Geometry and Shape I

Evolutionary Fronts for Topology-Independent Shape Modeling and Recovery *

R. Malladi,[1] J. A. Sethian,[1] and B. C. Vemuri[2]

[1] University of California, Berkeley, CA 94720, USA
[2] University of Florida, Gainesville, FL 32611, USA

Abstract. This paper presents a novel framework for shape modeling and shape recovery based on ideas developed by Osher & Sethian for interface motion. In this framework, shapes are represented by propagating fronts, whose motion is governed by a "Hamilton-Jacobi" type equation. This equation is written for a function in which the interface is a particular level set. Unknown shapes are modeled by making the front adhere to the object boundary of interest under the influence of a synthesized halting criterion. The resulting equation of motion is solved using a narrow-band algorithm designed for rapid front advancement. Our techniques can be applied to model arbitrarily complex shapes, which include shapes with significant protrusions, and to situations where no *a priori* assumption about the object's topology can be made. We demonstrate the scheme via examples of shape recovery in 2D and 3D from synthetic and low contrast medical image data.

1 Introduction

An important goal of computational vision is to recover the shapes of objects in 2D and 3D from various types of visual data. One way to achieve this goal is via model-based techniques. Broadly speaking, these techniques, as the name suggests, involve the use of a model whose boundary representation is matched to the image to recover the object of interest. In this paper, we present a new approach to dynamic shape modeling. In the present context the word shape model stands for a boundary (surface) representation of an object shape.

Shape reconstruction typically precedes the symbolic representation of surfaces. Shape models are expected to aid the recovery of detailed structure from noisy data using only the weakest of the possible assumptions about the observed shape. To this end, several variational shape reconstruction methods have been proposed and there is abundant literature on the same [2, 13, 17, 3, 21, 9]. Generalized spline models with continuity constraints are well suited for fulfilling the goals of shape recovery (see [3, 4, 17]). Following the advent of the dynamic shape modeling paradigm [18], considerable research activity followed, with numerous

* [1] Supported in part by the Applied Mathematical Sciences Subprogram of the Office of Energy Research, U.S. Dept. of Energy under Contract DE-AC03-76SD00098 and by the NSF ARPA under grant DMS-8919074.
[2] Supported in part by NSF grant ECS-9210648

Lecture Notes in Computer Science, Vol. 800
Jan-Olof Eklundh (Ed.)
Computer Vision – ECCV '94

application specific modifications to the modeling primitives, and external forces derived from data constraints [8, 5, 19, 20, 6, 22].

In this paper, we further these techniques by developing a scheme that can accurately model significant protrusions in complex shapes, and leads to numerical algorithms whose convergence to the desired shape is relatively independent of the shape initialization. The framework of energy minimization ("snakes") has been used successfully in the past for extracting salient image contours such as edges and lines by Kass *et al.* [8]. Despite a good initialization, the active contour model, due to its arc-length and curvature minimization properties, cannot be forced to extrude through any significant protrusions that a shape may possess. We propose a dynamic shape modeling method that can start with a single instance of the model and sprout branches during the evolutionary process.

Most existing surface modeling techniques require that the topology of the object be known before the shape recovery can commence. However, it is not always possible to specify the topology of an object prior to its recovery. In the context of static problems, particle systems have been used to model surfaces of arbitrary topology [16]. The scheme described in this paper can be applied to static as well as dynamic situations, where no *a priori* assumption about the object's topology is made. A single instance of our model, when presented with an image having more than one shape of interest (see figure 2(l)), has the ability to split freely to represent each shape [10, 11]. We show that by using our approach, it is also possible to extract the bounding contours of *shapes with holes* in a seamless fashion (see figure 2(l)–(p)).

Our method is inspired by ideas first introduced in Osher & Sethian [12, 15], which grew out of work in Sethian [14], to model propagating fronts with curvature-dependent speeds. Two such examples are flame propagation and crystal growth, in which the speed of the moving interface normal to itself depends on transport terms modified by the local curvature. The challenge in these problems is to devise numerical schemes for the equations of the propagating front which will accurately approximate these highly unstable physical phenomena. Osher & Sethian [12, 15] achieve this by viewing the propagating surface as a specific level set of a higher-dimensional function. The equation of motion for this function is reminiscent of an initial value "Hamilton-Jacobi" equation with a parabolic right-hand side and is closely related to a viscous hyperbolic conservation law.

