# Level Set Techniques for Tracking Interfaces; Fast Algorithms, Multiple Regions, Grid Generation, and Shape/Character Recognition

### J.A. Sethian

Abstract. We describe new applications of the level set approach for following the evolution of complex interfaces. This approach is based on solving an initial value partial differential equation for a propagating level set function, using techniques borrowed from hyperbolic conservation laws. Topological changes, corner and cusp development, and accurate determination of geometric properties such as curvature and normal direction are naturally obtained in this setting. In this paper, we review some recent work, including fast level set methods, extensions to multiple fluid interfaces, generation of complex interior and exterior body-fitted grids, and applications to problems in shape and character recognition.

#### 1.Introduction

In this paper, we review some recent work which extends the capabilities of level set methods for tracking the evolution of complex interfaces, and summarize some new recent applications of these techniques. Level set methods, introduced by Osher and Sethian [23], offer highly robust and accurate methods for tracking interfaces moving under complex motions. Their major virtue is that they naturally construct the fundamental weak solution to surface propagation posed by Sethian [25, 26]. They work in any number of space dimensions, handle topological merging and breaking naturally, and are easy to program. They approximate the equations of motion for the underlying propagating surface, which resemble Hamilton-Jacobi equations with parabolic right-hand sides. The central mathematical idea is to view the moving front as a particular level set of a higher dimensional function. In this setting, sharp gradients and cusps can form easily, and the effects of curvature may be easily incorporated. The key numerical idea is to borrow the technology from the numerical solution of hyperbolic conservation laws and transfer these ideas to the Hamilton-Jacobi setting, which then guarantees that the correct entropy satisfying solution will be obtained. As initially designed in [23], the level set technique is designed to track an interface where there is a clear distinction between an "inside" and "outside". This is because the interface is assigned the zero level value between the two regions. Additionally, calculations in the original technique were performed over all the level sets, not just the one corresponding to the zero interface. Consquently, to track a one-dimensional curve moving in two space, calculations were performed over the entire two-dimensional domain, leading to an  $O(N^2)$ calculation where N is the number of grid points in each direction. In many cases this is an unnecessary expense in both memory and computational labor, since only the motion of the interface itself is of interest. In this paper, we describe a fast level set approach introduced in [1] which limits computation to a narrow band around the interface, significantly reducing the computational labor. We also briefly describe a new extension of the level set approach given in [32] which can be used to track an arbitrary number of interfaces in some cases.

We then summarize the application of level set techniques to two new areas. First, we show how interface tracking can be used to generate logically rectangular body-fitted grids around complex bodies in both two and three space dimensions. We generate internal and external grids around a variety of objects, with the ability to body-fit deep into oscillatory bodies. Second, we show how these same techniques can be applied to shape detection and optical character recognition; providing algorithms that detect and fit shapes in MRI and CAT scans, and exploiting accurate and efficient feature vectors for shape recognition.

# **II.** Numerical Algorithms for Propagating Fronts

The fundamental aspects of front propagation in our context can be illustrated as follows. Let  $\gamma(0)$  be a smooth, closed initial curve in  $\mathbb{R}^2$ , 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). Here, F(K)is a given scalar function of the curvature K. Thus,  $n \cdot x_t = F(K)$ , where x is the position vector of the curve, t is time, and n is the unit normal to the curve. It can be shown that a curve collapsing under its curvature shrinks to a circle, see [15, 16, 18].

Consider a speed function of the form  $1 - \epsilon K$ , where  $\epsilon$  is a constant. An evolution equation for the curvature K, see [26], is given by

$$K_t = \epsilon K_{\alpha\alpha} + \epsilon K^3 - K^2 \tag{1}$$

where we have taken the second derivative of the curvature K with respect to arclength  $\alpha$ . This is a reaction-diffusion equation; the drive toward singularities due to the reaction term  $(\epsilon K^3 - K^2)$  is balanced by the smoothing effect of the diffusion term  $(\epsilon K_{\alpha\alpha})$ . Indeed, with  $\epsilon = 0$ , we have a pure reaction equation  $K_t = -K^2$ . In this case, the solution is K(s,t) = K(s,0)/(1 + tK(s,0)), which is singular in finite t if the initial curvature is anywhere negative. Thus, corners can form in the moving curve when  $\epsilon = 0$ .

