Geometric Shock-Capturing ENO Schemes for Subpixel Interpolation, Computation, and Curve Evolution

Kaleem Siddiqi, Benjamin B. Kimia and Chi-Wang Shu
Brown University
Providence, RI 02912

Abstract

Subpixel interpolation methods often use local surface fits or structural models in a local neighborhood to obtain the interpolated curve. Whereas their performance is good in smooth regions of the curve, it is typically poor in the vicinity of singularities. Similarly, when geometric estimates are regularized, discontinuities are often blurred over, leading to poor estimates in their vicinity. In this paper we propose a geometric interpolation technique to overcome these limitations by: 1) not blurring across discontinuities, and 2) explicitly and accurately placing them. The essential idea is to prevent the propagation of information across singularities by explicitly placing a "shock"; information is only allowed to propagate from the smoother side. The placement of shocks is guided by geometric continuity constraints, resulting in subpixel interpolation with accurate geometric estimates. The interpolations are shown to be better than spline-like interpolations in smooth regions, and far better in discontinuous ones. We demonstrate the usefulness of the technique in capturing not only smooth evolving curves, but also discontinuous ones, even when multiple or entire curves are present in the same pixel.

1 Introduction

Hyperacuity, or the ability of the human visual system to detect features at resolutions an order of magnitude better than retinal resolution, is a remarkable phenomenon. This phenomenon is particularly impressive since locating curves, *e.g.*, for image registration [4], as well as obtaining reliable estimates of geometric quantities, such as orientation and curvature, *e.g.*, for stereo and optical flow, have traditionally proven to be difficult problems. We follow [9] in exploiting *two* main sources of information to obtain measurements of differential structure in images: 1) normal conditions to utilize information along profiles *orthogonal* to image curves, and 2) tangential conditions to take advantage of continuity *along* image curves. Our goal is to achieve robust localization and accurate geometric estimates, not only at regular, but also at *subpixel* resolution.

First, observe that image structure is rarely present in binary form. It contains "contrast" information that can be used to locate curves with subpixel accuracy [8, 13]. In other work, curves have been viewed as level sets of a surface, *e.g.*, edges as the zero-crossings of the Laplacian of a Gaussian operator [17], isophotes of image intensity for scale [14], *etc.* In addition, more recently, curve evolution applications have used an embedding surface to represent the evolving curve, thus utilizing the additional dimension to regularize computations [18, 5], *e.g.*, shape representation [10, 11], shape from shading [12], image smoothing [3], affine invariant curve evolution [22, 1], shape modeling [2, 16, 26], optical flow [15], *etc.* In the above examples, the process of locating the curve and computing its geometric properties, *e.g.*, orientation and curvature, can benefit from the information contained in the embedding surface in the direction *orthogonal* to the curve.

Second, since slight variations in pixel data can cause large variations in computations of orientation and curvature, smoothing *along* the boundary is often employed to make the estimation procedure more robust, *e.g.*, spline interpolation, regularization, *etc.* While in smooth regions regularization achieves excellent geometric estimates, this is often at the expense of blurring across singularities, as observed in [8], thus leading to grossly inaccurate estimates in their vicinity, Figure 1 (middle column). However, observe that discontinuities are usually finite in number and are isolated, while the object boundaries are smooth between them. Thus far, it has not been clear how to exploit these constraints without also blurring across singularities; see [21] for a promising approach in the domain of curve evolution. In recent years, a large number of nonlinear approaches to smoothing shapes and images have been introduced, with the goal of preserving discontinuities, *e.g.*, see the articles in [27] for an overview. The ideas developed here can be utilized for curve localization and geometric estimation.

