GRAPHICAL MODELS AND IMAGE PROCESSING
Vol. 60, No. 2, March, pp. 112–124, 1998
ARTICLE NO. IP970461

Planar Shape Enhancement and Exaggeration

Ami Steiner*

Electrical Engineering Department, Technion, Haifa 32000, Israel

Ron Kimmel†

Lawrence Berkeley National Laboratory and Department of Mathematics, University of California, Berkeley, California 94720

and

Alfred M. Bruckstein‡

Computer Science Department, Technion, Haifa 32000, Israel

Received December 3, 1996; revised June 30, 1997; accepted October 29, 1997

A local smoothing operator applied in the reverse direction is used to obtain planar shape enhancement and exaggeration. Inversion of a smoothing operator is an inherently unstable operation. Therefore, a stable numerical scheme simulating the inverse smoothing effect is introduced. Enhancement is obtained for short time spans of evolution. Carrying the evolution further yields shape exaggeration or caricaturization effect. Introducing attraction forces between the evolving shape and the initial one yields an enhancement process that converges to a steady state. These forces depend on the distance of the evolving curve from the original one and on local properties. Results of applying the unrestrained and restrained evolution on planar shapes, based on a stabilized inverse geometric heat equation, are presented showing enhancement and caricaturization effects. © 1998 Academic Press

1. INTRODUCTION

In this paper we consider possible ways to design an automatic procedure for enhancing and caricaturizing planar shapes. Different caricaturists relate to similar inputs differently and end up with very different caricatures [4]. Yet there is a common trend in all caricatures: special, unusual, or uncommon features in objects are detected and magnified. In [4], Susan Brennan proposed a caricaturization algorithm based on exaggerating the differences between a given object and an "average" one. Her algorithm requires a priori knowledge of a set of items from the input class and the correspondence points between them. Without these reference items, the input item cannot be exaggerated. Even when a set of reference shapes was available, the strict demands on the correspondence between the shapes led the author to represent them by simple polygons which made it practically impossible to generate smooth caricatures (see results in [4, 5]). We here propose an exaggeration algorithm that needs no a priori data and can perform exaggeration on single input images.

To generate an approach for shape exaggeration that does not rely on some average or typical object of each class, we propose to utilize planar curve evolution theory. Further motivation for introducing the feature enhancement and exaggeration for planar shapes via curve evolution comes from a closely related approach for deblurring and feature enhancement in 2D images. Indeed, in the field of image processing one often needs to enhance an image that was blurred or smoothed by some known operator. In this context it is possible, in some cases, to invert the blurring operator, thereby deblurring or reconstructing the original image [13, 7].

As shown in [8, 10, 11], the curvature flow for planar curve evolution, also known as the geometric heat equation (GHE), shrinks any planar curve into a circular point with the fastest rate of shortening the curve's total length. In [25], an affine invariant GHE that shrinks planar curves into ellipses is analyzed. Such evolutions can be considered the ultimate simplification or smoothing flows for shapes.[1] Therefore, for caricature generation based on local properties of the shape itself and for shape enhancement, all that is needed is to invert these flows. An inverse geometric heat equation would have desired specifications of locally exaggerating features of planar curves. But inverting the time direction of the partial differential equation describing the GHE must be done carefully. Involving positive feedback, such an evolution is inherently unstable. In this paper we propose methods for controlling the propagation of curves so that

* E-mail: steiner@tx.technion.ac.il.
† E-mail: ron@math.lbl.gov.
‡ E-mail: freddy@cs.technion.ac.il.

[1] Shape interpolation based on a similar "scale space" was used in [9] for morphing between polygonal shapes.

1077-3169/98 $25.00

stability is achieved. A steady state can be achieved in some models by introducing restraining forces in addition to the evolution forces. This idea resembles the so-called deformable templates relaxation approach [14, 27]. Before introducing the proposed methods, let us briefly review the classical deblurring approach in image processing.

Assume an image $I_0(x, y)$ is smoothed in time by the differential equation

$$I_t(x, y; t) = \nabla^2 I(x, y; t),$$

where $I(x, y; 0) = I_0(x, y)$ is the original image, $I(x, y; \Delta t)$ is the image distorted by the smoothing process at time Δt, $I_t \equiv \partial I/\partial t$, and $\nabla^2 I(x, y; t) \equiv I_{xx} + I_{yy}$ is the Laplacian linear operator operating on I.

Given the blurred image $I(x, y; \Delta t)$ the goal is to reconstruct the original image $I(x, y; 0)$. Using the Taylor expansion one can write

$$I(x, y; 0) = I(x, y; \Delta t) - \Delta t \nabla^2 I(x, y; \Delta t) + \mathcal{O}(\Delta t^2).$$

One-dimensional deblurring of a smoothed step function is illustrated in Fig. 1. We could try to use the same idea for the nonlinear partial differential equation describing the GHE deforming planar curves by

$$C_t = C_{ss}. \tag{1}$$

Here, $C(s, t) : [0, L] \times [0, T] \to \mathbb{R}^2$ is a planar curve deforming in time t according to its second derivative with respect to its arc-length s; e.g., for $C(s, t) = (x(s, t), y(s, t))$ the GHE may be read as $\frac{\partial}{\partial t}(x(s, t), y(s, t)) = \frac{\partial^2}{\partial s^2}(x(s, t), y(s, t))$. Again, by using the Taylor expansion we could propose to reconstruct $C(0)$ from $C(\Delta t)$ via

$$C(0) = C(\Delta t) - \Delta t C_{ss}(\Delta t) + \mathcal{O}(\Delta t^2).$$

In the linear case, one could predict the effect of the heat flow by convolution with a Gaussian kernel in which the variance corresponds to the “time” of the flow. Deblurring the Gaussian blur could just as well be performed in the Fourrier domain by a multiplication of the signal with the corresponding inverse filter. Numerical care should be taken in the linear case as well: The Gaussian tail approaches zero in an exponential rate; computing its inverse introduces numerical difficulty of dividing by zero.

Here we deal with a nonlinear heat equation in which the parameter (arc-length) is not preserved during the evolution. Thus, the effect of the flow on the curve cannot be predicted with one convolution operation. Indeed $\partial^2/\partial s^2$ is a nonlinear operator, and the arc-length parameter s of $C(\Delta t)$ may differ from that of $C(0)$. Since we aim at propagating a curve for long periods of time in order to obtain the exaggeration effects, we should search for an iterative numerical approximation that is easy to control.