As an example, consider the periodic initial cosine curve

$$\gamma(0) = (-s, [1 + \cos 2\pi s]/2) \tag{2}$$

propagating with speed  $F(K) = 1 - \epsilon K$ ,  $\epsilon > 0$ . As the front moves, the troughs at s = n + 1/2,  $n = 0, \pm 1, \pm 2, \dots$  are sharpened by the negative reaction term (because K < 0 at

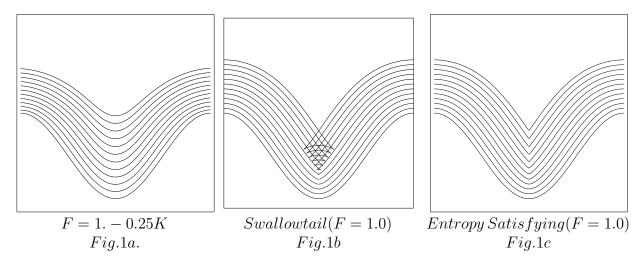


Figure 1: Propagating Cosine Curve.

such points) and smoothed by the positive diffusion term (see Figure 1a). For  $\epsilon > 0$ , it can be shown (see [26, 23]) that the moving front stays  $C^{\infty}$ .

On the other hand, for  $\epsilon = 0$ , the front develops a sharp corner in finite time as discussed above. In general, it is not clear how to construct the normal at the corner and continue the evolution, since the derivative is not defined there. One possibility is the "swallowtail" solution formed by letting the front pass through itself (see Figure 1b). However, from a geometrical argument it seems clear that the front at time t should consist of only the set of all points located a distance t from the initial curve. (This is known as the Huygens principle construction, see [26]). Roughly speaking, we want to remove the "tail" from the "swallowtail". In Figure 1c, we show this alternate weak solution. Another way to characterize this weak solution is through the following "entropy condition" posed by Sethian (see [26]): If the front is viewed as a burning flame, then once a particle is burnt it stays burnt. Careful adherence to this stipulation produces the Huygens principle construction. Furthermore, this physically reasonable weak solution is the formal limit of the smooth solutions  $\epsilon > 0$  as the curvature term vanishes, (see [26]).

As further illustration, we consider the case of a V-shaped front propagating normal to itself with unit speed (F = 1). In [25], the link between this motion and hyperbolic conservation laws is explained. In Figure 2a, the point of the front is downwards; as the moves inwards with unit speed, a shock develops as the front pinches off, and an entropy condition is required to select the correct solution to stop the solution from being doublevalued and to produce the limit of the viscous case. Conversely, in Figure 2b, the point of the front is upwards; in this case the unit normal speed results in a rarefaction fan which connects the left state with slope +1 to the right state which has slope -1. Extensive discussion of the role of shocks and rarefactions in propagating fronts may be found in [25].

The key to constructing numerical schemes which adhere to both this entropy condition and rarefaction structure comes from the link between propagating fronts and hyperbolic conservation laws. Consider the initial front given by the graph of f(x), with f and f'periodic on [0, 1], and suppose that the propagating front remains a function for all time. Let  $\phi$  be the height of the propagating function at time t, thus  $\phi(x, 0) = f(x)$ . The normal at  $(x, \phi)$  is  $(1, \phi_x)$ , and the equation of motion becomes  $\phi_t = F(K)(1 + \phi_x^2)^{1/2}$ . Using the

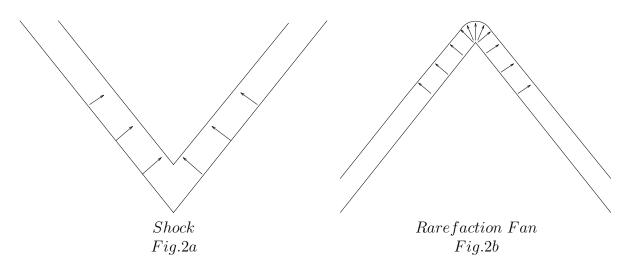


Figure 2: Front Propagating with Unit Normal Speed

speed function  $F(K) = 1 - \epsilon K$  and the formula  $K = -\phi_{xx}/(1 + \phi_x^2)^{3/2}$ , we get

$$\phi_t - (1 + \phi_x^2)^{1/2} = \epsilon \frac{\phi_{xx}}{1 + \phi_x^2}$$
(3)

Differentiating both sides of this equation yields an evolution equation for the slope  $u = d\phi/dx$  of the propagating front, namely

$$u_t + [-(1+u^2)^{1/2}]_x = \epsilon [\frac{u_x}{1+u^2}]_x.$$
(4)

Thus, the derivative of the Hamilton-Jacobi equation with parabolic right-hand-side for the changing height  $\phi$  is a viscous hyperbolic conservation law for the propagating slope u(see [28]). Our entropy condition is in fact equivalent to the one for propagating shocks in hyperbolic conservation laws. Thus, we exploit the numerical technology from hyperbolic conservation laws to build consistent, upwind schemes which select the correct entropy conditions. For details, see [23, 27].

Our goal then is to choose an appropriate speed function that yields front motion away from the body that remains smooth for all time, and thus can act to define one set of body-fit coordinate lines. Before doing so, we must extend the above ideas to include propagating fronts which are not easily written as functions. This is the level set idea introduced by Osher and Sethian [23], which we now describe.

