}+log\iint_{\mathcal{R}_\mathcal{M}}e^{\frac{-(I(D(\mathbf{q};\mathbf{x}))-I(\mathbf{y}))^2}{2\sigma^2}}d\mathbf{y}\Big]d\mathbf{x}
\end{aligned}
\tag{15}
$$

During model evolution, when the model is still far away from the object boundary, this ML term generates very little force to influence the model deformation. When the model gets close to the boundary, the ML term helps the model to converge and is particularly useful in preventing the model from leaking through large gaps (e.g. in Fig. 6).

C. Model Dynamic Evolution

In our formulation above, both the edge data term and the region data terms are differentiable with respect to the model deformation parameters $\mathbf{q}$, thus a unified gradient-descent based parameter updating scheme can be derived using both edge and region information. Based on the energy term definitions, one can derive the following evolution equation for each element $\mathbf{q}_i$ in the deformation parameters $\mathbf{q}$:

$$
\frac{\partial E}{\partial\mathbf{q}_i} = \frac{\partial E_E}{\partial\mathbf{q}_i} + k\Big(\frac{\partial E_{R_l}}{\partial\mathbf{q}_i}+b\frac{\partial E_{R_m}}{\partial\mathbf{q}_i}\Big)
\tag{16}
$$

- The motion due to the edge data term is:

$$
\frac{\partial E_E}{\partial\mathbf{q}_i} = \frac{1}{S(\partial\mathcal{R}_\mathcal{M})}\iint_{\partial\mathcal{R}_\mathcal{M}}2\Phi(D(\mathbf{q};\mathbf{x}))\cdot\big(\nabla\Phi(D(\mathbf{q};\mathbf{x}))\cdot\frac{\partial}{\partial\mathbf{q}_i}D(\mathbf{q};\mathbf{x})\big)d\mathbf{x}
$$

- And the motion due to the region data terms are:

$$\frac{\partial E_{R_l}}{\partial \mathbf{q}_i} = \frac{1}{S(\partial \mathcal{R}_{\mathcal{M}})} \iint_{\partial \mathcal{R}_{\mathcal{M}}} \Phi_r(\mathbf{x}) \big(\nabla \Phi_{\mathcal{M}}(D(\mathbf{q}; \mathbf{x})) \cdot \frac{\partial}{\partial \mathbf{q}_i} D(\mathbf{q}; \mathbf{x})\big) d\mathbf{x}$$

$$\frac{\partial E_{R_m}}{\partial \mathbf{q}_i} = - \frac{1}{S(\partial \mathcal{R}_{\mathcal{M}})} \iint_{\partial \mathcal{R}_{\mathcal{M}}} \Big[\big(\iint_{\mathcal{R}_{\mathcal{M}}} e^{\frac{-(I(D(\mathbf{q};\mathbf{x}))-I(\mathbf{y}))^2}{2\sigma^2}} d\mathbf{y} \big)^{-1} \iint_{\mathcal{R}_{\mathcal{M}}} e^{\frac{-(I(D(\mathbf{q};\mathbf{x}))-I(\mathbf{y}))^2}{2\sigma^2}} \cdot$$

$$\big(- \frac{(I(D(\mathbf{q};\mathbf{x}))-I(\mathbf{y}))}{\sigma^2} \cdot \big(\nabla I(D(\mathbf{q};\mathbf{x})) \cdot \frac{\partial}{\partial \mathbf{q}_i} D(\mathbf{q};\mathbf{x})\big) \big) d\mathbf{y} \Big] d\mathbf{x}$$

In the above formulas, the partial derivatives with respect to the IFFD deformation parameters, $\frac{\partial}{\partial \mathbf{q}_i} D(\mathbf{q}; \mathbf{x})$, can be easily derived from the deformation formula for $D(\mathbf{q}; \mathbf{x})$ [Eq. (9)].

D. Summary of the Model Fitting Algorithm

Having defined the energy terms, the overall model fitting algorithm consists of the following steps:

1) Initialize the deformation parameters $\mathbf{q}$ to be $\mathbf{q}^0$, which indicates no deformation.

2) Compute $\frac{\partial E}{\partial \mathbf{q}_i}$ for each element $\mathbf{q}_i$ in the deformation parameters $\mathbf{q}$.

3) Update the parameters $\mathbf{q}'_i = \mathbf{q}_i - \lambda \cdot \frac{\partial E}{\partial \mathbf{q}_i}$. λ is the gradient descent step size.

4) Using the new parameters, compute the new model $\mathcal{M}' = D(\mathbf{q}'; \mathcal{M})$.

5) Update the model. Let $\mathcal{M} = \mathcal{M}'$, re-compute the implicit shape representation $\Phi_{\mathcal{M}}$, and the new partition of the image domain by the new model: $[\mathcal{R}_{\mathcal{M}}]$, $[\Omega - \mathcal{R}_{\mathcal{M}}]$, and $[\partial \mathcal{R}_{\mathcal{M}}]$. Also re-initialize a regular FFD control lattice to cover the new model, and update the ROI "shape image" ϕ_r based on the new model interior.

6) Repeat steps 1-5 until convergence.

In the algorithm, after each iteration, both model shape and model-interior intensity statistics get updated, and deformation parameters get re-initialized for the new model. This allows continuous, both large-scale and small-scale deformations for the model to converge to the energy minimum.

E. Multiple Model Initialization and Merging

When multiple models are initialized in an image, each model evolves based on its own dynamics. To allow merging and competition of the multiple models, a collision detection step is applied after every few iterations to check whether the interiors of more than one models overlap. Although collision detection requires complicated algorithms in parametric deformable models [38], it is straightforward in Metamorphs because of the implicit model

shape representation. Suppose the implicit representations for two models being tested are: $\Phi_{\mathcal{M}_a}(\mathbf{x})$ and $\Phi_{\mathcal{M}_b}(\mathbf{x})$. According to the definition of implicit shape representation (Eq. 3), $\Phi_{\mathcal{M}_a}$ and $\Phi_{\mathcal{M}_b}$ have positive values for pixels inside the model, negative values outside, and zero on the model. So to detect collision, we test every pixel $\mathbf{x}$ that has positive value in $\Phi_{\mathcal{M}_a}$. If for any such pixel $\mathbf{x}$ ($\Phi_{\mathcal{M}_a}(\mathbf{x}) > 0$), $\Phi_{\mathcal{M}_b}(\mathbf{x})$ is also positive, then a collision is detected, because $\mathbf{x}$ is inside both model $\mathcal{M}_a$ and model $\mathcal{M}_b$. Upon completion of each collision detection step, all models that collide are checked to see whether their interior intensity statistics are close; the colliding models are merged only if their statistics are sufficiently close.

