# Using deformable surfaces to segment 3-D images and infer differential structures

Isaac Cohen, Laurent D. Cohen, Nicholas Ayache

CVGIP: IMAGE UNDERSTANDING
Vol. 56, No. 2, September, pp. 242–263, 1992

# Using Deformable Surfaces to Segment 3-D Images and Infer Differential Structures

ISAAC COHEN,* LAURENT D. COHEN,§ AND NICHOLAS AYACHE*

* *INRIA, Domaine de Voluceau, Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France; and § CEREMADE, U.R.A. CNRS 749, Université Paris IX–Dauphine, Place du Marechal de Lattre de Tassigny, 75775 Paris Cedex, France*

Received August 7, 1991; accepted February 20, 1992

In this paper, we use a 3-D deformable model, which evolves in 3-D images, under the action of internal forces (describing some elasticity properties of the surface), and external forces attracting the surface toward some detected edgels. Our formalism leads to the minimization of an energy which is expressed as a functional. We use a variational approach and a conforming finite element method to actually express the surface in a discrete basis of continuous functions. This leads to reduced computational complexity and better numerical stability. The power of the approach to segmenting 3-D images is demonstrated by a set of experimental results on various complex medical 3-D images. Another contribution of this approach is the possibility to infer easily the differential structure of the segmented surface. As we end up with an analytical description of class $\mathcal{C}^\infty$ of the surface almost everywhere, this allows us to compute, for instance, its first and second fundamental forms. From this, one can extract a curvature primal sketch of the surface, including some intrinsic features which can be used as landmarks for 3-D image interpretation. 

## 1. INTRODUCTION

We propose a deformable 3-D shape model which can be used to extract reliable surfaces in 3-D images and infer a differential structure on them.

Usually, 3-D images are given as a set of intensity voxels (volume elements). A 3-D edge detector, after a *local* image analysis [27, 36], provides a set of 3-D edgels (edge elements). These edgels can be considered as the trace of a certain number of surfaces. One is then confronted with a dual problem:

1. Select edgels belonging to the same surface trace; this is the segmentation problem.

2. Recover a continuous and differentiable description of each surface; this yields a path between the original sparse discrete data and the ability to compute a differential structure useful for interpretation [1, 19, 20].

Both questions were analyzed by Sander and Zucker [31], who proposed solving (1) by a connectivity analysis and (2) by the fitting of a set of local quadratic models. But difficulties arise when the connectivity analysis fails because edges are too sparse and also when the model is too local to reliably describe a complex shape.

Another approach to solving a similar problem in 2-D consists of introducing an active deformable model [22], which solves the segmentation problem (1) assuming that an initial estimate is provided (an initial solution might be provided by several means, including user interactivity, which is usually encouraged in medical applications) and the interpolation problem (2) when the curve is expressed in a basis of continuous functions [14, 28]. Such models were generalized in $2\frac{1}{2}$-D and 3-D [34, 33], where the deformable surface is evolving under the forces computed on a 2-D image or a set of 2-D images.

In contrast with the methods of reconstruction based on a 2-D slice-by-slice approach [6, 13, 14], we use, in this paper, a 3-D deformable model, which evolves in 3-D images, under the action of internal forces (describing some elasticity properties of the surface), and external forces attracting the surface toward some detected edgels. This formalism leads to the minimization of an energy which is expressed as a functional. We use a variational approach and a conforming finite element method to express the surface in a discrete basis of continuous functions. This method allows us to perform an "adaptive subdivision" of the parametrization domain without adding nodal points and consequently without increasing the size of the linear system we solve. This leads to reduced computational complexity and better numerical stability than those of a classical finite difference method.

The power of the approach to segment 3-D images is demonstrated by a set of experimental results on complex medical 3-D images (see also [11]).

Another contribution of this approach is the ability to easily infer the differential structure of the segmented surface. As we end up with an analytical description of class $\mathcal{C}^\infty$ of the surface almost everywhere (except between two finite element patches, where the representa-



1049-9660/92 $5.00

tion is only of class $\mathcal{C}^1$; i.e., the tangent plane is continuous), we can compute for instance, its first and second fundamental forms [17]. From this, one can extract a curvature primal sketch of the surface [30, 8], including some intrinsic features which can be used as landmarks for 3-D image interpretation [2, 19].

Last but not least, a careful analysis of our external forces (those which attract the deformable surface toward the edges) shows some intriguing connections with the properties of minimal surfaces: if the deformable surface is a minimal surface (i.e., a surface whose mean curvature is everywhere zero), then it satisfies also the definition of an edge surface.

The paper is organized as follows: We first define the 3-D deformable model (Section 2) and then give an appropriate external force (Section 3) and its relationship with 3-D edge points (Section 4). We show then how to solve this minimization problem by a conforming finite element method (Section 5) and give a correct choice of the regularization parameters (Section 6). An algorithmic complexity comparison between the conforming finite element method and the finite difference method is given in Section 7. Section 8 indicate how to infer the differential structure of the 3-D images from the obtained surface. Finally Section 9 describes a set of experimental results on synthetical and complex medical 3-D images.

## 2. ENERGY-MINIMIZING SURFACES

A 3-D image is given by a set of intensity voxels, or as a set of successive 2-D cross sections. In our first work [13, 14] we processed 3-D images as a set of successives 2-D images. This is a familiar approach which is also used in tracking methods [3, 12, 21], but this method is not effective and cannot take into account the spatial homogeneity of the data. In the following we consider the 3-D image data as a set of pixel and the boundaries of a 3-D image are described by a set of surfaces.

A deformable surface model allows us to characterize these surfaces [33, 34]. This characterization consists of determining the location and the shape of the surface.

In the following we restrict ourselves to parameterized surfaces, since any 3-D surface has a local parameterization [17].

This model is defined by a space of admissible deformations Ad and a functional $E$ to minimize. This functional $E$ represents the energy of the model. A surface $v$ is defined by the mapping

$$\Omega = [0, 1] \times [0, 1] \longrightarrow R^3$$
$$(s, r) \longmapsto v(s, r) = (x(s, r), y(s, r), z(s, r))$$

and the associated energy $E$ is given by

$$E\colon \mathrm{Ad} \longrightarrow R$$
$$v \longmapsto E(v) = \int_\Omega w_{10} \left|\frac{\partial v}{\partial s}\right|^2 + w_{01} \left|\frac{\partial v}{\partial r}\right|^2 + 2w_{11} \left|\frac{\partial^2 v}{\partial s \partial r}\right|^2 + w_{20} \left|\frac{\partial^2 v}{\partial s^2}\right|^2 + w_{02} \left|\frac{\partial^2 v}{\partial r^2}\right|^2 + P(v(s, r))ds\, dr,$$

where $P$ is the potential associated with the external forces. The external forces refer to the forces which allow the surface to localize the image attributes. So, if we want the surface to be attracted by 3-D edge points, the potential $P$ is expressed in terms of a 3-D gradient image. The internal forces acting on the shape of the surface depend on the coefficients $w_{ij}$ such that the elasticity is determined by $(w_{10}, w_{01})$, the rigidity by $(w_{20}, w_{02})$, and the resistance to twist by $w_{11}$. Therefore the coefficients $w_{ij}$ determine the mechanical properties of the surface. These coefficients are also called regularization parameters. We can also constrain the surface structure by adjusting boundary conditions (for instance, to create a cylinder or a torus).

A local minimum $v$ of $E$ satisfies the associated Euler–Lagrange equation

$$-\frac{\partial}{\partial s}\left(w_{10}\frac{\partial v}{\partial s}\right) - \frac{\partial}{\partial r}\left(w_{01}\frac{\partial v}{\partial r}\right) + 2\frac{\partial^2}{\partial s \partial r}\left(w_{11}\frac{\partial^2 v}{\partial s \partial r}\right) + \frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 v}{\partial s^2}\right) + \frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 v}{\partial r^2}\right) = -\nabla P(v) \qquad (1)$$
$$+ \text{boundary conditions},$$

which represents the necessary condition for a minimum ($E'(v) = 0$). A solution of Eq. (1) can be seen as either realizing the equilibrium between internal (or regularizing) and external forces or reaching a minimum of the energy $E$.

The boundary conditions constrain the surface structure by specifying different properties of the surface at the boundaries of the parameterization domain $\Omega = [0, 1] \times [0, 1]$. For example, setting $v(s, 0) = v(s, 1)$, $\forall s \in [0, 1]$, or $v(0, r) = v(1, r)$, $\forall r \in [0, 1]$, constrains the surface $v$ to have a cylindrical structure. The other structures such torus or sphere can be obtained in a similar way.

Since the energy function is not convex, there may be many local minima of $E$. The Euler–Lagrange equation (1) may characterize any such local minimum. But as we are interested in finding a 3-D contour in a given area, we assume in fact that we have a rough *prior* estimation of the surface. This estimate is used as initial data for the associated evolution equation in which we add a temporal parameter $t$:

$$\frac{\partial v}{\partial t} - \frac{\partial}{\partial s}\left(w_{10}\frac{\partial v}{\partial s}\right) - \frac{\partial}{\partial r}\left(w_{01}\frac{\partial v}{\partial r}\right) + 2\frac{\partial^2}{\partial s \partial r}\left(w_{11}\frac{\partial^2 v}{\partial s \partial r}\right) + \frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 v}{\partial s^2}\right) + \frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 v}{\partial r^2}\right) = -\nabla P(v) \quad (2)$$

$v(0, s, r) = v_0(s, r)$ initial estimation
+ boundary conditions.

A solution to the static problem is found when the solution $v(t, s, r)$ converges as $t$ tends to infinity: then the term $\partial v/\partial t$ vanishes, providing a solution of the static problem.

This evolution equation can also be seen as a gradient descent algorithm starting with the initial estimate $v_0$ (see Section 5.3).

## 3. DEFINING THE POTENTIAL $P$

The potential $P$ is such that the force $F(v) = -\nabla P(v)$ must attract the surface to the image attributes that we are looking for. Our main goal is the extraction of "good" edge points (i.e., to be able to remove spurious edge points, while ensuring connected contours). Thus the surface must be attracted by edge points and minimize the energy

$$E_{ext} = \int\int_{\Omega} P(v(s, r))ds\, dr. \quad (3)$$

For this purpose the authors of [22] set the potential $P = -|\nabla \mathcal{I}|^2$ so that edge points will minimize $E_{ext}$, where $\mathcal{I}$ is the 3-D image convolved with a Gaussian function. For numerical stability (a complete discussion is given in [13, 14]) we normalize the force

$$F(v) = -k\frac{\nabla P}{\|\nabla P\|},$$

where $k$ is a parameter which allows us to tune the attraction force. Now, all the edge points including spurious ones have the same ability to attract the surface. But spurious points generally form small connected components in 3-D images; consequently, when the surface converges toward the real contours, all these points first attract the surface and then are ignored by the regularization effect of the algorithm.

Another way to make the edge points attract the surface is by using a Chamfer distance [7] or an Euclidean distance [16] image. The distance image is obtained by computing at each image point (pixel) the distance to the nearest edge point (these points are obtained by a local edge detector). These distances allow us to compute at each image point the attraction force to the nearest edgels. This force can be computed in different ways, allowing us to test different forces according to the rate of convergence of the algorithm. For instance,

$$P(v(s, r)) = -e^{-d(v(s,r))^2}$$

produces a slow convergence whereas

$$P(v(s, r)) = \frac{-1}{d(v(s, r))} \quad \text{and} \quad P = -1 \text{ if } d(v(s, r)) = 0$$

produces a faster convergence [$d(v(s, r))$ denotes the distance between $v(s, r)$ and the nearest edge whereas the smallest distance between two distinct points is one pixel]. The use of such attraction potentials is detailed in [15].

## 4. MINIMIZING SURFACES AND 3-D IMAGE EDGE POINTS

In the previous section we showed how to choose correctly the potential $P$ such that the surface will localize accurately the edge points. Here we comment on the relationship between the surface minimizing the energy of external forces $E_{ext}$ and 3-D edge points. This is a generalization to 3-D surfaces of the result given by Fua and Leclerc [18] for 2-D curves. We use the following definition of the 3-D edges, as proposed by Canny [9].

DEFINITION. A 3-D edge is a surface $\mathcal{S}$ whose points have a maximal gradient magnitude in the direction normal to the surface. All points along the surface $\mathcal{S}$ (called Canny's edge points) satisfy

$$\frac{d|\nabla \mathcal{I}(x(s, r))|}{dN(x(s, r))} = 0, \quad (4)$$

where $N(x(s, r))$ is the normal to the surface $\mathcal{S}$ parameterized by the application $x(s, r)$ and $\mathcal{I}$ denotes the image $I(x, y, z)$ convolved with a Gaussian.

As in [18], to establish the relation between the energy-minimizing surfaces and this definition, let us define the energy associated to the external forces as

$$E_P(\mathcal{S}) = -\frac{1}{|\mathcal{S}|}\int_{\Omega} |\nabla \mathcal{I}(x(s, r))| dA, \quad (5)$$

where $|\mathcal{S}| = \int\int_{\Omega} |x_s \times x_r|\, ds\, dr$ is the surface area and $dA = \sqrt{EG - F^2}\, ds\, dr$ is the standard surface area measure.

In Appendix A we show that a surface $\mathcal{S}$ is a local minimum of $E_P$, with respect to infinitesimal deformation, if

$$\frac{d|\nabla\mathcal{I}(x(s,r))|}{dN(x(s,r))} = \frac{eG - 2fF + gE}{EG - F^2} \left(|\nabla\mathcal{I}(x(s,r))| - \frac{1}{|\mathcal{S}|}\int_{\mathcal{S}} |\nabla\mathcal{I}(x(s,r))| dA\right), \quad (6)$$

where $E(s, r)$, $F(s, r)$, $G(s, r)$, $e(s, r)$, $f(s, r)$, and $g(s, r)$ are the coefficients of the first and second fundamental forms in the basis $\{x_s, x_r, N\}$ (using the same notations as that in [17]). A remarkable result is that the quotient $(1/2)$ $[(eG - 2fF + gE)/(EG - F^2)]$ is the mean curvature of the surface $\mathcal{S}$.

Equation (6) shows that there exists two very interesting special cases:

1. minimal surfaces (i.e., surfaces with a mean curvature which is everywhere zero) and

2. surfaces whose trace is made of edgels with constant gradient magnitude (then the term within parentheses in Eq. (6) vanishes).

Indeed, in both cases, the second member of Eq. (6) vanishes to zero, which means that Canny's edge points coincide with the minimal external energy of a deformable model.

In practice these are interesting but exceptional academic situations, and the deformable model simply converges toward a solution which is an equilibrium between the applied external forces (corresponding to the energy $E_P$) and the internal forces, parameterized by the elasticity coefficients $w_{ij}$.

