To appear in the Journal of Computerized Medical Imaging and Graphics, 1994.

A Dynamic Finite Element Surface Model for Segmentation and Tracking in Multidimensional Medical Images with Application to Cardiac 4D Image Analysis

Tim McInerney and Demetri Terzopoulos

Department of Computer Science, University of Toronto, Toronto, ON, Canada M5S 1A4

Abstract

This paper presents a physics-based approach to anatomical surface segmentation, reconstruction, and tracking in multidimensional medical images. The approach makes use of a dynamic "balloon" model—a spherical thin-plate under tension surface spline which deforms elastically to fit the image data. The fitting process is mediated by internal forces stemming from the elastic properties of the spline and external forces which are produced from the data. The forces interact in accordance with Lagrangian equations of motion that adjust the model's deformational degrees of freedom to fit the data. We employ the finite element method to represent the continuous surface in the form of weighted sums of local polynomial basis functions. We use a quintic triangular finite element whose nodal variables include positions as well as the first and second partial derivatives of the surface. We describe a system, implemented on a high performance graphics workstation, which applies the model fitting technique to the segmentation of the cardiac LV surface in volume (3D) CT images and LV tracking in dynamic volume (4D) CT images to estimate its nonrigid motion over the cardiac cycle. The system features a graphical user interface which minimizes error by affording specialist users interactive control over the dynamic model fitting process.

Keywords: 3D/4D Medical Image Analysis, Deformable Models, Finite Elements, Dynamics, Cardiac LV Segmentation, Nonrigid Motion Tracking, Visualization, Interaction.

1 Introduction

CT, MRI, PET and other noninvasive medical imaging technologies can provide exceptional views of internal anatomical structures, but the computer aided visualization, manipulation, and quantitative analysis of the multidimensional image data they produce is still limited. State-of-the-art medical imagers generate massive databases of static volume (3D) and dynamic volume (4D) images. These data sets, which usually suffer from sampling artifacts, spatial aliasing, and noise, are essentially "blocks of granite" with meaningful embedded structures. An important problem is to extract the surface elements belonging to an anatomical structure (the segmentation step) and to integrate these surface elements into a globally coherent surface model of the structure (the reconstruction step). Certain diagnostic procedures also require the tracking and deformation analysis of nonrigidly moving anatomical surfaces; e.g., the stretching of the left ventricle (LV) during the cardiac cycle is directly related to heart condition. The ease and accuracy of such procedures can be critically dependent upon the model used. Dynamic models are needed which are robust against noise-corrupted data and which are capable of accurately representing the complex geometries of anatomical surfaces while permitting the quantitative measurement of highly nonrigid tissue kinematics.

This paper describes a physics-based modeling approach that addresses the surface segmentation and reconstruction problems, as well as the geometric analysis and nonrigid motion estimation problems. We develop a dynamic, elastically deformable surface model whose deformation is governed by basic laws of nonrigid motion. The formulation of the motion equations includes a strain energy, simulated forces, and other physical quantities. The surface strain energy stems from a thin-plate under tension variational spline. Deformation results from the action of internal spline forces which impose surface continuity constraints and external forces which fit the surface to the image data. The inherently dynamic formulation of the model makes it suitable both for static anatomical surface reconstruction and for problems involving the reconstruction and tracking of nonrigidly moving organs.

To deal with closed anatomical surfaces, we formulate a deformable "balloon" model that is topologically isomorphic to a sphere. We employ the finite element method to spatially discretize the balloon, uniformly tessellating it into a set of connected triangular element domains. The finite element method provides an analytic, piecewise polynomial surface representation that is (C^1) continuous across triangles. We use a quintic finite element whose nodal variables include not only the nodal positions, but also the first and second parametric partial derivatives of the surface. The element is naturally suited to the surface energy functional because these same partial derivatives occur in the thin-plate under tension energy expression. The existence of parametric derivative nodal variables facilitates the computation of the differential properties of the modeled surface. In particular, the nodal variables and their time derivatives can be useful for computing the surface curvature, enclosed volume, and motion properties of anatomical surfaces.

We have implemented a system on a high performance graphics workstation which applies the dynamic model fitting technique to the segmentation of the LV surface in cardiac volume (3D) CT images and LV tracking in dynamic volume (4D) CT images in order to estimate nonrigid LV motion over the cardiac cycle. The system includes a graphical user interface which provides interactive visualization and affords control over the model fitting process. The interface allows a user to select the initial size and location of the model and to exert interactive forces on the model as it deforms to fit the data. This type of interactive control is desirable in medical image analysis applications where there is low tolerance for inaccuracy, because it allows specialist users to exploit their knowledge to correct model fitting errors.

2 Background

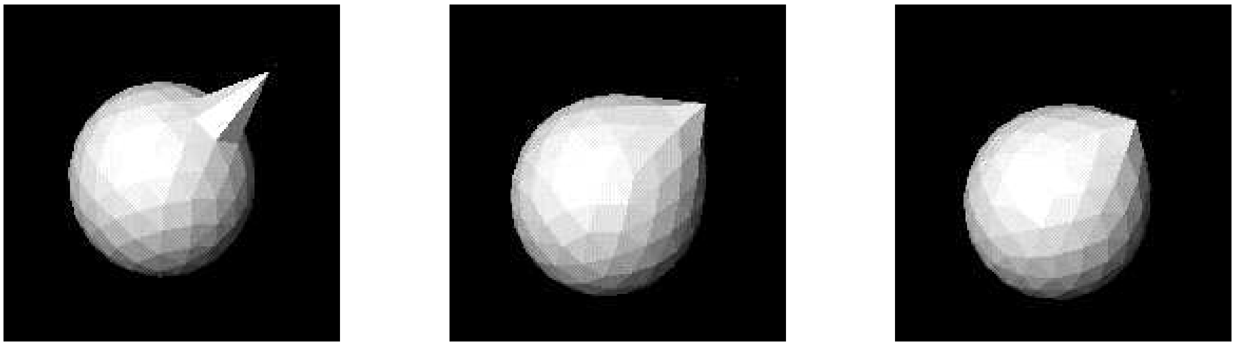
The literature on segmentation and surface reconstruction in 3D medical images includes both manual and automatic techniques. The dominant manual method is slice-editing. In manual slice-editing a skilled operator, using a computer mouse, pen, or trackball traces the region of interest on each slice of the volume. This labor intensive method suffers from many drawbacks, such as difficulties in achieving reproducible segmentation results, difficulties in comparing measurements from different operators, and difficulties deducing 3D structure from 2D slices. The technique can be speeded up and made more reproducible, however, through the use of contour extraction methods such as interactive snakes [1, 2].

The traditional automatic segmentation methods, such as density thresholding and the application of (2D or 3D) edge operators, have many well-known problems. Edge detection and the more recent marching cube [3] technique reduce volume data into something that is more readily displayed through 3D graphics, such as surface elements. However, they employ only the local properties of the image data; hence, they raise the difficult problem of establishing the connectivity of surface trace elements in order to assemble sensible global surface structures [4]. These difficulties have prompted some researchers to settle for merely visualizing the volume data in its original form using morphology [5] or volume rendering techniques [6]. Unlike global surface models, however, these voxel-display representations do not attempt to capture the geometric structure of anatomical structures; hence, they do not treat the data in a manner consistent with the physical properties of the imaged objects.

Deformable surface models are a promising approach to extracting anatomically meaningful structures from volume data. The dynamic form of the deformable model fitting technique described in this paper was first introduced by Terzopoulos, Witkin, and Kass [7]. They proposed a dynamic deformable cylinder model constructed from generalized splines, along with force field techniques to fit the model to image data. This dynamic approach is being pursued by several researchers in computational vision [8, 9, 10, 11, 12, 13, 14]. The use of finite element representations for variational problems in vision were first explored in [15]. Our formulation applies the finite element method to the thin-plate under tension spline proposed in [16] in order to derive discrete nonrigid dynamics equations. The finite element representation yields piecewise continuous deformable surface models that generally require fewer variables for similar accuracy compared to finite difference approaches.

Our work is related to that of Young [17][18] and Cohen and Cohen [19, 20] who also develop 3D deformable surface models which are based on the thin-plate under tension spline. Young fits an open bicubic Hermitian finite element based surface to the 3D locations of the coronary arteries at diastasis. The parameters of the time–varying displacement field were then fitted to the tracked displacements of the bifurcation points of the coronary arteries. Cohen and Cohen fit a cylindrical, bicubic Hermitian finite element based surface to MRI images of the LV. Another relevant deformable model is the discrete model developed by Miller *et al.* [21], which is subdivided and fitted to CT volume images by a relaxation process that minimizes a set of constraints such as the distance to the data or the local curvature of the model.

In our work we develop a closed 3D surface model based on a quintic triangular finite element with position and derivative nodal variables. The model begins as a uniformly tesselated icosahedron which may subdivide repeatedly to attain the desired geometric resolution. Our model is dynamic in the sense that it undergoes deformations that are governed by non-



(a) (b) (c)

Figure 1: Balloon models with varying elasticity and pulled by a spring point force.

rigid Lagrangian mechanics. Note, however, that although these dynamics equations serve well in model fitting and tracking using multidimensional data sets, we make no attempt to model the actual biomechanical properties of the anatomical structure under consideration (such as the cardiac LV; see, e.g., [22]).

3 Dynamic Deformable Balloon Model

The balloon model that we develop in this paper is constructed of the simulated thin-plate material under tension. The deformation energy of the material serves as a constraint which compels the model to vary smoothly almost everywhere. The balloon is represented as a vector-valued parametric function $\mathbf{x}(u,v) = [x(u,v), y(u,v), z(u,v)]^\top$ where vector $\mathbf{x}$ represents the positions of material points (u,v) relative to a reference frame in Euclidean 3-space.