Suppose a collision is detected between model A and model B. Since the model interior appearances are represented using nonparametric p.d.f.s, the Kullback-Leibler Divergence can be used to measure the dissimilarity between two p.d.f.s. Suppose the intensity p.d.f. for model A is p_A and the p.d.f for model B is p_B, then the Kullback-Leibler Divergence between the two distributions is defined by:

$$D_{p_A \| p_B} = \int_U p_A(i) log \frac{p_A(i)}{p_B(i)} di \tag{17}$$

where U denotes the set of all intensity values. If this K-L distance is sufficiently small (less than 0.5), the algorithm decides the statistics of the two model-interior regions are sufficiently close, and the two models will be merged; otherwise, the two models will keep evolving on their own.

If two models in collision are to be merged, the new model's implicit representation can be easily derived from the representations of the two models before merging. Suppose the implicit representations for the two models to be merged are: $\Phi_{\mathcal{M}_a}(\mathbf{x})$ and $\Phi_{\mathcal{M}_b}(\mathbf{x})$. Then the implicit representation for the merged model will simply be: $\Phi_{\mathcal{M}}(\mathbf{x}) = \max\left(\Phi_{\mathcal{M}_a}(\mathbf{x}), \Phi_{\mathcal{M}_b}(\mathbf{x})\right)$. Thereafter this new model's interior intensity p.d.f. is updated and it evolves in place of the two old models.

Fig. 8 shows an image of the chest where we initialize multiple models inside the objects of interest including the left and right lungs, and the left and right ventricles. The models first evolve on their own, and if any two models collide, they merge into one new model if their interior intensity statistics are sufficiently close. The converged models are shown in Fig. 8(e) to demonstrate the segmentation result.

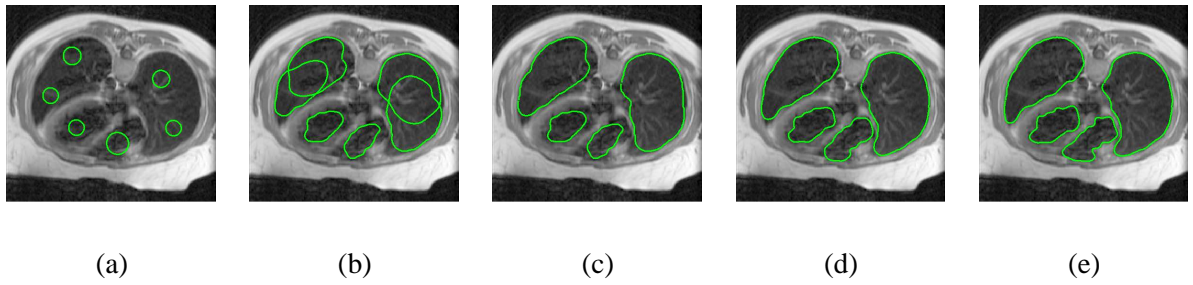(a)   (b)   (c)   (d)   (e)

Fig. 8. (a) Multiple initialized models. (b) Result after the models evolve on their own for 5 iterations. (c) Collision detection, and merging after passing the statistics tests. (d) Result after 5 more iterations. (e) Converged models after 16 iterations.
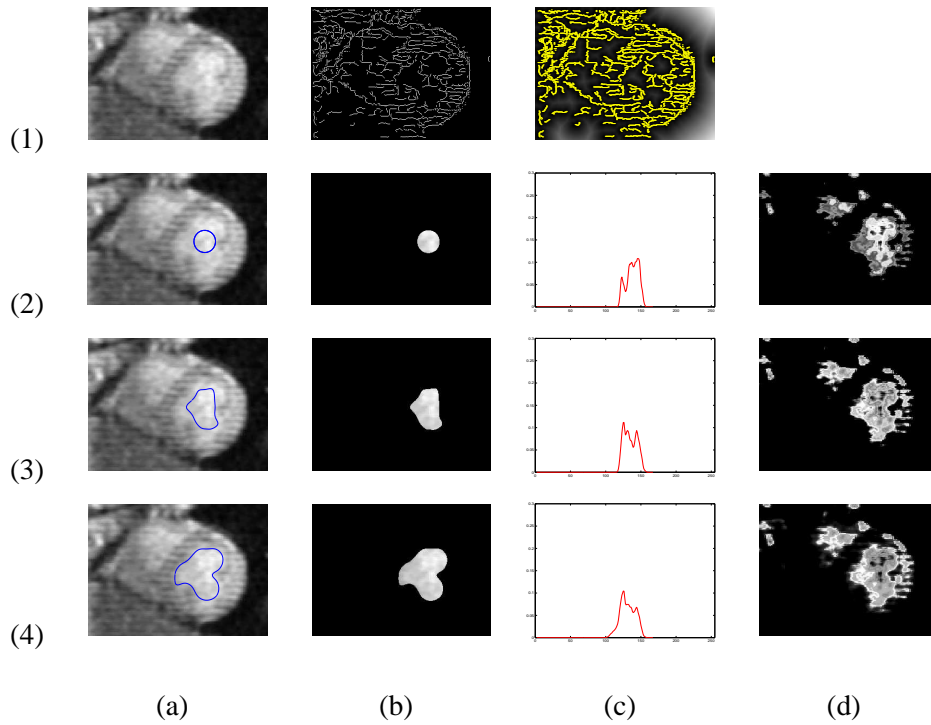

(a)   (b)   (c)   (d)

Fig. 9. Tagged MRI heart image example. (1.a) Original image. (1.b) Edge map. (1.c) "shape image" derived from the edge map. (2) Initial model. (3) Intermediate result. (4) Converged model (after 12 iterations). (2-4)(a) The evolving model. (2-4)(b) Model interior. (2-4)(c) Model interior intensity p.d.f. (2-4)(d) Intensity probability map according to the p.d.f. in (c).