The geometric heat equation is known to be a stable process; however, attempts to directly invert Eq. (1), i.e., to propagate

$$C_t = -C_{ss}, \tag{2}$$

result in an unstable, partial differential equation. In [22, 23], the

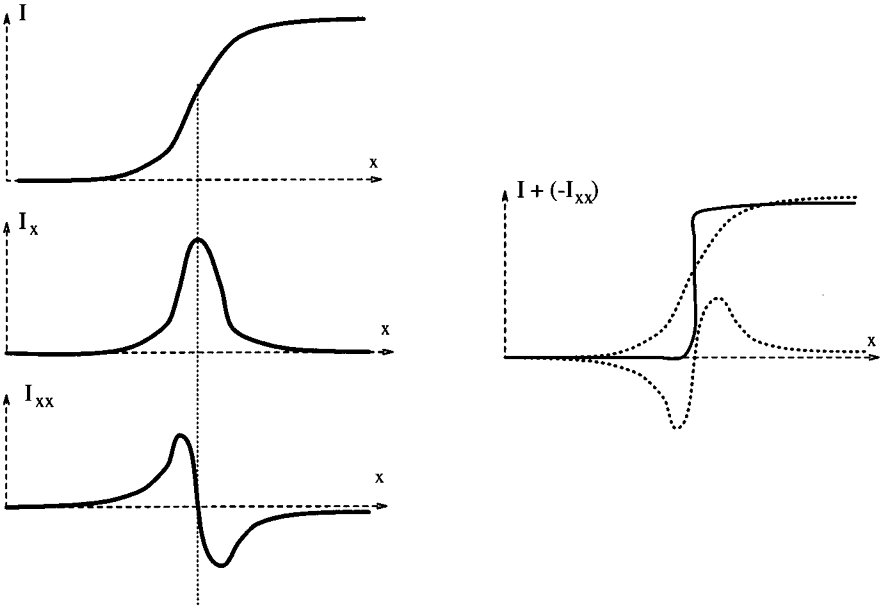

FIG. 1. Deblurring a step function, the $1D$ case.

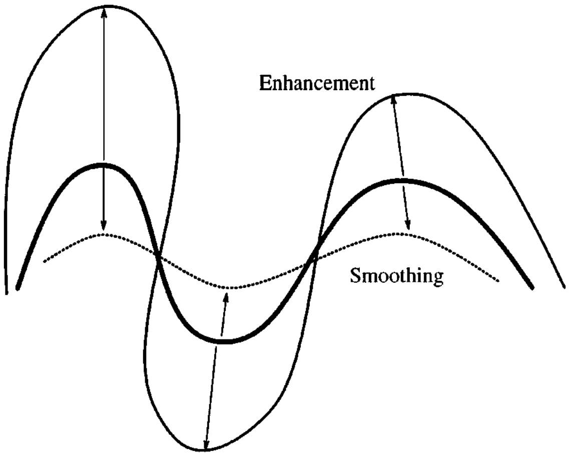

FIG. 2. Smoothing vs enhancement. A smoothing evolution used on the original (bold) curve shifts the high curvature parts inward (dashed). Evolution in the reverse direction will shift it outward yielding an enhanced curve (solid).

inverse smoothing of curves was indeed identified as an unstable evolution. Malladi and Sethian [18] simulated the inverse heat equation to exaggerate the borders of alphanumeric input and thereby improve classification results of handwritten characters. Figure 2 demonstrates the smoothing effects of the GHE and desired enhancement/exaggeration effects to be obtained from the inverse flow.

We introduce two new approaches for simulating the effects of the inverse GHE on outlines of shapes given as polygons or implicitly as level sets of gray-level images on a grid of pixels. Section 2 introduces a stabilized curve enhancement evolution. Careful numerical treatment is suggested to stabilize this inherently unstable process. The proposed evolution enhances the curve for short time, but it does not reach a steady state. To approximate the evolution of continuous curves, we utilize a numerical method for level-set evolution introduced by Osher and Sethian in [19] and elaborated in [3, 15, 20, 24, 26]. For curves approximated by polygons, we use a generalized form of the discrete evolution given in [1] and [2]. In Section 3, we formulate "restrained" evolutions by introducing restraining forces that are added to the original evolution. Results of the application of the unrestrained and restrained evolutions based on the inverse GHE of some planar shapes are demonstrated on pixel grid images as well as on polygonal curves.

2. UNRESTRAINED EVOLUTION

In this section we present two approaches for simulating the inverse smoothing operation for curves and describe the numerical schemes for approximating the process. For continuous curves, the level-set Eulerian formulation [19] is used and a generalized model which controls the intensity of the exaggeration is suggested. For polygonal approximation of curves, two discrete evolutions are applied: the discrete analogue to the reverse GHE evolution and an evolution based on inverting the evolution equations analyzed in [1].

2.1. Simulating the Continuous Case

We embed the curve as a level set of a higher dimensional function and evolve this implicit representation of the curves.

Let us first modify the planar curve evolution via Eq. (2) into a controllable one. Geometrically, Eq. (2) is identical to

$$C_t = -\kappa(s,t)\hat{N}(s,t),$$

where κ is the curvature and $\hat{N}$ is a unit vector normal to the curve. We follow the Osher–Sethian Eulerian formulation [19], creating a bivariate function $\phi : \mathbb{R}^2 \times [0, T] \to \mathbb{R}$ and evolving each of its level sets $\phi(x, y; t) = l$ (also denoted as $C = \phi^{-1}(l)$) according to

$$C_t = F(\kappa(s,t))\hat{N}(s,t). \tag{3}$$

It was shown in [19] (see also Sethian's recent book [23]) that the ϕ-surface evolution equivalent to Eq. (3) is

$$\phi_t = F(\kappa(\phi))\,|\nabla\phi|, \tag{4}$$

where $|\nabla\phi| = \sqrt{\phi_x^2 + \phi_y^2}$.

Let the zero level set correspond to the curve of interest (($l = 0$), i.e., $C = \phi^{-1}(0)$). This means that given the set of points $\{(x, y) \mid \phi(x, y; 0) = 0\}$ that corresponds to the initial curve $C(s, 0)$, then the set of points $\{(x, y) \mid \phi(x, y; t) = 0\}$ of the propagated ϕ function will correspond to the curve $C(s, t)$. Thereby, we can simulate the evolution of C by propagating its implicit representation ϕ.