## 5. NUMERICAL SOLUTION BY A CONFORMING FINITE ELEMENT METHOD

We consider the same evolution equation as that in Section 2:

$$\frac{\partial v}{\partial t} - \frac{\partial}{\partial s}\left(w_{10}\frac{\partial v}{\partial s}\right) - \frac{\partial}{\partial r}\left(w_{01}\frac{\partial v}{\partial r}\right) + 2\frac{\partial^2}{\partial s\partial r}\left(w_{11}\frac{\partial^2 v}{\partial s\partial r}\right) + \frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 v}{\partial s^2}\right) + \frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 v}{\partial r^2}\right) = -\nabla P(v) \quad (7)$$

$v(0, s, r) = v_0(s, r)$ initial estimation

+ boundary conditions.

The boundary conditions allow us to constrain the topology of the surface (for instance, to consider the topology of a cylinder, a torus, or a sphere). In the following, to simplify the notation, we consider Eq. (7) with zero-boundary conditions (more general cases can be handled by a simple change of variables). In our case, we consider deformable models of fixed topology. One could refer to [24, 25] to find varying topology models.

The solution of Eq. (7) is performed in two steps. First we solve the static problem (1) with a conform Finite Element Method (FEM) [10] and then solve the evolution problem with a finite difference method. This can be done since the space variables $(s, r)$ and the temporal variable $t$ are independent.

### 5.1. *Solution of the Static Problem: The Variational Problem*

The solution of Eq. (1) is performed through a variational method. This consists of defining a bilinear form $a(u, v)$ and a linear form $L(v)$ such that solving Eq. (1) is equivalent to solving the associated variational problem: Find a function $v$ in the Sobolev space $H_0^2(\Omega)$ such that

$$a(v, u) = L_v(u), \quad \forall u \in H_0^2(\Omega). \quad (8)$$

where $H_0^2(\Omega)$ is the space of functions such that $\int_\Omega |D^m v|^2 < +\infty$ for $m = 0, 1, 2$, where $D^m v$ is the $m$th-order derivative of function $v$.

One can easily remark that solving the variational problem (8) is equivalent to minimizing the functional

$$J(u) = \tfrac{1}{2}\, a(u, u) - L_v(u).$$

Thus, by setting

$$a(u, v) = \int_\Omega w_{10}\frac{\partial u}{\partial s}\frac{\partial v}{\partial s} + w_{01}\frac{\partial u}{\partial r}\frac{\partial v}{\partial r} + w_{20}\frac{\partial^2 u}{\partial^2 s}\frac{\partial^2 v}{\partial s^2} + 2w_{11}\frac{\partial^2 u}{\partial s\partial r}\frac{\partial^2 v}{\partial s\partial r} + w_{02}\frac{\partial^2 u}{\partial r^2}\frac{\partial^2 v}{\partial r^2}\, ds\, dr$$

(remark that $a(u, v)$ is defined only for $u, v \in H_0^2(\Omega)$) and

$$L_v(u) = -\int_\Omega \nabla P(v)\, u\, ds\, dr,$$

solving (8) is equivalent to solving the Euler–Lagrange equation (1). We can further show that the problem (8) has a unique solution as long as the bilinear form $a(u, v)$ is symmetric and positive definite provided that the parameters $w_{ij}$ are positive.

### 5.2. *Discretizing the Variational Problem: The Conforming FEM*

So far, we have dealt with the continuous form of the variational problem to show the existence and the uniqueness of the solution of (8). In the following we define an approximate problem of (8) which is called the discrete variational problem. This allows us to search for an approximate solution $v_h$ of (8) in a finite-dimensional space $V_h \subset H_0^2(\Omega)$ (the solution $v_h$ is such that $\|v - v_h\|_H \to 0$ as $h \to 0$; furthermore, for the FEM used in the following the order of convergence of $\|v - v_h\|_H$ is $O(h^2)$).

These two conditions ensure good numerical properties of the discretization schemes and are the basis of the *conforming* FEM [10]. This is different from the FEM mentioned by Terzopoulos [32, 33] or Pentland and Sclaroff [29], who refer in general to a nonconforming discretization scheme of a set of differential equations, such as that of Bathe [4] for instance.

A well-known approach to approximating such problems is Galerkin's method [10], which consists of defining a similar discrete problem, over a finite-dimensional subspace $V_h$ of the Sobolev space $H_0^2(\Omega)$. The associated discrete problem for (8) is

$$\text{find } v_h \in V_h \text{ such that } a(v_h, u_h) = L_h(u_h), \quad \forall u_h \in V_h, \tag{9}$$

which leads to solving a linear system over the space $V_h$. The conforming finite element method provides an efficient tool for defining the space $V_h$. It is characterized by three aspects:

- A tessellation is established over the parameterization set $\Omega = [0, 1] \times [0, 1]$.
- The functions $v_h \in V_h$ are piecewise polynomials.
- There exists a basis in the space $V_h$ whose functions have small support.

This last feature is very important since it defines the structure of the linear system that we solve. Choosing functions with small support induces a very sparse linear system and leads to a reduced computational complexity.

Details on the tessellation of the domain $\Omega$ and the construction of the subspace $V_h$ with the Bogner–Fox–Schmit (BFS) elements [5] are given in Appendix B.

Expressing $v_h \in V_h$ in the BFS basis leads to the identity

$$\begin{aligned} v_h = \sum_{i,j=0}^{N_x-1,N_y-1} & v_h(a_{ij})\varphi_{ij} + \frac{\partial v_h}{\partial s}(a_{ij})\psi_{ij} \\ & + \frac{\partial v_h}{\partial r}(a_{ij})\eta_{ij} + \frac{\partial^2 v_h}{\partial s \partial r}(a_{ij})\zeta_{ij}, \end{aligned} \tag{10}$$

where $a_{ij} = (ih_x, jh_y)$ are the nodal points. Equation (10) gives us a $\mathcal{C}^1$ analytic solution over the set $\Omega$.

Finally the solution of the discrete problem associated with Eq. (1) is equivalent to a solution of the linear system

$$A \cdot V = L, \tag{11}$$

where the matrix $A$ is symmetric, definite positive, and tridiagonal per bloc. The reader can find all details on the variational problem and linear system (11) in Appendix B.

The FEM described above is a *conform* FEM. Some other numerical methods for solving this kind of partial differential equations are often used: the Finite Difference Method (FDM) and the *nonconforming* FEM. The FDM is easy to implement but has some drawbacks:

- the solution is known only at the nodal points and
- the applied forces are taken into account only at nodal points, and consequently, one must have a large number of nodal points to compute accurately the applied forces. This leads to a larger numerical complexity (see Section 7).

In the nonconforming FEM, the finite-dimensional space $V_h$ does not belong to the space $H_0^2(\Omega)$; this may alter the smoothness of the solution. These methods do not propagate the stress and the strain correctly [35] and do not provide a decomposition of the solution in a discrete basis of continuous functions.

### 5.3. *Solution of the Evolution Problem*

In the previous section we showed that solving Eq. (1) can be done by solving a linear system $A \cdot V = L$; consequently the discrete form of the evolution of Eq. (7) is

$$\frac{\partial V}{\partial t} + A \cdot V = L_V.$$

This equation is solved by an implicit scheme,

$$\begin{aligned} & \frac{V^t - V^{t-1}}{\tau} + A \cdot V^t = L_{V^t} \\ & V^0 = v_0 \text{ initial estimation,} \end{aligned} \tag{12}$$

where $\tau$ is the time step.

This scheme is difficult to solve since the term $L$ is complex. Thus we have chosen an implicit scheme for $V^t$ and an explicit scheme for the forces $L$. This leads to the solution of the linear system

$$(Id + \tau A) \cdot V^t = V^{t-1} + \tau L_{V^{t-1}}. \tag{13}$$

Finally to find a solution to Eq. (7) we must solve the linear system $M \cdot V = N$ at each time step, for which the matrix $M = (Id + \tau A)$ is banded, symmetric, and positive definite. This linear system is solved with a Conjugate Gradient (CG) method, in which the solution $V^{t-1}$ is taken as an initial guess at time $t$. At each time step, the CG method converges in a few iterations (3 to 10 iterations). This approach appears to have a convergence faster than that of a Cholesky factorization and take less memory for storage.

### 5.4. *Computation of the Vector L*

The vector $L$, where

$$L_{ij}(e_{ij}) = -\int_{\Omega} \nabla P(v_{t-1}(s, r)) e_{ij}(s, r)\, ds\, dr$$

($e_{ij} = (\varphi_{ij}, \psi_{ij}, \eta_{ij}, \zeta_{ij})$ and $\varphi_{ij}$, $\psi_{ij}$, $\eta_{ij}$, and $\zeta_{ij}$ are the basis functions of the BFS element), represents the contribution of the external forces (which attract the surface toward the edges) in the linear system that we solve at each iteration. Thus, the more we weight the potential $P$ the more accurate the result and the faster the convergence.

Since the potential $P$ is known only at integer values (discrete image data) we must compute the $L_{ij}(e_{ij})$ with a numerical integration. Consequently we compute $\nabla P$ at any point $(x, y, z) \in R^3$ by a trilinear interpolation of its eight neighbors.

To take into account all the contributions of the external forces, we modified the numerical integration formula such that every image point in the set $v([(i-1)h_s, (i+1)h_s] \times [(j-1)h_r, (j+1)h_r])$ is taken into account in the computation of each term $L_{ij}(e_{ij})$. Let $F(s, r) = -\nabla P(v(s, r))$, and for each rectangle $K_{ij} = [ih_s, (i+1)h_s] \times [jh_r, (j+1)h_r]$,

$$d_1 = \mathrm{Sup}[d(v(ih_s, jh_r), v((i+1)h_s, jh_r)), d(v(ih_s, (j+1)h_r), v((i+1)h_s, (j+1)h_r))]$$

and

$$d_2 = \mathrm{Sup}[d(v(ih_s, jh_r), v(ih_s, (j+1)h_r)), d(v((i+1)h_s, jh_r), v((i+1)h_s, (j+1)h_r))],$$

where $d(\cdot, \cdot)$ denotes the 3-D Euclidean distance. Thus

$$\int_{K_{ij}} F(s, r)\, ds\, dr = \sum_{l=0}^{d_1} \sum_{m=0}^{d_2} \int_{lh_s}^{(l+1)h_s} \int_{mh_r}^{(m+1)h_r} F(s, r)\, ds\, dr$$

$$= \frac{h_s h_r}{d_1 d_2} \sum_{l=0}^{d_1} \sum_{m=0}^{d_2} F\left(ih_s + \frac{2l+1}{2d_1} h_s, jh_r + \frac{2m+1}{2d_2} h_r\right).$$

This method allows us to perform an adaptive subdivision of the rectangle $K_{ij}$ without adding nodal points and consequently without increasing the size of the linear system that we solve. This method significantly reduces algorithmic complexity while increasing accuracy and convergence speed.

## 6. REGULARIZATION PARAMETERS

The elasticity and rigidity coefficients $w_{ij}$ play an important role in the convergence process of the surface toward the image edges [11]. These coefficients must be chosen in a correct way such that the internal forces generated by the energy

$$E_{\mathrm{int}}(v) = \int_{\Omega} w_{10} \left|\frac{\partial v}{\partial s}\right|^2 + w_{01} \left|\frac{\partial v}{\partial r}\right|^2 + 2w_{11} \left|\frac{\partial^2 v}{\partial s\, \partial r}\right|^2 + w_{20} \left|\frac{\partial^2 v}{\partial s^2}\right|^2 + w_{02} \left|\frac{\partial^2 v}{\partial r^2}\right|^2 ds\, dr$$

have the same magnitude as the external forces $F(v)$. In this case a minimum of the energy $E$ will be a trade-off between the internal and external energy, and the obtained surface will fit the edge points while being smooth and regular. If the internal energy is preponderant, the surface will tend to collapse on itself without detecting image edges, whereas if the external energy predominates, the surface will converge along the image edges with any degree of smoothing.

To ensure that both internal and external energy have the same order of magnitude we have found it sufficient to choose the coefficients $w_{ij}$ such that the linear system of Eq. (7) is well conditioned. This leads to

$$w_{10} = w_{01} = h_s^2 h_r^2 \quad \text{and} \quad w_{20} = w_{11} = w_{02} = h_s^3 h_r^3,$$

where $h_s$ and $h_r$ are the discretization step of $\Omega$.

We set $w_{10} = w_{01}$ and $w_{20} = w_{11} = w_{02}$ since the 3-D image data were isotropic and consequently all directions have the same weight.

## 7. NUMERICAL COMPLEXITY

In the following we compare the algorithmic complexity of the conforming FEM and the FDM. In both cases the discretization of the evolution equation (2) leads to the solution of the linear system (13), but the number of unknowns is different. In the conforming FEM at each nodal point $a_{ij}$ ($i = 0 \cdots N_s - 1$ and $j = 0 \cdots N_r - 1$), we must compute $v_h(a_{ij})$, $(\partial v_h/\partial s)(a_{ij})$, $(\partial v_h/\partial r)(a_{ij})$, and $(\partial^2 v_h/\partial s\, \partial r)(a_{ij})$; consequently the matrix $A$ is $4 \times N_s \times N_r$ and it has a tridiagonal per bloc structure (its bandwidth is $2N_s - 1$; see Appendix B.3).

In the FDM, the matrix $A$ has a pentadiagonal per bloc structure (its bandwidth is $4N_s - 1$) and its size is $N_s \times N_r$ since at each nodal point $a_{ij}$ ($i = 0 \cdots N_s - 1$ and $j = 0 \cdots N_r - 1$) we must determine $v_h(a_{ij})$.

The linear system is solved by a CG method. This method is an iterative method. At each CG iteration we must perform (44 *additions* + 42 *multiplications* + 1 *division*) $\times$ 4 $\times$ $N_s$ $\times$ $N_r$ for the conforming FEM and (18 additions + 16 multiplications + 1 division) $\times$ $N_s$ $\times$ $N_r$ for the FDM.

Consequently, if we consider the same number of nodal points $a_{ij}$ and that the necessary number of CG iterations is the same for both matrices, the numerical complexity of the conforming FEM is approximatively 12 times larger than that of the FDM. In practice, it appears [13, 14] that for 2-D deformable curves the number of points in the FDM must be at least equal to the length $l$, in pixels, of the initial guess, to compute accurately the attraction force. With the FEM, the number of points can typically be reduced to the order of $l/6$. If we assume that this result is still valid for the deformable surfaces (we have not implemented the FDM for deformable surfaces), we must consider, for the FDM, a number of discretizations points 36 times larger than that of the conforming FEM. In this case, the numerical complexity is $36/12 = 3$ times greater than the complexity of the conforming FEM.

This advantage is due to the computation of the vector $L_{V^{t-1}}$ in (13) and the nature of the method. In the classical FDM, we follow the evolution of a set of points. In the conform FEM, we actually deform the surface which is between the points of the grid, and the image forces between the grid points are also considered since the numerical integration is made at the pixel size so that no information is lost (see Section 5.4).