The paper is organized as follows. In Section 2 we review the *essentially non-oscillatory* (ENO) interpolation method, originally proposed to overcome the problem of smoothing across discontinuities in fluid dynamics applications. In Section 3 we suggest a modification of the scheme to explicitly capture and represent the discontinuities, leading to more accurate geometric estimates in their vicinity. In Section 4 we extend the technique for interpolating 1D functions to one for interpolating a

0-8186-7190-4/95 $4.00 © 1995 IEEE

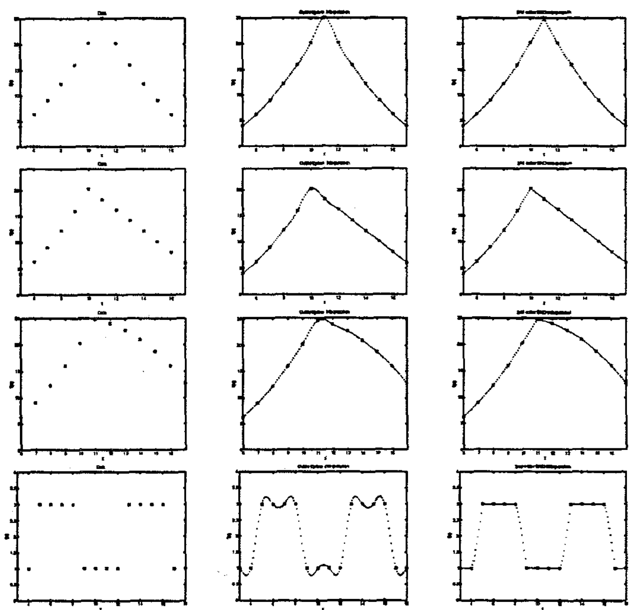

Figure 1: Cubic spline interpolation versus ENO interpolation. LEFT COLUMN: The data points are marked with "*"s. MIDDLE COLUMN: The cubic spline interpolation is overlaid on the data. RIGHT COLUMN: The second-order ENO interpolation is overlaid on the data. Observe that in contrast to cubic spline interpolation, ENO interpolation does not blur across singularities or introduce spurious oscillations.

2D curve, via a two stage approach, the first to obtain a set of ordered subpixel sample points, and the second to interpolate between them using geometric basis functions. Finally, in Section 5 we illustrate the advantages of this *geometric* ENO (GENO) interpolation scheme in application to curve evolution.

2 Background: ENO

Splines or polynomials are commonly used to interpolate discrete data and provide geometric estimates (orientation, curvature, *etc.*) in a variety of computer vision and graphics applications. Whereas the estimates are generally reliable in regions where the data is smooth, they are prone to error in the vicinity of discontinuities. This follows because such fitting techniques blur over discontinuities by propagating information across them, Figure 1 (middle column). Recently, a class of schemes have been proposed in the numerical analysis literature to address this problem, in application to the numerical solution of conservation laws and the propagation of fronts. These *Essentially Non-Oscillatory* schemes were introduced by Harten *et al.* [6, 7], were adapted to the numerical solution of Hamilton-Jacobi equations by Osher *et al.* [18, 19], and were later made more efficient by Shu and Osher [24].

The key feature of ENO schemes is an *adaptive stencil* high order interpolation which tries to avoid shocks or high gradient regions whenever possible [24]: at regions

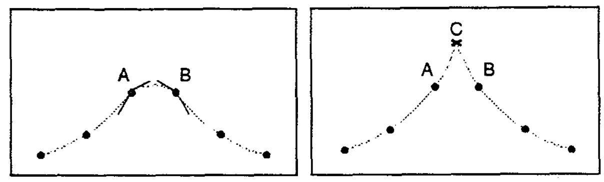

Figure 2: LEFT: The interval AB supports two discontinuities, one at A and one at B. RIGHT: As an alternate interpolation, one may "relieve" these two shock points by smoothly continuing the neighboring interpolations and introducing a single singularity at C.

neighboring discontinuities, the smoothing is always from the side *not* containing the discontinuity. The basic idea is to select between two contiguous sets of data points for interpolation the one which gives the lower variation. To illustrate, to find the polynomial approximation between the grid points x_j and x_{j+1}, we start by interpolating a first-order polynomial between x_j and x_{j+1}. A second-order polynomial is constructed by adding either x_{j-1} or x_{j+2}, whichever produces the smoother polynomial, *i.e.*, that which has a lower coefficient for the highest order term. A third-order polynomial is interpolated by choosing an additional data point, and so on, for higher degree polynomial interpolations. The procedure is known to give very good numerical results: sharp, non-oscillatory transitions at shocks and high order accuracy in regions where the data is smooth, Figure 1.