The reverse GHE is given by

$$\phi_t = -\kappa(x, y; t)|\nabla\phi|, \tag{5}$$

where $C(t) = \phi^{-1}(0)$ for all t. One possible way of choosing the initial $\phi(x, y; 0)$ is the distance from the curve $C(s, 0)$, with negative signs in the interior and positive signs in the exterior of the curve. The curvature of each level set curve $\phi^{-1}(l)$ is given by

$$\kappa(x, y; t) = \kappa(\phi) = \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) = \frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{\left(\phi_x^2 + \phi_y^2\right)^{3/2}}. \tag{6}$$

But the Eulerian formulation, (5), suffers from the inherent instability of the original planar evolution. We therefore need to make some modifications that will enable us to monitor the evolution.

For a small neighborhood near the zero level set $\phi^{-1}(0)$ of a simple convex curve, the outer level sets having low curvature propagate outward slowly while the inner level sets, having higher curvature, propagate outward with higher velocity. Without any numerical control on the function behavior, this causes

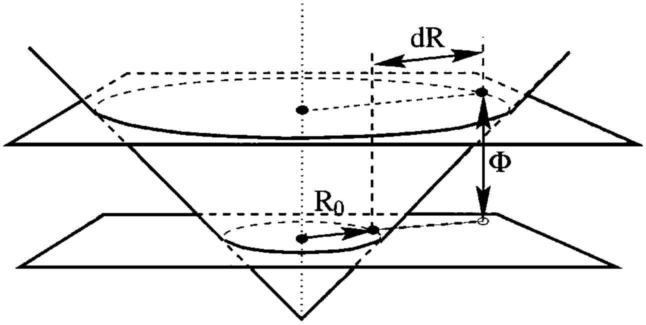

FIG. 3. For a circle of radius R_0, the distance map $\phi(x, y; 0)$ is a cone. The radius of the level set $\phi(x, y; 0) = l$ is $R_0 + dR$. Note that $dR = \phi$, therefore, $\kappa(\phi = l)$ can be modified to allow the same propagation speed as the that of the zero level set.

discontinuities to form in the propagating ϕ. Given ϕ as a distance map, it is possible to change the evolution law given in Eq. (4) so that the distance to the zero level set is preserved along the propagation. Each level set should now evolve in lockstep with the zero level set. This is achieved by first observing that in a distance map, the curvature radius $R = 1/\kappa$ of each level set $\phi^{-1}(l)$ changes linearly with l, as shown in Fig. 3.

Using this observation, Eq. (4) can be modified to

$$\phi_t = F(K)|\nabla\phi|,$$

and the modified inverse geometric heat equation becomes

$$\phi_t = -K|\nabla\phi|, \tag{7}$$

where K is set to be

$$K = K(x, y, t) = \frac{1}{\frac{1}{\kappa(x,y,t)} - \phi(x, y, t)} \tag{8}$$

in the attempt to assign to the point (x, y) the curvature of its closest point on the zero level set. Equation (7) is still not stable numerically. Recalling the explicit representation of the curve, we further modify it by actually fixing the flow field to its initial value throughout the evolution,

$$C_t = -K(x, y, 0)\hat{N}(s, t), \tag{9}$$

where $K(x, y, 0) : \mathbb{R}^2 \to \mathbb{R}$ is the extended curvature of $C(s, 0)$,

$$K(x, y, 0)$$
$$= \{\kappa(s, 0) \mid s \text{ minimizes the distance } |C(s, 0) - (x, y)|\}.$$

This is achieved by setting $t = 0$ in Eq. (8). The curvature of the initial curve is thus extended to the whole plane, so that each point on the plane assigned a value corresponding to the curvature of the initial curve point closest to it. Having a flow field K of propagation velocity fixed in time, we use the Eulerian formulation to implement the curve evolution. The relaxed evolution equation for ϕ equivalent to Eq. (9) is given by

$$\phi_t = -K(x, y, 0)|\nabla\phi|. \tag{10}$$

The above result, obtained by geometrical reasoning, can also be obtained via the following considerations from curve evolution theory. Observe that the level sets of any distance function $\phi(x, y; 0)$ can be described as curves evolving according to the classical "prairie-fire" rule,

$$\begin{cases} C_\tau = \hat{N}(p, \tau) \\ C(p, 0) = \{\text{Curve extracted from } \phi(x, y; 0) \text{ zero level set}\}, \end{cases} \tag{11}$$

where p is an arbitrary parameterization and $\hat{N}(p, \tau)$ is a unit vector normal to the curve at (p, τ). This means that the set of points of each level set of ϕ, i.e., $\{(x, y) \mid \phi(x, y; 0) = l\}$, is given by the set of points of the curve $C(p, \tau = l)$. As shown by Sethian [22], the curvature of $C(p, \tau)$ evolves in this case according to

$$\kappa_\tau(p, \tau) = -\kappa^2(p, \tau), \quad \text{given } \kappa(p, 0),$$

a Riccati equation for which the explicit solution is

$$\kappa(p, \tau) = \frac{\kappa(p, 0)}{1 + \kappa(p, 0)\tau},$$

and therefore we have that

$$\kappa(p, 0) = \frac{\kappa(p, \tau)}{1 - \kappa(p, \tau)\tau}. \tag{12}$$

We next show that for the curve evolution given by Eq. (11), one can trace back the correspondence between any point $(x, y) \in C(p, \tau)$ and its origin $(x', y') \in C(p, 0)$. This can be done using the following Lemma (see [19] or [16]):

LEMMA 1. *For a simple closed curve $C(p, \tau)$ evolving according to*

$$\frac{\partial}{\partial\tau} C(p, \tau) = \hat{N}(p, \tau),$$

where $\hat{N}(p, \tau)$ is a unit vector normal to the curve at (p, τ), the direction of the normal is a "conserved quantity," i.e., it does not change in time prior to shock formation.