Given a moving closed hypersurface  $\Gamma(t)$ , that is,  $\Gamma(t = 0) : [0, \infty) \to \mathbb{R}^N$ , we wish to produce an Eulerian formulation for the motion of the hypersurface propagating along its normal direction with speed F, where F can be a function of various arguments, including the curvature, normal direction, etc. The main idea is to embed this propagating interface as the zero level set of a higher dimensional function  $\phi$ . Let  $\phi(x, t = 0)$ , where  $x \in \mathbb{R}^N$  be defined by

$$\phi(x,t=0) = \pm d \tag{5}$$

where d is the distance from x to  $\Gamma(t = 0)$ , and the plus (minus) sign is chosen if the point x is outside (inside) the initial hypersurface  $\Gamma(t = 0)$ . Thus, we have an initial function

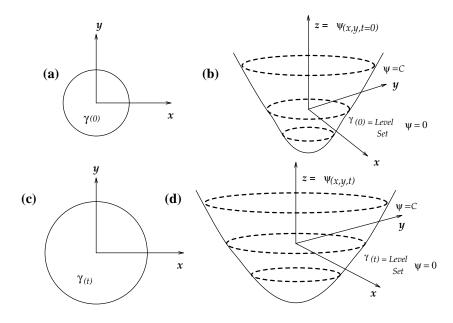


Figure 3: Propagating Circle

 $\phi(x,t=0): \mathbb{R}^N \to \mathbb{R}$  with the property that

$$\Gamma(t=0) = (x|\phi(x,t=0) = 0)$$
(6)

Our goal is to now produce an equation for the evolving function  $\phi(x, t)$  which contains the embedded motion of  $\Gamma(t)$  as the level set  $\phi = 0$ . Let  $x(t), t \in [0, \infty)$  be the path of a point on the propagating front. That is, x(t = 0) is a point on the initial front  $\Gamma(t = 0)$ , and  $x_t = F(x(t))$  with the vector  $x_t$  normal to the front at x(t). Since the evolving function  $\phi$  is always zero on the propagating hypersurface, we must have

$$\phi(x(t),t) = 0 \tag{7}$$

By the chain rule,

$$\phi_t + \nabla \phi(x(t,t)) \cdot x'(t) = 0 \tag{8}$$

Since F already gives the speed in the outward normal direction, then  $x'(t) \cdot n = F$  where  $n = \nabla \phi / |\nabla \phi|$ . Thus, we then have the evolution equation for  $\phi$ , namely

$$\phi_t + F|\nabla\phi| = 0 \tag{9}$$

$$\phi(x,t=0) \quad given \tag{10}$$

We refer to this as a Hamilton-Jacobi "type" equation because, for certain forms of the speed function F, we obtain the standard Hamilton-Jacobi equation.

In Figure 3, (taken from [29]), we show the outward propagation of an initial curve and the accompanying motion of the level set function  $\phi$ . In Figure 3a, we show the initial circle, and in Figure 3b, we show the circle at a later time. In Figure 3c, we show the initial position of the level set function  $\phi$ , and in Figure 3d, we show this function at a later time.

There are four major advantages to this Eulerian Hamilton-Jacobi formulation. The first is that the evolving function  $\phi(x, t)$  always remains a function as long as F is smooth.

However, the level surface  $\phi = 0$ , and hence the propagating hypersurface  $\Gamma(t)$ , may change topology, break, merge, and form sharp corners as the function  $\phi$  evolves, see [23].

The second major advantage of this Eulerian formulation concerns numerical approximation. Because  $\phi(x, t)$  remains a function as it evolves, we may use a discrete grid in the domain of x and substitute finite difference approximations for the spatial and temporal derivatives. For example, using a uniform mesh of spacing h, with grid nodes (i, j), and employing the standard notation that  $\phi_{ij}^n$  is the approximation to the solution  $\phi(ih, jh, n\Delta t)$ , where  $\Delta t$  is the time step, we might write

$$\frac{\phi_{ij}^{n+1} - \phi_{ij}^{n}}{\Delta t} + (F)(\nabla_{ij}\phi_{ij}^{n}) = 0$$
(11)

Here, we have used forward differences in time, and let  $\nabla_{ij}\phi_{ij}^n$  be some appropriate finite difference operator for the spatial derivative. As discussed above, the correct entropy-satisfying approximation to the difference operator comes from exploiting the technology of hyperbolic conservation laws. Following [23], given a speed function F(K), we update the front by the following scheme. First, separate F(K) into a constant advection term  $F_0$  and the remainder  $F_1(K)$ , that is,

$$F(K) = F_0 + F_1(K)$$
(12)

The advection component  $F_0$  of the speed function is then approximated using upwind schemes, while the remainder is approximated using central differences. In one space dimension, we have

$$\phi_i^{n+1} = \phi_i^n - \Delta t F_0 \left[ \left( \max(D_i^-, 0)^2 + \min(D_{i,0}^+)^2 \right)^{1/2} - |F_1(K) \nabla \phi_i^n| \right]$$
(13)

Extension to higher dimensions are straightforward; we use the version introduced in [31].