3 Shock Placing ENO

The ENO interpolation algorithm effectively deals with areas where the underlying data is smooth, and with areas neighboring discontinuities. However, since singularities are not explicitly placed, geometric estimates in an interval containing a singularity are still error prone. In seeking a solution to this problem we follow Harten's idea of explicitly placing a "shock", but rather than use "conservation' to guide its placement [6], we suggest the use of geometric constraints for the vision applications we have in mind. Two questions are addressed: what constraints can be employed to signal the presence of a singularity, and, how should the singularity be placed?

Observe that the placement of shocks can be signaled by the ENO interpolation algorithm: in smooth regions, ENO interpolation in one interval continuously follows to interpolations in neighboring intervals, with little or no change in orientation or curvature, Figure 2. However, in an interval appearing to contain a discontinuity, *e.g.*, AB, the orientation and curvature limits at each boundary point are different when approached from the left than when approached from the right. Therefore, the interval AB typically supports a discontinuity at each boundary point. As an alternate interpolation, one may "relieve" these two discontinuities (shock points) and introduce a single shock C by smoothly extrapolating neighboring interpolations, Figure 2 (right). Such a choice is guided by minimizing curvature and orientation variations. To

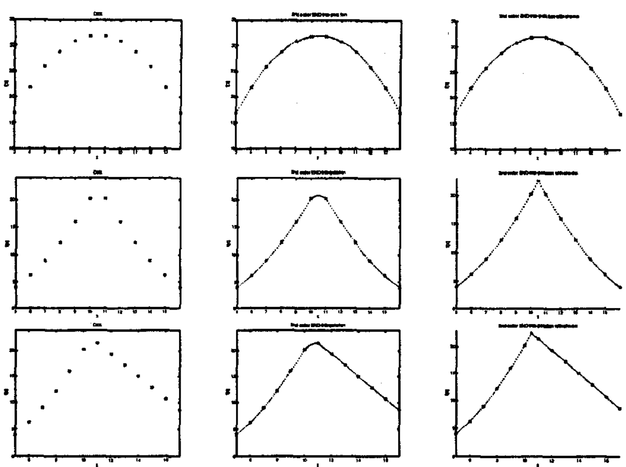

Figure 3: ENO interpolation versus ENO interpolation with shock placement. LEFT COLUMN: The data points are marked with "*"s. MIDDLE COLUMN: The second-order ENO interpolation is overlaid on the data. Note that singularities that lie within an interval are not captured. RIGHT COLUMN: In a modified algorithm a shock is placed (marked with a cross) when there are significant orientation and curvature changes at the end points of an interval, (MIDDLE AND BOTTOM ROWS), but not when the data is smooth (TOP ROW).

formalize these ideas we make two assumptions explicit:

Assumption 1 *The underlying curve is piecewise smooth with finite total curvature.*

Assumption 2 *The curve has a finite number of singularities, with the singularities being at least one interval apart.*

Our strategy for shock placement is as follows. First, all intervals where singularities may occur are "flagged." Second, the constraint of piecewise smoothness between singularities is enforced by extrapolating the interpolations from the neighboring intervals, and placing the singularity at the point of their intersection. Thus, the interpolating curve on one side of the shock is obtained solely from the neighboring interval on that side. Examples illustrating this strategy are shown in Figure 3 (right column). The resulting shocks, which are marked by crosses, are intuitive. Observe also that for smooth data, no shocks are placed, and the interpolation agrees with the unmodified ENO interpolation, Figure 3 (top). We now extend this "shock placing" ENO scheme to one for 2D data.