Proof. Let us prove that $T_\tau = \frac{\partial}{\partial\tau} T(p, \tau) = 0$ and therefore $\hat{N}_\tau = 0$, where $T(p, \tau)$ is a unit vector tangent to the curve at (p, τ). First we calculate

$$\langle C_{p\tau}, \hat{N}\rangle = \langle C_{p\tau}, C_\tau\rangle = \frac{1}{2}\frac{\partial}{\partial p}\langle C_\tau, C_\tau\rangle$$
$$= \frac{1}{2}\frac{\partial}{\partial p}\langle \hat{N}, \hat{N}\rangle = \frac{1}{2}\frac{\partial}{\partial p} 1 = 0$$

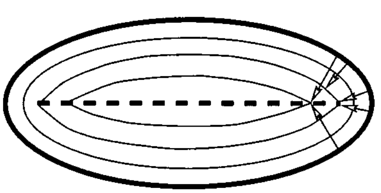

FIG. 4. One way to compute the distance from the (bold) ellipse is to propagate the curve along its normal with constant velocity. The set of curvature singularities (shocks) of the propagating wave is one possible way to define the symmetry axis of the shape (dashed bold line).

which means that $C_{p\tau}$ has only a tangent component. Now

$$\begin{aligned}\frac{\partial}{\partial\tau}T &= \frac{\partial}{\partial\tau}\frac{C_p}{|C_p|} = \frac{|C_p|C_{p\tau} - \frac{\langle C_p, C_{p\tau}\rangle C_p}{|C_p|}}{|C_p|^2}\\ &= \frac{C_{p\tau} - \langle T, C_{p\tau}\rangle T}{|C_p|} = 0. \qquad \blacksquare\end{aligned}$$

To conclude, given $\phi(x, y; 0)$ as a distance map, one can calculate the "extended" curvature map $\kappa(x, y; 0)$ directly from $\phi(x, y; 0)$ using Eq. (6), define $K(x, y; 0)$ using Eq. (8), and evolve the surface according to Eq. (10).

The flow field $K(x, y; 0)$ suffers from discontinuities along symmetry points (at the (x, y) locations of the ridges in the distance map, see Fig. 4).[2] We have therefore smoothed the flow field by convolving it with a Gaussian kernel. This smoothing also suppresses the effects of small perturbations on the propagating curves. A reasonable assumption is that curvature values of the outline contour of shapes given on a pixel grid do not exceed the value of 2, i.e., a curvature radius of half a pixel. Hence, we limit K so that $-|K| < 2$ and set higher values to 2. This limit on $|K|$ also allows us to maintain the CFL condition without forcing very short time steps in the numerical approximation (see [17]).[3]

Propagating ϕ for a short time ΔT, then computing the distance function $D(x, y; \Delta T)$ of each point (x, y) on the plane, from the zero level set of $\phi(x, y; \Delta T)$, it is possible to compute the new $K(x, y; \Delta T)$ and proceed with the propagation:

$$\begin{cases}\phi_t = -K(x, y; \Delta T)|\nabla\phi|\\ \phi(x, y; \Delta T) = D(x, y; \Delta T).\end{cases}$$

Repeating the same procedure at $t = 2\Delta T, 3\Delta T, \ldots$ will enable monitoring of the stability of the evolution process while preserving consistency with the continuous case.

The first step of the numerical approximation involves taking forward finite difference approximation in time and the slope limiters [17, 21, 26] described below. For each time step Δt, we calculate $\Phi_{i,j}^{(n+1)} \approx \phi(i\Delta x, j\Delta y, (n+1)\Delta t)$ to be

$$\Phi_{i,j}^{(n+1)} = \Phi_{i,j}^{n} - \Delta t K(i, j, n)\big(\big(\Delta_x\Phi_{i,j}^n\big)^2 + \big(\Delta_y\Phi_{i,j}^n\big)^2\big)^{1/2}, \quad (13)$$

where

$$\big(\Delta_x\Phi_{i,j}^n\big)^2 = \begin{cases}\max(\min(D_+^x(i, j), 0)^2, \max(D_-^x(i, j), 0)^2)\\ \qquad\qquad\qquad\qquad \text{if } K(i, j) > 0\\ \max(\max(D_+^x(i, j), 0)^2, \min(D_-^x(i, j), 0)^2)\\ \qquad\qquad\qquad\qquad \text{if } K(i, j) \le 0,\end{cases}$$

$$\big(\Delta_y\Phi_{i,j}^n\big)^2 = \begin{cases}\max(\min(D_+^y(i, j), 0)^2, \max(D_-^y(i, j), 0)^2)\\ \qquad\qquad\qquad\qquad \text{if } K(i, j) > 0\\ \max(\max(D_+^y(i, j), 0)^2, \min(D_-^y(i, j), 0)^2)\\ \qquad\qquad\qquad\qquad \text{if } K(i, j) \le 0,\end{cases}$$

and

$$\begin{aligned}D_+^x(i, j) &= (\Phi_{i+1,j} - \Phi_{i,j})/\Delta x\\ D_-^x(i, j) &= (\Phi_{i,j} - \Phi_{i-1,j})/\Delta x\\ D_+^y(i, j) &= (\Phi_{i,j+1} - \Phi_{i,j})/\Delta y\\ D_-^y(i, j) &= (\Phi_{i,j} - \Phi_{i,j-1})/\Delta y.\end{aligned}$$

Equation (13) is a stable numerical approximation for the evolution given by Eq. (10). We use Eq. (13) with $K(i, j, n) = K(i, j, 0)$. For K fixed in time, Φ does not remain a distance map while evolving. We therefore adjust it to be a distance map, with respect to its zero-level-set, after every few iterations. Figure 5 demonstrates some curve exaggerations using this procedure with Eq. (13).

2.2. The Generalized Continuous Case

Equation (4) defines the evolution low to be proportional to the initial curvature $\kappa(\phi)$. In principle, ϕ_t can depend on any parameter derived from the data in $\{\phi(x, y; t)\}$. Thus Eq. (4) can be generalized to

$$\phi_t = f(\phi(x, y; t))|\nabla\phi|.$$

We suggest an evolution low that controls the intensity of enhancement using the following observation: Since $\phi(x, y; t)$ is a distance map with respect to its zero-level-set, it follows that if at time t_0, $\phi(x_0, y_0; t_0) = 0$, then the point (x_0, y_0) is on the curve $C(s, t_0)$ and is exactly a distance of $\phi(x_0, y_0, 0)$ away from the original curve $C(s, 0)$. Replacing $K(x, y; 0)$ in Eq. (10) by a

[2] Generally speaking, 'shocks' are the set of points at which characteristics collide. Let us measure the distance form the bold ellipse curve in Fig. 4 by evolving an equal distance curve with constant velocity in the normal direction. Curvature singularities are formed along the front as it propagates, causing discontinuities at the distance function. This set of points is the 'symmetry axis' often know as the skeleton of the shape. It is the set of points from which there is more than one line of shortest distance connecting to the boundary.