## 8. INFERRING THE DIFFERENTIAL STRUCTURE FROM 3-D IMAGES

### 8.1. *With the Previous Active Model*

In the previous sections we showed how to use the deformable surface to segment and fit some 3-D image edge points. In the following we assume that the surface has localized accurately the 3-D image edges, which means that we have reached a minimum of $E$. We now use this surface to compute the differential characteristics of the 3-D image surface boundary. This computation can be done analytically at each point of the surface since the use of conform FEM gives an analytic representation of the surface $v(s, r)$,

$$v = \sum_{i,j=0}^{N_s,\,N_r-1} v_{ij}(a_{ij})\varphi_{ij} + \frac{\partial v_{ij}}{\partial s}(a_{ij})\psi_{ij} + \frac{\partial v_{ij}}{\partial r}(a_{ij})\eta_{ij} + \frac{\partial^2 v_{ij}}{\partial s\,\partial r}(a_{ij})\zeta_{ij}, \tag{14}$$

where $\varphi_{ij}$, $\psi_{ij}$, $\eta_{ij}$, and $\zeta_{ij}$ are the basis functions and the coefficients $v_{ij}(a_{ij})$, $(\partial v_{ij}/\partial s)\,(a_{ij})$, $(\partial v_{ij}/\partial r)\,(a_{ij})$, and $(\partial^2 v_{ij}/\partial s\,\partial r)\,(a_{ij})$ are computed by solving the linear system (12) (Fig. 1). Another major contribution of the analytic repre-

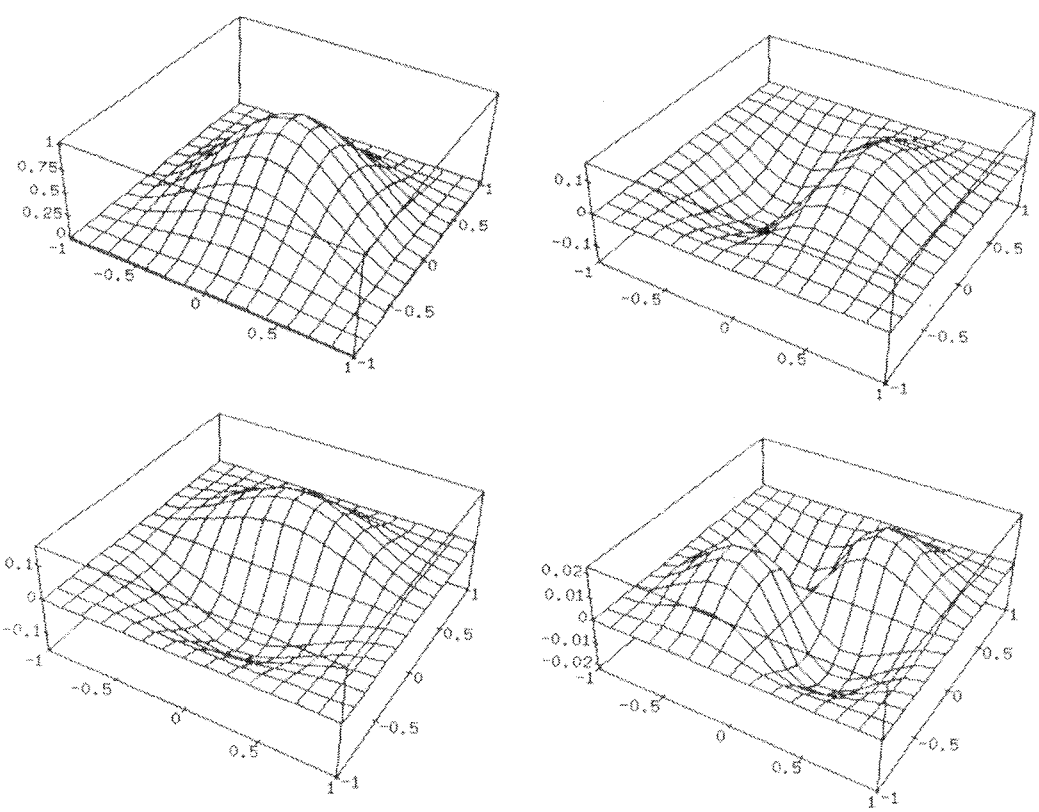


FIG. 1. Surface plots of the four basis functions $\varphi$, $\psi$, $\eta$, and $\zeta$.

sentation is that at each point of the surface, the tangent plane is given by the vectors $v_s = \partial v/\partial s$ and $v_r = \partial v/\partial r$ (in the following the subscripts $s$, $r$, $ss$, $rr$, and $sr$ denote the first and second derivatives). This allows us to compute all the differential characteristics in local coordinates and consequently handle more general situations.

Let us consider the basis $\{v_s, v_r, N\}$, where $N = (v_s \times v_r)/|v_s \times v_r|$, and let $T_p(\mathcal{S})$ denote the tangent plane to $\mathcal{S}$ at the point $p \in R^3$. Since $N_s$ and $N_r$ belong to $T_p(\mathcal{S})$ we can write

$$N_s = a_{11}v_s + a_{12}v_r$$
$$N_r = a_{21}v_s + a_{22}v_r,$$

and therefore $dN_p$, which denotes the differential of $N$ computed at point $p$, is given by the matrix $(a_{ij})_{i,j=1,2}$ in the basis $\{v_s, v_r, N\}$. This matrix can be expressed in terms of the first and second fundamental forms of $\mathcal{S}$ as [17]

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \frac{-1}{EG - F^2} \begin{pmatrix} e & f \\ f & g \end{pmatrix} \begin{pmatrix} G & -F \\ -F & E \end{pmatrix},$$

where $E$, $F$, $G$, $e$, $f$, and $g$ are the coefficients of the first and second fundamental forms in the basis $\{v_s, v_r, N\}$ defined by

$$E = \langle v_s, v_s \rangle, \quad F = \langle v_s, v_r \rangle, \quad G = \langle v_r, v_r \rangle,$$
$$e = \langle N, v_{ss} \rangle, \quad f = \langle N, v_{sr} \rangle, \quad g = \langle N, v_{rr} \rangle.$$

This gives the relations

$$a_{11} = \frac{fF - eG}{EG - F^2}, \quad a_{12} = \frac{gF - fG}{EG - F^2},$$
$$a_{21} = \frac{eF - fE}{EG - F^2}, \quad a_{22} = \frac{fF - gE}{EG - F^2},$$

known as the *equations of Weingarten*. Thus the Gaussian curvature $K$ and the mean curvature $H$ of $\mathcal{S}$ at $p$ are

$$K = \det(dN_p) = \det(a_{ij}) = \frac{eg - f^2}{EG - F^2},$$
$$H = -\frac{1}{2}(a_{11} + a_{22}) = \frac{1}{2}\frac{eG - 2fF + gE}{EG - F^2}.$$

The principal curvatures are the opposites of the eigenvalues of $dN_p$; they satisfy the equation

$$\det(dN_p + kI) = k^2 - 2Hk + K = 0$$
$$\text{and therefore} \quad k = H \pm \sqrt{H^2 - K}.$$

The principal curvature field can be used to extract the maxima of the larger curvature in the direction of the larger curvature [26]. This yields the ridges of the surface, which are very useful for matching. In Fig. 16, one can visualize the value of the larger principal curvature. The result appears to be qualitatively correct and can be compared to those obtained in [26] by another method, consisting of fitting locally a quadratic surface model. Our results are a little more noisy, but the advantage is that the segmentation and curvature are computed simultaneously. Also, our approach appears to be computationally much less expensive.

### 8.2. *Using the Surface Normals to Improve the Model*

In the previous example, the computation of the curvatures was based only on the location of the surface (i.e., the sole data given were the 3-D points $(X, Y, Z)$, which are the 3-D image edge points) and the obtained curvatures are qualitatively correct, but the curvature does not vary smoothly. We believe that the use of additional information such as the normals of the surface improves the result of the computation of the curvatures. Monga *et al.* already used this property in [26] to fit local quadrics models to 3-D edgels. Lee [23] used positions and normals to fit surfaces fusing binocular stereo and photometric informations. Terzopoulos [32] also used the normal information for surface fitting.

Indeed, if we assume that the sought surfaces are noisy *isointensity* surfaces (as is usually the case in medical images), then the image gradient (estimated by a 3-D edge detector [27]) is colinear to the surface normal $N$ and provides this information.

If we use a parameterized surface $v(s, r) = (x(s, r), y(s, r), z(s, r))$, taking into account the normals leads to a nonlinear minimization problem [31]. To avoid this, we use a property of regular surfaces which states that for each surface point there exists a neighborhood such that the surface can be expressed in the form $v(s, r) = (s, r, g(s, r))$ [17, p. 164]. This is done by choosing locally a reference frame such that the first two vectors of the basis belong to the tangent plane. We call this reference frame the tangent plane reference frame.

If $v$ is a surface of the form $(s, r, v(s, r))$, a vector colinear to the unit normal has the simple form $N(s, r) = (-v_s, -v_r, 1)$. To constrain the surface $v$ to fit the data points $(X, Y, Z)$ (obtained by a previous application of a simple deformable model) and the normals $N$ (estimated by a 3-D edge detector [27]), we transform locally these data in the tangent plane reference frame of the surface. Once this is done, we divide the first two coordinates of the normal vector by the third one, obtaining a measured normal of the form $(N_1, N_2, 1)$ (this is always possible

within a neighborhood of a regular surface point in the tangent plane reference frame).

We can now define a new energy given by $E = E_{\text{smooth}} + E_{\text{location}}$, where

$$E_{\text{location}}(v) = \frac{1}{2} C_1 \int_\Omega |v(s, r) - z|^2 \, ds \, dr + \frac{1}{2} C_2 \left( \int_\Omega |v_s(s, r) + N_1|^2 \, ds \, dr + \int_\Omega |v_r(s, r) - N_2|^2 \, ds \, dr \right), \quad (15)$$

where $z$ represent the third component of the projection of the data point $(X, Y, Z)$ in the tangent plane reference frame.

Here, once again, the energy $E_{\text{smooth}}$ constrains the surface to be smooth the $E_{\text{location}}$ measures the discrepancy between the data and the surface. The coefficients $C_1$ and $C_2$ allow us to weigh differently the data points and the normals. This allows us to choose the coefficients $C_1$ and $C_2$ depending on the reliability of the data points or the normals (up to now we set $C_1 = C_2 = 1$).

A minimum of $E$ is obtained by solving the equation

$$-\frac{\partial}{\partial s}\left(w_{10}\frac{\partial v}{\partial s}\right) - \frac{\partial}{\partial r}\left(w_{01}\frac{\partial v}{\partial r}\right) + 2\frac{\partial^2}{\partial s \partial r}\left(w_{11}\frac{\partial^2 v}{\partial s \partial r}\right) + \frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 v}{\partial s^2}\right) + \frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 v}{\partial r^2}\right) \quad (16)$$

$$= -C_1(v - z) - C_2\frac{\partial}{\partial s}(v_s + N_1) - C_2\frac{\partial}{\partial r}(v_r - N_2)$$

+ boundary conditions.

This equation is solved with a conforming FEM as described in Section 5.

## 9. EXPERIMENTAL RESULTS

We now give results of the algorithms presented so far. Using a 3-D deformable model to segment a 3-D image provides better results than the iterated application of a 2-D deformable model to successive 2-D cross sections [14]. Indeed, the 3-D model easily bridges edge gaps in 3-D, i.e. not only within a cross section, but also between cross sections, ensuring that the result is globally a smooth surface and not only a collection of smooth planar curves. This significantly improves the robustness of the segmentation; for instance, it is even possible to remove all the edges of a single cross section (assuming that the edges are correctly detected in the other ones) without degrading the final result too much.

We present in Fig. 2 an example with artificial data. This figure represents overlays of some horizontal cross sections of the initial surface with the original data. The 3-D image here is a cylinder where we have removed some edges in three successive cross sections to compare the results obtained by a 2-D model applied to successive cross sections with those of a full 3-D deformable model. With the deformable surface we can restore the lost edges and obtain a perfect reconstruction of the cylinder (Fig. 3), whereas a 2-D model [14] cannot restore the lost edges even if we use the same attraction force as that for the 3-D model.

The deformable model can also handle noisy data. In Fig. 4, we have added Gaussian noise ($\sigma^2 = 0.8$) to the edge locations of a cylinder with an elliptic base. This figure shows some cross sections of the obtained surface with the data and a representation of the surface. We have also conducted some experiments on the accuracy of the model and its ability to handle noisy data. We have considered a set of data points (a surface plot is given in Fig. 17) obtained by sampling the analytical function $(s, r, z = \sqrt{1.0 - (s - 50.0)^2/2500.0})$ and corrupting by a

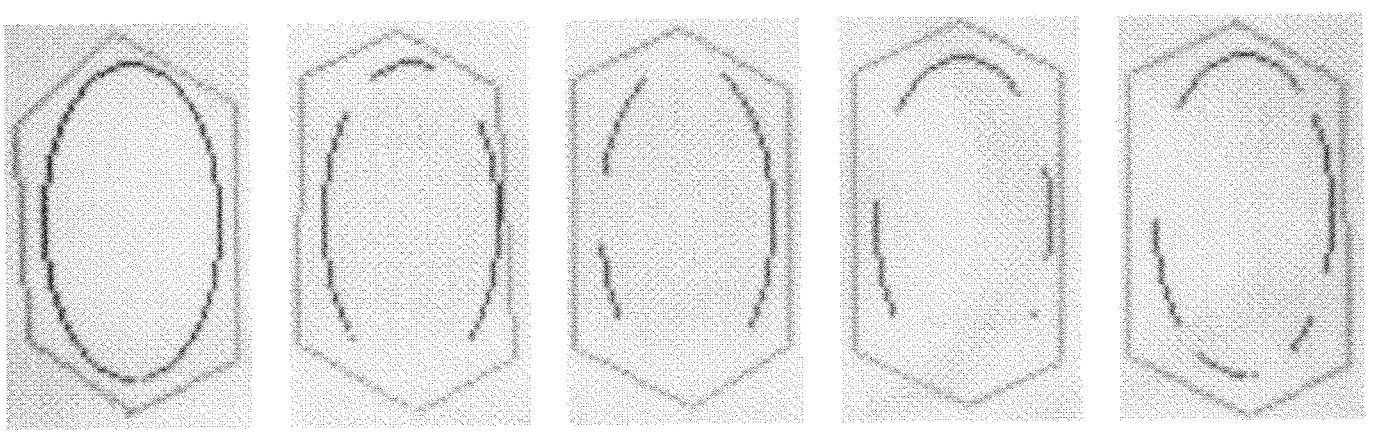

FIG. 2. Some successive cross sections of image edges (in black) and initial surface (in gray) given by the user.