IV. EXPERIMENTAL RESULTS

Some boundary finding examples using Metamorphs have been shown in Fig. 3, Fig. 6, and Fig. 8. Fig. 9 shows another example of left ventricle segmentation in a noisy tagged MRI image. Note that, due to the tagging lines and intensity inhomogeneity, the detected edges are fragmented, and there are spurious edges inside the region. In this case, the integration of edge and region information was critical in helping the model converge to the true object boundary.
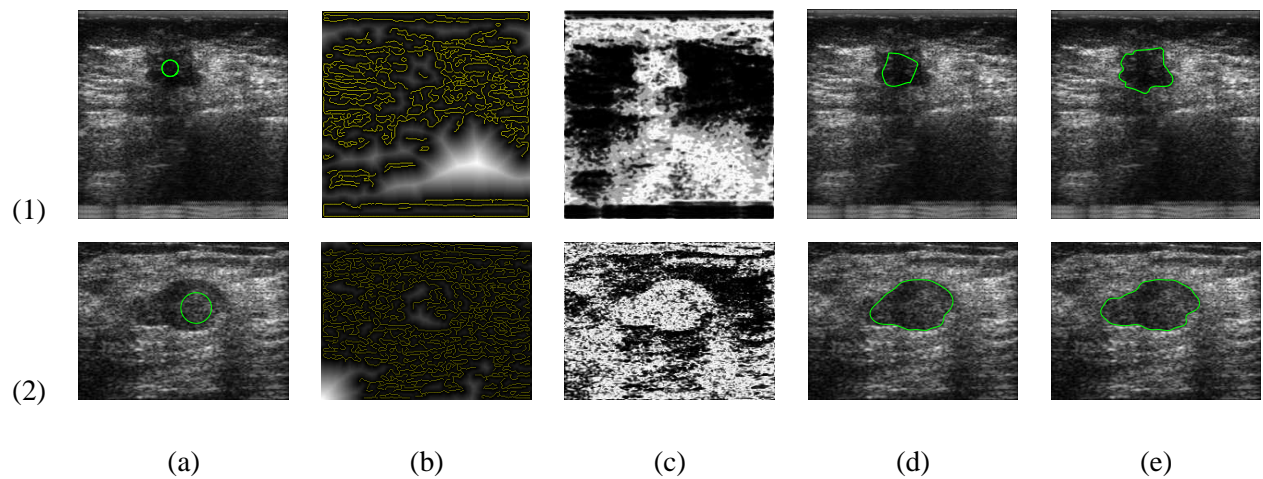
Fig. 10. Segmenting lesions in ultrasound breast images. (a) The original ultrasound image, with the initial model drawn on top, (b) The shape image derived from the edge map, (c) Intensity likelihood map, (d) Intermediate model after 4 iterations for example (1), and 13 iterations for example (2), (e) Final converged model after 11 iterations for (1) , and 20 iterations for (2).

To test the ability of Metamorphs in coping with objects whose interiors have a high level of speckle noise, we also apply the algorithm to breast lesion segmentation in ultrasound images. Fig. 10 shows two such examples. Because of the nature of ultrasound images, there is no clear contrast edges that separate a lesion from its surrounding normal tissue. The criterion in locating a lesion is usually that a lesion contains less speckle than its surroundings. One can see from Fig. 10(1.c) and 10(2.c) that the likelihood map computed based on model-interior intensity distribution captures pretty well the difference in speckle density between a lesion and its surroundings. This appearance-based region information is the key for the model to converge to the accurate boundary (Fig. 10(e)) despite very noisy speckle edges (Fig. 10(1.b)) inside the lesion.

Other than the medical images, we tested our algorithm on natural images in which specularity, shadow, non-uniform illumination, texture, occlusion and other conditions are common. Fig. 11 shows the segmentation result on an image of peppers. Several circular models are initialized, and their interiors capture the intensity variations of the three foreground peppers due to illumination, shadow and specularity. The models evolve, merge and finally converge to the result shown in Fig. 11(d-e). During model evolution, the two models initialized on the elongated pepper quickly merge after a few iterations because their interior intensities are very close; note that, although one model was poorly initialized that a small part of it covers the background, the dominant interior statistics (representing object) wins out, and the model quickly deforms to recover from the poor initialization and continues

Fig. 11. Boundary finding in the pepper image. (a) Original image, with initial models drawn on top. (b) The shape image derived from the edge map. (c) Intermediate result showing the models after 10 iterations. (d) Final converged models after 14 iterations. (e) The three pepper segments enclosed by the three converged models.
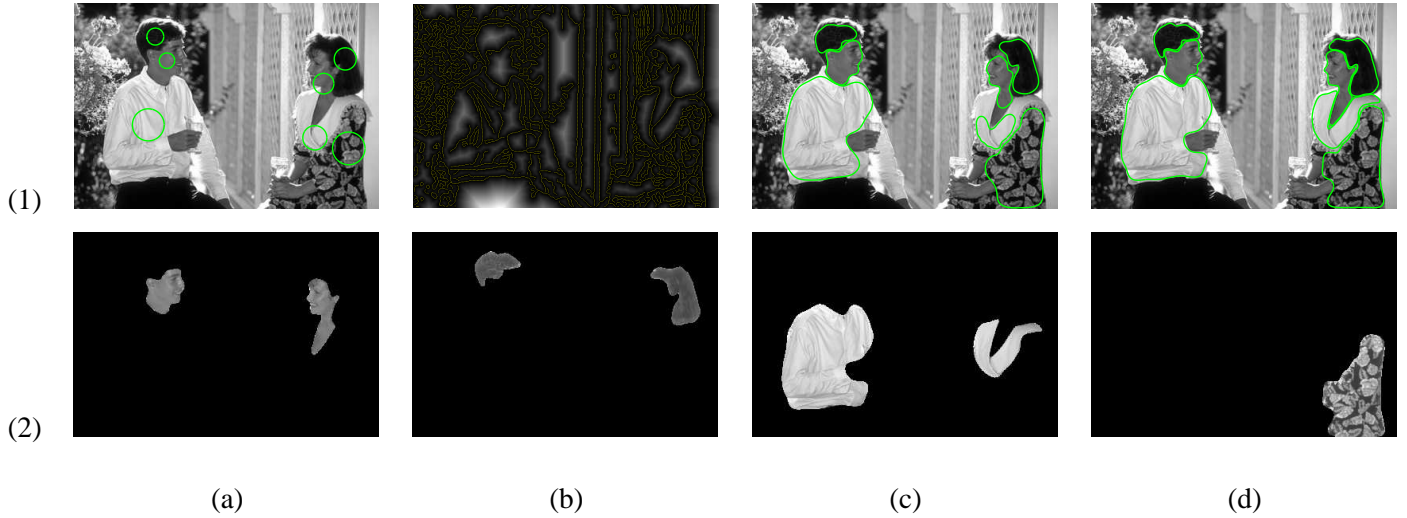