4 Geometric ENO (GENO)

The approach developed thus far applies to 1D data that is ordered. In extending the scheme to one for 2D data, we face at least two difficulties. First, in general the 2D data is not ordered until the shape's boundary has been traced. Second, the independent variable for interpolation, *e.g.*, arc-length $\tilde{s}$, is not readily available

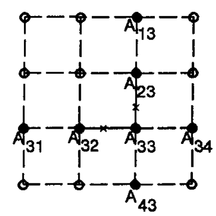

Figure 4: The 1D shock placing ENO interpolation is performed along gridlines. The zeros of the interpolation polynomials provide subpixel sample points, *e.g.*, the crosses in the intervals $A_{32}A_{33}$ and $A_{23}A_{33}$.

prior to obtaining the interpolation itself[1]. In the following we propose a two-stage solution where: (i) ordered subpixel sample points of the boundary are obtained, and (ii) the 2D sample points are interpolated, using a *geometric* generalization of the 1D ENO scheme.

First Stage: Subpixel Samples The goal of the first stage is to obtain samples of the trace of the boundary with subpixel accuracy, and later to order these points. Subpixel sample points can be obtained by 1D ENO interpolation of the surface in the direction of any line passing through 2D grid points[2]: for each interval along such a line, we apply the shock-placing ENO algorithm, Section 3. When no shock is placed, the sample points correspond to the zeros of the interpolation polynomial that lie *within* the interval under consideration. When a shock is placed there are two polynomials to consider, *i.e.*, one from the left and one from the right. The sample points correspond to the zeros of each polynomial that lie *within* the interval under consideration *and* on the appropriate side of the shock.

To illustrate, consider shock-placing ENO interpolation using second-order polynomials, Figure 4. The interpolation in the interval $A_{32}A_{33}$ will use the points A_{32}, A_{33} and either A_{31} or A_{34}, depending upon which leads to the smaller coefficient for the second-order term. Similarly, the interpolation in the interval $A_{23}A_{33}$ will use the points A_{23}, A_{33} and either A_{13} or A_{43}. The valid zeros will be stored for further processing, as indicated by the crosses in Figure 4. Note that the subpixel samples are found *without explicitly representing* a high resolution sub-grid between any two neighboring grid points.

The sample points must now be ordered such that they lie consecutively along the boundary of the shape. This can be done by generalizing a standard contour tracing algorithm such that instead of grid points, it uses the high resolution sample points. The details of this procedure are described in [25]. We should mention that the final algorithm merges the procedures for obtaining sam-

[1] See [20] for a similar dilemma in the problem of curve detection from images.

[2] In practice we have found that *grid lines*, or lines that pass through (normal resolution) grid points in both the horizontal and vertical directions, give excellent results, and there is limited used for interpolation along diagonals.

ple points and tracing them, thus achieving significant computational savings.

Second Stage: Geometric ENO interpolation The goal of the second stage is to interpolate between the subpixel sample points, while simultaneously placing singularities when required. Note that there are difficulties with applying the shock placing ENO algorithm to the 2D samples. A description of the shape's boundary as a function in some local coordinate frame is infeasible because: i) the coordinate system, *e.g.*, the *Frenet Frame*, may not be computable prior to obtaining the interpolation, ii) the function can become multivalued in intervals containing high curvature points, and iii) the results are not invariant to rotations of the data. On the other hand, interpolating the original (extrinsic) x and y coordinates of the boundary points as separate functions of the arc length parameter $\tilde{s}$ is also infeasible because: i) the measure of arc length is not available prior to obtaining the fit, ii) unless a joint criterion is designed for selection, the data points used for ENO interpolation may not coincide for $x(\tilde{s})$ and $y(\tilde{s})$, and iii) once again, the results are not invariant to rotations of the data set.