The third major advantage of the above formulation is that intrinsic geometric properties of the front may be easily determined from the level function  $\phi$ . For example, at any point of the front, the normal vector is given by

$$\vec{n} = \frac{\nabla\phi}{|\nabla\phi|} \tag{14}$$

and the curvature is easily obtained from the divergence of the gradient of the unit normal vector to front, i.e.,

$$K = \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} = -\frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_x\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}}$$
(15)

Finally, the fourth major advantage of the above level set approach is that there are no significant differences in following fronts in three space dimensions. By simply extending the array structures and gradients operators, propagating surfaces are easily handled.

As an example of the application of level set methods, consider once again the problem of a front propagating with speed  $F(K) = 1 - \epsilon K$ . In Figure 4, we show two cases of a propagating initial triple sin curve. For  $\epsilon$  small (Fig. 4a), the troughs sharpen up and will result in transverse lines that come too close together. For  $\epsilon$  large (Fig. 4b), parts of the

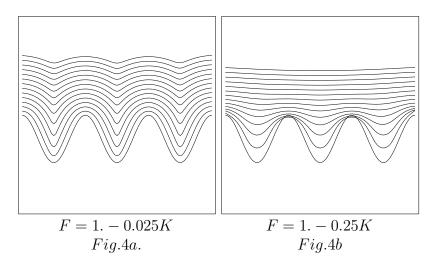


Figure 4: Propagating Triple Sine Curve.

boundary with high values of positive curvature can initially move inwards, and concave parts of the front can move quickly up.

Since its introduction in [23], the above level set approach has been used in a wide collection of problems involving moving interfaces. Some of these applications include the generation of minimal surfaces [8], singularities and geodesics in moving curves and surfaces in [10], flame propagation [24, 33], fluid interfaces [4, 7], crystal growth and dendritic solidification [31], detection of self-similar surfaces [9] and shape reconstruction [21]. Extensions of the basic technique include fast methods in [1] and grid generation in [29, 30]. The fundamental Eulerian perspective presented by this approach has since been adopted in many theoretical analyses of mean curvature flow, see in particular [12, 6], and related work in [2, 11, 13, 14, 17, 19].

#### **III.** Fast Level Set Methods

The main issue in the level set approach is the extension of the speed function F to all of space in order to move all the level sets, not simply the zero level set on which the speed function is naturally defined. While this may be straightforward in some cases, it is not efficient, since one must perform considerable computational labor away from the front to advance the other level sets.

In [1], an approach introduced by Chopp in [8] and used in recovering images in [21], was refined and analyzed extensively. The central idea is to focus computational effort in a narrow band about the zero level set. We only update the values of the level set function  $\phi$  in this thin zone around the interface. Thus, in two dimensions, an  $O(N^2)$  calculation, where N is the number of grid points per side, reduces to an O(kN) calculation, where k is the number of cells in the narrow band. This reduction of labor makes the method typically much faster than marker particle methods, due to the need for many marker points per mesh cell in order to obtain acceptable accuracy. As the front moves, the narrow band must occasionally be rebuilt (known as "re-initialization") of the interface. For details, see [8, 21, 1].

Scheme	40 Cells	80 Cells	16 Cells
Narrow Band 1st Order	125.1	243.5	507.9
Full Matrix 1st Order	330.9	1367.5	5657.6
Narrow Band 2nd Order	118.8	250.5	547.8

Figure 5: Comparative Timings of Schemes

Briefly, the entire two-dimensional grid of data is stored in a square array. A onedimensional object is used to keep track of which points in this array correspond to the tube, and the values of  $\phi$  at those points are updated. When the front moves half the distance towards the edge of the tube boundary, the calculation is stopped, and a new tube is built with the zero level set interface boundary at the center. Details on the accuracy, typical tube sizes, and number of times a tube must be rebuilt may be found in [1].

As an example of the speed up possible using this approach, we cite the results given in [1]. On a typical two-dimensional interface tracking problem, we compare timings of a first and second order narrow band approach with the the full matrix approach; calculations are performed over various grid sizes. Results are measured in a rough manner, with optimization turned off and timing compared using the Unix time command. Thus, the important feature are the ratios. The narrow band calculation is around 10 times faster for the finest calculation than the full matrix solution.