(a) (b) (c) (d)

Fig. 12. Boundary finding in a picture of people. (1) The evolution of models; (2) Segmented patches. (1.a) Original image, with initial models drawn on top. (1.b) The shape image derived from the edge map. (1.c) Intermediate result showing the models after 8 iterations. (1.d) Converged models after 22 iterations. (2.a) Face. (2.b) Hair. (2.c) White clothing. (2.d) Textured dress.

deforming as an "object" model. The two models initialized on the dark-colored pepper do not merge until the top model evolves and includes part of the specular region (see Fig. 11(c)) because only then the two model interiors have sufficiently close statistics (Eq. 17).

Fig. 12 demonstrates the experiment on a picture of people. Several circular models are initialized on the face, hair, clothes of the two people. The converged models are shown in Fig. 12(1.d). On the faces, small interior structures such as the eyes and eyebrows did not stop the models from converging to the face boundaries. The texture of the women's dress consists of large-scale patterns and multiple colors; by initializing a model whose interior captures the color changes within the texture, the model accurately converges to the texture boundary without getting stuck
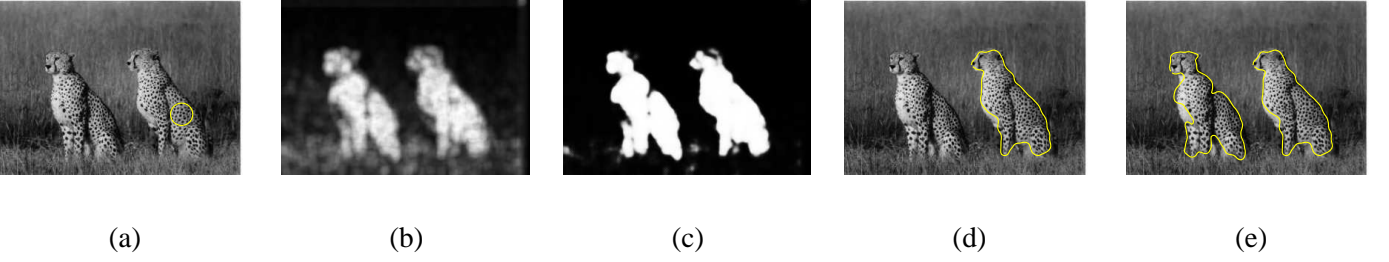
Fig. 13. (a) Original image with initial model. (b) Likelihood map based on comparing statistics of Gabor filter responses between model interior and the overall image. (c) The improved likelihood map after applying belief propagation. (d) The converged model finds the boundary of one cheetah. (e) Both cheetah boundaries detected after initializing another model in the other high-likelihood area.

at interior edges produced by the changing colors. The model on the man's white shirt stopped before including the left arm because of the strong appearance and edge boundary generated by the glass and bright sunlight. In the second row, Fig. 12(2) shows the segments enclosed by the converged models. In Fig. 12(2.a-c), the segments that are displayed together have similar intensity statistics according to the Kullback-Leibler Divergence criterion.

While the examples presented so far are based on nonparametric intensity statistics, the Metamorphs framework can be generalized to automatically deal with textures that consist of large-scale periodic patterns. The basic idea is to first compute the natural scale of the model-interior patterns (or texons). If the scale is large which means the texture consists of large-scale periodic patterns, a small bank of Gabor filters can be applied both to the model-interior and to the overall image, then the nonparametric statistics and likelihood map can be evaluated on the filter responses. The belief propagation (BP) algorithm can also be applied to improve the texture likelihood map. Suppose the likelihood map thus derived is $\mathcal{L}$, instead of explicitly finding a ROI for the balloon term as in Eq. 14, we use the following likelihood-modulated anisotropic balloon term for segmenting textured image:

$$E_{R_l} = \frac{1}{V(\partial \mathcal{R}_{\mathcal{M}})} \iint_{\partial \mathcal{R}_{\mathcal{M}}} \mathcal{L}(\mathbf{x})\big(\Phi_{\mathcal{M}}(D(\mathbf{q}; \mathbf{x}))\big) d\mathbf{x}$$

This technique has produced promising results and one example using a cheetah image is demonstrated in Fig. 13. More details on Metamorphs texture segmentation and extensive experimental results can be found in [39], [40].

A. Performance and Parameters

The Metamorphs model evolution is computationally efficient. On a typical image of size 256×256, the segmentation process takes less than $200ms$ to converge on a 2GHz PC work station with 1GB RAM. Several reasons contribute to this. First, the IFFD parameterization of model deformations significantly reduces the number of local deformation parameters, while guaranteeing the model's smoothness properties. Second, most computation only involves pixels that are either within a narrow band surrounding the model or inside the model. The only whole-image computation, which is the intensity probability map, is done efficiently using the Fast Gauss Transform [26] in linear time.

A hierarchical IFFD implementation, which uses coarse-to-fine control lattices, is able to represent both large- and small-scale deformations. In Metamorphs, the resolution of the IFFD control lattice used to cover the model, $M \times N$ in Eq. 4, is initially set to be 10×10 for all examples, and it is dynamically increased during model evolution based on feedback from the overall lattice deformation magnitude. When the magnitude is large and the model is far-away from boundary, a coarse-level IFFD enables the model to evolve quickly with global smoothness constraints. As the model gets nearer to the boundary and the lattice deformation magnitude gets smaller, the resolution of a reinitialized control lattice (see section III-D step 5) is gradually increased, to account for smaller-scale deformations so that the model can fit into detailed convexities and concavities of the object boundary. The capability of Metamorphs to converge within a small number of iterations as well as to capture high curvature features on boundaries is demonstrated through examples in Fig. 3, Fig. 8 (e.g. tips of the lung), and Fig. 12 (e.g. ears, clothing-generated corners).