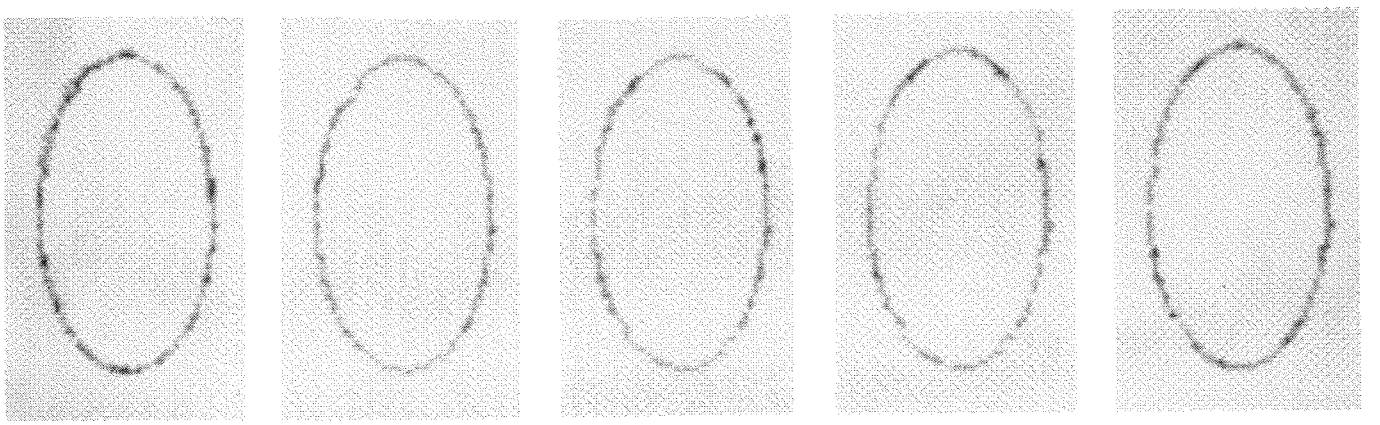

FIG. 3. Here we show how the deformable surface (in gray) can reconstruct the lost edges while having a 3-D homogeneity. In this example, a 2-D model cannot reconstruct the missing edges even if we use a 3-D potential.

Gaussian noise of variance $\sigma^2 = 0.01$, $\sigma^2 = 0.05$, and $\sigma^2 = 0.1$. Table 1 represents the obtained values, at six different points, of the surface approximating the function $z$. The mean error and the variance error are computed over the whole data set (these data vary between 0 and 1).

Figure 5 shows a set of cross sections of a vertebra (out of a total of 50 cross sections), obtained with an X-ray scanner and the initial segmentation provided by the user (the external curves show the intersection of the given surface with the corresponding cross sections).

Figure 6 shows the resulting surface obtained after 40 iterations, in the same set of cross sections. It is interesting to note the remarkable accuracy of the segmentation, although the detected contours were often incomplete (due to noise effect). Figure 7 shows a wire-frame representation of the resulting surface.

Figure 8 represents some cross sections of the 3-D edge image of a human heart (obtained with Magnetic Resonance Imaging (MRI)) with the initial surface (in gray). Figure 9 shows some cross sections of the surface, once it has reached the minimum of $E$. We can note the good localization of the surface on the 3-D edge points. Figure 10 gives a 3-D representation of the inside cavity of the left ventricle.

Figure 11 represents some cross sections of a human head (out of 70) obtained with MRI. Figure 12 shows some cross sections of the surface with the 2-D images. A representation of the surface is given in Fig. 13. Figure 14 shows the cross sections of the surface obtained on another set of MRI images and Fig. 15 a rendered representation of the surface.

We now present the results of the computation of curvature on the face surface. Figure 16 shows the computation of the larger value of the two principal curvatures, without using the normal information (the black areas correspond to high-curvature regions and the light gray areas to low-curvature regions). We can easily remark that the black areas tend to characterize the structures of the surface. These characteristics could be used to recog-

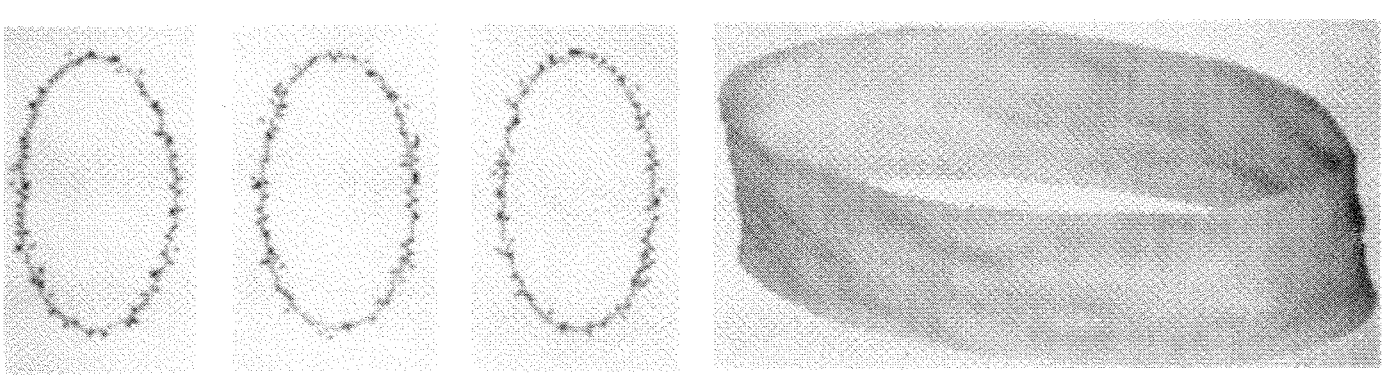

FIG. 4. This figure illustrate the use of deformable surface to segment noisy data. We added a Gaussian noise ($\sigma^2 = 0.8$) to the edge location of a cylinder with an elliptic base. This figure represents some cross sections of the solution (in gray) with the image plane and the complete surface.

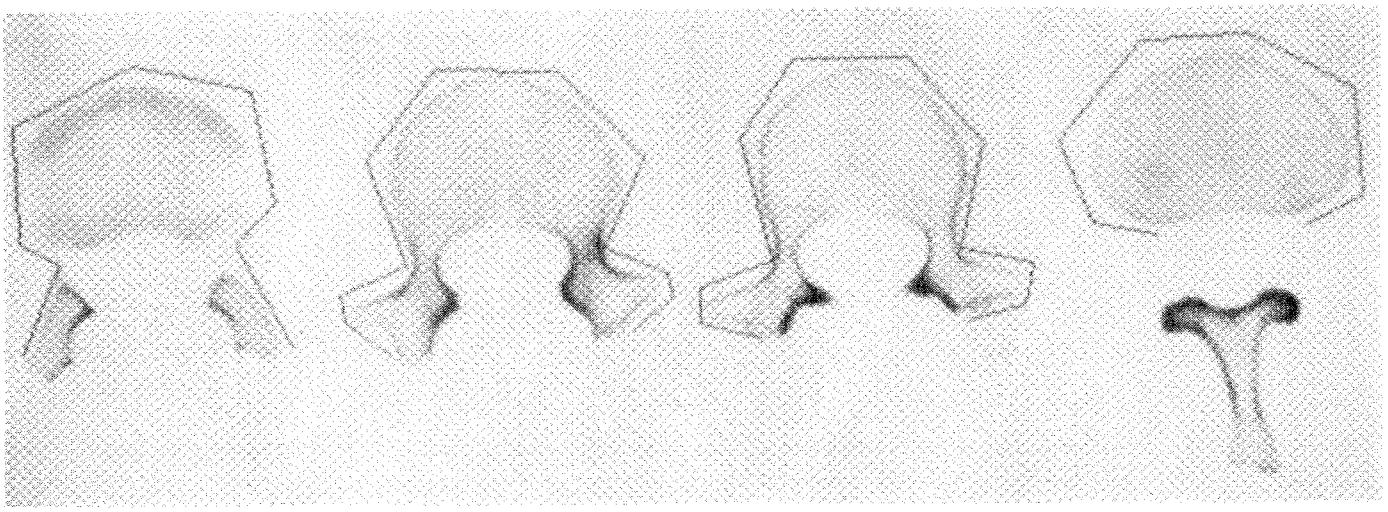

FIG. 5. Some cross sections of the initial surface given by the user.

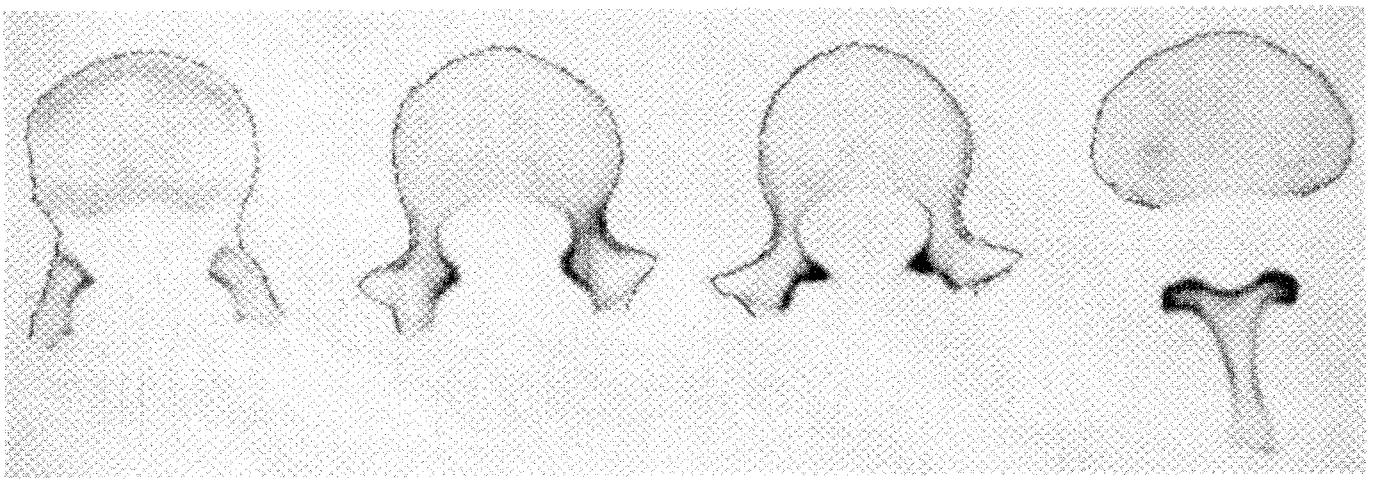

FIG. 6. The corresponding cross sections of the solution.

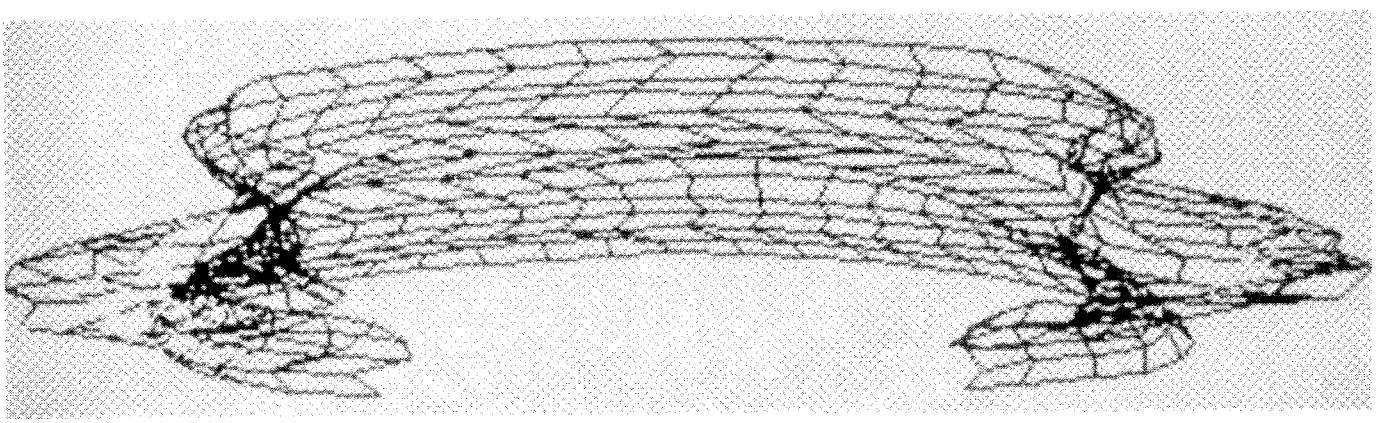

FIG. 7. A wire-frame representation of the vertebra.

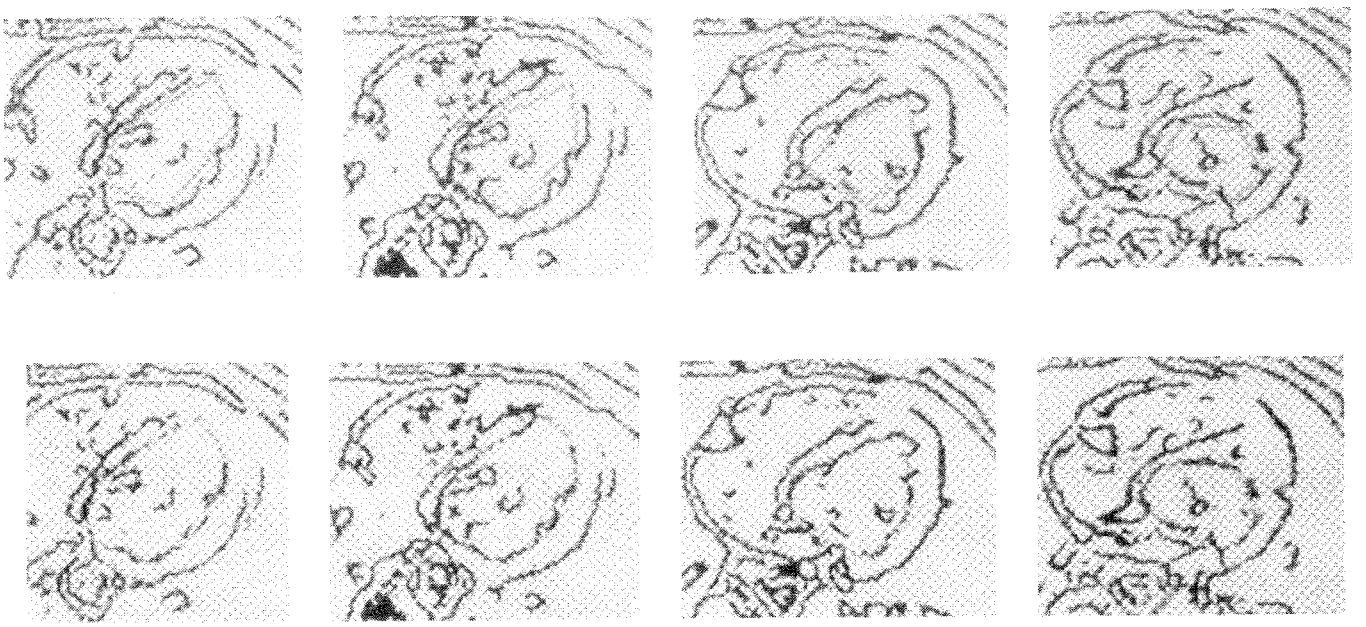

FIG. 9. Here we give a further example where we use a deformable surface constrained by boundary conditions (cylinder type) to segment the inside cavity of the left ventricle. Overlays of some cross sections of the surface (in gray) are shown with the contour image to show the accurate localization of the surface.