#### IV. Extension to Multiple Interfaces

As discussed above, the level set technique was initially designed for tracking an interface between two regions, where the notion of an "inside" and "outside" is clear. Some work has been done on extending the approach to multiple fluids; mostly notably in [3] where an extensive study of the motion of triple points was made. In the approach presented there, at each time step the calculation stops, the zero level set is found, and the entire level set function is rebuilt using a reinitialization technique.

In many cases, such an approach is not necessary; in [32] a level approach is given for tracking an arbitrary number of interfaces in two and three dimensions which includes the motion of triple points in some cases. The technique presented does not rely on any reinitialization, and retains the essential characteristic of the original approach; the front is only explicitly constructed for display purposes. Here, we briefly review the approach, for details, see [32].

The key idea lies in recasting the interface motion as the motion of one level set function for each material. In some sense, this is what was done in the re-ignition idea given in [24]. In that approach, the front was a flame which propagated downstream under a fluid flow, and was re-ignited at each time step at a flame holder point. This "re-ignition" was executed by taking the minimum of the advancing flame and its original configuration around the flame holder, thus ensuring that the maximum burned fluid is achieved.

Consider two-dimensions and let Region A occupy the left half plane while Region B occupies the right half plane. Consider the case where the interface between the two regions,

that is, the y axis, propagates in its normal direction with speed 1, where the normal is defined to point in the positiv x direction. Thus, the interface moves to the right with unit speed. Our central perspective is as follows:

We imagine that Region A is propagating in its outward normal direction with speed 1, and imagine that Region B does not move and is overtaken by Region A.

This is accomplished as follows. Let  $\phi_A^*, \phi_B^*$  be the trial positions of the level sets functions obtained by advancing interface A ahead with speed 1 for one time step, and interface B ahead with speed 0 for one time step; these advances are obtained by using the hyperbolic conservation law methodology described above. Of course, evolution with speed 0 corresponds to no motion, but we describe it this way for ease of explanation. Our only job now is to combine these values in such a way as to obtain the new values at time n + 1. Let

$$\phi_A^{n+1} = \phi_A^* \qquad \phi_B^{n+1} = max(\phi_B^*, -\phi_A^*) \tag{16}$$

The technique may be extended easily to multiple regions as follows. Imagine that we have N separate regions, and a full set of all possible pairwise speed functions  $F_{IJ}$  which describe the propagation speed of region I into region J; F is taken as zero if Region I cannot penetrate J. The idea is to advance each interface to obtain a trial value for each interface with respect to motion into every other region, and then combine the trial values in such a way as to obtain the maximum possible motion of the interface.

In general then, we proceed as follows. Given a Region I, we obtain N - 1 trial level set functions  $\phi_{IJ}^*$  by moving the Region I into each possible Region J, J=1,N ( $J \neq I$ ) with speed  $F_{IJ}$ . During the motion of Region I into Region J, we assume that all other regions are impenetrable. We then test the penetrability of the Region J itself, leaving the value of  $\phi_{IJ}^*$ unchanged if  $F_{IJ} \neq 0$ , else modifying it with the maximum of itself and  $-\phi_{JI}^*$ . Finally, to allow Region I to evolve as much as possible, we take the minimum over all possible motions to obtain the new position; this is the re-ignition idea described earlier. Complete details of the approach may be found in [32].

As illustration of our approach, we study the motion of a triple point between Regions A, B, and C. Region A is the disk on the left, Region B the disk on the right, and Region C the remaining material. We assume that Region A penetrates B with speed 1, B penetrates C with speed 1, and C penetrates A with speed 1. The exact solution to this is given by a spiral with no limiting tangent angle as the triple point is approached. The triple point does not move; instead, the regions spiral around it. In Figure 6, we show the results calculated on a 98x98 grid. Starting from the initial configuration, the regions spiral around each other, with the leading tip of each spiral controlled by the grid size. In other words, we are unable to resolve spirals tighter than the grid size, and hence that controls the fine scale description of the motion. However, we note that the triple point remains fixed.

#### V. Generation of Body-fitted Logically Rectangular Grids

The generation of logically rectangular grids around and inside complex bodies is still