The two weight factors that balance the contributions from different energy terms, k and b in Eq. (16), are estimated automatically for each Metamorphs model based on its surrounding image information. The weight factor between the edge term and the region terms, k, is determined by a confidence measure, C_e, of the computed edge map. To decide this confidence value, we compute the ROI (see section III-B.1) after initializing a model, then C_e is determined by the complexity of image gradient or edges within this ROI. The confidence value is low if there are many small edges inside the ROI; the value is high otherwise. More specifically, suppose the percentage of edge points inside the ROI is f, then the edge confidence measure $C_e = \frac{1}{f}$. Considering that the weight factor
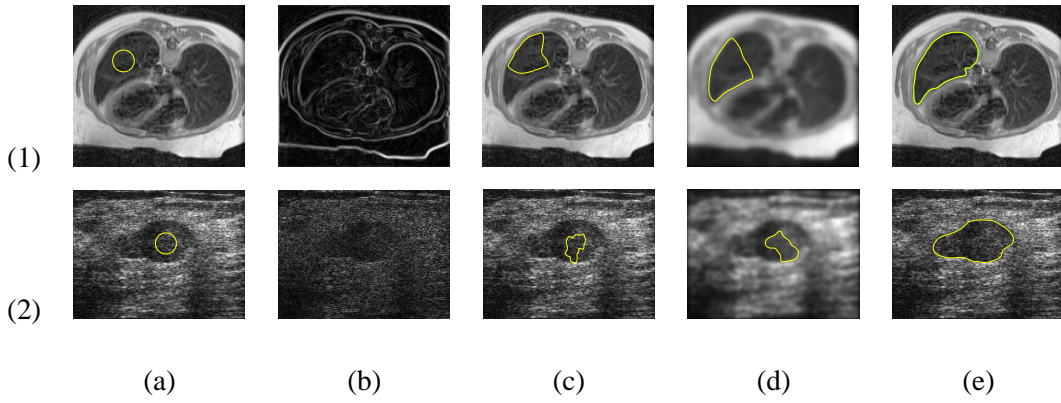
Fig. 14. Comparison between GVF Snakes and Metamorphs. (a) Original image with initial model drawn on top. (b) Gray-level edge map. (c) Result using GVF snake on original image. (d) Result using GVF snake on smoothed image. (e) Result using Metamorphs without image smoothing.

k should be inversely proportional to C_e, we set k to be proportional to f (e.g. $k = 10f$). For the other weight factor, b, we always set it to be greater than 1 (e.g. $b = 10$) so that the Maximum Likelihood term E_{R_m} is given more weight. The importance of the ML term can be seen in Fig. 6 and 10.

When multiple models are initialized on one image, the same set of parameters are used for all models. In the beginning, all models are active, and during evolution, if any model converges, its status is changed to inactive. The algorithm runs until all models converge and are no longer active.

B. Comparison with Other Boundary Finding Methods

1) Comparison with other snake models: Compared with various snake (deformable) models in the literature, the main contributions of the Metamorphs model lie in its novel way of integrating shape and appearance fundamentally at the modeling level, its unified shape and intensity representation in the pixel space, its FFD deformation parameterization, and the novel variational framework for robust boundary finding. The advantage of Metamorphs is most significant when models are initialized far away from the object boundary. Using circular model initializations in Fig. 14(1.a) and Fig. 14(2.a) for a chest MR and a breast ultrasound image respectively, we compare Metamorphs with the Gradient Vector Flow (GVF) snakes [11]. We choose to compare with GVF because they were proposed to make snake models less sensitive to initialization, and the GVF code from the original authors is available. We use the default parameter settings in the original code ($\alpha = 0.5, \beta = 0.1, \mu = 1, \kappa = 0.2$), except for the amount of image smoothing. For the chest MR image, without smoothing the original image and when initialized as in Fig.

14(1.a), a GVF snake converged after 120 iterations and the result is shown in Fig. 14(1.c). After applying Gaussian Smoothing to the image (with Gaussian kernel $\sigma = 3.0$), a GVF snake using the same initialization converged after 150 iterations and the result is shown in Fig. 14(1.d). We tried the same tasks on the breast ultrasound image with speckle noise. Without smoothing, a GVF snake converged after 80 iterations and the result is shown in Fig. 14(2.c). With Gaussian smoothing, a GVF snake converged after 100 iterations as shown in Fig. 14(2.d). In both examples, given the far-away initializations, the GVF snakes stopped early before reaching the true boundary. This can partly be understood by the essence of the GVF potential field, which is a laplacian diffusion of the gray-level edge map's gradient vectors. So when strong image gradients (edges) or small islands remain inside the object even after smoothing, the GVF snake gets attracted to them and gets stuck in local minima. This behavior is also typical of other types of deformable models that rely on image gradient or edge information alone.

For comparison, we show the segmentation result using Metamorphs with the same initializations. The models are run on the original images without smoothing. The Metamorphs model on the chest MR image reached convergence after 24 iterations (Fig. 14(1.e)), and the model on the breast ultrasound image reached convergence after only 18 iterations (Fig. 14(2.e)).

2) Comparison with region-based segmentation methods: Region-based segmentation methods such as region growing [41], Markov Random Fields [42], and graph cuts [43] group pixels whose intensities follow consistent statistics, hence they are less sensitive to localized image noise than edges. Metamorphs has this grouping property because of its appearance test based on nonparametric statistics and its region-based energy terms. Unlike region-based methods however, Metamorphs is an efficient model-based approach that includes implicit smoothness constraints on the model surface and thus directly generates smooth boundary, without the need for additional post-processing steps to extract boundary from pixel clusters and to smooth the boundary. Furthermore, Metamorphs is particularly good at segmenting an object that has gradually changing and nonuniform intensity or texture appearance; this is because Metamorphs also takes into account edge information and the adaptive nature of Metamorphs enables the model to learn on-line an accurate representation for the appearance statistics of the object.