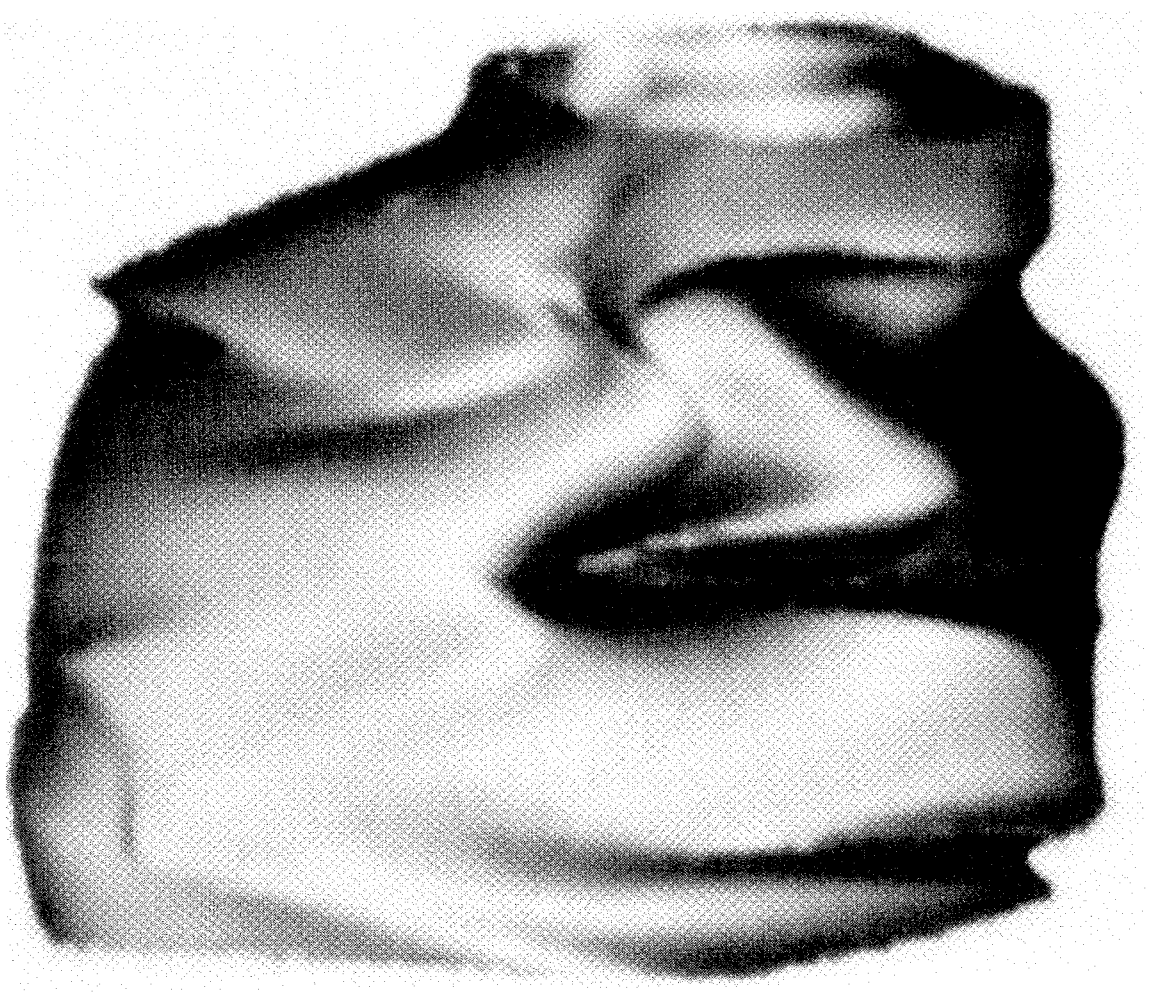

FIG. 10. A representation of the inside surface of the left ventricle.

TABLE 1
Results Obtained on Some Points of the Computed Surface on Noisy Data for Three Different Values of $\sigma$

| Points | Theoretical value $z_t$ | $\sigma^2 = 0.01$ Compued value $z_c$ | $\sigma^2 = 0.01$ Relative error $\|z_t - z_c\|$ | $\sigma^2 = 0.05$ Computed value $z_c$ | $\sigma^2 = 0.05$ Relative error $\|z_t - z_c\|$ | $\sigma^2 = 0.1$ Computed value $z_c$ | $\sigma^2 = 0.1$ Relative error $\|z_t - z_c\|$ |
|---|---|---|---|---|---|---|---|
| (0.375, 0.5) | 0.9682 | 0.8870 | 0.0812 | 0.8609 | 0.1073 | 0.8333 | 0.1349 |
| (0.5, 0.5) | 1.0000 | 0.9183 | 0.0817 | 0.8973 | 0.1027 | 0.8513 | 0.1487 |
| (0.75, 0.5) | 0.8660 | 0.8090 | 0.0570 | 0.7690 | 0.0970 | 0.6805 | 0.1855 |
| (0.75, 0.5) | 0.9682 | 0.8796 | 0.0886 | 0.8302 | 0.1380 | 0.7803 | 0.1879 |
| (0.5, 0.25) | 1.0000 | 0.9077 | 0.0923 | 0.8445 | 0.1555 | 0.7598 | 0.2402 |
| (0.75, 0.75) | 0.8660 | 0.8243 | 0.0417 | 0.8313 | 0.0347 | 0.7981 | 0.0679 |
| Mean error | | | 0.0658 | | 0.0878 | | 0.1409 |
| Variance error | | | 0.0175 | | 0.0316 | | 0.0820 |

nize anatomical structures from the 3-D images. Nevertheless, as discussed before, the results are qualitatively correct, but noisy.

To validate the use of the normal, we conducted a series of experiments on synthetic data. In Fig. 17 a rendered representation of the data and the theoretical value of the larger value of the two principal curvatures are shown. We have added Gaussian noise ($\sigma^2 = 0.01$) to

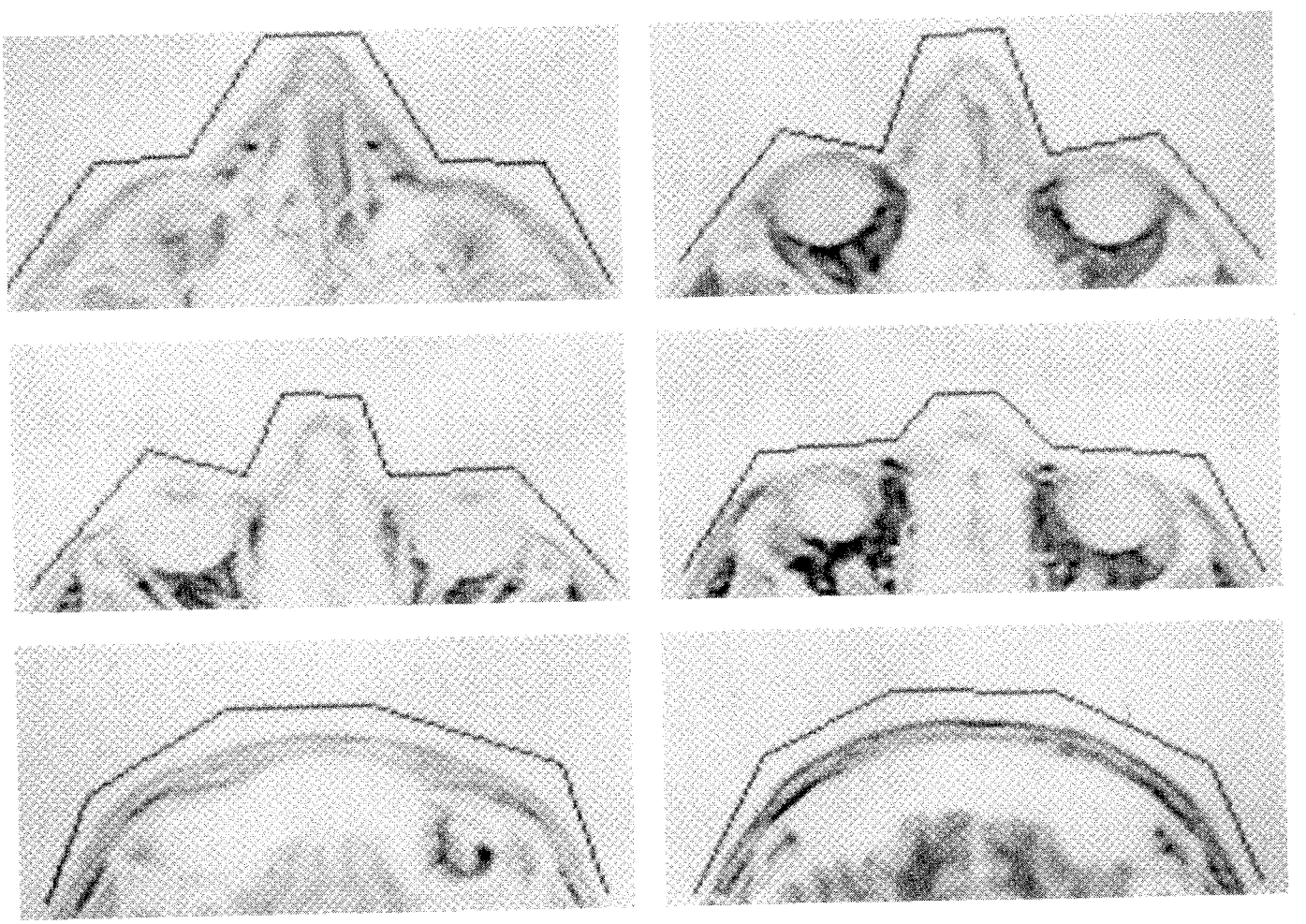

FIG. 11. Overlays of some cross sections of the initial surface given by the user with the original data.

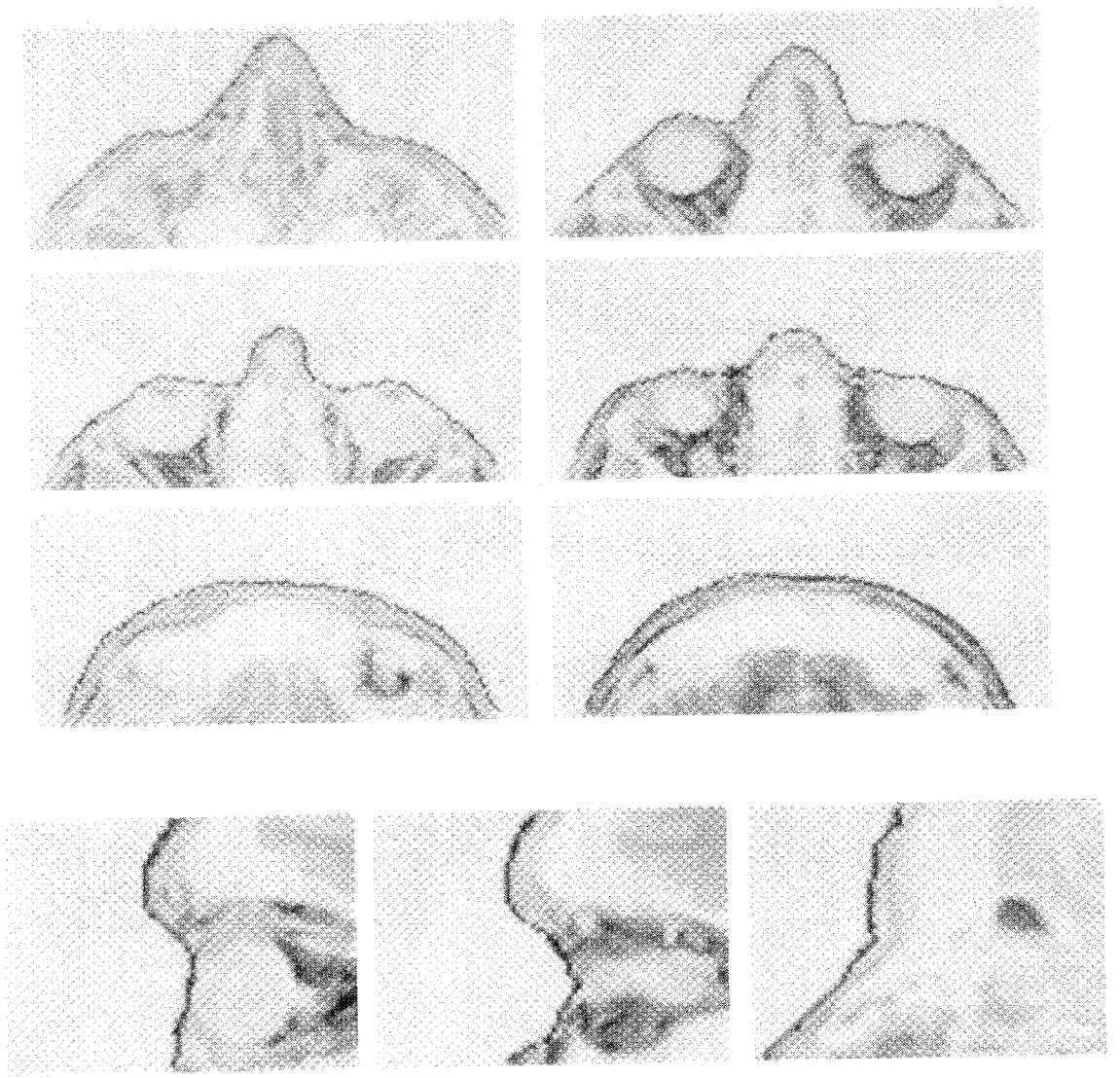

FIG. 12. Overlays of some horizontal and vertical cross sections of the surface obtained by the algorithm with the original data.

these data (Fig. 18). Figure 19 (left) shows the result of approximating the surface to the noisy data and the computation of the curvature information (Fig. 19, right) without the use of the normals. Figure 20 (right) shows the same quantities computed with the information of the surface normals and demonstrates the importance of this information for regularizing the result. This visual qualitative improvement is confirmed by a qualitative comparison of the computed second-order derivatives at different points on the surface. Typically, a gain of 10% in the accuracy of the second-order derivatives is provided by the normal information. Table 2 shows the larger value of the principal curvatures at three distinct points.