The solution we propose rests in replacing extrinsic polynomials with a set of *geometric* basis "functions" for interpolation. The essential idea is to use fits that are not dependent on the choice of a coordinate system, but rather depend only on the geometry of the underlying curve. Recall that in ENO interpolation using polynomials, all but the highest order derivative of the interpolant are zero. In a geometric sense, the variables of interest are orientation and its derivatives: curvature, curvature variation, *etc.* In analogy to the algebraic case, the basis functions can be found by setting various order derivatives to zero. For example, $\theta_{\tilde{s}} = 0$ yields a straight line, $\theta_{\tilde{s}\tilde{s}} = 0$ gives a circle, and $\theta_{\tilde{s}\tilde{s}\tilde{s}} = 0$ results in an Euler spiral [3]. These constitute our geometric interpolation bases, with an associated sense of order.

The *Geometric Essentially Non Oscillatory* (GENO) interpolation begins, therefore, with a straight line, interpolating the boundary $C(s) = (x(s), y(s))$ between two consecutive sample points $C(s_n)$ and $C(s_{n+1})$. If we stopped here, we would have the first-order (line) interpolation. A second-order (circular arc) interpolation is constructed by adding either of the points $C(s_{n-1})$ or $C(s_{n+2})$, whichever provides the circular arc fit with lower curvature, and so on for higher order interpolations[4]. Further, to deal with the presence of singularities *within* an interval, the same shock placing algorithm developed in Section 3 is used, with the exception that geometric basis functions are used in place of polynomials for interpolation.

[3]The Euler spiral is a curve with linear curvature variation, studied by Euler in 1781, which has found practical use in applications such as laying railroad tracks.

[4]In this paper we have used second order (circular arc) interpolations, for which the accuracy of the results is generally excellent. Higher order interpolations will lead to improved accuracy when curvature varies significantly within an interval.

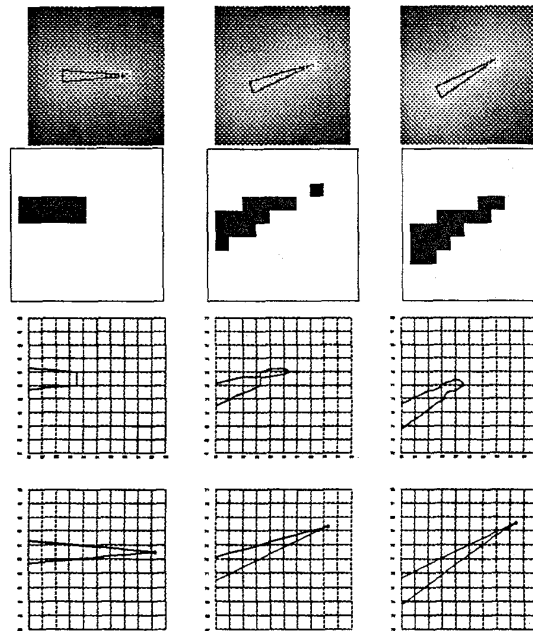

Figure 5: A comparison of interpolation techniques. TOP: The original images are 128x128 signed distance transforms of a triangle with a 10 degree right vertex, oriented at 0, 20, and 30 degrees respectively. The box depicts a 10x10 region under examination. SECOND ROW: The shape's interior at the resolution of the grid. THIRD ROW: Bilinear interpolation truncates or rounds corners, and introduces gross artifacts. FOURTH ROW: Geometric ENO interpolation preserves corners (marked with circles) and accurately places them with subpixel resolution.

In Figures 5 and 6 we illustrate GENO interpolation on straight-edged and curved shapes, comparing it with the commonly used bilinear interpolation technique. Observe that: 1) the GENO fits are intuitive both in smooth regions, and in the vicinity of singularities, 2) GENO explicitly captures singularities and places them accurately, and 3) GENO is robust to rotations and does not truncate corners or introduce artifacts in the vicinity of singularities. We conclude by illustrating the advantages of GENO in application to curve evolution.