The deformation energy of the thin-plate under tension material is given by the energy functional

$$\mathcal{E}_{\mathrm{p}}(\mathbf{x}) = \iint \alpha_{10}\left|\frac{\partial \mathbf{x}}{\partial u}\right|^2 + \alpha_{01}\left|\frac{\partial \mathbf{x}}{\partial v}\right|^2 + \beta_{20}\left|\frac{\partial^2 \mathbf{x}}{\partial u^2}\right|^2 + \beta_{11}\left|\frac{\partial^2 \mathbf{x}}{\partial u \partial v}\right|^2 + \beta_{02}\left|\frac{\partial^2 \mathbf{x}}{\partial v^2}\right|^2 du\, dv. \quad (1)$$

$\mathcal{E}_{\mathrm{p}}$ is a controlled-continuity spline defined in [16]. The nonnegative weighting functions $\alpha_{ij}(u,v)$ and $\beta_{ij}(u,v)$ control the elasticity of the material. The α_{10} and α_{01} functions control the tensions in the u and v directions, respectively, while the β_{02} and β_{20} functions control the corresponding bending rigidities, and the β_{11} function controls the twisting rigidity. Increasing the α_{ij} has a tendency to decrease the surface area of the material, while increasing the β_{ij} tends to make it less flexible. In general, the weighting functions may be used to introduce depth and orientation discontinuities in the material. In this paper, however, we do not make use of this capability and set the functions to constant values $\alpha_{ij}(u,v) = \alpha_{ij}$ and $\beta_{ij}(u,v) = \beta_{ij}$. Figure 1 shows the thin plate under tension balloon pulled radially by a spring point force (in (a) $\alpha_{ij} = 0.8$ and $\beta_{ij} = 0$, in (b) $\alpha_{ij} = \beta_{ij} = 0.5$, and in (c) $\alpha_{ij} = 0$ and $\beta_{ij} = 0.8$):

A general and elegant approach to fitting deformable surface models to data, especially when the data are time-varying, is to make the models *dynamic*. A dynamic formulation imposes a natural temporal continuity on the model, thereby permitting a smoothly animated display of the data fitting process. It also allows a user to interact with the model by applying constraint forces to pull it out of local minima towards the correct solution.

In a Lagrangian dynamics formulation, the positions of material points becomes a time-dependent function $\mathbf{x}(u, v, t)$ and we imbue the simulated material with mass and damping densities. The deformation energy yields internal elastic forces, and $\mathcal{E}_{\mathrm{p}}(\mathbf{x})$ is minimized when these forces equilibrate against externally applied forces and the model stabilizes: $\partial\mathbf{x}/\partial t = \partial^2\mathbf{x}/\partial t^2 = \mathbf{0}$.

The dynamic behavior of the balloon model during the fitting process is governed by the second-order partial differential equations

$$\mu\frac{\partial^2\mathbf{x}}{\partial t^2} + \gamma\frac{\partial\mathbf{x}}{\partial t} + \delta_{\mathbf{x}}\mathcal{E}_{\mathrm{p}} = \mathbf{f}, \tag{2}$$

where the first term represents the inertial forces due to the mass density $\mu(u, v)$, the second term represents the damping forces due to the damping density $\gamma(u, v)$, the third term represents the elastic force which resist deformation, and finally $\mathbf{f}(u, v, t)$ represents the external forces derived from the image data. The (generally nonlinear) data forces may be formalized as stemming from a data functional

$$\mathcal{E}_{\mathrm{d}}(\mathbf{x}) = -\iint \mathbf{x}^\top\mathbf{f}\, du\, dv. \tag{3}$$

4 Finite Element Representation

The finite difference method or the finite element method are applicable to computing numerical solutions to the function $\mathbf{x}(u, v, t)$. Finite difference solutions approximate the continuous function $\mathbf{x}$ as a set of discrete nodes in space. A disadvantage of the finite difference approach is that the continuity of the solution between nodes is not made explicit. The finite element method, on the other hand, provides continuous surface approximations; that is, the method approximates the unknown function $\mathbf{x}$ in terms of combinations of local basis functions [23].

To apply the finite element method to our models, we tessellate the continuous material domain (u, v) into a mesh of M element subdomains E_j. We approximate $\mathbf{x}$ as a weighted sum of piecewise polynomial basis functions $\mathbf{N}_i$:

$$\mathbf{x}(u, v, t) \approx \hat{\mathbf{x}}(u, v, t) = \sum_{i=1}^{n} \mathbf{N}_i(u, v)\mathbf{q}_i(t), \tag{4}$$

where $\mathbf{q}_i$ is a vector of nodal variables associated with mesh node i.

Substituting (4) into (2) yields the discrete equations of motion

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f}_q, \tag{5}$$

with $\mathbf{q} = [\mathbf{q}_1^\top, \ldots, \mathbf{q}_i^\top, \ldots, \mathbf{q}_n^\top]^\top$, where the mass matrix $\mathbf{M}$, damping matrix $\mathbf{C}$, and stiffness matrix $\mathbf{K}$ are sparse, symmetric matrices and vector $\mathbf{f}_q$ are nodal data forces. These *global* matrices may be assembled from their associated *local* element matrices by expanding each element matrix appropriately into a $q \times q$ matrix and then summing:

$$\mathbf{M} = \sum_{\mathrm{j=1}}^{\mathrm{M}} \mathbf{M}^{\mathrm{j}}_{\mathrm{q\times q}};\ \mathbf{C} = \sum_{\mathrm{j=1}}^{\mathrm{M}} \mathbf{C}^{\mathrm{j}}_{\mathrm{q\times q}};\ \mathbf{K} = \sum_{\mathrm{j=1}}^{\mathrm{M}} \mathbf{K}^{\mathrm{j}}_{\mathrm{q\times q}};\ \mathbf{f}_{\mathrm{q}} = \sum_{\mathrm{j=1}}^{\mathrm{M}} \mathbf{f}^{\mathrm{j}}_{\mathrm{q}}, \tag{6}$$

where $\mathbf{M}^j$, $\mathbf{C}^j$, $\mathbf{K}^j$, and $\mathbf{f}^j_q$ are element mass, damping, and stiffness matrices, and nodal data forces associated with element E_j, $\ j = 1, \ldots, M$.

We now derive expressions for $\mathbf{M}^j$, $\mathbf{C}^j$, $\mathbf{K}^j$, and $\mathbf{f}^j_q$ from element kinetic and potential energy functionals. Let $\mathbf{x}^j(u, v, t)$ be the position of material point (u, v) within E_j, and let $\mathbf{q}^j$ denote the concatenation of nodal variables for all the nodes of E_j. Following (4), we write the element trial function

$$\hat{\mathbf{x}}^j(u, v, t) = \mathbf{N}^j(u, v)\mathbf{q}^j(t) \approx \mathbf{x}^j(u, v, t), \tag{7}$$

where $\mathbf{N}^j$ are the element shape functions. Recall that the basis functions $\mathbf{N}_i$ are obtained by superposing the shape functions associated with node i. The element velocity is $\partial\hat{\mathbf{x}}^j/\partial t = \mathbf{N}^j\dot{\mathbf{q}}^j$, where $\dot{\mathbf{q}}^j(t)$ is the rate of change of the nodal variables.

The kinetic energy associated with element E_j is

$$\frac{1}{2}\iint_{E_j} \mu \frac{\partial\hat{\mathbf{x}}^{j\top}}{\partial t}\frac{\partial\hat{\mathbf{x}}^j}{\partial t}\, du\, dv = \frac{1}{2}\dot{\mathbf{q}}^{j\top}\Big[\iint_{E_j} \mu\mathbf{N}^{j\top}\mathbf{N}^j\, du\, dv\Big]\dot{\mathbf{q}}^j = \frac{1}{2}\dot{\mathbf{q}}^{j\top}\mathbf{M}^j\dot{\mathbf{q}}^j, \tag{8}$$

where the element mass matrix is given by

$$\mathbf{M}^j = \iint_{E_j} \mu\mathbf{N}^{j\top}\mathbf{N}^j\, du\, dv. \tag{9}$$

We introduce a simple velocity-proportional kinetic energy dissipation according to the (Raleigh) dissipation functional

$$\frac{1}{2}\iint_{E_j} \gamma \frac{\partial\hat{\mathbf{x}}^{j\top}}{\partial t}\frac{\partial\hat{\mathbf{x}}^j}{\partial t}\, du\, dv = \frac{1}{2}\dot{\mathbf{q}}^{j\top}\mathbf{C}^j\dot{\mathbf{q}}^j. \tag{10}$$

The element damping matrix is proportional to the mass matrix and is given by

$$\mathbf{C}^j = \iint_{E_j} \gamma\mathbf{N}^{j\top}\mathbf{N}^j\, du\, dv. \tag{11}$$