## 10. CONCLUSION AND FUTURE RESEARCH

We have shown how a deformable surface can be used to segment 3-D images by minimizing an appropriate energy. The minimization process is done by a variational method with a conforming finite element. Our method, including the use of an adaptive subdivision for force computation, has the following advantages:

1. it requires less discretization points and consequently produces a smaller linear system to be solved, thus reducing significantly the algorithmic complexity, and

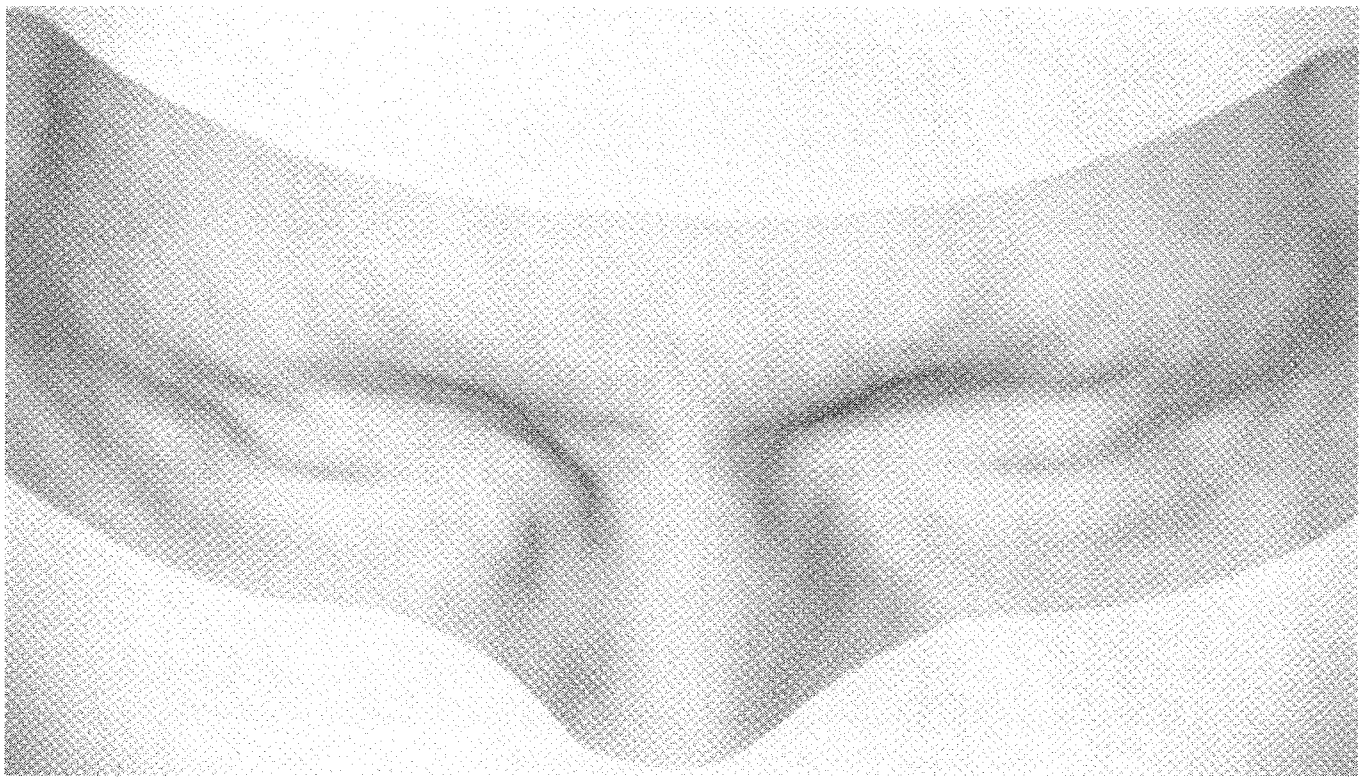

FIG. 13. A 3-D representation of the surface.

2. it provides an analytical representation of the surface.

This last feature is the most important one for inferring differential structures of the surface, and we showed how to compute the first and second fundamental forms of the deformable model. These characteristics provide a helpful tool for recognizing 3-D objects. They will be used soon to match the deformable surface to an anatomical atlas [2].

## APPENDIX A: SURFACE AND 3-D EDGE POINTS

In this first appendix we give a necessary and sufficient condition for a surface to produce a local extremum of the energy

$$E_P(\mathcal{S}) = -\frac{1}{|\mathcal{S}|}\int_\Omega |\nabla \mathcal{I}(x(s,r))| dA,$$

where $|\mathcal{S}| = \int\int_\Omega |x_s \times x_r|\, dsdr$ and $dA = \sqrt{EG - F^2}\, dsdr$ is the standard surface area measure. A necessary and sufficient condition for the surface $\mathcal{S}$ to produce a local extremum of $E_P$ with respect to infinitesimal deformations is

$$\frac{d|\nabla \mathcal{I}(x(s,r))|}{dN(x(s,r))} = \frac{eG - 2fF + gE}{EG - F^2} \left(|\nabla \mathcal{I}(x(s,r))| - \frac{1}{|\mathcal{S}|}\int\int_\Omega |\nabla \mathcal{I}(x(s,r))|\, dsdr\right), \quad (17)$$

TABLE 2
The Comparison between the Larger Value of the Principal Curvature Computed on Noisy Data with and without the Use of Normals

| | | Perfect data | Noisy data | | | |
|---|---|---|---|---|---|---|
| Points | Theoretical $\kappa_1$ | Computed curvature $\kappa_c$ | Without normal $\kappa_c$ | $\|\kappa_1 - \kappa_c\|$ | With normals $\kappa_c$ | $\|\kappa_1 - \kappa_c\|$ |
| (0.25, 0.5) | 2.658 | 2.418 | 1.846 | 0.812 | 1.924 | 0.734 |
| (0.5, 0.75) | 1.609 | 1.687 | 1.627 | 0.018 | 1.605 | 0.004 |
| (0.65, 0.5) | 2.141 | 1.276 | 1.746 | 0.409 | 1.802 | 0.340 |

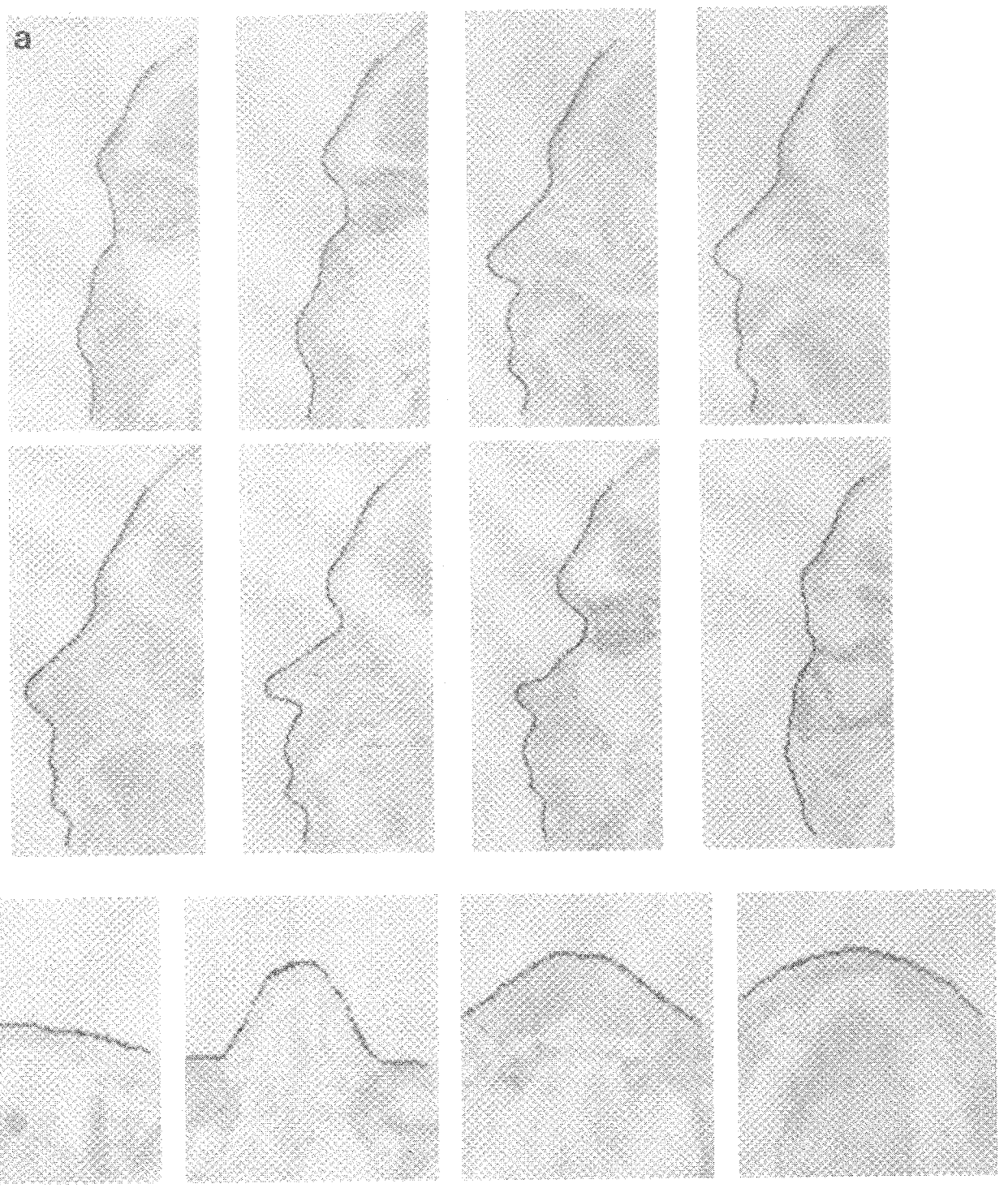


FIG. 14. Another example of the segmentation of the human face from MR images. Overlays of some vertical cross sections of the surface obtained by the algorithm with the original data.

where $E(s, r)$, $F(s, r)$, $G(s, r)$, $e(s, r)$, $f(s, r)$, and $g(s, r)$ are the coefficients of the first and second fundamental form in the basis $\{x_s, x_r, N\}$ (see [17] for details about the notations), $\Omega = [0, L] \times [0, M]$, and $P = |\nabla \vartheta|$.

Let us consider $\mathcal{S}_\lambda$, a small deformation of the surface $\mathcal{S}$ such that the parametrization of $\mathcal{S}_\lambda$ is

$$x^\lambda = x + \lambda(\alpha x_s + \beta x_r + \gamma N), \tag{18}$$

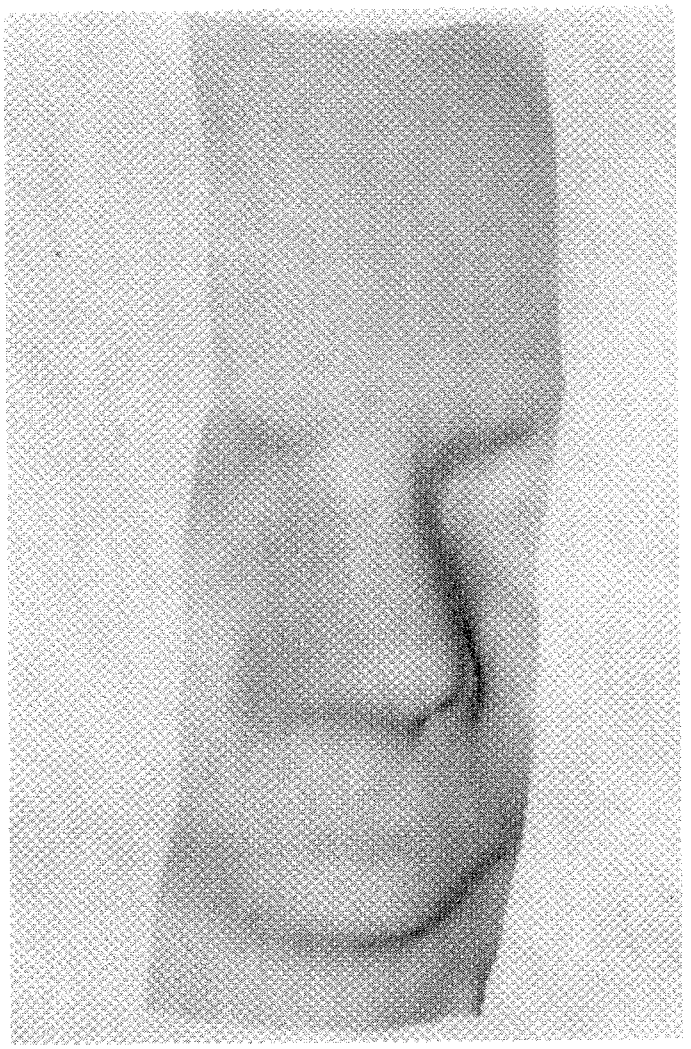

FIG. 15. A 3-D representation of the surface.

where $\alpha(s, r)$, $\beta(s, r)$, and $\gamma(s, r)$ are arbitrary continuous and differentiable functions and $x_s$, $x_r$, and $N$ are the derivatives of $x$ and the normal to the surface.

$\mathcal{S}$ is a local extremum of $E_P$ if and only if

$$\left.\frac{dE_P(\mathcal{S}_\lambda)}{d\lambda}\right|_{\lambda=0} = 0, \tag{19}$$

for all $\alpha$, $\beta$, and $\gamma$.

We show that (19) holds if and only if (17) is satisfied.

By definition

$$E_P(\mathcal{S}_\lambda) = -\frac{\int_\Omega P(x^\lambda(s, r))dA}{\int_\Omega dA} = -\frac{\int\int_\Omega P(x^\lambda(s, r))\sqrt{EG - F^2}\,dsdr}{\int\int_\Omega \sqrt{EG - F^2}\,dsdr}, \tag{20}$$

where $E$, $F$, and $G$ are the coefficients of the first fundamental form of $\mathcal{S}_\lambda$.

To compute the derivative $(dE_P(\mathcal{S}_\lambda)/d\lambda)|_{\lambda=0}$, we need to compute the derivatives of the vectors $x_s$, $x_r$, and $N$. For this purpose we use the equations [17, Sect. 4.3, p. 231]

$$\begin{aligned} x_{ss} &= \Gamma^1_{11}x_s + \Gamma^2_{11}x_r + eN, \\ x_{sr} &= x_{rs} = \Gamma^1_{12}x_s + \Gamma^2_{12}x_r + fN, \\ x_{rr} &= \Gamma^1_{22}x_s + \Gamma^2_{22}x_r + gN. \end{aligned}$$

where the coefficients $\Gamma^k_{ij}$ are the Christoffel symbols of $\mathcal{S}$ in the parametrization $x$, and $e$, $f$, $g$ are the coefficients of the second fundamental form of $\mathcal{S}$. In the following the Christoffel symbols $\Gamma^k_{ij}$ have been replaced in terms of the coefficients of the first fundamental form $E$, $F$, $G$ and their derivatives.