In our work, we adopt these level set techniques to the problem of shape recovery. To isolate a shape from its background, we first consider a closed, non-intersecting, initial hypersurface placed inside (or outside) it. This hypersurface is then made to *flow* along its gradient field with a speed $F(K)$, where K is the curvature of the hypersurface. Unknown shapes are recovered by making the front adhere to the object boundaries. This is done by synthesizing a speed term from image data which acts as a halting criterion.

The remainder of this paper is organized as follows: section 2 introduces the level set formulation for curvature-dependent front propagation. In section 3 we explain our level set algorithm for shape recovery and in section 4 we outline the details of our fast numerical algorithm. In section 5 we present some 2D and

3D shape recovery results of applying our method to synthetic and low contrast medical images. We conclude in section 6.

2 Level set formulation for front propagation

In this section we present the level set technique due to Osher & Sethian [12]. For details and an expository review, see Sethian [15]. As a starting point, consider a closed curve moving in the plane, i.e., let $\gamma(0)$ be a smooth, closed initial curve in Euclidean plane R^2, and let $\gamma(t)$ be the one-parameter family of curves generated by moving $\gamma(0)$ along its normal vector field with speed $F(K)$, a given scalar function of the curvature K. The central idea in the level set approach is to represent the front $\gamma(t)$ as the level set $\{\psi = 0\}$ of a function ψ. The equation of motion is written for the function ψ in such a way that at any time instant t, the level set $\{\psi = 0\}$ yields the moving front. Topological changes in the front can be handled naturally in this approach by exploiting the property that the level set $\{\psi = 0\}$ need not be simply connected.

Specifically, let $\gamma(0)$ be a closed, nonintersecting, $(N-1)$ dimensional hypersurface. Let $\psi(\mathbf{x}, t)$, $\mathbf{x} \in R^N$, be the scalar function such that $\psi(\mathbf{x}, 0) = \pm d(\mathbf{x})$, where $d(\mathbf{x})$ is the signed distance from $\mathbf{x}$ to the hypersurface $\gamma(0)$. We use the plus sign if $\mathbf{x}$ is outside $\gamma(0)$ and minus sign if $\mathbf{x}$ is inside. Each level set of ψ flows along its gradient field with speed $F(K) = 1 - \varepsilon K$, and its equation of motion is given by

$$\psi_t + \mid \nabla\psi \mid = \varepsilon K \mid \nabla\psi \mid, \tag{1}$$

with an initial condition $\psi(\mathbf{x}, 0) = \pm d(\mathbf{x})$. We refer to equation (1) as the level set "Hamilton-Jacobi" formulation. This equation can be solved using the stable, entropy-satisfying finite difference schemes, borrowed from the literature on hyperbolic conservation laws (see [12]).

3 Shape recovery with front propagation

In this section, we describe how the level set formulation for the front propagation problem discussed in the previous section can be used for shape recovery. In the context of shape recovery, the front represents the boundary of an evolving shape. Since the idea is to extract object shapes from a given image, the front should be forced to stop in the vicinity of the desired object boundaries under the influence of an image-based halting criterion. We define the final shape to be the configuration when all the points on the front come to a stop, thereby bringing the computation to an end.

Our goal now is to define a speed function from the image data that can be applied on the propagating front as a stopping criterion. In general, the function F can be split into two components: $F = F_A + F_G$. The term F_A, referred to as the advection term, is independent of the moving front's geometry. Depending on its sign the front uniformly expands or contracts with speed F_A. The second

term F_G, is the part which depends on the geometry of the front such as its local curvature. This (diffusion) term smooths out the high curvature regions of the front and has the same regularizing effect on the front as the internal deformation energy term in thin-plate-membrane splines [17, 8].

Let the front move with a speed $F = F_A + F_G$. Define an image-based speed term k_I to be

$$k_I(x,y) = \frac{1}{1+ \mid \nabla G_\sigma * I(x,y) \mid}, \tag{2}$$

where $G_\sigma * I$ denotes the image convolved with a Gaussian smoothing filter whose characteristic width is σ. This term has values that are closer to zero in regions of high image gradient and values that are closer to unity in regions with relatively constant intensity. More sophisticated stopping criteria can be synthesized by using the orientation dependent "steerable" filters [7].