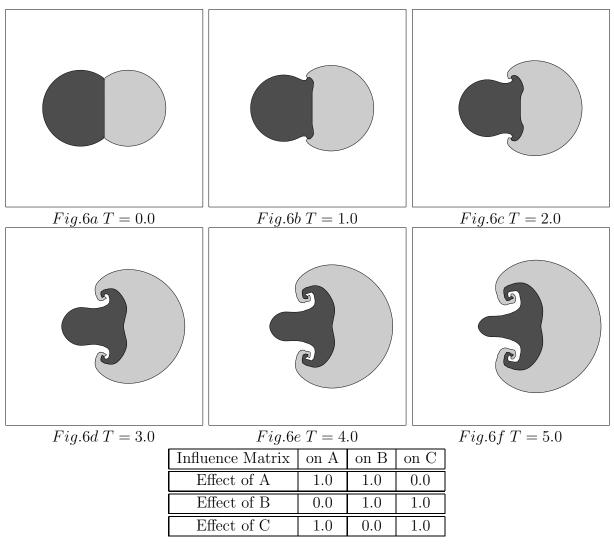


Figure 6: Spiraling Triple Point: 98x98 Grid

an art. While unstructured meshes may be obtained in a relatively automated fashion, many calculations require the accuracy of a logically rectangular, body-fitted grid. For example, high Reynolds turbulent flow requires an accurate, body-fit grid in the boundary layer where gradients are steep and a highly accurate scheme is critical. Standard techniques in logical rectangular boundary-fitted grid generation fall under four general categories. Hyperbolic grids march out from the boundary. Algebraic grids adjust nodes until a desired shape is achieved. Elliptic grids solve an associated elliptic partial differential equation, and variational methods minimize certain functionals. Grids obtained through these techniques can be plagued by colliding grid lines, inability to handle sharp corners in the bodies, and difficulty extending to three space dimensions.

Recently, the level set technique have been applied to grid generation in two and three dimensions, [29]. Here, we review some of that work; for details see [29]. The technique hinges on viewing the boundary of the body as a propagating front. The front is then allowed to propagate with a speed law that ensures that it will smoothly evolve from the body in such a way that the position of the front yields one set of coordinate lines. The judicious choice of speed function produces a geometry-flowing interface which is guaranteed to handle cusps and corners, produce smooth contours, and trivially extend to three space dimensions. Lines orthonormal to the propagating, body-fitted, level-set lines are obtained by following the trajectories of particles propagating with a speed function dependent on the local curvature and emanating from the boundary.

Using this approach, the resulting algorithm generates two and three dimensional interior and exterior grids around reasonably complex bodies which may contain sharp corners and significant variations in curvature. We have also used these techniques to produce non-uniform solution-adaptive meshes and boundary-fitted moving grids. The algorithm is completely automatic; the only user-supplied grid-dependent parameters are the shape of the initial boundary, and the time step spacing for the evolving front function. In Figure 7, we show the results of this application. In Figure 7a, a two-dimensional external grid is produced around a fairly complex object; in Figure 7b, an internal grid is computed. Finally, in Figure 7c, a three-dimensional grid is obtained. In the exterior grid cases, a speed function was chosen of the generic form  $1 - \epsilon K$ , where K is the curvature; in the case of the interior grid, the speed function was chosen as  $F = \min(\delta, K)$ , where  $\delta$  is a threshold value chosen to ensure that every point of the body moves inwards with some minimum speed. Without such a factor, points on non-convex boundaries would first evolve outwards at some points, which violates the ability of the algorithm to create a grid.

The above approach is automatic, inexpensive, and requires no complex alteration in three-dimensions. To extend it to multiple bodies, our current work is aimed at using the above level set approach to grid generation to generate a logically rectangular grid near each body (for example, in the boundary layer and past), and then take these grids as input to more traditional techniques which can grid multiple bodies, but cannot access complex shapes in an automatic fashion. Thus, the approach is a marriage of the level set algorithm to generate an accurate near-body grid which yields a smoother, almost convex shape, and a traditional methodology which will patch these smooth grids together. For details of this approach, see [30].

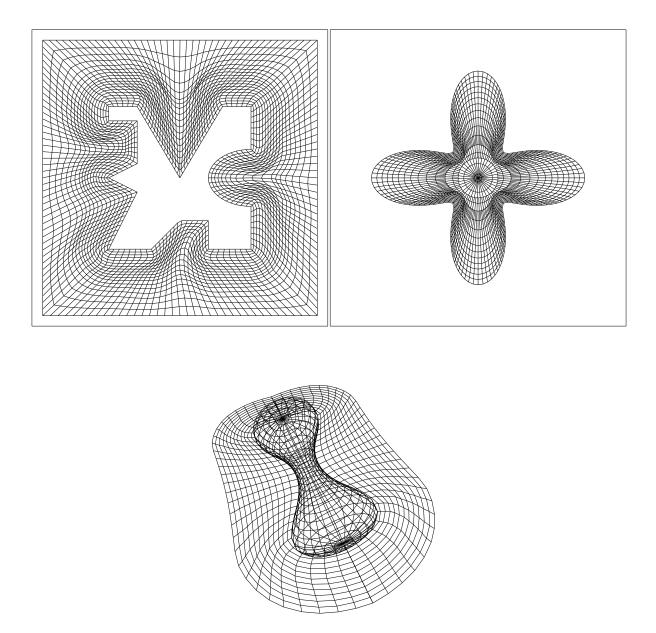


Figure 7: Level Set Approach to Grid Generation

## VI. Shape Representation and Recognition

Roughly speaking, image processing and analysis includes four stages:

- *Enhancement*. In this step, the original image, which may be given by an image intensity at grid points, or in fact represent range data, is enhanced and modified. This enhancement includes techniques to sharpen features, heighten contrasts, remove noise, remove blur, etc.
- Segmentation. In this step, the goal is to isolate a particular desired feature within an image. For example, in medical images such as MRI and CAT scans, one may want to isolate a particular shape, such as those corresponding to tumors, which are suggested by sharp contrasts in the image intensity; thus the goal is to "segment" a shape from an image. Traditional techniques often exploit "snakes" or balloon-methods which attempt to follow a parameterized curve as it deforms to fit the boundary of the desired shape.
- *Representation*. In this step, the goal is to represent the segmented shape in some convenient way, such as splines or cylindrical patches, such that various geometric properties (area, curvature variation, etc.) can be easily calculated.
- *Recognition*. Given the above representation, the goal is to somehow match the representation to a given library of shapes. Techniques often exploit neural nets or direct distance function comparisons of carefully chosen "feature-vectors" which characterize information about the desired shape.

Partial differential-based approaches have played a major role in revolutionizing these four stages. To begin, enhancement schemes based on applying shock filtering schemes and curvature-driven flows, approximated using the level set approach have provided significant improvement in techniques to enchance images. In these approaches, the idea is to view the image intensity map as something akin to a "level set function", and evolve the function under geometry-driven flows which depend on the curvature and both sharpen features and anisotropically smooth. A level set approach to segmentation was introduced in [21] and independently in [5]. In [21], an initial front was placed inside a field, and an intensitydriven speed law devised to attract this interface to the boundary of the shape. The front is again moved using the level set approach, providing the ability to change topology and evolve into sharp corners and limbs of the desired shape. Once segmented by the level set approach, the resulting distance function representation can be used to compare the found shape with a library of test images; in [22], these techniques were used to perform optical character recognition.

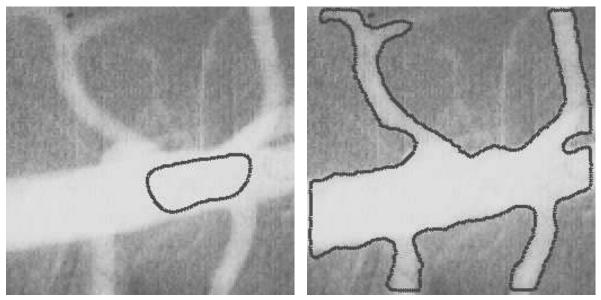
As examples of the level set approach applied to the last three stages of the above process (segmentation, representation, and recognition), in Figure 8, we show two different examples. In Figure 8a, we show a level set technique applied to extract the arterial structure from a digital subtraction angiogram (DSA). The level set approach starts with a small front inside the artery, and expands easily as it evolves into the intricate channel structure; for details, see [21]. In Figure 8b, we show our level set technique applied to character recognition; using the NIST Database 3 and 7 as our training sets, we use a neural net based approach

and train on the signed distance function as the feature vector. Those shapes which are not identified within the given threshold are then altered under flow rules based on our hyperbolic conservation law approach to bring them into a given test category. Figure 8b shows handwritten characters which have been identified using this approach; for details, see [22].

Acknowledgements: All calculations were performed at the University of California at Berkeley and the Lawrence Berkeley Laboratory.

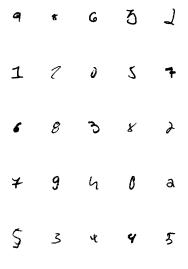
# References

- [1] Adalsteinsson, D., and Sethian, J.A., A fast level set method for propagating interfaces, submitted for publication, Jour. Comp. Phys., 1994.
- [2] Altshuler, S., Angenent, S.B., and Giga, Y., Mean Curvature Flow through Singularities for Surfaces of Rotation, preprint, 1993.
- [3] Merriman, B., Bence, J., and Osher, S.J., *Motion of Multiple Junctions: A Level Set Approach*, to appear, Jour. Comp. Phys., 1994.
- [4] Bourlioux, A., and Sethian, J.A., A Conservative Level Set Method for Fluid Interfaces, to be submitted, 1994.
- [5] Casselles, V., Catte, F., Coll, T., and Dibos, F., A Geometric Model for Active Contours, Numerische Mathematic, Vol. 60, pp. 1-31, 1993.
- [6] Chen, Y., Giga, Y., and Goto, S., Uniqueness and Existence of Viscosity Solutions of Generalized Mean Curvature Flow Equations, J. Diff. Geom, Vol. 33, 749, 1991.
- [7] Chang, Y.C., Hou, T.Y., Merriman, B., and Osher, S.J., A Level Set Formulation of Eulerian Interface Capturing Methods for Incompressible Fluid Flows, submitted for publication, Jour. Comp. Phys., 1994.
- [8] Chopp, D. L., Computing minimal surfaces via level set curvature flow, Journal of Computational Physics, Vol. 106, pp. 77–91, 1993.
- [9] Chopp, D. L., Construction of Self-Similar Surfaces Need reference, 1994.
- [10] Chopp, D.L. and Sethian, J.A., Flow under Curvature: Singularity Formation, Minimal Surfaces, and Geodesics, to appear, Jour. Exper. Math, 1994.
- [11] Evans, L.C., Sonar, H.M., and Souganidis, P.E., *Phase Transitions and Generalized Motion by Mean Curvature*, Communications on Pure and Applied Mathematics, Vol. 45, 1097, 1992.