According to (1) the element deformation matrix may be expressed as

$$\mathcal{E}^j_{\mathrm{p}}(\mathbf{x}) = \iint_{E_j} \boldsymbol{\sigma}^{j\top}\boldsymbol{\epsilon}^j\, du\, dv, \tag{12}$$

where the strain vector is

$$\boldsymbol{\epsilon}^j = \left[\frac{\partial \mathbf{x}^{j\top}}{\partial u}, \frac{\partial \mathbf{x}^{j\top}}{\partial v}, \frac{\partial^2 \mathbf{x}^{j\top}}{\partial u^2}, \frac{\partial^2 \mathbf{x}^{j\top}}{\partial u \partial v}, \frac{\partial^2 \mathbf{x}^{j\top}}{\partial v^2}\right]^\top = \mathbf{L}\mathbf{x}^j \tag{13}$$

and the stress vector is

$$\boldsymbol{\sigma}^j = \begin{bmatrix} \alpha_{10}^j \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha_{01}^j \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \beta_{20}^j \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \beta_{11}^j \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \beta_{02}^j \mathbf{I} \end{bmatrix} \boldsymbol{\epsilon}^j = \mathbf{D}^j \boldsymbol{\epsilon}^j, \tag{14}$$

with $\mathbf{I}$ a 3×3 unit matrix. Using (7), we can write

$$\boldsymbol{\epsilon}^j = \mathbf{L}\mathbf{N}^j \mathbf{q}^j = \mathbf{B}^j \mathbf{q}^j, \tag{15}$$

where $\mathbf{B}^j$ is the element strain matrix. Inserting the expressions for $\boldsymbol{\epsilon}^j$ and $\boldsymbol{\sigma}^j$ into (12) yields

$$\mathcal{E}_{\mathrm{p}}^j(\mathbf{x}) = \mathbf{q}^{j\top} \mathbf{K}^j \mathbf{q}^j, \tag{16}$$

where the element stiffness matrix is given by

$$\mathbf{K}^j = \iint_{E_j} \mathbf{B}^{j\top} \mathbf{D}^j \mathbf{B}^j \, du \, dv. \tag{17}$$

Finally, according to (3), the potential energy in element E_j due to data forces $\mathbf{f}^j(u, v, t)$ is

$$-\iint_{E_j} \hat{\mathbf{x}}^{j\top} \mathbf{f}^j \, du \, dv = -\mathbf{q}^{j\top} \iint_{E_j} \mathbf{N}^{j\top} \mathbf{f}^j \, du \, dv = -\mathbf{q}^{j\top} \mathbf{f}_q^j, \tag{18}$$

where the nodal data forces are given by

$$\mathbf{f}_q^j = \iint_{E_j} \mathbf{N}^{j\top} \mathbf{f}^j \, du \, dv. \tag{19}$$

5 Model Structure

The balloon model is a closed surface in Euclidean 3-space which is topologically isomorphic to a sphere. We initially discretize the balloon in the material coordinates (u, v) by tessellating it into a set of 20 triangular elements to form an icosahedron. We chose the icosahedron because it has a simple representation in material coordinates and it has a regular structure in Euclidean 3-space, with each of its 12 nodes connected to five neighboring nodes.

The parametric equation which initially maps the material (u_i, v_i) coordinates of the 12 icosahedron nodes into 3-space is given by

$$\mathbf{x}(u_i, v_i) = a \begin{pmatrix} \cos u_i \, \cos v_i \\ \cos u_i \, \sin v_i \\ \sin u_i \end{pmatrix}, \tag{20}$$

where $-\pi/2 \leq v \leq \pi/2$ and $-\pi \leq u < \pi$ and $a \geq 0$ is a radius parameter.

5.1 Triangular C^1 Finite Element

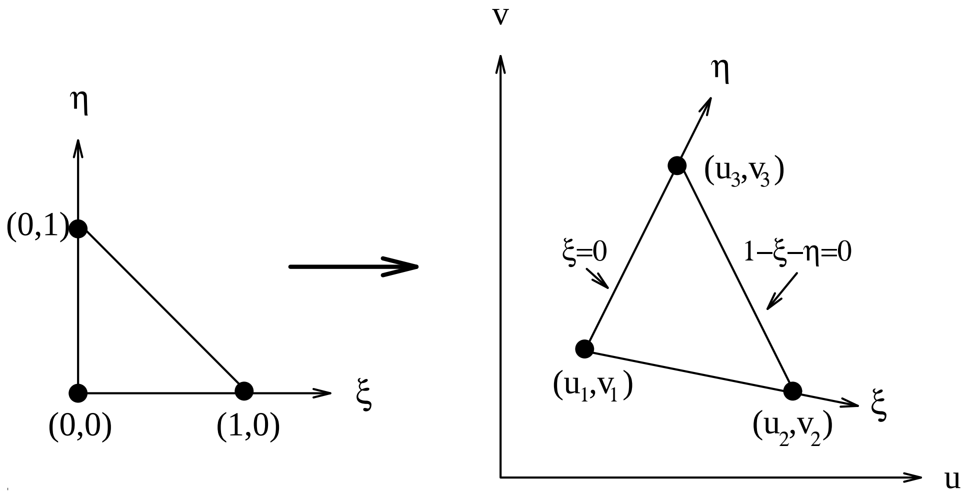


Figure 2: C^1 continuous triangular element.

We use a fifth-order triangular finite element to implement the balloon model [23]. In view of the form of the deformation energy (1) which leads to the strain vector (13), it is natural to choose as nodal variables $\mathbf{x}$, along with its first and second parametric partial derivatives evaluated at each node i. The nodal variable vector for the balloon is therefore

$$\mathbf{q}_i(t) = \left[\mathbf{x}_i^\top, \left(\frac{\partial \mathbf{x}}{\partial u}\right)_i^\top, \left(\frac{\partial \mathbf{x}}{\partial v}\right)_i^\top, \left(\frac{\partial^2 \mathbf{x}}{\partial u^2}\right)_i^\top, \left(\frac{\partial^2 \mathbf{x}}{\partial u \partial v}\right)_i^\top, \left(\frac{\partial^2 \mathbf{x}}{\partial v^2}\right)_i^\top\right]^\top. \tag{21}$$

Figure 2 shows the C^1 continuous element defined locally in the dimensionless oblique coordinates (ξ, η). In this local coordinate system the material coordinates (u, v) can be expressed as

$$\begin{aligned} u &= (1-\xi-\eta)u_1 + \xi u_2 + \eta u_3, \\ v &= (1-\xi-\eta)v_1 + \xi v_2 + \eta v_3, \end{aligned} \tag{22}$$

where (u_i, v_i) are the material coordinates at the nodes (as numbered in the figure), and the local nodal variable vector becomes

$$\mathbf{q}_{i_\xi}(t) = \left[\mathbf{x}_i^\top, (\mathbf{x}_\xi)_i^\top, (\mathbf{x}_\eta)_i^\top, (\mathbf{x}_{\xi\xi})_i^\top, (\mathbf{x}_{\xi\eta})_i^\top, (\mathbf{x}_{\eta\eta})_i^\top\right]^\top. \tag{23}$$

The transformation from global to local coordinates is

$$\mathbf{q}_i = \mathbf{T}_i \mathbf{q}_{i_\xi} \tag{24}$$

where the transformation matrix $\mathbf{T}_i$ is specified in [23] (pp. 100–101).

Concatenating the $\mathbf{q}_{i_\xi}$ at each of the three nodes of element j, we obtain the 18-dimensional element nodal vector $\mathbf{q}_\xi^j = [\mathbf{q}_{1_\xi}^\top, \mathbf{q}_{2_\xi}^\top, \mathbf{q}_{3_\xi}^\top]^\top$. According to (7), we can write the local trial function as

$$\hat{\mathbf{x}}^j(\xi, \eta, t) = \mathbf{N}^j(\xi, \eta)\mathbf{q}_\xi^j(t). \tag{25}$$

The nodal shape functions $\mathbf{N}_i(\xi, \eta)$ which are contained in the 18×18 matrix $\mathbf{N}^j$ are, for node 1

$$\begin{array}{rclrcl}
\mathbf{N}_1 & = & \lambda^2(10\lambda - 15\lambda^2 + 6\lambda^3 + 30\xi\eta(\xi + \eta)), & \mathbf{N}_2 & = & \xi\lambda^2(3 - 2\lambda - 3\xi^2 + 6\xi\eta) \\
\mathbf{N}_3 & = & \eta\lambda^2(3 - 2\lambda - 3\eta^2 + 6\xi\eta), & \mathbf{N}_4 & = & \frac{1}{2}\xi^2\lambda^2(1 - \xi + 2\eta) \\
\mathbf{N}_5 & = & \xi\eta\lambda^2, & \mathbf{N}_6 & = & \frac{1}{2}\eta^2\lambda^2(1 + 2\xi - \eta),
\end{array}$$

for node 2,

$$\begin{array}{rclrcl}
\mathbf{N}_7 & = & \xi^2(10\xi - 15\xi^2 + 6\xi^3 + 15\eta^2\lambda), & \mathbf{N}_8 & = & \frac{\xi^2}{2}(-8\xi + 14\xi^2 - 6\xi^3 - 15\eta^2\lambda) \\
\mathbf{N}_9 & = & \frac{\xi^2\eta}{2}(6 - 4\xi - 3\eta - 3\eta^2 + 3\xi\eta), & \mathbf{N}_{10} & = & \frac{\eta^2}{4}(2\xi(1 - \xi)^2 + 5\eta^2\lambda) \\
\mathbf{N}_{11} & = & \frac{\xi^2\eta}{2}(-2 + 2\xi + \eta + \eta^2 - \xi\eta), & \mathbf{N}_{12} & = & \frac{\xi^2\eta^2\lambda}{4} + \frac{\xi^3\eta^2}{2},
\end{array}$$

and for node 3,

$$\begin{array}{rclrcl}
\mathbf{N}_{13} & = & \eta^2(10\eta - 15\eta^2 + 6\eta^3 + 15\xi^2\lambda), & \mathbf{N}_{14} & = & \frac{\xi\eta^2}{2}(6 - 3\xi - 4\eta - 3\xi^2 + 3\xi\eta) \\
\mathbf{N}_{15} & = & \frac{\eta^2}{2}(-8\eta + 14\eta^2 - 6\eta^3 - 15\xi^2\lambda), & \mathbf{N}_{16} & = & \frac{\xi^2\eta^2\lambda}{4} + \frac{\xi^2\eta^3}{2} \\
\mathbf{N}_{17} & = & \frac{\xi\eta^2}{2}(-2 + \xi + 2\eta + \xi^2 - \xi\eta), & \mathbf{N}_{18} & = & \frac{\eta^2}{4}(2\eta(1 - \eta)^2 + 5\xi^2\lambda),
\end{array}$$

where $\lambda = 1 - \xi - \eta$.