In Fig. 15, we compare the results from Markov Random Fields (MRF) with that from Metamorphs, using a

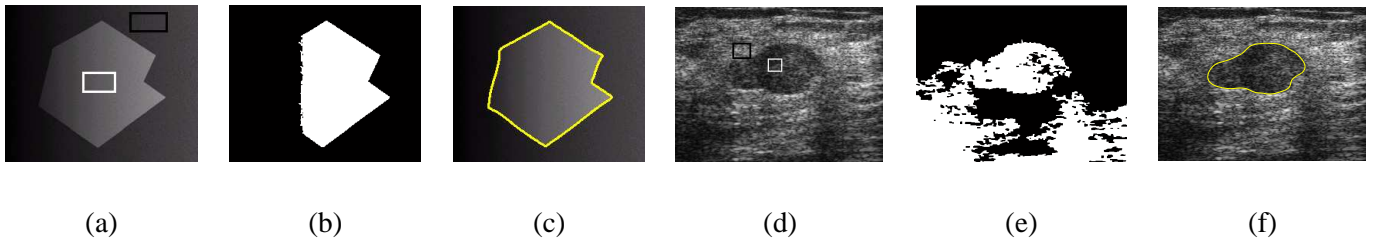| (a) | (b) | (c) | (d) | (e) | (f) |

Fig. 15. Comparing Markov Random Fields (MRF) with Metamorphs. (a) & (d) two-class initialization for MRF: object-class sample patches are enclosed by white rectangles, and background-class sample patches are enclosed by black rectangles. (b) & (d) MRF segmentation results using the algorithm described in [42]. The main connected component of the object class is rendered in white, and the background class is rendered in black. (c) & (f) Metamorphs segmentation results for comparison.

synthetic example which contains an object with gradually-changing intensity from left to right (Fig. 15(a)-(c)), and an ultrasound image with speckle noise (Fig. 15(d)-(f)). The MRF implementation is based on the supervised Bayesian MRF image classification algorithm described in [42]. We specified the images consisting of two classes: the *object* class and the *background* class. Given a sample patch for each class (Fig. 15(a) and Fig. 15(d)), the algorithm computes the intensity mean and variance for each class and applies MRF to perform classification. For the synthetic image, the MRF segmentation result after 268 iterations (with final temperature value $T = 0.018$) is shown in Fig. 15(b), and the result after 346 iterations (with final temperature value $T = 0.019$) for the breast ultrasound image is shown in Fig. 15(e). The temperature values and number of iterations are automatically determined by the MRF algorithm. Note that we only display the main connected component of the *object* class from the MRF result for fair comparison with Metamorphs. One can see that the MRF segmentation failed on the synthetic image because of the heavily inhomogeneous intensity. Other than the initialization sample patches shown in Fig. 15(a), we also tried many other sample patch choices including patches that cover the entire spectrum of intensity for the background or the object but none could produce the correct segmentation. MRF also failed on the ultrasound image since it did not separate the lesion object from part of the background that has similar statistics and it generated small holes/islands inside the object. In comparison, our model-based method can accurately locate the object boundaries by combining edge information, adaptive object appearance learning and model smoothness constraints; the clean boundaries found by Metamorphs can be seen in Fig. 15(c) and Fig. 15(f).

3) Comparison with other works that integrate boundary and region information: Compared with previous works that integrate region and boundary information for image segmentation [15], [44], [45], [24], Metamorphs is more of a novel deformable modeling framework that fundamentally unifies the representation and deformation schemes for shape and intensity, and as a result, the models naturally integrate region and boundary information when applied to segmentation. Experimentally we compare Metamorphs with a recent integration work, Geodesic Active Regions (GAR) [24].

Although both GAR and Metamorphs use the implicit shape representation, their segmentation and model deformation frameworks are very different. First, GAR is a frame partition framework that integrates edge and region based modules to partition an image into conspicuous regions, while Metamorphs is a deformable model based object segmentation framework that finds the boundary of a foreground object without being interfered by the possibly cluttered background. Second, GAR is a supervised segmentation method that requires knowing a *priori* the number of segments in an image and learning the intensity statistics of each region off-line using a Mixture-of-Gaussian approximation. In contrast, Metamorphs does not need prior training, and it learns adaptively the model (and foreground object) interior's intensity statistics using a nonparametric kernel-based method as the model deforms. Third, GAR uses a completely implicit level set technique to solve for a set of smooth curves that separate regions in an image, while Metamorphs combines the implicit shape representation with explicit FFD-based parametric deformations. Having the explicit FFD deformations enables our method to keep track of correspondences during model evolution, which could prove useful in applications such as tracking [46]. The parametric FFD deformations also make Metamorphs more computationally efficient. As far as the computational cost is concerned, on a 256×256 image, GAR has a modeling (learning) stage that takes less than one second, and a narrow-band level set segmentation stage that takes 23 to 25 seconds on a ultra-Sun work station with 250MB RAM and 299 MHZ processor [47]; Metamorphs segmentation converges in less than 30 iterations and takes less than $200ms$ on a Dell PC work station with 1GB RAM and 2GHz Pentium Processor, which is about *three* times faster than GAR. In terms of accuracy, we compare the supervised GAR segmentation with Metamorphs segmentation on a Cheetah image as shown in Fig. 16. The GAR result is from the original authors. Using Metamorphs, we do not need a prior training step, instead we directly initialize circular models inside the cheetah object, then let the
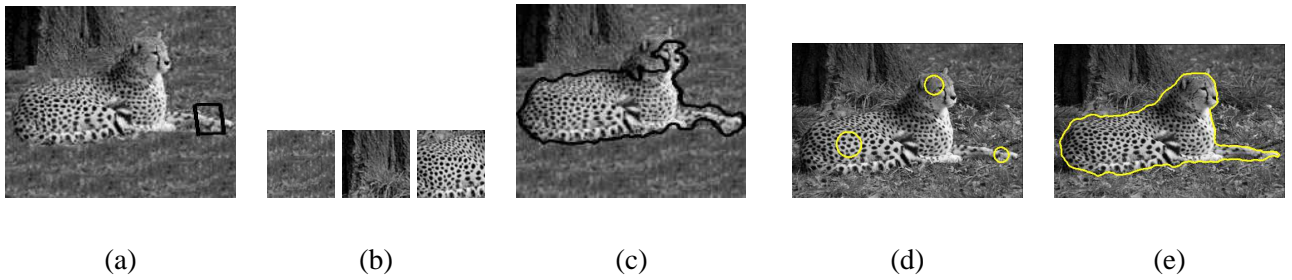
Fig. 16. Comparison between Geodesic Active Regions and Metamorphs. (a) Original Cheetah image from [47] with initial contour. (b) Three window patterns selected to represent three different texture regions in the image for GAR supervised learning. (c) GAR segmentation result based on models learned from the three patterns in (b). (d) Cheetah image with initial circular Metamorphs models. (e) Metamorphs model converged after 33 iterations.

models evolve toward the boundary.