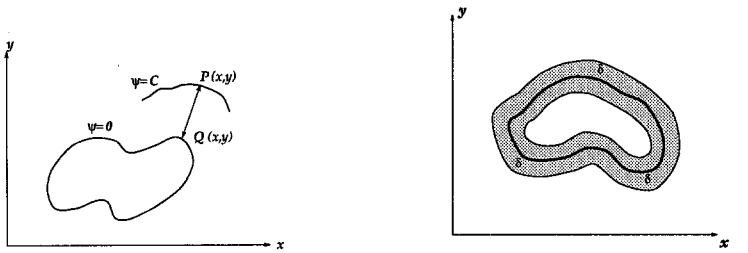
The image-based speed term k_I has meaning only on the boundary $\gamma(t)$, i.e. on the level set $\{\psi = 0\}$. This follows from the fact that it is designed to force the propagating level set $\{\psi = 0\}$ to a complete stop in the neighborhood of the object boundary. However, the level set equation of motion is written for the function ψ defined over the entire domain. Consequently, we require that the evolution equation has a consistent physical meaning for all the level sets, i.e., at every point $(x,y) \in \Omega$. The speed term derives its meaning not from the geometry of ψ but from the configuration of the level set $\{\psi = 0\}$ in the image plane. Thus, our goal is to construct an image-based speed function $\hat{k}_I$ that is globally defined. We call it an *extension* of k_I off the level set $\{\psi = 0\}$ because it extends the meaning of k_I to other level sets. Note that the level set $\{\psi = 0\}$ lies in the image plane and therefore $\hat{k}_I$ must equal k_I on $\{\psi = 0\}$. With this extension $\hat{k}_I$, the equation of motion is given by

$$\psi_t + \hat{k}_I(F_A + F_G) \mid \nabla\psi \mid = 0. \tag{3}$$

If the level curves are moving with a constant speed, i.e. $F_G = 0$, then at any time t, a typical level set $\{\psi = C\}$, $C \in R$, is a distance C away from the level set $\{\psi = 0\}$. Observe that the above statement is a rephrased version of *Huygen's principle* which, from a geometrical standpoint, stipulates that the position of a front propagating with unit speed at a given time t should consist of only the set of points located a distance t away from the initial front (see [14]). On the other hand, if $F_G \neq 0$, the level sets will violate the property that they are a constant distance away from each other. With the above remarks in mind we stipulate the following:

Property. Any external (image-based) speed function that is used in the equation of motion written for the function ψ should not cause the level sets to collide and cross each other during the evolutionary process.

We construct one such extension to the image-based speed function as follows: (see figure 1(a)).



(a) Extension (b) Narrow band

Fig. 1. (a) Extension of image-based speed terms off the level set $\{\psi = 0\}$. (b) A narrow band of width δ around the zero set.

Construction. *The value of $\hat{k}_I$ at a point P lying on a level set $\{\psi = C\}$ is exactly the value of k_I at a point Q, such that point Q is closest to P and lies on the level set $\{\psi = 0\}$.*

First, note that $\hat{k}_I$ reduces to k_I on $\{\psi = 0\}$. Note also that if the definition of a speed function adheres to the above construction, then the level sets will stay a constant distance away from each other. Thus, having defined the intent of the equation (3) in the context of shape modeling, we solve it using finite difference numerical schemes given below.

4 Numerical solution

The equation (3) poses an initial valued problem. As shown in Osher & Sethian [12], numerical techniques for computing hyperbolic conservation laws may be exploited to pick out the correct entropy satisfying weak solution. We use the following upwind scheme to advance the position of the front given in equation (3), namely

$$\psi_{i,j}^{n+1} = \psi_{i,j}^{n} - \Delta t F_A(\hat{k}_I)_{i,j}\{(\max(D_x^- \psi_{i,j}, 0))^2 + (\min(D_x^+ \psi_{i,j}, 0))^2 + (\max(D_y^- \psi_{i,j}, 0))^2 + (\min(D_y^+ \psi_{i,j}, 0))^2\}^{1/2} - \Delta t F_G \hat{k}_I \mid \nabla\psi \mid, \quad (4)$$

where D^+ and D^- are standard forward and backward difference operators. Here, we have not approximated the second term $F_G \hat{k}_I \mid \nabla\psi \mid$; one may use a straightforward central difference approximation to this term.