5 Subpixel Curve Evolution

Geno interpolation can benefit curve evolution by: i) providing for more accurate geometric estimates, thus leading to more accurate numerical simulations, and ii) allowing for a refined description of the evolving curve. First, we have theoretically and numerically compared geometric estimates of curvature using GENO with those from two other techniques [25], and have found that GENO provides more accurate estimates in smooth regions, and far better ones in regions containing singularities. Second, we illustrate the advantages of using GENO for locating the curve for a particular application of curve evolution, namely, shape representation in

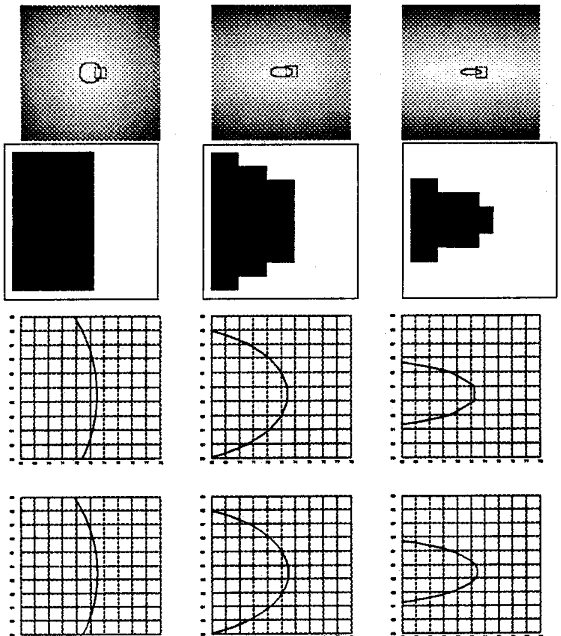

Figure 6: A comparison of interpolation techniques. TOP: The original images are 128x128 surfaces for an ellipse with major/minor axes equal to 20/20, 20/10, and 20/5 respectively. The box depicts a 10x10 region under examination. SECOND ROW: The shape's interior at the resolution of the grid. THIRD ROW: Bilinear interpolation. FOURTH ROW: Geometric ENO interpolation.

computer vision [11]. Here deformations of the shape are given by $\frac{\partial C}{\partial t} = \beta(.)\vec{N}$, where C is the boundary vector of curve coordinates, $\vec{N}$ is the outward normal, t is the time duration (magnitude) of the deformation, and β is an arbitrary function. For numerical [18] as well as theoretical [5] reasons, the evolution is embedded in a higher dimension, *i.e.*, C is taken to be the zero level set of an evolving surface ϕ, $\phi(x, y, t) = 0$. Typically ϕ is taken to be the distance transform of the shape. Its evolution is given by $\frac{\partial \phi}{\partial t} + \beta(.)|\nabla\phi| = 0$, and after each iteration the evolving curve C must be recovered. To this end, Figure 7 illustrates the advantages of using GENO over: 1) straightforward discretization, and 2) bilinear interpolation[5]. Observe that in contrast to the other techniques, GENO interpolation deals with the presence of *corners* and also multiple curve segments per cell, both of which can be significant aspects of the evolution.

Acknowledgements The support of grants NSF IRI-9305630, NSF DMS-9211820, ARO DAAH04-94-G-0205, and AFOSR 93-0090, is gratefully acknowledged.

References

[1] L. Alvarez and J.-M. Morel. Formalization and computational aspects of image analysis. *Acta Numerica*, pages 1–59, 1994.

[2] V. Caselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours in image processing. Tech. Report 9210, CEREMADE, 1992.

[5]This method is very similar to that used in [23].

[3] F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. of Num. Anal.*, 29(1):182–193, Feb. 1992.

[4] I. J. Cox, J. B. Kruskal, and D. A. Wallach. Predicting and estimating the accuracy of a subpixel registration algorithm. *IEEE PAMI*, 12(8):721–734.

[5] L. C. Evans and J. Spruck. Motion of level sets by mean curvature I. *J. of Diff. Geometry*, 33(3):635–681, May 91.

[6] A. Harten. ENO schemes with subcell resolution. *J. of Comp. Physics*, 83:148–184, 1989.