Digital Image

Final Position



Identified Characters

Figure 8: Level Set Approach to Image Capturing and Recognition

- [12] Evans, L.C., and Spruck, J., Motion of Level sets by Mean Curvature, J. Diff. Geom, Vol. 33, pp. 635-681, 1991.
- [13] Evans, L.C., and Spruck, J., Motion of Level sets by Mean Curvature II, Transactions of the American Mathematical Society, Vol. 330, 91, 1992.
- [14] Falcone, M., Giorgi, T., and Loretti, P., Level Sets of Viscosity Solutions and Applications, Instituto per le Applicazioni del Calcolo, Rome, preprint, 1990.
- [15] Gage, M., Curve Shortening Makes Convex Curves Circular, Inventiones Mathematica, Vol. 76, 357, 1984.
- [16] Gage, M., and Hamilton, R., The Equation Shrinking Convex Planes Curves, J. Diff. Geom, Vol. 23, 69, 1986.
- [17] Giga, Y., and Goto, S., Motion of Hypersurfaces and Geometric Equations, Journal of the Mathematical Society of Japan, Vol. 44, 99, 1992.
- [18] Grayson, M., The heat equation shrinks embedded plane curves to round points, J. Diff. Geom., Vol. 26, 285 (1987).
- [19] Huisken, G., Flow by mean curvature of convex surfaces into spheres, J. Diff. Geom., Vol. 20, 237, (1984).
- [20] Kimmel, R., and Bruckstein, A., Shape from Shading via Level Sets, Center for Intelligent Systems Report No.9209, Technion- Israel Institute of Technology, June 1992,
- [21] Malladi, R., Sethian, J. A., and Vemuri, B. C., Shape modeling with front propagation: A level set approach, Center for Pure and Applied Mathematics, Report PAM-589, Univ. of California, Berkeley, August 1993.
- [22] Malladi, R., Sethian, J.A., A Unified Framework for Shape Segmentation, Representation, and Recognition, submitted August 1994, CVGIP - Image Understanding.
- [23] Osher, S., and Sethian, J. A., Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation, Jour. Comp. Phys., Vol. 79, pp. 12-49, 1988.
- [24] Rhee, C., Talbot, L. and Sethian, J.A., Dynamical Study of a Premixed V Flame, Submitted for Publication, Jour. Fluid Mech. 1994.
- [25] Sethian, J.A., An Analysis of Flame Propagation, Ph.D. Dissertation, Mathematics, University of California, Berkeley, 1982.
- [26] Sethian, J.A., Curvature and the evolution of fronts, Commun. in Math. Physics, Vol. 101, pp. 487–499, 1985.
- [27] Sethian, J.A., Numerical algorithms for propagating interfaces: Hamilton-Jacobi equations and conservation laws, Jour. of Diff. Geom., Vol. 31, pp. 131-161, 1990.

- [28] Sethian, J.A., Numerical methods for propagating fronts, in Variational methods for free surface interfaces, edited by P. Concus and R. Finn, (Springer-Verlag, New Work, 1987).
- [29] Sethian, J.A., Curvature Flow and Entropy Conditions Applied to Grid Generation, to appear, J. Comp. Phys. 1994.
- [30] Sethian, J.A., Three-Dimensional Grid Generation Using Curvature Flow and Level Sets. work in progress.
- [31] Sethian, J.A. and Strain, J.D., Crystal Growth and Dendritic Solidification J. Comp. Phys., Vol. 98, pp. 231-253, (1992).
- [32] Sethian, J.A., Tracking the Evolution of Multiple Interfaces and Multi-Junctions submitted for publication, 1994, J. Comp. Phys.
- [33] Zhu, J. and Sethian, J.A., Projection Methods Coupled to Level Set Interface Techniques J. Comp. Phys., Vol. 102, pp.128-138, 1992.

James A. Sethian Dept. of Mathematics University of California Berkeley, California, 94720