4.1 Narrow-band update mechanism

The computation of *extension* for image-based speed term is very expensive. This is because at each grid point, we must search for the closest point lying on

the level set $\{\psi = 0\}$. Moreover, if $F_G = 0$, then the stability requirement for the explicit method for solving our level set equation is $\Delta t = O(\Delta x)$. For the full equation (4), the stability requirement is $\Delta t = O(\Delta x^2)$. This could potentially force a very small time step for fine grids. These two effects, individually and compounded, make the computation exceedingly slow.

Instead, we observe that the front can be moved by updating the level set function at a small set of points in the neighborhood of the zero set instead of updating it at all the points on the grid. In figure (1(b)) the bold curve depicts the level set $\{\psi = 0\}$ and the shaded region around it is the narrow-band. The narrow-band is bounded on either side by two curves which are a distance δ apart, i.e., the two curves are the level sets $\{\psi = \pm\delta/2\}$. The value of δ determines the number of grid points that fall within the narrow-band. A complete discussion of the narrow-band techniques for interface propagation may be found in [1].

As a consequence of our update strategy, the front can be moved through a maximum distance of $\delta/2$, either inward or outward, at which point we must rebuild an appropriate (a new) narrow band. We reinitialize the ψ function by treating the current zero set configuration, i.e., $\{\psi = 0\}$, as the initial curve $\gamma(0)$. Note that the reinitialization procedure must account for the case when $\{\psi = 0\}$ changes topology. This procedure will restore the meaning of ψ function by correcting the inaccuracies introduced as a result of our update algorithm. Once a new ψ function is defined on the grid, we can create a new narrow-band around the zero set, and go through another set of, say l, iterations in time to move the front ahead by a distance equal to $\delta/2$. The value of l is set to the number of time steps required to move the front by a distance roughly equal to $\delta/2$. Thus, a faster algorithm for shape recovery consists of the following steps:

Algorithm

1. Set the iteration number $m = 0$ and go to step 2.
2. At each grid point (i, j) lying inside the narrow-band, compute the *extension* $\hat{k}_I$ of image-based speed term.
3. With the above value of extended speed term $(\hat{k}_I^m)_{i,j}$ and $\psi_{i,j}^m$, calculate $\psi_{i,j}^{m+1}$ using the upwind, finite difference scheme given in equation (4).
4. Construct a polygonal approximation for the level set $\{\psi = 0\}$ from $\psi_{i,j}^{m+1}$. A polygonal approximation is required in step 2 for the evaluation of image-based speed term and more importantly, in step 6 for reinitializing the ψ function.
5. Increment m by one. If the value of m equals l, go to step 6, else, go to step 2.
6. Compute the value of signed distance function ψ by treating the polygonal approximation of $\{\psi = 0\}$ as the initial contour $\gamma(0)$. Go to step 1.

In the narrow-band approach, since we only update ψ at points lying in the narrow-band, the issue of specifying boundary conditions for points lying on the edge of the band becomes pertinent. With our relatively simple speed motion, the free-end boundary condition is adequate, however, in more complex appli-

cations such as crystal growth, and flame propagation, accurate specification of boundary conditions becomes necessary [1].

5 Shape recovery results

In this section we present several shape recovery results that were obtained by applying the narrow-band level set algorithm to image data. Given an image, our method requires the user to provide an initial contour $\gamma(0)$. The initial contour can be placed anywhere in the image plane. However, it must be placed inside a desired shape or enclose all the constituent shapes. Our front seeks the object boundaries by propagating inward or outward in the normal direction. This choice is made at the time of initialization. The level set function ψ is discretized on the image plane and its initial value, i.e., $\psi(\mathbf{x}, 0)$ is computed from $\gamma(0)$. Once the value of $\psi_{i,j}$ is computed at time $t = 0$, we use the update equation (4) to move the front.

We now present our shape recovery results in 2D. First, we consider a CT (computed tomography) image of an abdominal section shown in figure 2(a), with the goal of recovering the shape of the stomach in this particular slice. The function ψ has been discretized on a 128×128 mesh. In figure 2(a), we show the closed contour that the user places inside the desired shape at time $t = 0$. The front is then made to propagate in the normal direction with speed $F = \hat{k}_I(-1.0 - 0.025K)$. We employed the narrow-band update algorithm to move the front with a time step size $\Delta t = 0.0005$ and the ψ function was recomputed after every 50 time steps. In figures 2(b) and 2(c) we depict the configuration of the front at two intermediate time instants. The final result is achieved after 575 time iterations and is shown in figure 2(d). We emphasize that our method does not require that the initial contour be placed close to the object boundary.