[7] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. of Comp. Physics*, 71:231–303, 1987.

[8] P. D. Hyde and L. S. Davis. Subpixel edge estimation. *Pattern Recognition*, 16(4):413–420, 1983.

[9] L. A. Iverson and S. W. Zucker. Logical/linear operators for image curves. *IEEE PAMI*, To Appear, 1995.

[10] B. B. Kimia, A. R. Tannenbaum, and S. W. Zucker. Toward a computational theory of shape: An overview. In *ECCV '90*, France, 1990.

[11] —. Shapes, shocks, and deformations, I: The components of shape and the reaction-diffusion space. *Int. J. of Computer Vision*, To Appear, 1995.

[12] R. Kimmel, K. Siddiqi, B. B. Kimia, and A. Bruckstein. Shape from shading: Level set propagation and viscosity solutions. *Int. J. of Computer Vision*, In Press, 1994.

[13] N. Kiryati and A. Bruckstein. Gray levels can improve the performance of binary image digitizers. *CVGIP: GMIP*, 53(1):31–39, Jan. 91.

[14] J. J. Koenderink and A. J. van Doorn. Dynamic shape. *Biol. Cybernetics*, 53:383–396, 1986.

[15] A. Kumar, A. R. Tannenbaum, and G. J. Balas. Optical flow: A curve evolution approach. *IEEE Trans. on Image Processing*, To Appear, 1994.

[16] R. Malladi, J. A. Sethian, and B. C. Vemuri. Evolutionary fronts for topology-independent shape modelling and recovery. In *ECCV '94*, pp 3–13.

[17] D. Marr and E. Hildreth. Theory of edge detection. Tech. Report MIT AI Memo 518, MIT AI Lab, 1979.

[18] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. of Comp. Physics*, 79:12–49, 1988.

[19] S. Osher and C.-W. Shu. High-order essentially non-oscillatory schemes for Hamilton-Jacobi equations. *SIAM J. of Num. Anal.*, 28:907–922, 1991.

[20] P. Parent and S. W. Zucker. Trace inference, curvature consistency and curve detection. *IEEE PAMI*, 11(8):823–839.

[21] E. J. Pauwels, P. Fiddelaers, and L. J. V. Gool. Geometry-driven evolution and functional optimization for curves. Tech. Report KUL/ESAT/MI2/9414A, Nov. 94.

[22] G. Sapiro and A. Tannenbaum. Affine invariant scale-space. *Int. J. of Computer Vision*, 10:25–44, 1993.

[23] J. A. Sethian and J. Strain. Crystal growth and dendritic solidification. *J. of Comp. Physics*, 98:231–253, 1992.

[24] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. of Comp. Physics*, 77:439–471, 1988.

[25] K. Siddiqi, B. B. Kimia, and C. Shu. Geometric shock-capturing eno schemes for subpixel interpolation, computation and curve evolution. Tech. Report LEMS 142, Feb. 95.

[26] H. Tek and B. B. Kimia. Image segmentation by reaction-diffusion bubbles. In *ICCV '95*.

[27] B. M. ter Haar Romeny, editor. *Geometry-Driven Diffusion in Computer Vision*. Kluwer, September 1994.