[3] The Courant, Friedrichs, and Lewy (CFL) condition is a necessary stability condition for any numerical scheme: The domain of dependence of each point in the domain of the numerical scheme should include the domain of dependence of the PDE itself.

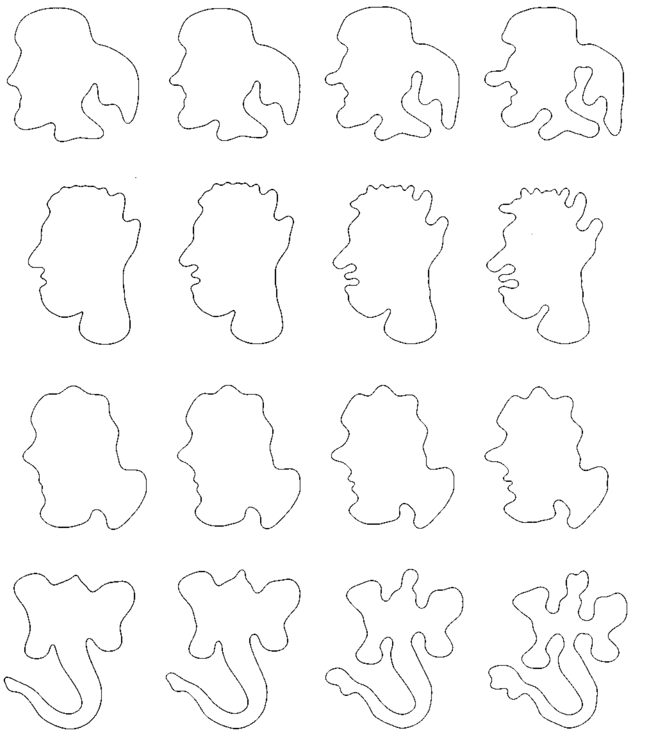

FIG. 5. Exaggeration of several curves (original curves are at the left side). The evolving curves are sampled at times $\lfloor n\Delta t \rfloor = 10, 30,$ and 50.

“modified” $K_G(x, y)$ that makes use of the above observation, the amount of enhancement can be controlled. For example, substituting $K(x, y; 0)$ in Eq. (10) by

$$K_G = K(x, y; 0)(1 + \alpha|\phi(x, y; 0)|^{\beta}), \quad \alpha, \beta \geq 0,$$

increases the enhancement effect, since

$$\frac{K_G(x, y)}{K(x, y; 0)} \geq 1$$

for all (x, y) and it increases as the point (x, y) departs from the initial curve. Figure 6 shows results obtained with the above K_G.

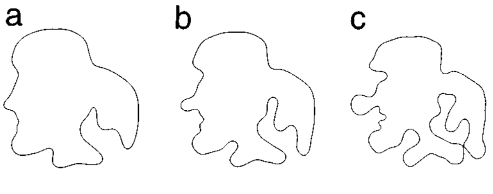

FIG. 6. (a) Original curve, (b) normal exaggeration, and (c) enhanced exaggeration ($\alpha = 0.6$, $\beta = 1$). Both evolutions are shown after $\lfloor n\Delta t \rfloor = 30$ iterations.

2.3. Simulating the Polygonal Case

For curves given in a polygonal form we can use a discrete nonlinear evolution rule analogue of the continuous case. Alternatively, a linear affine invariant evolution similar to the reverse GHE can be introduced by reversing the direction of the discrete smoothing transformation introduced in [6] and discussed in [1, 2]. We will first derive the discrete analogue of the reverse GHE as given by Eq. (3), compare it with the suggested linear evolution, and show the relation between the two.

2.3.1. *Direct Approximation of the Reverse GHE*

Let a polygonal contour be defined by its vertices, $\{P_i\}_{i=1}^{M}$. The discrete evolution analogue to the GHE shifts each vertex $P_i = (x_i, y_i)$, according to

$$p_i^{(n+1)} = P_i^{(n)} + \kappa_i^{(n)} \cdot \hat{N}_i^{(n)}. \tag{14}$$

Exaggeration is achieved by inverting the direction of movement. That is

$$P_i^{(n+1)} = P_i^{(n)} - \kappa_i^{(n)} \cdot \hat{N}_i^{(n)}. \tag{15}$$

Here $P_i^{(n)} = (x_i^{(n)}, y_i^{(n)})$ indicates the location of vertex i after n iterations, $\hat{N}_i^{(n)}$ is a unit normal to the curve at vertex i, defined as a unit vector in the direction of the bisector of that vertex, and $\kappa_i^{(n)}$ is the curvature at vertex i defined in [1] as

$$\kappa_i^{(n)} \overset{def}{=} c \cdot \theta_i^{\text{ext}},$$

where θ_t^{ext} is the external angle between the two edges, for which $P_i^{(n)}$ is a common vertex, and c is a normalization factor. Figure 7 demonstrates polygon exaggeration using Eq. (15).

2.3.2. *Other Affine and Euclidean Approximations to the GHE*

In [2] the following general smoothing operator is proposed,

$$P_i^{(n+1)} = (1 - \alpha)P_i^{(n)} + \alpha\Gamma_{-}P_{i-1}^{(n)} + \alpha\Gamma_{+}P_{i+1}^{(n)}$$

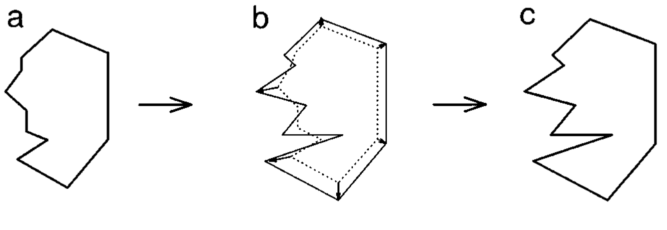

FIG. 7. Polygon exaggeration. For each vertex of the initial polygon (a), the normal $\hat{N}_i^{(n)}$ and curvature $\kappa_i^{(n)}$ are calculated. Then, (b) each vertex is moved in the direction of $\hat{N}_i^{(n)}$ by a step proportional to $\kappa_i^{(n)}$ thus creating a caricature effect (c).

or, in matrix form

$$P^{(n+1)} = MP^{(n)}, \tag{16}$$

where $\{P_i^{(n)}\}_{i=1}^N$ are the polygons N vertices after n iterations and M is an N by N matrix. For $\Gamma_- = \Gamma_+ = \frac{1}{2}$, this operator is circulant, linear, and affine invariant. It evolves an arbitrary closed polygon to a simple concave one, and finally the polygon vanishes to a point having an elliptic polygonal limiting shape (see proof in [1]). For

$$\Gamma_- = \frac{d_+}{d_+ + d_-}, \quad \Gamma_+ = \frac{d_-}{d_+ + d_-},$$

where $d_- = |P_i^{(n)} - P_{i-1}^{(n)}|$ and $d_+ = |P_{i+1}^{(n)} - P_i^{(n)}|$ are the two edge lengths, the evolution is nonlinear and Euclidean invariant. It smoothes shapes but may lead to nonelliptic limiting shapes.