In our second experiment we recover the structure of an arterial tree. The real image has been obtained by clipping a portion of a digital subtraction angiogram. In this experiment we compare the performance of our scheme with the active contour model with inflation forces [5]. Despite a good initialization in figure 2(e), the ballooning snake model barely recovers the main stem of the artery and completely fails to account for the branches (see 2(g)). We now apply our level set algorithm to recover the same shape. After initialization in figure 2(h), the front is made to propagate in the normal direction. We employ the narrow-band updation scheme with a band width of $\delta = 0.075$ to move the front. It can be seen that in subsequent frames the front evolves into the branches and finally in 2(k) completely reconstructs the complex tree structure. Thus, a single shape model sprouts branches to recover all the connected components of a given shape. Calculations were carried out on a 64×64 grid with a time step $\Delta t = 0.001$.

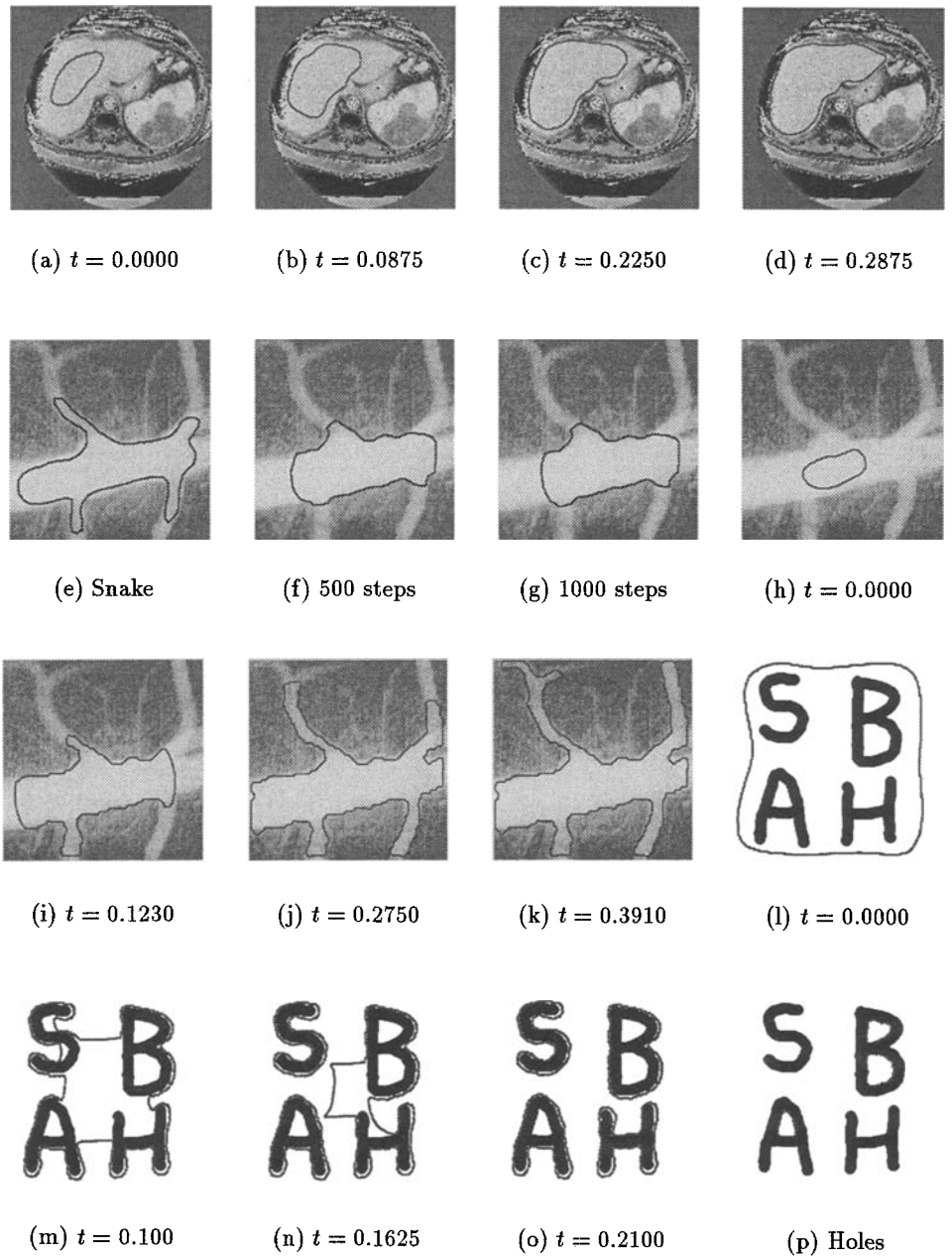
In the next experiment, we use this approach to recover shapes with holes. We also show that the front can undergo a topological transformation to recover the constituent shapes in a given image. We do this by applying our method to extract the shapes of hand-printed characters. The outer and inner boundaries