Note that the polynomial basis of the element is complete up to fourth-order terms and contains three fifth-order terms [23]. The trial functions are C^∞ within elements and they ensure C^1 continuity between elements. Since (1) contains up to second order derivatives, the element is conforming.

The shape functions are expressed in terms of the local coordinates (ξ, η) and it is convenient to work with these coordinates. Thus, the required derivatives of the shape functions in the strain matrix $\mathbf{B}$ are computed using repeated applications of the chain rule and equation (23). Also, a function $f(u, v)$ may be integrated over E_j by transforming to the local coordinate system as follows:

$$\iint_{E_j} f(u, v)\, du\, dv = \iint_{E_j} f(u(\xi, \eta), v(\xi, \eta)) \det \mathbf{J}\, d\xi\, d\eta, \tag{26}$$

where

$$\mathbf{J} = \begin{bmatrix} \frac{\partial u}{\partial \xi} & \frac{\partial v}{\partial \xi} \\ \frac{\partial u}{\partial \eta} & \frac{\partial v}{\partial \eta} \end{bmatrix} \tag{27}$$

is the Jacobian matrix. These integrals are approximated using Gauss-Legendre quadrature rules [23].

5.2 Model Refinement

Our implementation allows the balloon model to be refined during the fitting process by subdividing the triangular elements. Each element spawns 4 child elements by connecting the midpoints of its 3 edges (Fig. 3). This process may be applied recursively to each child element. The connectivity of all new vertices formed in this fashion is six, while the original 12 vertices of the icosahedron remain five-connected. Thus a low resolution model may be

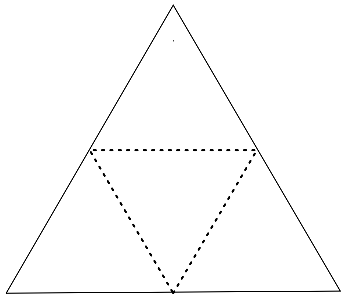
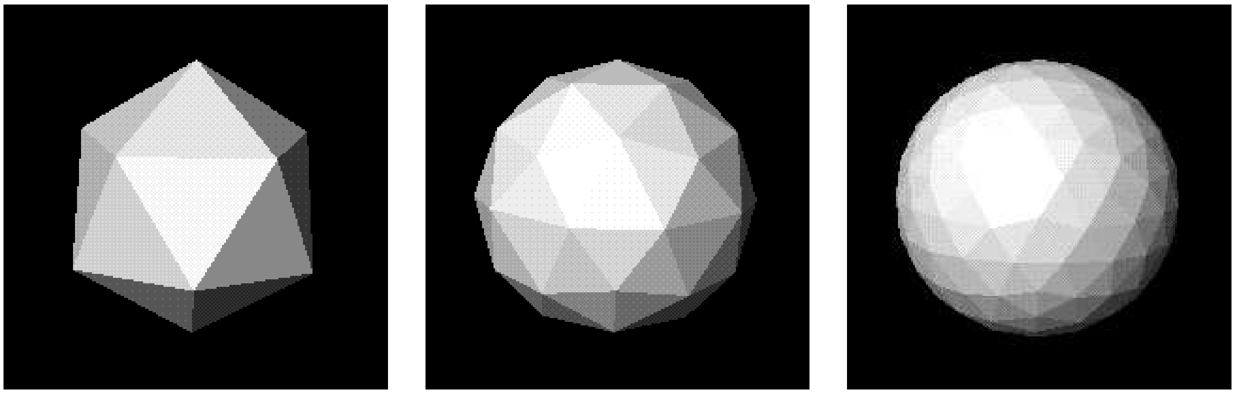


Figure 3: Subdividing an equilateral triangular element.
Four smaller elements are formed by connecting the midpoints of the edges.



(a) 20 triangles (b) 80 triangles (c) 320 triangles

Figure 4: Balloon model mesh in Euclidean 3-space at three subdivision levels.

initially fit to the data, efficiently reconstructing the rough overall shape, and subsequently refined in steps as necessary to capture the detail. This approach greatly reduces the overall computation time required for reconstruction.

Since each global subdivision of the balloon model increases the number of element nodes by approximately fourfold, this scheme has its limits. A better scheme is to locally subdivide the model in areas where the shape implied by the data varies considerably. Local subdivision is not pursued in this paper.

6 Applied Forces

Our dynamic model fitting paradigm applies data constraints to the model as external force distributions $\mathbf{f}(u, v, t)$. The contribution of the force distribution to each element E_j is converted through (19) to generalized forces $\mathbf{f}_q^j$ associated with the nodal variables of the element. Two types of data forces are applied to the balloon model– forces obtained through gradients of image potential functions and forces based on distances between data points and the model surface.

6.1 3D Image Forces

When extracting and reconstructing surfaces from 3D image data, we design forces that localize salient image features. For example, to attract our model towards significant 3D intensity edges (gradients) in some region of an image function $I(x, y, z)$ we construct a 3D potential function

$$P(x, y, z) = \lambda_1 \|\nabla(G_\sigma * I)\| + \lambda_2 \|O_{\mathrm{MD}} * I\| \tag{28}$$

whose potential valleys (minima) coincide with the object surface [7]. In the first term on the right hand side of (28), G_σ denotes a 3D Gaussian smoothing filter of characteristic width σ. This filter broadens or narrows the potential valleys of this term thus determining the extent of the region of attraction of the intensity gradient. Typically, the attraction has a relatively short range. In the second term, a 3D edge detector, the 3D Monga-Deriche (MD) operator [24], is applied to the image data to produce a 3D intensity edge field. The potential valleys of this term tend to be narrow and deep, complementing (and coinciding with) the wider but more shallow valleys produced by the first term. A weighted combination of these terms is formed so the model will "slide down" the shallow valleys and then drop into the deeper valleys thus locking onto image edges.

The potential function produces a force distribution

$$\mathbf{f}(\mathbf{x}) = \kappa \frac{\nabla P(\mathbf{x})}{\|\nabla P(\mathbf{x})\|} \tag{29}$$

on the model, where κ controls the strength of the force. We normalize the image force as for better numerical stability [10]. Consequently, all significant edge voxels, including spurious edges, will attract the model equally. However, once the model converges towards the true 3D edges of the object, the smoothing effect of the model will give it a tendency to ignore spurious 3D edges.

Note that to compute ∇P at any model point $\mathbf{x}(u, v)$ from a discrete 3D image data set $I(i, j, k)$, we tri-linearly interpolate $I(\mathbf{x})$ using values at the eight surrounding pixels.

6.2 Balloon Inflation Force

When extracting object surfaces from 3D image data, the balloon model must first be initialized within the object. If the model is not close enough to the surface of the object, the short-range image forces defined previously may not attract it. For this reason, an internal pressure force is used to "inflate" the balloon model towards the object surface [7] [10]. The force takes the form

$$\mathbf{f} = \kappa_1 \mathbf{n}(u, v), \tag{30}$$

where $\mathbf{n}(u, v)$ is the unit normal vector to the model surface at the point $\mathbf{x}(u, v)$, and κ_1 is the amplitude of this force. If κ_1 is negative, the force will deflate the balloon. We usually set the image force scale parameter κ and κ_1 to be of the same order, with κ slightly larger than κ_1 so that a significant 3D edge will stop the inflation, but with κ_1 large enough for the model to pass through weak or spurious edges.

6.3 User and Constraint Forces

Accurate measurement of medical image structures is important in a clinical setting. For this and other reasons, visualization and manual interaction are likely to remain essential in 3D biomedical image scenarios. Our dynamic modeling approach provides a facile interface to the models through the use of force interaction tools. For example, as the model is deforming, the user may use the mouse to specify spring forces which pull the model towards significant image features, or to specify "pins" which constrain the model to interpolate fiducial features in the data that a specialist can identify.

Both the mouse and pin forces are implemented as long-range spring-like point forces

$$\mathbf{f}(u,v) = \kappa \left\| \mathbf{p} - \mathbf{x}(u_p, v_p) \right\| \tag{31}$$

proportional to the separation between the mouse or pin point $\mathbf{p}$ in space and the point of influence (u_p, v_p) of the force on the model's surface.

We approximate (u_p, v_p) as the model node with minimal distance to the point $\mathbf{p}$, using a heuristic local neighborhood search to find the nearest model node.

6.4 Parallel Numerical Integration

Our dynamic surface model is easiest to manipulate interactively during the fitting process if its motion is critically damped to minimize vibrations. Critical damping can be achieved by appropriately balancing the mass and damping distributions. Another way of eliminating vibration while preserving useful dynamics is to set the mass density $\mu(u,v)$ to zero, thus reducing (5) to

$$\mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f}_q. \tag{32}$$

This first-order dynamical system governs a model which has no inertia and comes to rest as soon as all the forces equilibrate. Although (32) is simpler to implement and numerically more efficient, a model lacking inertia can experience difficulty tracking moving objects if external forces are not persistently reliable due to weak, noisy, or missing data. Nonetheless, we have successfully employed the first-order dynamic model (32) in our cardiac reconstruction and tracking system which is presented in the next section.