We achieve shape enhancement by inverting Eq. (16). We may calculate the shift from the original polygon to the smoothed one, $(M - I)P^{(n)}$, and then move the vertex in the opposite direction:

$$P^{(n+1)} = (I - (M - I))P^{(n)} = (2I - M)P^{(n)}. \tag{17}$$

The relation between the evolution given by Eq. (14) and the evolution given by Eq. (16) for the case

$$\Gamma_- = \frac{d_+}{d_+ + d_-}, \quad \Gamma_+ = \frac{d_-}{d_+ + d_-}$$

is readily derived. The reverse Eqs. (15) and (17) are similarly related. Defining $\hat{v}_-$ and $\hat{v}_+$ as unit vectors from $P_i^{(n)}$ to $P_{i-1}^{(n)}$ and $P_{i+1}^{(n)}$, respectively, we have

$$\begin{aligned} P_i^{(n+1)} &= (1 - \alpha)P_i^{(n)} + \alpha\Gamma_- P_{i-1}^{(n)} + \alpha\Gamma_+ P_{i+1}^{(n)} \\ &= P_i^{(n)} + \alpha \cdot \frac{d_+}{d_+ + d_-}\left(P_{i-1}^{(n)} - P_i^{(n)}\right) \\ &\quad + \alpha \cdot \frac{d_-}{d_+ + d_-}\left(P_{i+1}^{(n)} - P_i^{(n)}\right) \\ &= P_i^{(n)} + \alpha\frac{d_+ \cdot d_-}{d_+ + d_-}(\hat{v}_- + \hat{v}_+) \\ &= P_i^{(n)} + \mu_i^{(n)} \cdot \hat{N}_i^{(n)}, \end{aligned}$$

where $\mu_i^{(n)}$ is the coefficient multiplying the "normal" vector of vertex i after the nth iteration. Note that $\hat{v}_- + \hat{v}_+$ is indeed a vector in the direction of the bisector of the angle at vertex P_i which was the above defined "normal" to the polygon at P_i. We thus showed that both Eqs. (16) and (14) move the vertices in the direction of the bisector (see Fig. 8), but differ in the amount of movement. Equation (16) propagates vertex i by $\mu_i^{(n)}$ while Eq. (14) propagate it by $\kappa_i^{(n)}$. Figures 9–11 show results obtained by applying the two evolution laws (Eqs. (16) and (17)) in their linear form (i.e., $\Gamma_- = \Gamma_+ = \frac{1}{2}$).

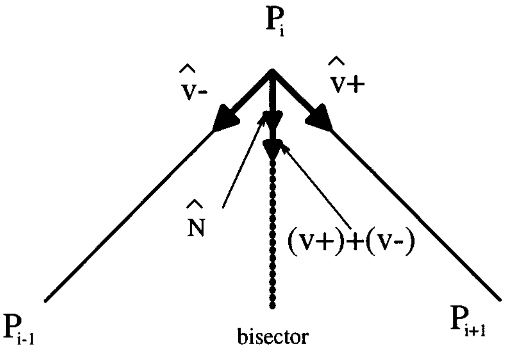

FIG. 8. The vectors $\hat{v}_-$ and $\hat{v}_+$ are defined by the vertices P_{i-1}, P_i, and P_{i+1}. The dashed line indicates the bisector of P_i which coincides with the direction of the vector $\hat{v}_- + \hat{v}_+$.

3. RESTRAINED EVOLUTION

So far we have defined several stable, yet nonconverging, shape enhancing evolutions. Applying the above evolution laws (in both continuous and polygonal simulations) for infinite time spans expands the initial curve to infinity. In this section we define evolution processes that converge to steady states. We introduce "imaginary strings" that connect the original curve with its evolving "image" so that each point on the evolving curve is attracted back to its initial position. For continuous curves, attraction forces between the original and evolving ϕ functions will implicitly restrain the evolution of the embedded curve.

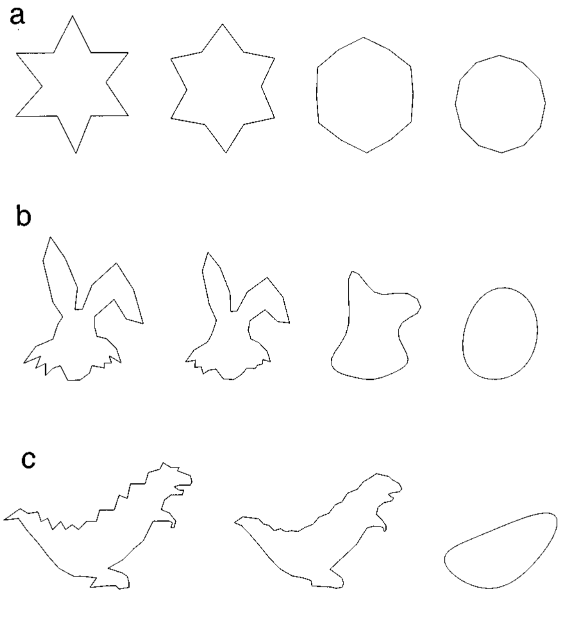

FIG. 9. Some examples of the smoothing evolution. (The scaling is different in each image.) Further iterations result in an infinitesimal polygon of elliptic shape. (a) Star after 0, 2, 10, and 20 iterations, (b) bunny after 0, 3, 30, and 300 iterations, and (c) dino after 0, 5, and 500 iterations.