Thus computing $dE_P(\mathcal{S}_\lambda)/d\lambda$ and evaluating it at $\lambda = 0$ leads to

$$\begin{aligned} |\mathcal{S}|\left.\frac{dE_P(\mathcal{S}_\lambda)}{d\lambda}\right|_{\lambda=0} = 0 = &\int\int_\Omega \sqrt{EG - F^2}\alpha\left[|\mathcal{S}|\frac{dP}{dx_s}\right. \\ &\left. + (|\mathcal{S}|P - P_m)(\Gamma^1_{11} + \Gamma^2_{12})\right]dsdr \\ &+ \int\int_\Omega \alpha_s\sqrt{EG - F^2}(|\mathcal{S}|P - P_m)dsdr \\ &+ \int\int_\Omega \sqrt{EG - F^2}\beta\left[|\mathcal{S}|\frac{dP}{dx_r}\right. \\ &\left. + (|\mathcal{S}|P - P_m)(\Gamma^1_{12} + \Gamma^2_{22})\right]dsdr \\ &+ \int\int_\Omega \beta_r\sqrt{EG - F^2}(|\mathcal{S}|P - P_m)dsdr \\ &+ \int\int_\Omega \gamma\left[\sqrt{EG - F^2}|\mathcal{S}|\frac{dP}{dN}\right. \\ &\left. - (|\mathcal{S}|P - P_m)\frac{eG - 2fF + gE}{\sqrt{EG - F^2}}\right]dsdr \end{aligned}$$

where $P_m = \int_\Omega PdA$.

Integrating by parts the integral (except the last one) yields (17) as a necessary and sufficient condition for (19) to be satisfied for all $\alpha$, $\beta$, and $\gamma$.

## APPENDIX B: DETAILS ON THE NUMERICAL SOLUTION

### B.1. *Variational Formulation*

Let $\varphi \in H^2_0(\Omega)$ be a smooth function. If $v$ is a solution of Eq. (7), we have

FIG. 16. A representation of the larger value of the principal curvatures. The high values are in black and the low values are in light gray. These values characterize some structures of the human face such as the eyebrows and the nose.

$$\int_\Omega \frac{\partial v}{\partial t}\varphi ds dr - \int_\Omega \frac{\partial}{\partial s}\left(w_{10}\frac{\partial v}{\partial s}\right)\varphi ds dr - \int_\Omega \frac{\partial}{\partial r}\left(w_{01}\frac{\partial v}{\partial r}\right)\varphi ds dr + 2\int_\Omega \frac{\partial^2}{\partial s \partial r}\left(w_{11}\frac{\partial^2 v}{\partial s \partial r}\right)\varphi ds dr + \int_\Omega \frac{\partial^2}{\partial s^2}\left(w_{20}\frac{\partial^2 v}{\partial s^2}\right)\varphi ds dr + \int_\Omega \frac{\partial^2}{\partial r^2}\left(w_{02}\frac{\partial^2 v}{\partial r^2}\right)\varphi ds dr = -\int_\Omega \nabla P(v)\varphi ds dr,$$

where the function $v$ depends on $t$, $s$, and $r$. We remark that the variables $(s, r)$ and $t$ are independent; we can separate them (for more details see [10]). Green's formula yields

$$\frac{d}{dt}\int_\Omega v\varphi ds dr + \int_\Omega w_{10}\frac{\partial v}{\partial s}\frac{\partial \varphi}{\partial s}ds dr + \int_\Omega w_{01}\frac{\partial v}{\partial r}\frac{\partial \varphi}{\partial r}ds dr + 2\int_\Omega w_{11}\frac{\partial^2 v}{\partial s \partial r}\frac{\partial^2 \varphi}{\partial s \partial r}ds dr + \int_\Omega w_{20}\frac{\partial^2 v}{\partial s^2}\frac{\partial^2 \varphi}{\partial s^2}ds dr + \int_\Omega w_{02}\frac{\partial^2 v}{\partial r^2}\frac{\partial^2 \varphi}{\partial r^2}ds dr = -\int_\Omega \nabla P(v)\varphi ds dr.$$

Let us set

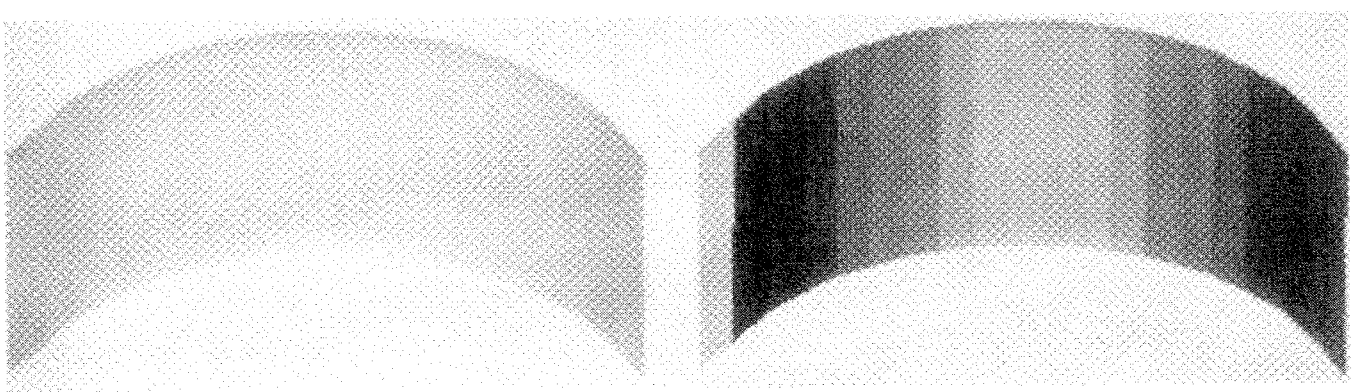

FIG. 17. A rendered representation of the theoretical data (left) and the value of the larger of the two principal curvatures (right).

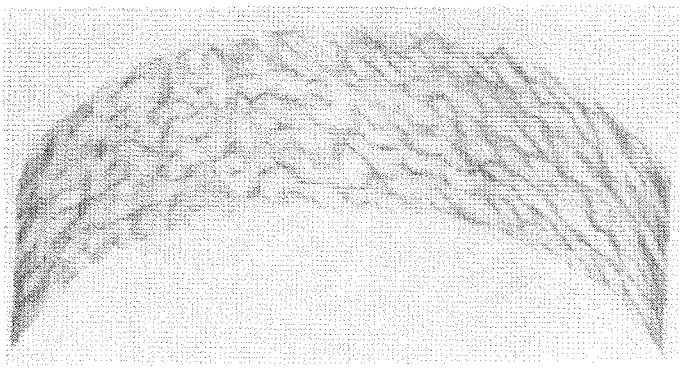

FIG. 18. A rendered representation of the noisy data ($\sigma^2 = 0.01$).

$$a(u, v) = \int_\Omega w_{10} \frac{\partial u}{\partial s} \frac{\partial v}{\partial s} + w_{01} \frac{\partial u}{\partial r} \frac{\partial v}{\partial r} + w_{20} \frac{\partial^2 u}{\partial s^2} \frac{\partial^2 v}{\partial s^2} + 2w_{11} \frac{\partial^2 u}{\partial s \partial r} \frac{\partial^2 v}{\partial s \partial r} + w_{02} \frac{\partial^2 u}{\partial r^2} \frac{\partial^2 v}{\partial r^2} \, ds dr$$

and

$$L_v(u) = -\int_\Omega \nabla P(v) u \, ds dr.$$

This leads to a new formulation of the problem given $v_0 \in L^2(\Omega)$ and $\nabla P \in L^2(0, T, L^2(\Omega))$, find a function $v \in L^2(0, T, H_0^2(\Omega)) \cap \mathscr{C}^1(0, T, L^2(\Omega))$ satisfying

$$\begin{aligned} &\frac{d}{dt}(v(t), u) + a(v(t), u) = L_v(u), \quad \forall u \in H_0^2(\Omega), \\ &v(0) = v_0(s, r) \\ &w_{ij} \in L^\infty(\Omega) \quad \text{and} \quad w_{ij}(s) \geq \alpha > 0. \end{aligned} \tag{21}$$

Since the variables, $s$, $r$, and $t$ are independent, we can solve Eq. (21) in two steps. First solve the static equation find $v \in H_0^2(\Omega)$ such that

$$a(v, u) = L_v(u), \quad \forall u \in H_0^2(\Omega), \tag{22}$$

where $L_v$ is not supposed to depend on $v$ (we remind the reader that there exists a unique solution to this equation, since the bilinear form $a(u, v)$ is symmetric and positive definite as long as $w_{ij} > 0$), and then solve the evolution equation (21). This yields Eq. (13).

### B.2. *Tessellation of* $\Omega$ *and the Basis Functions*

Given the numbers of discretization points in the two axes of parametrization $N_s$, $N_r > 1$, we set $h_s = 1/(N_s - 1)$, $h_r = 1/(N_r - 1)$ and consider a uniform subdivision of $\Omega$ of step size $h_s$ and $h_r$, composed of the nodes $a_{i,j} = (x_i, y_j) = (ih_s, jh_r)$, $0 \leq i \leq N_s - 1$, $0 \leq j \leq N_r - 1$. Thus

$$\Omega = [0, 1] \times [0, 1] = \bigcup_{i,j=0}^{N_s-1, N_r-1} K_{i,j}$$

$$= \bigcup_{i,j=0}^{N_s-1, N_r-1} [ih_s, (i + 1)h_s] \times [jh_r, (j + 1)h_r].$$

Since the higher derivatives appearing in Eq. (1) are of fourth order, the conform finite element space $V_h$ must satisfy $V_h \subset \mathscr{C}^1 \cap H_0^2(\Omega)$ (for details see [10]). For this purpose the space $H_0^2(\Omega)$ is approximated with the Bogner–Fox–Schmit elements [5, 10] defined by

- The rectangles $K_{ij}$, defined by the vertices $c_k$, $1 \leq k \leq 4$,
- The set $P_{K_{ij}}$ of polynomials containing the basis functions

$$P_{K_{ij}} = Q_3(R^2) = \left\{ p, \, p(s, r) = \sum_{0 \leq k,l \leq 3} \gamma_{k,l} s^k r^l \right\},$$

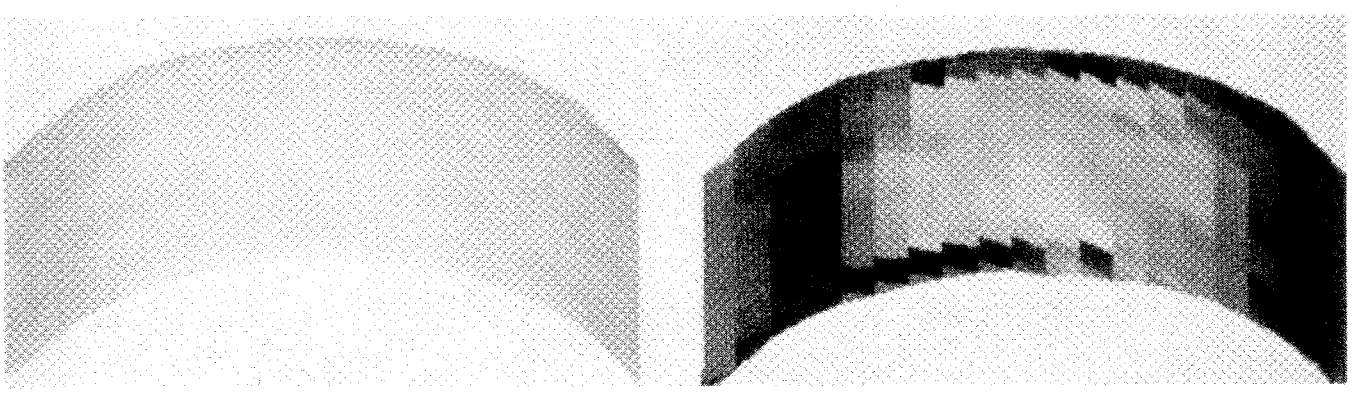

FIG. 19. A rendered representation of the surface fitted to the noisy data ($\sigma^2 = 0.01$) without normals (left) and the value of the larger of the two principal curvatures (right).

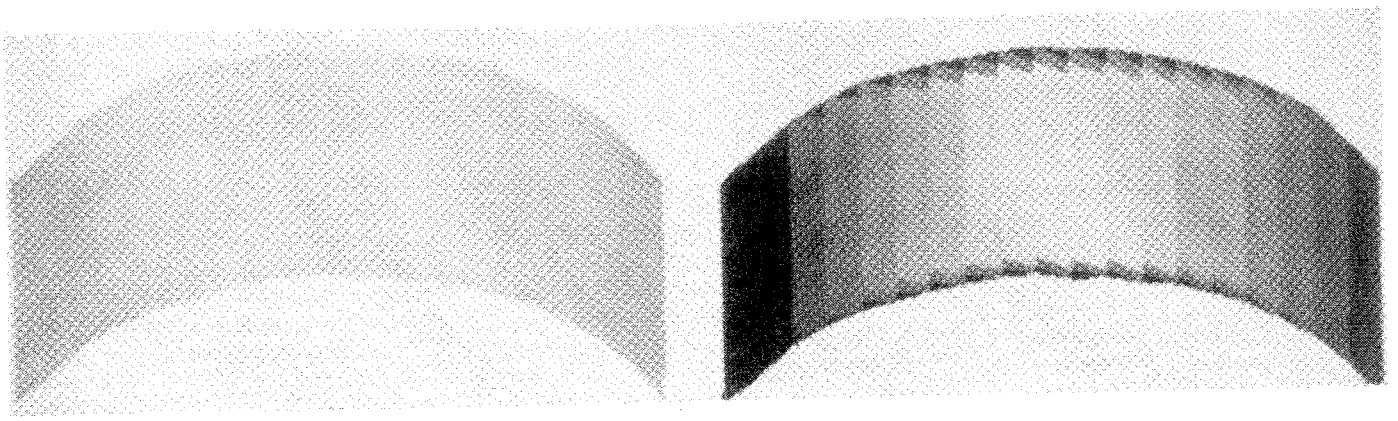

FIG. 20. A rendered representation of the surface fitted to the noisy data ($\sigma^2 = 0.01$) with taking into account the normals (left) and the value of the larger of the two principal curvatures (right).