We integrate equation (32) forward through time using an explicit first-order Euler method. This method approximates the temporal derivatives with forward finite differences. It updates the degrees of freedom $\mathbf{q}$ of the model from time t to time $t + \Delta t$ according to the formula

$$\mathbf{q}^{(t+\Delta t)} = \mathbf{q}^{(t)} + \Delta t (\mathbf{C}^{(t)})^{-1} \left(\mathbf{f}_q^{(t)} - \mathbf{K}\mathbf{q}^{(t)} \right). \tag{33}$$

In our implementation, we do not explicitly assemble and factorize a global stiffness matrix $\mathbf{K}$ as is common practice in applied finite element analysis. Instead, we update the nodal vectors $\mathbf{q}_i^{(t+\Delta t)}$ iteratively by computing the product $\mathbf{K}^j \mathbf{q}^j$ on an element-by-element basis using the element stiffness matrices $\mathbf{K}^j$. This approach makes the model fitting process easily parallelizable.

The deformable model is implemented as a list of finite element data structures and a list of node data structures. The element structures contain pointers to their associated node structures. The following actions are repeated at each time step of the model fitting process:

- For each model node, compute externally applied forces $\mathbf{f}_q$.

- For each element, accumulate the internal forces on the element nodal vectors $\mathbf{q}^j$ by computing the product $\mathbf{K}^j\mathbf{q}^j$.

- For each model node, update the position based on the applied and internal forces on the node using Euler time integration.

These operations can be readily parallelized on a shared memory multiprocessor (such as our 4 processor Silicon Graphics Iris 4D/340VGX workstation) by partitioning the element and node lists into equal sized sublists according to the number of processors available. Each processor then independently executes the loops using its assigned lists of elements and nodes.

7 A System for 3D/4D Medical Image Analysis

This section describes an interactive system, implemented on a Silicon Graphics Iris 4D/340 VGX workstation, that uses the deformable balloon model to extract (segment), reconstruct, and track surfaces of biological structures in volume images. The design of this experimental system is geared towards cardiac image analysis.

The system provides views of the data and model in separate windows to facilitate the interactive initialization and visualization of the data and the balloon model. One window displays a 3D view of the model embedded in three orthogonal image slices of the volume data (Fig. 5 right), which are $118 \times 128 \times 128$ pixel CT images of a canine heart. The user can interactively rotate this 3D view in any direction as well as change the image slice of any image plane. In addition, the user may translate the model in any of the (x, y, z) directions, or scale the model. This capability is useful in initializing the balloon. The other window displays a 2D image slice of one of the three orthogonal image planes overlaid with the corresponding cross section of the balloon model (Fig. 5 left). Note that our finite element surface representation makes it possible to compute any cross section of the balloon model to obtain a continuous planar contour. The user can quickly change or scan through the image slices of this orthogonal view or change to another orthogonal view.

The user can interact with the 3D model in the 2D image slice window by applying forces to a cross-sectional contour as if it were a deformable contour; i.e., a snake [1]. By positioning the mouse at some point in the window and depressing a mouse button, the mouse position is determined and the closest model point on the cross-sectional contour is calculated. A spring force is then applied to the balloon model along the vector from the model point to the mouse position. The force is applied while the mouse button is depressed, and its direction can be changed by dragging the mouse to a new position in the window. The user can also interact with the model by positioning the mouse and depressing a mouse button to specify a pin point. Pin constraints apply a sustained spring force to the closest model point on the cross–sectional contour, forcing the model to adhere closely to the pin. This mechanism allows the user to reinforce or create an object edge. Furthermore, the user may interactively alter the surface tension and rigidity parameters α_{ij} and β_{ij} as well as alter the time step, alter the balloon and image forces, and initiate a global subdivision of the model. Once the model has been fitted to the object surface, the user may choose menu items which compute and display surface curvature, perform enclosed volume calculations, etc.

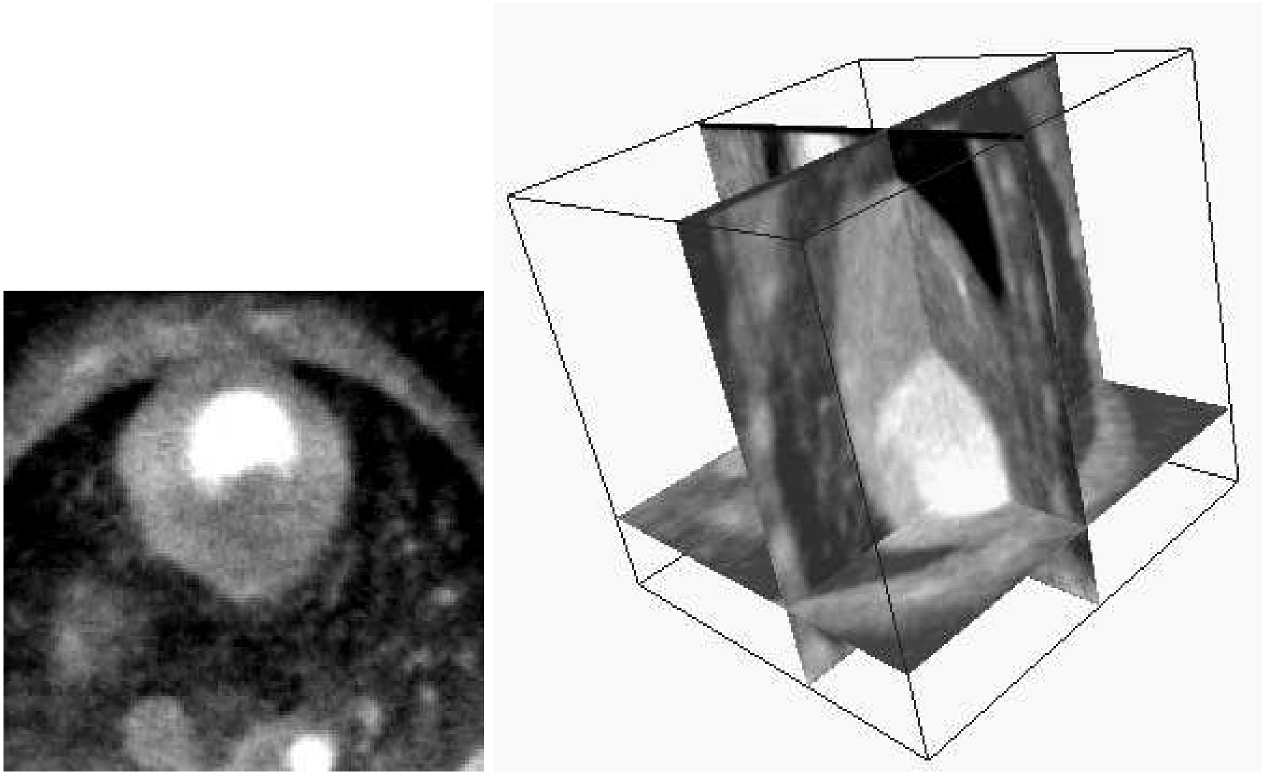
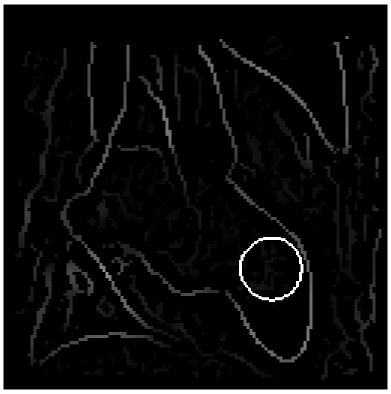
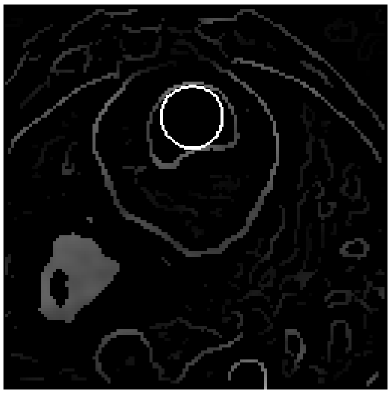


Figure 5: Left: Image slice window. Right: Image Volume Window.



(a)



(b)

Figure 6: YZ and XY view edge detected image slices showing initial cross sections of the balloon model.
The edge maps are generated by the application of the 3D Monga-Deriche (MD) operator.

7.1 The Segmentation/Reconstruction Process

To initiate the surface extraction/reconstruction process, the user scans through the image slices in the 2D image window to locate the approximate center of the object to be extracted. This process is repeated for the two other orthogonal views. The user can observe the 3D volume view window during this procedure to aid in determining the object center. The user then uses the mouse to specify the initial size and location of the model in the 2D image window (Fig. 6). The initial model will then be constructed and will appear embedded in the image slices in the 3D window. The user can further adjust the size and location of the model in either of the windows.

The user may specify an initial model resolution level. Typically we begin with a low resolution model and then use the mouse to globally subdivide the model. It is also useful to initially set the tension parameters α_{ij} to be significantly smaller than the rigidity parameters β_{ij}. This allows an initially coarse balloon model to stretch more easily and quickly towards the edges of the heart. Once the model reaches the edges and the model has been subdivided, β_{ij} is then increased to smooth the finer resolution model.

Once the initial model has been specified, the user may begin the model fitting (Fig. 7). Before or during the model fitting procedure the user may specify any number of pin constraints on the model. We determine by visual inspection when the fitting process is completed. The user can quickly scan the image slices in the 2D window to ascertain how well the cross sections of the model fit the object edges. A possible automatic stopping criterion might monitor the average position change of the model nodes at each iteration to assess whether the model has achieved equilibrium.