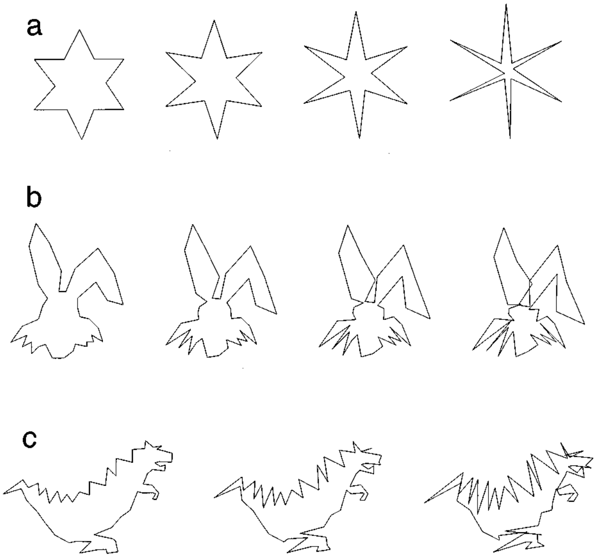

FIG. 10. Some examples of the exaggerating evolution. (The scaling is different in each image.) After a few more iterations the polygons "explode." (a) Star after 0, 2, 4, and 6 iterations, (b) bunny after 0, 4, 6, and 8 iterations, and (c) dino after 3, 6, and 9 iterations.

For polygonal approximations of curves, a set of strings binding each vertex of the original polygon with the corresponding vertex in the evolving polygon will create the desired attraction. The condition for the existence of steady states and an explicit formula for the steady state in the polygonal case are given.

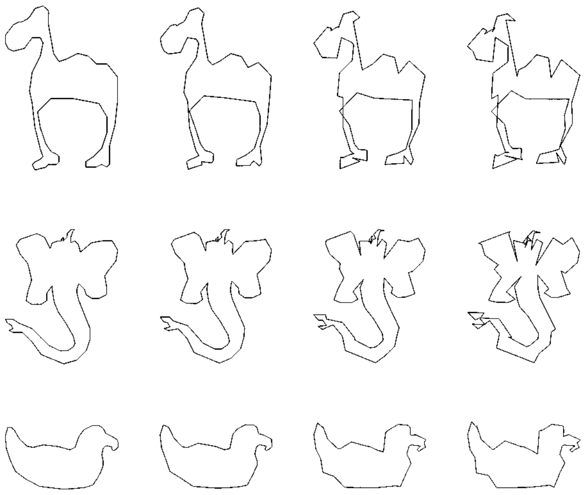

FIG. 11. A few more exaggeration examples.

3.1. Restraining Continuous Curve Evolution

We modify Eq. (10) to include attraction forces between the initial distance map, $\phi(x, y; 0)$, and the evolving one, $\phi(x, y; t)$:

$$\phi_t = -K(x, y; 0)\cdot|\nabla\phi| - \beta\cdot(\phi(x, y; t) - \phi(x, y; 0)); \quad \beta > 0. \tag{18}$$

The first term is the basic evolution force as defined in Eq. (10). The second term defines attraction forces proportional to the deviation of the evolving map from the initial one and directed toward the initial map. This way, parts that do not evolve rapidly are mostly influenced by the basic evolution while rapidly evolving parts are exposed to increasing attraction forces that impede their deviation from the original curve. Initially, for small t's, $\phi(x, y; t) \cong \phi(x, y; 0)$, and Eq. (18) looks like Eq. (10). As t increases, the restraining term becomes more and more influential. For large β, the attraction forces increase rapidly and the deviation of $\phi(x, y; t)$ from its initial shape $\phi(x, y; 0)$ is strongly restrained. Different choices of $\beta = \beta(x, y; t)$ define different evolution laws thereby controlling the evolution of the curve and its steady state (if it exists). For $\beta \to 0$ the attraction element vanishes. As with the unrestrained evolution, $\phi(x, y; t)$ does not remain a distance map while evolving. We therefore adjust it to be a distance map, with respect to its zero-level-set, every few iterations.

The problem of finding when $\phi(x, y; t)$ evolves to a steady state remains to be solved. If however there exists a steady-state solution $\phi^{(\infty)}(x, y)$ to Eq. (18), with ϕ adjusted to remain a distance map with respect to its zero-level-set during the evolution, then it must satisfy

$$\begin{cases} |\nabla\phi| = 1, \\ \phi_t = 0. \end{cases}$$

Thus, from Eq. (18) the steady state can be expressed in terms of the initial conditions:

$$\phi^{(\infty)}(x, y) = \lim_{t\to\infty} \phi(x, y; t) = \phi(x, y; 0) - \frac{K(x, y; 0)}{\beta(x, y)}. \tag{19}$$

In general, β can be a function of any parameters derived from $\phi(x, y; t)$ (such as local curvature). For $\beta \propto 1/$*local curvature*, low-curvature parts of the curve are restrained to their initial position while high-curvature parts depart rapidly from the original curve.

Figure 12 shows an example of restrained evolution using Eq. (18) compared with the unrestrained evolution using Eq. (10).

3.2. Restraining Polygon Evolutions

In the polygonal case, the attraction forces are assumed to act at each vertex. As before, let $\{P_i^0\}_{i=1}^N$ define the initial polygon and let $\{P_i^n\}_{i=1}^N$ be the evolved versions at discrete time steps $n = 1, 2, \ldots,$ the evolution being governed by Eq. (16). We introduce N attracting strings so that string (i) is attached on one

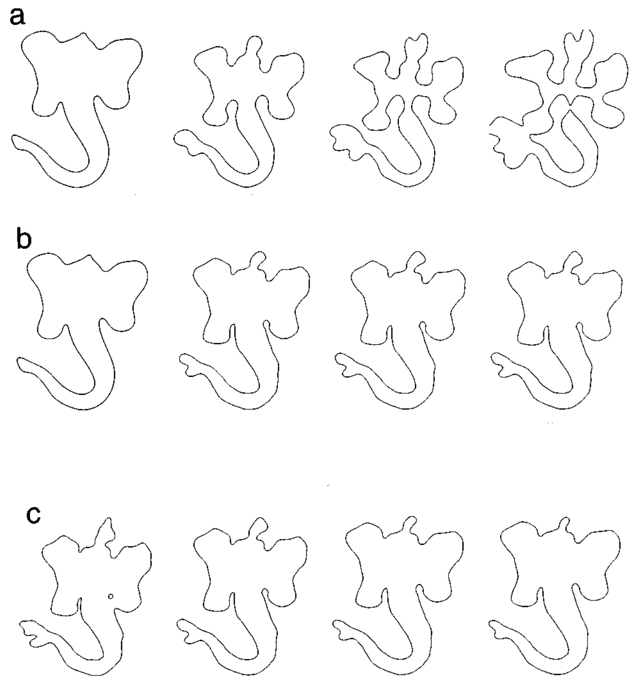

FIG. 12. Restrained versus unrestrained evolutions—the continuous case. (a) Unrestrained continuous evolution after 0, 30, 60, and 120 iterations. (b) Restrained continuous evolution after 0, 30, 60, and 120 iterations, with restraining factor: $\beta_0 = 0.001/K(x, y; t)$. (c) Restrained continuous evolution, all after 120 iterations, with restraining factors: $0.75\beta_0, \beta_0, 2\beta_0$, *and* $4\beta_0$.