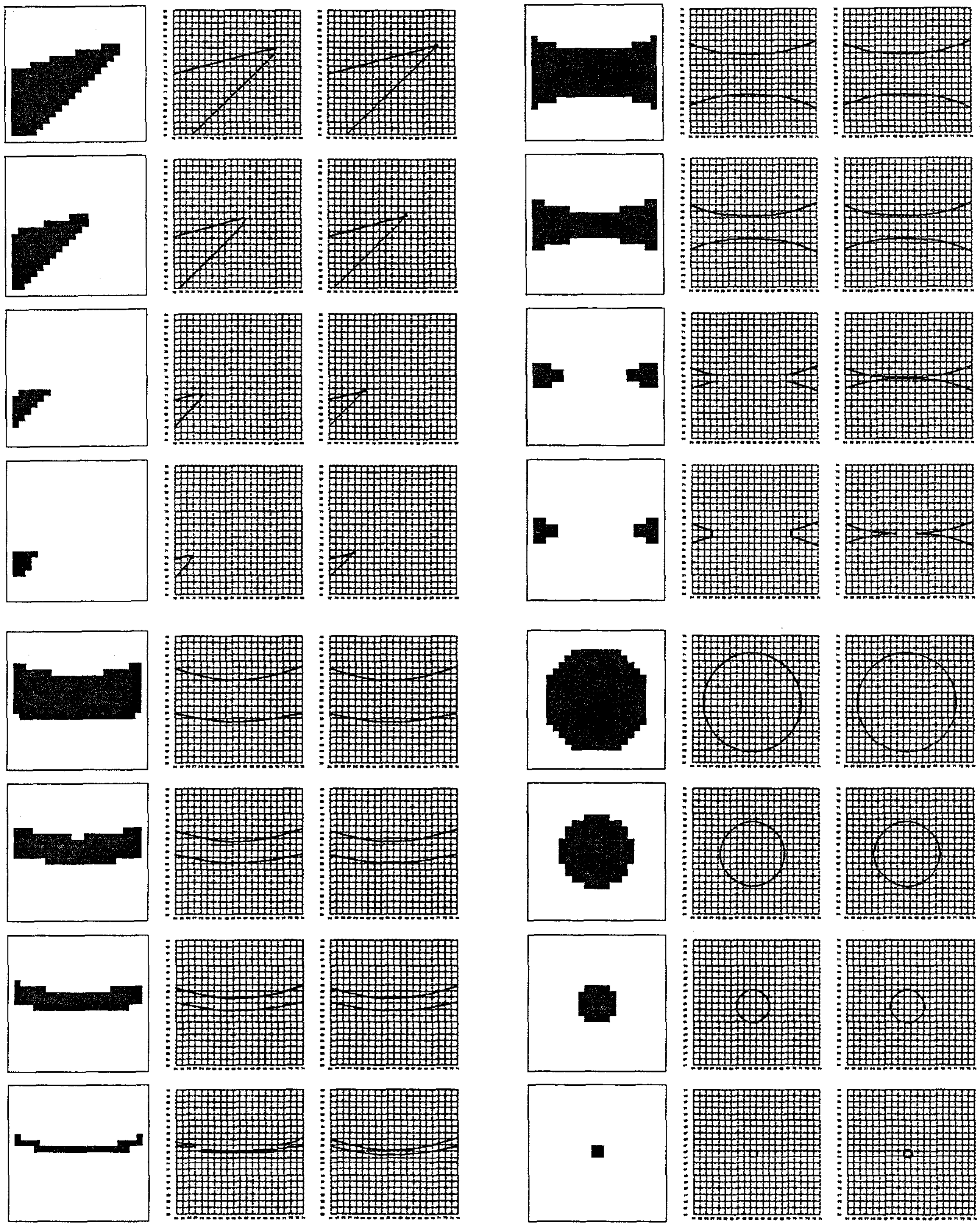

Figure 7: A comparison of interpolation techniques for four evolved shapes. Each row depicts the interior of the shape at the resolution of the grid (left), the bilinear interpolation of the boundary (middle) and the GENO interpolation of the boundary, with detected corners marked with circles (right). Whereas straightforward discretization results in a "jagged" appearance, GENO and bilinear interpolation are comparable in smooth regions. However, in contrast to bilinear interpolation, GENO is able to: i) capture the corner of the evolving triangle with subpixel resolution and without introducing artifacts (first-order shock, TOP LEFT), ii) detect and represent the topological split at the neck, followed by the formation of cusps on either side (second-order shock, TOP RIGHT), iii) represent the collapse of the bend without introducing artifacts (third-order shock, BOTTOM LEFT), and iv) preserve the "circular" shape of the circle even when it becomes very small (fourth-order shock, BOTTOM RIGHT).