7.2 Experiments

We used the interactive system to extract and reconstruct the left-ventricular chamber from 3D CT images of a canine heart. The data set was acquired by the dynamic spatial re-

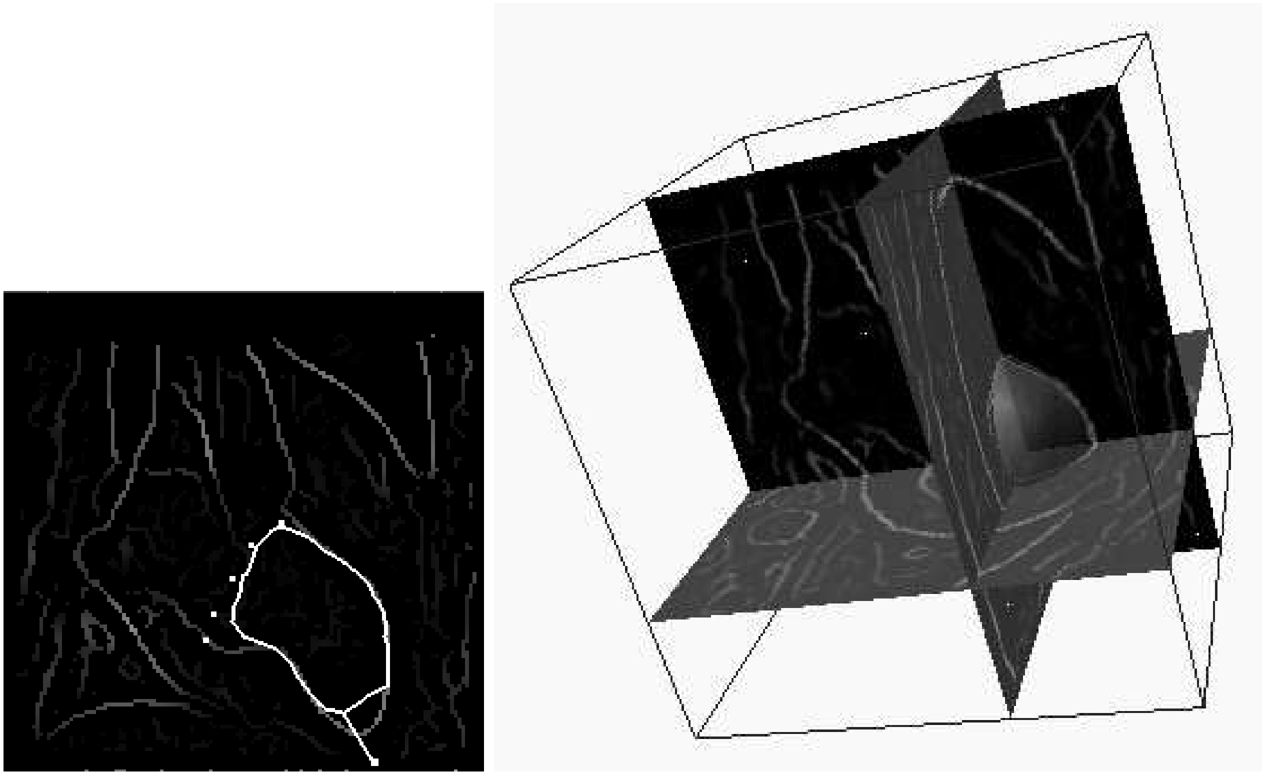


Figure 7: Segmentation of LV.
Left: Cross–section of balloon model deforming towards LV edges, influenced by "pin" constraints and a "spring" pulling the model towards an edge. Right: Balloon model embedded in volume image deforming towards LV edges.

(a) (b) (c)

Figure 8: Intensity and edge detected CT image slice of left ventricle.
(a) Intensity image YZ plane slice 68. (b) Edge detected image. (c) Cross section of fitted balloon model deforming to left ventricle.

constructor (DSR), a high speed volumetric X-ray CT scanner [25]. Sixteen volume images representing a complete cardiac cycle were used in the experiments, with each volume image containing 118 slices of 128×128 pixels. Each slice represents an approximately 0.9 mm thick transverse cross section of the scanned anatomy, with each voxel representing a $(0.9mm)^3$ cube of tissue.

Fig. 8(a) shows a sagittal (y-z plane) slice of the canine heart. In a canine heart the valves may appear directly connected to the LV chamber and aorta, frustrating all attempts to fully automate the reconstruction process. Our semi-automatic approach allows a user to interact with the model as it is deforming. As mentioned previously, the user may use the mouse to apply spring forces that pull the model away from spurious edges, or to specify pins which constrain the model to interpolate fiducial features in the data. A few pins are required on one or two image slices of the canine heart data to provide an effective separation between the LV chamber and the aorta. The smoothness of the elastic surface prevents the model from straying very far in neighboring image slices.

Figure 9 shows a cross section of the balloon model in an image slice deforming to fit the edge of the ventricle. The final LV reconstruction is shown in Fig. 10. The initial model consisted of 20 triangular elements. Four global subdivisions of the model were performed during the fitting process to increase the accuracy of the reconstruction. The final model contains 5120 elements and the fitting process takes on the order of 5 minutes to complete. As the balloon model deforms in 3D, it can potentially reconstruct a globally more consistent surface than can easily be reconstructed in serial sections using deformable contours [1]. It is also a potentially more robust technique—missing slices do not seriously reduce the quality of the fit—and it is far less time consuming than the traditional manual slice-editing technique.

7.3 Estimating the LV Motion

We can use the balloon model to estimate the nonrigid motion of the LV over successive CT volumes in the cardiac cycle. We begin by fitting the model to the first volume in

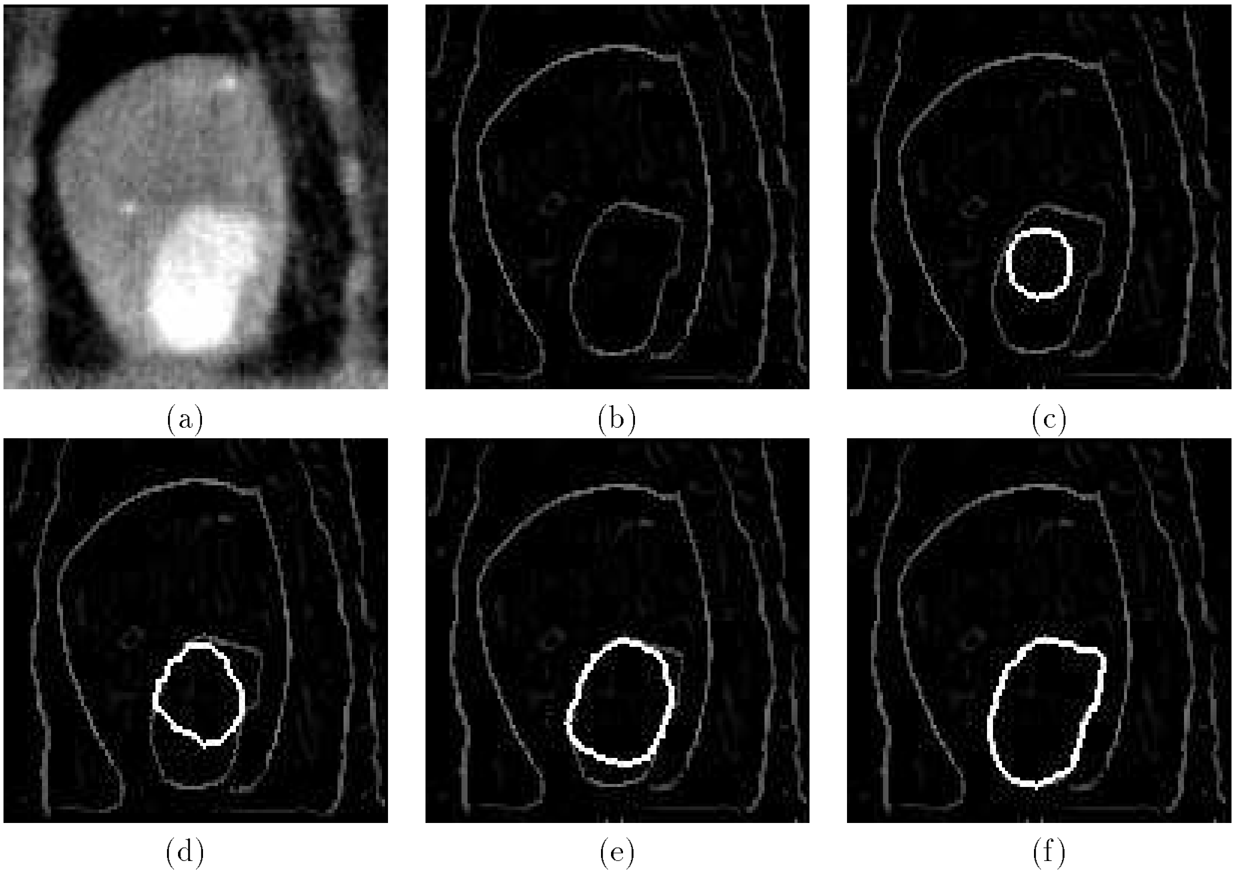


Figure 9: Intensity and edge detected CT image slice of left ventricle.
(a) Intensity image XZ plane slice 91. (b) Edge detected image. (c) Cross section of initial balloon model. (d)-(f) Cross section of balloon deforming to left ventricle.

Figure 10: Reconstruction of Left Ventricle.
Model parameters: $\alpha_{ij} = 0.8$, $\beta_{ij} = 0.2$, $\kappa = 111.0$, $\kappa_1 = 110.0$, $\Delta t = 0.004$.

the sequence and use this fitted model as the starting point for the reconstruction of the LV in the next volume. We continue this process for all 16 volumes in the cardiac cycle. The tracking process allows the model to be "continuously" deformed by the time-varying external data forces induced by the stream of volume images. The continuous finite element representation enables us to track the approximate motion of any point of the LV surface through the cardiac cycle (not just the nodal points).