• The set $\Sigma_{K_{ij}} = \{p(c_k), \partial p(c_k)/\partial s, \partial p(c_k)/\partial r, \partial^2 p(c_k)/\partial s \partial r, 1 \le k \le 4\}$, which allows us to define in a unique way the basis functions over each rectangles $K_{ij}$.

The subspace $V_h$ is then defined by

$$V_h = \{v \in \mathcal{C}^1(\Omega),\ v_{|K_{ij}} \in Q_3(K_{ij})\},$$

where $Q_k(I)$ is the vector space of the restrictions to an interval $I \subset R^2$ of the polynomials whose degree is less than $k$ for each variable, and $v_{ij}$ is the restriction of the function $v$ to the subset $I$. The basis functions of the finite element subspace $V_h$ are $\varphi_{ij}$, $\psi_{ij}$, $\eta_{ij}$, and $\zeta_{ij}$, and they are defined in a unique way over each rectangle $K_{ij}$ by

$$\begin{aligned}
&\varphi_{ij}(a_{kl}) = \delta_{ij,kl}, \quad \frac{\partial \varphi_{ij}}{\partial s}(a_{kl}) = \frac{\partial \varphi_{ij}}{\partial r}(a_{kl}) = \frac{\partial^2 \varphi_{ij}}{\partial s \partial r}(a_{kl}) = 0,\\
&\frac{\partial \psi_{ij}}{\partial s}(a_{kl}) = \delta_{ij,kl}, \quad \psi_{ij}(a_{kl}) = \frac{\partial \psi_{ij}}{\partial r}(a_{kl}) = \frac{\partial^2 \psi_{ij}}{\partial s \partial r}(a_{kl}) = 0,\\
&\frac{\partial \eta_{ij}}{\partial r}(a_{kl}) = \delta_{ij,kl}, \quad \eta_{ij}(a_{kl}) = \frac{\partial \eta_{ij}}{\partial s}(a_{kl}) = \frac{\partial^2 \eta_{ij}}{\partial s \partial r}(a_{kl}) = 0,\\
&\frac{\partial^2 \zeta_{ij}}{\partial s \partial r}(a_{kl}) = \delta_{ij,kl}, \quad \zeta_{ij}(a_{kl}) = \frac{\partial \zeta_{ij}}{\partial s}(a_{kl}) = \frac{\partial \zeta_{ij}}{\partial r}(a_{kl}) = 0,
\end{aligned} \tag{23}$$

where

$$\delta_{ij,kl} = \begin{cases} 1 & \text{if } i = k \text{ and } j = l \\ 0 & \text{otherwise.} \end{cases}$$

Thus $\forall v_h \in V_h$ we have the identity

$$v_h = \sum_{i,j=0}^{N_s-1,N_r-1} v_h(a_{ij})\varphi_{ij} + \frac{\partial v_h}{\partial s}(a_{ij})\psi_{ij} + \frac{\partial v_h}{\partial r}(a_{ij})\eta_{ij} + \frac{\partial^2 v_h}{\partial s \partial r}(a_{ij})\zeta_{ij},$$

providing a continuous representation of the solution over the space $\Omega$.

Equation (23) gives the expressions of the basis functions $\varphi_{ij}$, $\psi_{ij}$, $\eta_{ij}$, and $\zeta_{ij}$. This leads to analytical expressions which are too long to be reported here. Instead, we give a graphical representation of the four basis functions (Fig. 1).

### B.3. *Discrete Problem and Linear System*

Rewriting the discrete problem associated with Eq. (22) with the basis functions gives us the equations $\forall i, j = 0, \ldots, N_s - 1, N_r - 1$,

$$\begin{aligned}
a(v_h, \varphi_{ij}) &= L_v(\varphi_{ij})\\
a(v_h, \psi_{ij}) &= L_v(\psi_{ij})\\
a(v_h, \eta_{ij}) &= L_v(\eta_{ij})\\
a(v_h, \zeta_{ij}) &= L_v(\zeta_{ij})
\end{aligned} \tag{24}$$

and, using identity (10), $\forall i, j = 0, \ldots, N_s - 1, N_r - 1$,

$$\begin{aligned}
&\sum_{k,l=0}^{N_s-1,N_r-1} v_h(a_{kl})a(\varphi_{kl}, \varphi_{ij}) + \frac{\partial v_h}{\partial s}(a_{kl})a(\varphi_{kl}, \psi_{ij})\\
&\quad + \frac{\partial v_h}{\partial r}(a_{kl})a(\varphi_{kl}, \eta_{ij}) + \frac{\partial^2 v_h}{\partial s \partial r}(a_{kl})a(\varphi_{kl}, \zeta_{ij}) = L_v(\varphi_{ij})\\
&\sum_{k,l=0}^{N_s-1,N_r-1} v_h(a_{kl})a(\psi_{kl}, \varphi_{ij}) + \frac{\partial v_h}{\partial s}(a_{kl})a(\psi_{kl}, \psi_{ij})\\
&\quad + \frac{\partial v_h}{\partial r}(a_{kl})a(\psi_{kl}, \eta_{ij}) + \frac{\partial^2 v_h}{\partial s \partial r}(a_{kl})a(\psi_{kl}, \zeta_{ij}) = L_v(\psi_{ij})\\
&\sum_{k,l=0}^{N_s-1,N_r-1} v_h(a_{kl})a(\eta_{kl}, \varphi_{ij}) + \frac{\partial v_h}{\partial s}(a_{kl})a(\eta_{kl}, \psi_{ij})\\
&\quad + \frac{\partial v_h}{\partial r}(a_{kl})a(\eta_{kl}, \eta_{ij}) + \frac{\partial^2 v_h}{\partial s \partial r}(a_{kl})a(\eta_{kl}, \zeta_{ij}) = L_v(\eta_{ij})
\end{aligned}$$

$$\sum_{k,l=0}^{N_s-1,N_r-1} v_h(a_{kl})a(\zeta_{kl}, \varphi_{ij}) + \frac{\partial v_h}{\partial s}(a_{kl})a(\zeta_{kl}, \psi_{ij}) + \frac{\partial v_h}{\partial r}(a_{kl})a(\zeta_{kl}, \eta_{ij}) + \frac{\partial^2 v_h}{\partial s \partial r}(a_{kl})a(\zeta_{kl}, \zeta_{ij}) = L_v(\zeta_{ij}). \tag{25}$$

Equation (25) is a linear system where the unknowns are $v_h(a_{kl})$, $(\partial v_h/\partial r)(a_{kl})$, $(\partial v_h/\partial s)(a_{kl})$, and $(\partial^2 v_h/\partial s \partial r)(a_{kl})$.

Finally the solution of the discrete problem associated with (22) leads to the solution of the linear system $A \cdot V = L$, where $A = (\tilde{A}_{ij,kl})_{i,k=0,\ldots,Ns-1;j,l=0,\ldots,Nr-1}$ is a tri-diagonal bloc array.

$$\tilde{A}_{ij,kl} = \begin{pmatrix} a(\varphi_{ij}, \varphi_{kl}) & a(\varphi_{ij}, \psi_{kl}) & a(\varphi_{ij}, \eta_{kl}) & a(\varphi_{ij}, \zeta_{kl}) \\ a(\psi_{ij}, \varphi_{kl}) & a(\psi_{ij}, \psi_{kl}) & a(\psi_{ij}, \eta_{kl}) & a(\psi_{ij}, \zeta_{kl}) \\ a(\eta_{ij}, \varphi_{kl}) & a(\eta_{ij}, \psi_{kl}) & a(\eta_{ij}, \eta_{kl}) & a(\eta_{ij}, \zeta_{kl}) \\ a(\zeta_{ij}, \varphi_{kl}) & a(\zeta_{ij}, \psi_{kl}) & a(\zeta_{ij}, \eta_{kl}) & a(\zeta_{ij}, \zeta_{kl}) \end{pmatrix},$$

the $\tilde{A}_{ij,kl}$ array elements depending on the elasticity and rigidity coefficients.

## ACKNOWLEDGMENT

This work was partially supported by Digital Equipment Corp.

## REFERENCES


1. N. Ayache, J. D. Boissonnat, E. Brunet, L. Cohen, J. P. Chièze, B. Geiger, O. Monga, J. M. Rocchisani, and P. Sander, Building highly structured volume representations in 3-D medical images, in *Proceedings, Computer Aided Radiology, Berlin, West Germany 1989*.
2. N. Ayache, J. D. Boissonnat, L. Cohen, B. Geiger, J. Levy-Vehel, O. Monga, and P. Sander, Steps toward the automatic interpretations of 3-D images, in *3-D Imaging in Medicine* (H. Fuchs, K. Hohne and S. Pizer, Eds.), pp. 107–120, NATO ASI Series, Springer-Verlag, New York/Berlin, 1990.
3. N. Ayache, I. Cohen, and I. Herlin, Medical image tracking, in *Active Vision* (A. Blake and A. Yuille, Eds.), Chap. 20, MIT Press, Cambridge, MA, to appear.
4. K.-J. Bathe, *Finite Element Procedures in Engineering Analysis*, Prentice–Hall, Englewood Cliffs, NJ, 1982.
5. F. K. Bogner, R. L. Fox, and L. A. Schmit, The generation of interelement-compatible stiffness and mass matrices by the use of interpolation formulae, in *Proceedings, Conference on Matrix Methods in Structural Mechanics, Air Force Institute of Technology, Wright Patterson A.F. Base, OH, Oct. 1965*.
6. J. D. Boissonnat, Shape reconstruction from planar cross-sections, *Comput. Vision Graphics Image Process.* **44**, 1988, 1–29.
7. G. Borgefors, Distance transformations in arbitrary dimensions, *Comput. Vision Graphics Image Process.* **27**, 1984, 321–345.
8. M. Brady, J. Ponce, A. Yuille, and H. Asada, Describing surfaces, in *Proceedings, Second International Symposium on Robotics Research* (H. Hanafusa and H. Inoue, Eds.), pp. 5–16, MIT Press, Cambridge, MA, 1985.
9. J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intelligence*, **PAMI-8**(6), Nov. 1986, 679–698.
10. P. G. Ciarlet, *The Finite Element Methods for Elliptic Problems*, North-Holland, Amsterdam, 1987.
11. I. Cohen, *Modèles deformables 2-D et 3-D: Application à la segmentation d'images médicales*, PhD thesis, Université Paris-IX Dauphine, June 1992.
12. I. Cohen, N. Ayache, and P. Sulger, Tracking points on deformables objects, in *Proceedings, Second European Conference on Computer Vision, Santa Margherita Ligure, Italy, May 1992*.
13. L. D. Cohen, On active contour models and balloons, *CVGIP Image Understanding* **53**(2), Mar. 1991, 211–218.
14. L. D. Cohen and I. Cohen, A finite element method applied to new active contour models and 3-D reconstruction from cross sections, in *Proceedings, Third International Conference on Computer Vision, IEEE Computer Society Conference, Osaka, Japan, Dec. 1990*, pp. 587–591.
15. L. D. Cohen and I. Cohen, *Finite Element Methods for Active Contour Models and Balloons from 2-D to 3-D*, Tech. Rep. 9124, CEREMADE, U.R.A. CNRS 749, Université Paris IX—Dauphine, Cahiers de Mathematiques de la Decision, Nov. 1991.
16. P. E. Danielsson, Euclidean distance mapping, *Comput. Vision Graphics Image Process.* **14**, 1980, 227–248.
17. M. P. do Carmo, *Differential Geometry of Curves and Surfaces*, Prentice–Hall, Englewood Cliffs, NJ, 1976.
18. P. Fua and Y. G. Leclerc, Model driven edge detection, in *Proceedings, DARPA Image Understanding Workshop, 1988*.
19. G. G. Gordon, Face recognition from depth and curvature, in *Proceedings, SPIE Conference on Geometric Methods in Computer Vision, San Diego, CA, July 1991*.
20. A. Guéziec and N. Ayache, Smoothing and matching of 3D-space curves, in *Proceedings, Second European Conference on Computer Vision (ECCV), Santa Margherita Ligure, Italy, May 1992*.
21. I. L. Herlin and N. Ayache, Features extraction and analysis methods for sequences of ultrasound images, in *Proceedings, Second European Conference on Computer Vision (ECCV), Santa Margherita Ligure, Italy, May 1992*.
22. M. Kass, A. Witkin, and D. Terzopoulos, Snakes: Active contour models, in *Proceedings, First International Conference on Computer Vision, London, June 1987*, pp. 259–268.
23. S. Lee, *Visual Monitoring of Glaucoma*, PhD thesis, Robotics Research Group, Department of Engineering Science, University of Oxford, 1991.
24. F. Leitner and P. Cinquin, Dynamic segmentation: Detecting complex topology 3D-object, in *Proceedings, International Conference of the IEEE Engineering in Medicine and Biology Society*, Orlando, FL, Nov. 1991, pp. 295–296.
25. F. Leitner, I. Marque, S. Lavallée, and P. Cinquin, Dynamic segmentation: Finding the edge with snake-splines, in *Proceedings, International Conference on Curves and Surfaces, Chamonix, France, June 1990*, pp. 1–4, Academic Press, San Diego, CA, 1990.
26. O. Monga, N. Ayache, and P. Sander, From voxel to curvature, in *Proceedings, Computer Vision and Pattern Recognition, IEEE Computer Society Conference, Lahaina, Maui, HI, June 1991*, pp. 644–649.
27. O. Monga and R. Deriche, 3-D edge detection using recursive filtering, in *Proceedings, IEEE Conference on Vision and Pattern Recognition, San Diego, CA, June 1989*.
28. J. A. Noble, Finding corners, in *Proceedings, Alvey Vision Conference, June 1987*, pp. 267–274.

29. A. Pentland and S. Sclaroff, Closed-form solutions for physically based shape modelling and recognition, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-13**(7), July 1991, 715–729.

30. J. Ponce and M. Brady, Toward a surface primal sketch, in *Proceedings, IJCAI, 1985.*

31. P. T. Sander and S. W. Zucker, Inferring surface trace and differential structure from 3-D images, *IEEE Trans. Pattern Anal. Mach. Intelligence*, in press; a shortened version is available as Charting surface structure, in *Proceedings, First European Conference on Computer Vision, Antibes, Apr. 23–27.*

32. D. Terzopoulos, The computation of visible-surface representations, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-10**(4), July 1988, 417–438.

33. D. Terzopoulos, A. Witkin, and M. Kass, Constraints on deformable models: Recovering 3-D shape and nonrigid motion, *AI J.* **36,** 1988, 91–123.

34. D. Terzopoulos, A. Witkin, and M. Kass, Symmetry-seeking models for 3-D object reconstruction, in *Proceedings, First International Conference on Computer Vision, June 1987,* pp. 269–276.

35. O. C. Zienkiewicz, *The Finite Element Method*, McGraw–Hill, 3rd ed., New York, 1977.

36. S. W. Zucker and R. M. Hummel, A three-dimensional edge operator, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-3**(3), May 1981, 324–331.