V. DISCUSSION

In this paper, we assume user-guided model initialization. That is, the user initializes one or several circular models within an object of interest by clicking two points for each model: the first point is the centroid, and the distance between the first and the second points specifies the radius. Our method is robust to poor initializations where a model covers part of the background, as long as the majority of the model interior overlaps with the object of interest. We could also potentially automate the initialization process through supervised learning of object appearance statistics.

When presented with different initialization conditions for the same task, the converged Metamorphs models do have small differences, but the variations are mostly within a small range around the ground truth. Examples of such variations can be seen in two separate examples shown in Figs. 8(e), 14(1.e), and Figs. 10(2.e), 14(2.e). Since it is difficult even for humans to reach consensus on a unique ground truth for most segmentation tasks, we believe a good strategy is to take the average of several converged models that resulted from different initializations, when high segmentation accuracy is desired.

The use of an implicit model shape representation allows for topological changes. In Metamorphs, instead of curve evolution as in level-set segmentation, deformation is controlled by parametric FFD, hence pixel-wise displacements are mapped to FFD control point deformations. During the mapping, discontinuities are smoothed out, thus it is less likely for self-intersections and topological changes to happen in Metamorphs than in curve evolution. Topological

changes other than controlled merging (see Section III-E) could appear however, if the deformation step size is too large and the FFD control lattice folds over on itself. Since the FFD lattice is re-initialized after each iteration, a reasonable step size typically prevents control lattice fold-over and guarantees the preservation of topology during model evolution. In some applications where the object to be segmented has a known topology that must be preserved, explicit topology-preserving methods such as TLSM [48] can be used in Metamorphs to combine implicit representation and digital topology to guarantee the preservation of topology.

The Metamorphs framework can be extended to work on color images and 3D images. On color images, edges can be computed using color edge detectors, and the region-based likelihood maps can be considered in a color space (e.g. RGB, L*ab). In 3D, the implicit model shape representation, the kernel-based intensity p.d.f. estimation, and the definitions for all edge- and region-based energy terms remain the same. The IFFD use regular control lattices in 3D and a 3D tensor product of B-spline polynomials.

The Metamorphs formulation in this paper does not model compound objects with "holes" inside the deformable "disk" model. How to automatically detect holes and differentiate between holes and spurious islands caused by noise and texture is an interesting question to address. One possible solution is to keep a probabilistic mask that records which pixels inside the model have low likelihood values in the intensity probability map. If these pixels have consistently low likelihood values for a number of iterations and they connect to cover a relatively large area inside the model, they are likely part of a hole in the model. If a hole is detected, its interior will be excluded from computing the intensity statistics of the model and the hole boundary will not affect the model evolution.

It should be noted that all energy terms in Metamorphs are defined on the implicit representations (i.e. distance maps) of shape, edge, or ROI boundary. Because the implicit representation has the property that its first derivatives are continuous and the gradient of the distance function is a unit vector in the normal direction of the embedded shape/edge/boundary, gradient descent optimization of energy functions defined on such implicit functions are guaranteed to deform smoothly and converge quickly. Indeed, we have observed smooth model-deformation trajectories in all our experiments.

Since Metamorphs segmentation is an online algorithm without prior knowledge about the statistical shape distribution of the object of interest, it is generic and can be applied to segmenting arbitrary objects. On the

other hand, it is possible that Metamorphs will fail to find the accurate object boundary when there is neither edge nor obvious appearance difference between the foreground and part of the background objects; this happens, for instance, when segmenting ventricular structures in MRI brain scans [6]. In these cases, statistical shape and appearance prior models such as those introduced in [49] would help if training data for learning such models are available.

Another interesting future direction of our framework is to explore dynamically changing edge maps instead of one static edge map precomputed using canny edge detector. This is analogous to the existing capability of our framework to learn and exploit the dynamically changing model-interior intensity statistics, which distinguishes the method from other model-based region segmentation techniques.

VI. Conclusions

We have presented a new class of deformable models, Metamorphs, which possess both boundary shape and interior intensity statistics. The main contributions of the work lie in several aspects. First, the Metamorphs models represent a generalization of previous *parametric* and *geometric* deformable models by taking into account model-interior region information while being computationally efficient. Second, the proposed framework does not require learning statistical shape and appearance models *a priori*, but the model deformations are constrained such that interior statistics of the model after deformation are consistent with the statistics learned adaptively from the past history of the model interiors. Third, compared to other works that integrate edge and region information for segmentation, the integration in our framework is fundamental at the modeling level. By unifying shape and appearance representations in the pixel space, the framework does not have separate parameters for model shape and interior appearance. The only set of parameters is the IFFD parameters that specify model deformation, and as the model deforms, its implicit shape representation and its interior nonparametric intensity statistics get updated automatically. When applied to boundary finding, the Metamorphs dynamics are derived from edge and region energy terms that are both differentiable with respect to the IFFD deformation parameters in a common variational framework. The Metamorphs framework in general can also be applied to 3D and color segmentation, as well as to other applications such as tracking and shape reconstruction.

Rᴇғᴇʀᴇɴᴄᴇs

[1] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int'l Journal of Computer Vision*, vol. 1, pp. 321–331, 1987.

[2] L. H. Staib and J. S. Duncan, "Boundary finding with parametrically deformable models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1061–1075, 1992.

[3] L. D. Cohen and I. Cohen, "Finite-element methods for active contour models and balloons for 2-D and 3-D images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1131–1147, 1993.

[4] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," in *Proc. of IEEE Int'l Conf. on Computer Vision*, 1995, pp. 694–699.

[5] R. Malladi, J. Sethian, and B. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.

[6] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[7] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *Proc. of European Conf. on Computer Vision*, vol. 2, 1998, pp. 484–498.

[8] M. E. Leventon, E. L. Grimson, and O. Faugeras, "Statistical shape influence in geodesic active contours," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, 2000, pp. 1316–1323.

[9] D. Metaxas, *Physics-Based Deformable Models*. Kluwer Academic Publishers, 1996.

[10] T. McInerney and D. Terzopoulos, "A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis," *Computerized Medical Imaging and Graphics*, vol. 19, no. 1, pp. 69–83, 1995.

[11] C. Xu and J. L. Prince, "Snakes, shapes and gradient vector flow," *IEEE Trans. on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[12] A. Yezzi, S. Kichenssamy, A. Kumar, P. Olver, and A. Tannebaum, "A geometric snake model for segmentation of medical imagery," *IEEE Trans. on Medical Imaging*, vol. 16, no. 2, pp. 199–209, 1997.