Figure (11) shows sagittal slice 67 through the 16 successive CT volumes over one cardiac cycle. Figure (12) shows the reconstructed LV sequence. Each fitted model contains 1280 elements and the entire fitting process, including the time required to input the 4D DSR dataset, takes only about 100 minutes to complete. This demonstrates the enormous potential advantage of the dynamic deformable model approach compared with the time required to manually segment the LV. Once the initial 3D model has been fitted to the first volume, relatively small deformations are needed to fit subsequent volumes; consequently very little user intervention (i.e., application of pin constraints or spring forces) is necessary. Moreover, the fitting time per volume image should decrease as images are acquired at higher rates because the interframe motion will be smaller. This should lead to proportionally greater reductions in effort when the technique is applied to future image scanners capable of greater temporal resolution.

8 Discussion

The 3D deformable model provides an efficient, semi-automatic segmentation technique which reconstructs a globally coherent surface between image slices that does not suffer from the banding artifacts often seen in surfaces reconstructed by independently contouring each serial tomographic image. The surface model approximates the data across all the slices; hence, it is much less sensitive to noise than locally interpolatory segmentation schemes [3].

An extracted surface model with the aforementioned properties provides many options for quantitative analysis of the anatomic object. In cardiology, for instance, volumetric parameters (end-diastolic and end-systolic volumes, stroke volume, and ejection ratio) are diagnostically significant, while surface curvature extrema often have anatomical significance.

We know from differential geometry that smooth 3D surfaces are uniquely characterized (up to rigid-body transformations) by their first and second fundamental forms [26]. The parametric form of our surface model (i.e., $\mathbf{x}(u,v) = [x(u,v), y(u,v), z(u,v)]^\top$) and, in particular, the nodal variables (21) of its finite element representation contain all the information needed to compute the first and second fundamental forms of the fitted model surface. The intrinsic differential characteristics of the surface, such as the unit normal and the principal curvatures, can be conveniently computed from this information, as can mean and Gaussian curvatures. Furthermore, to compute the volume of the fitted balloon we can make use of Gauss's theorem which reduces a volume calculation problem to a surface integral of the form

$$\Psi = \iint_S F(\mathbf{x})\, d\sigma. \tag{34}$$

The balloon model is composed of M surface elements defined parametrically within an element in (25). Consequently, we can rewrite (34) as the sum of integrals over the surface

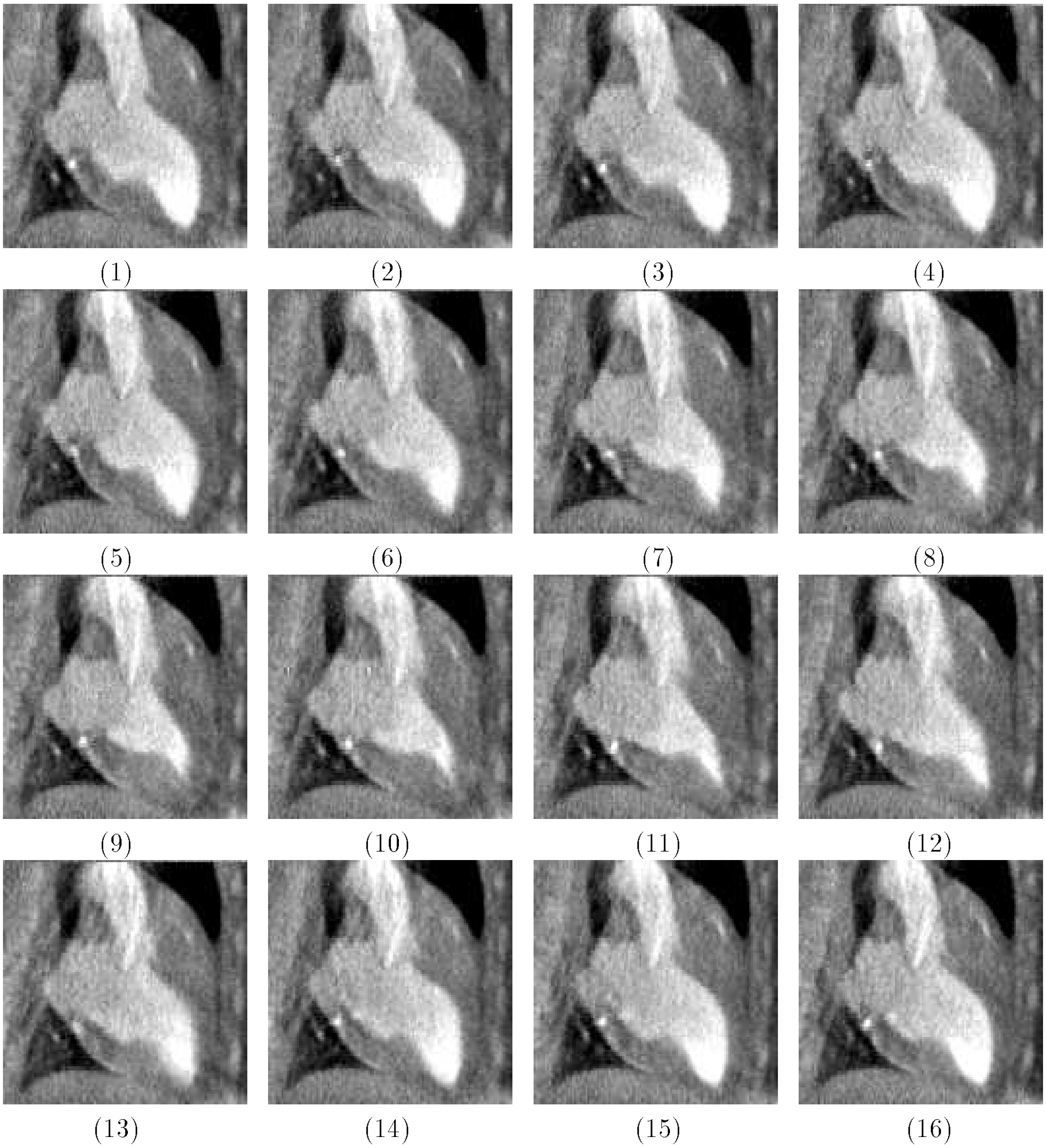


Figure 11: Sagittal slice of successive CT volumes over one cardiac cycle (1–16) showing motion of LV.

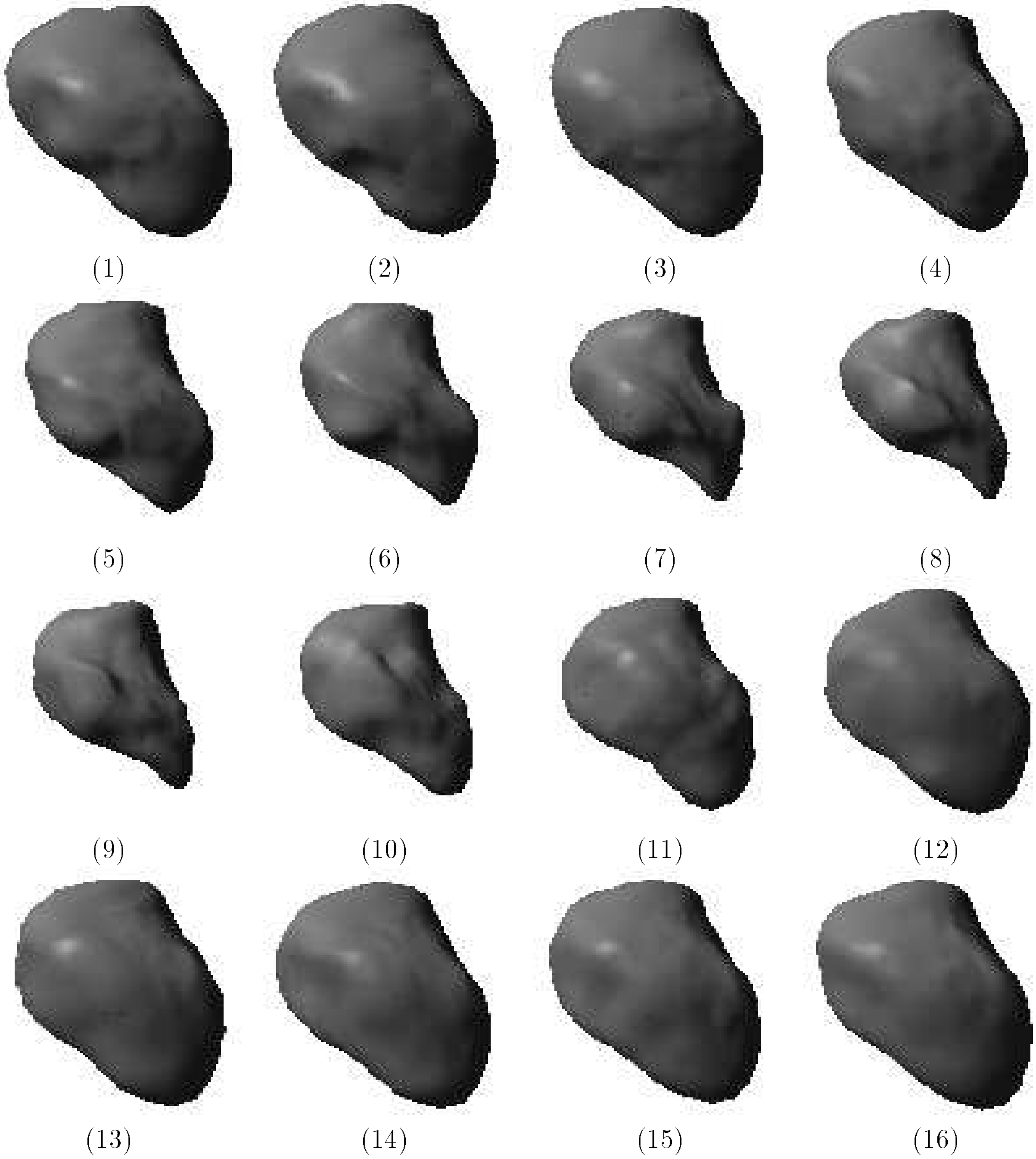


Figure 12: Tracking of the LV motion during one cardiac cycle (1–16).