of a given shape are recovered without separate initializations. In figure 2(l), we show the initial contour which encloses all the characters. This contour is then made to propagate inward with a constant speed. Figures 2(m) shows an intermediate stage in the front's evolution and in 2(n), the front splits into three separate contours. The calculation comes to a halt when in figure 2(o), the level set $\{\psi = 0\}$ recovers the outer boundaries of four separate characters. Unlike the characters "A" & "B" for which we need to extract the inner boundary for their complete shape description, for the characters "S" & "H", the outer boundary completely describes their shape. In the second stage of our computation, we treat the zero set configuration in figure 2(o) as an initial state, and propagate all the fronts inward by *momentarily relaxing the image-based speed term.* This causes the zero set to move into the character shapes and recover the holes in 2(p), thereby achieving a complete shape recovery. The calculations for this experiment were done on a 128×128 grid and the time step Δt was set to 0.00025.

Lastly, we recover the shape of a "flat" superquadric using the level set front propagation scheme in 3D. Volume data for this experiment consists of 32 slices each with a particular cross section of the superquadric. The image-based speed term k_I is computed from these images according to an equation in 3D which is analogous to equation (2). A sphere, which is the level surface $\{\psi = 0\}$ of the function $\psi(x, y, z) = x^2 + y^2 + z^2 - 0.01$, forms our initialization (see figure 3(a)). This initial surface is moved with speed $F = \hat{k}_I$ by updating the value of ψ on a discrete 3D grid. The initial surface expands smoothly in all directions until a portion of it collides with the superquadric boundary. At points with high gradient, the $\hat{k}_I$ values are close to zero and cause the zero set to locally come to a stop near the boundary of the superquadric shape. This situation is depicted in figures 3(b) and 3(c). Finally, in figure 3(d), the initial spherical shape completely transforms into a flat superquadric shape.

6 Concluding remarks

In this paper we presented a new shape modeling scheme. Our approach retains some of the desirable features of existing methods for shape modeling and overcomes some of their deficiencies. We adopt the level set techniques first introduced in Osher & Sethian [12] to the problem of shape recovery. With this approach, complex shapes can be recovered from images. The final result in our method is relatively independent of the initial guess. This is a very desirable feature to have, specially in applications such as automatic shape recovery from image data. Moreover, our scheme makes no *a priori* assumption about the object's topology. Other salient features of our shape modeling scheme include its ability to split and merge freely without any additional bookkeeping during the evolutionary process, and its easy extensibility to higher dimensions. We believe that this shape modeling algorithm will have numerous applications in the areas of computer vision and computer graphics.



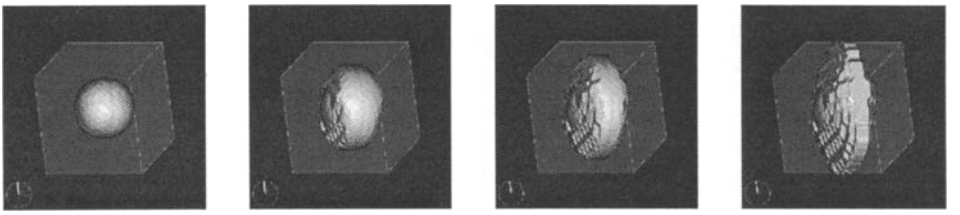
(a) $t = 0.0000$ (b) $t = 0.0875$ (c) $t = 0.2250$ (d) $t = 0.2875$

(e) Snake (f) 500 steps (g) 1000 steps (h) $t = 0.0000$

(i) $t = 0.1230$ (j) $t = 0.2750$ (k) $t = 0.3910$ (l) $t = 0.0000$

(m) $t = 0.100$ (n) $t = 0.1625$ (o) $t = 0.2100$ (p) Holes

Fig. 2. (a)–(d): Recovery of the stomach shape from a CT image of an abdominal section. (e)–(g): An unsuccessful attempt to recover the arterial structure using an active contour model. (h)–(k): Reconstruction of a shape with "significant" protrusions using level set front propagation algorithm. (l)–(p): Topological split and shapes with holes: a two-stage scheme is used to arrive at a complete shape description of both simple shapes and shapes with holes.