[13] T. Chan and L. Vese, "Active contours without edges," *IEEE Trans. on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[14] R. Ronfard, "Region-based strategies for active contour models," *Int'l Journal of Computer Vision*, vol. 13, no. 2, pp. 229–251, 1994.

[15] S. Zhu and A. Yuille, "Region Competition: Unifying snakes, region growing, and Bayes/MDL for multi-band image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884–900, 1996.

[16] T. Jones and D. Metaxas, "Automated 3D segmentation using deformable models and fuzzy affinity," in *Proc. of Information Processing in Medical Imaging*, 1997, pp. 113–126.

[17] T. Chen and D. Metaxas, "Image segmentation based on the integration of Markov Random Fields and deformable models," in *Proc. of Int'l Conf. on Medical Imaging Copmuting and Computer-Assisted Intervention*, 2000, pp. 256–265.

[18] R. Huang, V. Pavlovic, and D. Metaxas, "A graphical model framework for coupling mrfs and deformable models," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 739–746.

[19] S. Jehan-Besson, M. Barlaud, and G. Aubert, "Shape gradients for histogram segmentation using active contours," in *Proc. of IEEE Int'l Conf. on Computer Vision*, 2003, pp. 408–415.

[20] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson, "Image segmentation using active contours: Calculus of variations or shape gradients," *SIAM Applied Mathematics*, vol. 63, no. 6, pp. 2128–2154, 2003.

[21] A. J. Yezzi, A. Tsai, and A. Willsky, "A statistical approach to snakes for bimodal and trimodal imagery," in *Proc. of IEEE Int'l Conf. on Computer Vision*, vol. 2, 1999, pp. 898–903.

[22] C. Samson, L. Blanc-Feraud, G. Aubert, and J. Zerubia, "A level set model for image classification," *Int'l Journal of Computer Vision*, vol. 40, no. 3, pp. 187–198, 2000.

[23] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *Int'l Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.

[24] N. Paragios and R. Deriche, "Geodesic active regions and level set methods for supervised texture segmentation," *Int'l Journal of Computer Vision*, vol. 46, no. 3, pp. 223–247, 2002.

[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. Elgammal, R. Duraiswami, and L. S. Davis, "Efficient kernel density estimation using the Fast Gauss Transform with applications to color modeling and tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1499–1504, 2003.

[27] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[28] A. A. Amini, Y. Chen, M. Elayyadi, and P. Radeva, "Tag surface reconstruction and tracking of myocardial beads from SPAMM-MRI with parametric B-spline surfaces," *IEEE Trans. on Medical Imaging*, vol. 20, no. 2, pp. 94–103, 2001.

[29] P. Faloutsos, M. van de Panne, and D. Terzopoulos, "Dynamic free-form deformations for animation synthesis," *IEEE Trans. Visualization and Computer Graphics*, vol. 3, pp. 201–214, 1997.

[30] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on the Hamilton-Jacobi formulation," *J. of Computational Physics*, vol. 79, pp. 12–49, 1988.

[31] T. W. Sederberg and S. R. Parry, "Free-form deformation of solid geometric models," in *Proc. of the 13th Annual Conf. on Computer Graphics*, 1986, pp. 151–160.

[32] X. Huang, N. Paragios, and D. Metaxas, "Shape registration in implicit spaces using information theory and free form deformations,"

IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 28, no. 8, pp. 1303–1318, 2006.

[33] Y. S. Akgul and C. Kambhamettu, "A coarse-to-fine deformable contour optimization framework," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 174–186, 2003.

[34] R. O. Duda and P. Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

[35] D. W. Scott, *Multivariate Density Estimation*. Wiley-Interscience, 1992.

[36] A. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *Proc. of European Conf. on Computer Vision*, ser. LNCS, vol. 1843, 2000, pp. 751–767.

[37] R. M. Haralick and L. Shapiro, *Computer and Robot Vision*. Addison-Wesley, 1992.

[38] N. Govindaraju, D. Knott, N. Jain, I. Kabul, R. Tamstorf, R. Gayle, M. Lin, and D. Manocha, "Interactive Collision Detection between Deformable Models using Chromatic Decomposition," in *ACM Trans. on Graphics (TOG), Proc. of ACM SIGGRAPH 2005*, vol. 24, no. 3, 2005, pp. 991–999.

[39] X. Huang, Z. Qian, R. Huang, and D. Metaxas, "Deformable-model based textured object segmentation," in *4th Int'l Workshop on energy minimization methods in CVPR (EMMCVPR)*, ser. LNCS, vol. 3757, 2005, pp. 119–135.

[40] L. Wolf, X. Huang, I. Martin, and D. Metaxas, "Patch-based texture edges and segmentation," in *Proc. of European Conf. on Computer Vision*, ser. LNCS, vol. 3952, 2006, pp. 481–493.

[41] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*. Second Edition, PWS Publishing, 1999.

[42] M. Berthod, Z. Kato, S. Yu, and J. Zerubia, "Bayesian image classification using Markov Random Fields," *Image and Vision Computing*, vol. 14, pp. 285–295, 1996.

[43] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[44] A. Chakraborty, L. Staib, and J. Duncan, "Deformable boundary finding in medical images by integrating gradient and region information," *IEEE Trans. on Medical Imaging*, vol. 15, no. 6, pp. 859–870, 1996.

[45] A. Chakraborty and J. S. Duncan, "Game-theoretic integration for image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 1, pp. 12–30, 1999.

[46] K. Karantzalos and N. Paragios, "Implicit free-form-deformations for multi-frame segmentation and tracking," in *IEEE Workshop on Variational and Level Set Methods in Computer Vision*, 2005.

[47] N. Paragios, "Geodesic active regions and level set methods: Contributions and applications in artificial vision," *Ph.D. Dissertation, University of Nice Sophia Antipolis, School of Computer Engineering*, 2000.

[48] H. X., C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 755–768, 2003.

[49] X. Huang, Z. Li, and D. Metaxas, "Learning coupled prior shape and appearance models for segmentation," in *Proc. of Int'l Conf. on Medical Imaging Copmuting and Computer-Assisted Intervention*, vol. 1, 2004, pp. 60–69.

[50] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating

segmentation algorithms and measuring ecological statistics," in *Proc. of IEEE Int'l Conf. on Computer Vision*, vol. 2, 2001, pp. 416–423.