Figure 13: End-diastolic and end-systolic surfaces of LV during cardiac cycle.

elements as follows:

$$\Psi = \sum_{i=1}^{M} \iint_{S_i} F(\mathbf{x})\, d\sigma = \iint_{S_i} F(\mathbf{x}(\xi, \eta)) \det \mathbf{J}\, d\xi\, d\eta, \tag{35}$$

where $\det \mathbf{J} = \left\| \frac{\partial \mathbf{x}}{\partial \xi}(\xi, \eta) \times \frac{\partial \mathbf{x}}{\partial \eta}(\xi, \eta) \right\|$ is the Jacobian of transformation.

By tracking a parametric surface over time, the dynamic deformable model technique permits a direct analysis of the estimated nonrigid motion. For instance, the variation in the Gaussian curvature of the fitted model over time can be used to estimate the local stretching and shrinking of the LV surface during the cardiac cycle. It should be noted, however, that for the relatively smooth LV surface, the simple tracking scheme employed in this paper estimates the tangential component of the surface velocity field much less reliably than the normal component. A more accurate estimation of the tangential component would require additional data or *a priori* information. For example, SPAMM images [27] depict transient magnetic tags within the heart wall whose motion can be followed over several subsequent images, providing both orthogonal components of the local velocity in the image plane [28]. Our model can readily assimilate this type of information and that available from other sources. For instance,

- *a priori* information about nonrigidity could be included so that the model not only deforms to fit the data but also preserves some basic nonrigid constraints such as isometry or conformality [29].
- fiducial points can be extracted from the model surface and used as a guide when fitting the model to subsequent volumes in the sequence.
- the model can be generalized so that it subdivides elements in areas undergoing stretching or bending or merge elements in areas that are less curved (cf. adaptive meshes

[29, 30, 31]). This would enable the elements to better follow the motion of the data points and allow for correspondence recovery.

Obviously, it is difficult to assess the accuracy of our LV reconstruction and tracking results from a single 4D dataset. A complete error analysis would also require quantitative comparisons against images segmented manually by experts and is beyond the scope of this paper.

9 Summary

We proposed a 3D elastically deformable balloon model for segmentation, reconstruction, and tracking of anatomical structures in multidimensional images. The surface of the model is composed of C^1 triangular finite elements whose nodal variables include position and first and second parametric partial derivatives of the surface. Lagrangian equations of motion make the dynamic model responsive to forces, derived from the 3D data, which deform its surface to fit the data in an elegant and intuitive manner. The fitting is carried out through numerical time-integration of the motion equations. An iterative integration method is used that exploits the parallelism of shared-memory multiprocessor architectures. This low-latency method supports real-time 3D display of the model as it extracts and tracks an anatomical surface. Furthermore, the model features a recursive, global subdivision capability which can fit a high resolution surface at low overall computational cost.

We described an experimental interactive system that demonstrates some of the capabilities of our model by applying it to 4D cardiac CT data. The system semi-automatically segments, reconstructs, and tracks the LV, allowing the user to initialize the model in the region of interest, dynamically manipulate it during the data analysis, and alter the viewpoint, shading mode, and other visualization parameters at any time. The effective efficiency gains that can accrue from a system of this sort should be even more dramatic with dynamic imaging at higher spatial and temporal resolution. Additional refinements will increase the model's potential to support reliable quantitative analysis of volume, form, and nonrigid motion for diagnostic and other medical purposes. This is a promising direction for further work.

Acknowledgements

DT is a Fellow of the Canadian Institute for Advanced Research. We thank the following people for their cooperation: The cardiac CT images were made available by Eric Hoffman of the University of Pennsylvania Medical School and were redistributed to us courtesy of Dmitry Goldgof, University of South Florida. The Monga-Deriche edge detector was provided courtesy of Nicholas Ayache and Gregoire Malandain of INRIA, Paris, France. This research was supported by the Natural Sciences and Engineering Research Council of Canada and the Information Technologies Research Center of Ontario.

References

[1] I. Carlbom, D. Terzopoulos, and K.M. Harris. Reconstructing and visualizing models of neu-

ronal dendrites. In N.M. Patrikalakis, editor, *Scientific Visualization of Physical Phenomena*, pages 623–638. Springer–Verlag, New York, 1991.

[2] A. Singh, L. von Kurowski, and M.-Y. Chiu. Cardiac MRI Segmentation Using Deformable Models. In *Proc. IEEE Conf. on Computers and Cardiology*, London, Sept. 1993.

[3] W.E. Lorenson and H.E. Cline. Marching Cubes, A High Resolution 3D Surface Construction Algorithm. *Computer Graphics*, 21(4):163–169, 1987.

[4] P. Sander and S. Zucker. Inferring surface trace and differential structure from 3-D images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(9):833–854, 1990.

[5] S.R. Sternberg. Grayscale Morphology. *Computer Vision, Graphics, and Image Processing*, 35:333–355, 1986.

[6] R.A. Drebin, L. Carpenter, and P. Hanrahan. Volume Rendering. *Computer Graphics*, 22(4):65–74, 1988.

[7] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on Deformable Models: Recovering 3D Shape and Nonrigid motion. *Artificial Intelligence*, 36(1):91–123, 1988.

[8] D. Terzopoulos and D. Metaxas. Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):703–714, 1991.

[9] A. Pentland and B. Horowitz. Recovery of Nonrigid Motion and Structure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):730–742, July 1991.

[10] L.D. Cohen. On Active Contour Models and Balloons. In *CVGIP: Image Understanding*, volume 53(2), pages 211–218, March 1991.

[11] H. Delingette, M. Hebert, and K. Ikeuchi. Shape Representation and Image Segmentation Using Deformable Surfaces. In *Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, pages 467–472, June 1991.

[12] Y.F. Wang and J.F. Wang. Surface Reconstruction using Deformable Models with Interior and Boundary Constraints. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(5):572–579, May 1992.

[13] T. McInerney. Finite Element Techniques for Fitting Deformable Models to 3D Data. Master's thesis, Dept. of Computer Science, University of Toronto, Toronto, ON, Canada, 1992.

[14] D. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):580–591, 1993.

[15] D. Terzopoulos. Multilevel computational processes for visual surface reconstruction. *Computer Vision, Graphics, and Image Processing*, 24:52–96, 1983.

[16] D. Terzopoulos. Regularization of Inverse Visual Problems Involving Discontinuities. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(4):413–424, 1986.

[17] A.A. Young. Epicardial Surface Estimation from Coronary Cineangiograms. *Computer Vision, Graphics, and Image Processing*, 47:11–127, 1989.

[18] A. Young. Epicardial deformation from coronary cineangiograms. In L. Glass, P. Hunter, and A. McCulloch, editors, *in Theory of Heart*, pages 175–207. Springer–Verlag, Heidelberg, 1991.

[19] I. Cohen, L.D. Cohen, and N. Ayache. Introducing New Deformable Surfaces to Segment 3D Images. In *Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, pages 738–739, June 1991.

[20] L.D. Cohen and I. Cohen. Deformable Models for 3D Medical Images Using Finite Elements and Balloons. In *Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, pages 592–598, June 1992.

[21] J.V. Miller, D.E. Breen, W.E. Lorensen, R.M. O'Bara, and M.J. Wozny. Geometrically Deformed Models. In *Computer Graphics(SIGGRAPH'91)*, volume 25(4), pages 217–226, July 1991.

[22] L. Glass, P. Hunter, and A. McCulloch, editors. *Theory of Heart.* Springer–Verlag, New York, 1991.

[23] G. Dhatt and G. Touzot. *The Finite Element Method Displayed.* Wiley, New York, 1984.

[24] O. Monga and R. Deriche. 3D Edge Detection Using Recursive Filtering. In *Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, June 1989.

[25] E.L. Ritman, R.A. Robb, and L.D. Harris. *Imaging Physiological Functions: Experience with the Dynamic Spatial Reconstruction.* Praeger, New York, 1985.

[26] P. M. do Carmo. *Differential Geometry of Curves and Surfaces.* Prentice Hall, Englewood Cliffs, New Jersey, 1976.

[27] L. Axel and L. Dougherty. Heart Wall Motion: Improved Method of Spatial Modulation of Magnetization for MR Imaging. *Radiology*, 172:349–350, 1989.

[28] A. Young and L. Axel. Non-rigid Wall Motion Using MR Tagging. In *Proc. IEEE Conf. Comp. Vision and Pattern Recognition*, pages 399–404, June 1992.

[29] W.C. Huang and D.B. Goldgof. Left Ventricle Motion and Analysis by Adaptive-Size Physically-based Models. In *SPIE Proceedings*, volume 1660–30, pages 299–310, Feb. 1992.

[30] W.-C. Huang and D. B. Goldgof. Adaptive-size meshes for rigid and nonrigid shape analysis and synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):611–616, 1993.

[31] M. Vasilescu and D. Terzopoulos. Adaptive meshes and shells: Irregular triangulation, discontinuities, and hierarchical subdivision. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'92)*, pages 829–832, Champaign, IL, June 1992. IEEE Computer Society Press.