(a) $t = 0.0000$ (b) $t = 0.1000$ (c) $t = 0.1750$ (d) $t = 0.3000$

Fig. 3. Shape recovery in 3D: a flat superquadric shape. Calculations were done on a $32 \times 32 \times 32$ grid with a time step $\Delta t = 0.0025$.

References

1. D. Adalsteinsson and J. A. Sethian, "A fast level set method for propagating interfaces," submitted for publication, *Journal of Computational Physics*, 1994.
2. R. M. Bolle and B. C. Vemuri, "On three-dimensional surface reconstruction methods," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI 13, No. 1, pp. 1–13, 1991.
3. T.E. Boult and J.R. Kender, "Visual surface reconstruction using sparse depth data," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 1986, pp. 68–76.
4. A. Blake and A. Zisserman, *Visual Reconstruction*, MIT Press, Cambridge, MA.
5. L. D. Cohen, "On Active Contour Models and Balloons," *Computer Vision, Graphics, and Image Processing*, Vol. 53, No. 2, pp. 211–218, March 1991.
6. H. Delingette, M. Hebert, and K. Ikeuchi, "Shape representation and image segmentation using deformable models," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 467–472, Maui Hawaii, June 1991.
7. W. T. Freeman and E. H. Adelson, "Steerable filters for early vision, image analysis, and wavelet decomposition," in *Proceedings of ICCV*, pp. 406–415, Osaka, Japan, 1990.
8. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *International Journal of Computer Vision*, pp. 321–331, 1988.
9. D. Lee and T. Pavlidis, "One-dimensional regularization with discontinuities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI 10, pp. 822–829, 1986.
10. R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," to appear in *IEEE Trans. on Pattern Analysis and Machine Intelligence.*
11. R. Malladi, J. A. Sethian, and B. C. Vemuri, "A fast level set based algorithm for topology-independent shape modeling," to appear in the *Journal of Mathematical Imaging & Vision*, special issue on Topology & Geometry in Computer Vision.
12. S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation," *Journal of Computational Physics*, Vol. 79, pp. 12-49, 1988.

13. L.L. Schumaker, "Fitting Surfaces to Scattered data," in *Approximation Theory II*, G.G. Lorentz, C.K. Chui, and L.L. Schumaker, (eds.). New York: Academic Press, 1976, pp. 203–267.
14. J. A. Sethian, "Curvature and the evolution of fronts," *Commun. in Mathematical Physics*, Vol. 101, pp. 487–499, 1985.
15. J. A. Sethian, "Numerical algorithms for propagating interfaces: Hamilton-Jacobi equations and conservation laws," *Journal of Differential Geometry*, Vol. 31, pp. 131–161, 1990.
16. R. Szeliski and D. Tonnesen, "Surface modeling with oriented particle systems," *Computer Graphics SIGGRAPH*, Vol. 26, No. 2, pp. 185–194, July 1992.
17. D. Terzopoulos, "Regularization of inverse visual problems involving discontinuities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI 8, No. 2, pp. 413–424, 1986.
18. D. Terzopoulos, A. Witkin, and M. Kass, "Constraints on deformable models: Recovering 3D shape and nonrigid motion," *Artificial Intelligence*, 36, pp. 91–123, 1988.
19. B. C. Vemuri and R. Malladi, "Intrinsic Parameters for Surface Representation using Deformable Models," in *IEEE Trans. on Systems, Man & Cybernetics*, Vol. 23, No. 2, pp. 614–623, March/April 1993.
20. B. C. Vemuri and R. Malladi, "Constructing intrinsic parameters with active models for invariant surface reconstruction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 7, pp. 668–681, July 1993.
21. B. C. Vemuri, A. Mitiche, and J. K. Aggarwal, "Curvature-based representation of objects from range data," *Int. Journal of Image and Vision Computing*, 4, pp. 107–114, 1986.
22. Y. F. Wang and J. F. Wang, "Surface reconstruction using deformable models with interior and boundary constraints," in *Proceedings of ICCV*, pp. 300–303, Osaka, Japan, 1990.