side to the evolving vertex P_i^n and on the other side to the initial vertex P_i^0 and has an elasticity constant of β_i. The attraction force will be proportional to the distance of the evolving vertex from its original position. Adding these restraining forces to the smoothing evolution Eq. (16) we arrive at an evolution of the form

$$P_i^{(n+1)} = (1-\alpha)\cdot P_i^{(n)} + \alpha\Gamma_- P_{i-1}^{(n)} + \alpha\Gamma_+ P_{i+1}^{(n)} + \beta_i \cdot \left(P_i^{(0)} - P_i^{(n)}\right),$$

or in a matrix form

$$P^{(n+1)} = MP^{(n)} + B\cdot(P^{(0)} - P^{(n)}). \tag{20}$$

In a similar way, the reversed (exaggerating) evolution is given by

$$P^{(n+1)} = (I - (M - I))P^{(n)} + B\cdot(P^{(0)} - P^{(n)}). \tag{21}$$

The last term in Eqs. (20) and (21) is the restraining force. B is an $N \times N$ diagonal matrix: $B = \mathrm{diag}(\beta_0, \ldots, \beta_N)$. We are particularly interested in diagonal matrices with elements related to the curvature at each vertex. Figure 13 shows an example of restrained evolution using Eq. (21) compared with unrestrained evolution using Eq. (17).

We shall next derive conditions for the convergence of the evolving polygon to a steady-state polygonal shape and explicitly express the steady state for the linear case.

3.2.1. *Terms of Convergence for the Restrained Linear Smoothing Law*

THEOREM 1. *Given the polygon smoothing equation,*

$$P^{(n+1)} = MP^{(n)} + B(P^{(0)} - P^{(n)}), \tag{22}$$

where $P^{(n)}$ is an N-element vector of the polygon's coordinates after n iterations (in complex notation), M is an $N \times N$ circulant matrix with first row defined as $M_{1,(*)} = \{1-\alpha, \alpha/2, 0, \ldots, 0, \alpha/2\}$, and $B = \mathrm{diag}(\beta_0, \ldots, \beta_N)$, where β_i are the restraining

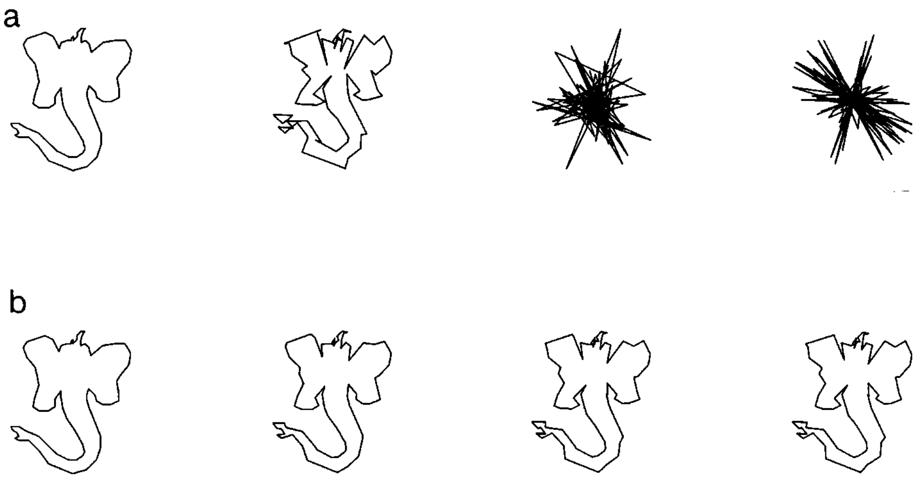

FIG. 13. Restrained versus unrestrained evolutions—the polygonal case. (a) Elephant after 0, 10, 30, and 100 unrestrained iterations (the "exploding" polygons are down scaled). (b) Elephant after 0, 10, 30, and 100 restrained iterations.

coefficients, so that $0 < \beta_i < 2(1-\alpha), \forall\, 0 \le i \le N-1$, *there exists a steady-state polygon defined by*

$$P^{(\infty)} = \lim_{n\to\infty} P^{(n)} = (I - M + B)^{-1} B P^{(0)}. \tag{23}$$

Proof. Following $P^{(n)}$'s evolution we obtain

$$\begin{aligned}
P^{(1)} &= \overbrace{(M - B)}^{Q} P^{(0)} + BP^{(0)} = QP^{(0)} + BP^{(0)}\\
P^{(2)} &= (M - B)P^{(1)} + BP^{(0)} = Q^2 P^{(0)} + BQP^{(0)} + BP^{(0)}\\
&\vdots\\
P^{(n)} &= (M - B)P^{(n-1)} + BP^{(0)} = \left[Q^n + B\sum_{i=0}^{n-1} Q^i \right] P^{(0)}.
\end{aligned} \tag{24}$$

Therefore, $P^{(\infty)}$ exists if and only if the right side of Eq. (24) is finite. We now utilize the following Lemmas, as stated in [12].

LEMMA 2. *Let A be a given n by n matrix. If there is a matrix norm* $|||\cdot|||$ *such that* $|||A||| < 1$, *then* $\lim_{k\to\infty} A^k = 0$; *that is, all the entries of* A^k *tend to zero as* $k \to \infty$ *(see [12], 298).*

LEMMA 3. *An n by n matrix A is invertible if there is a matrix norm* $|||\cdot|||$ *such that* $|||I - A||| < 1$. *If this condition is satisfied, then*

$$A^{-1} = \sum_{k=0}^{\infty} (I - A)^k$$

(see [12], p. 301).

From the above it follows that if there exists a matrix norm $|||\cdot|||$, such that $|||A||| < 1$, then

$$\sum_{l=0}^{\infty} A^l = (I - A)^{-1}$$
$$\lim_{k\to\infty} A^k = 0.$$

Evolving via Eq. (24), $P^{(n)}$ converges to a finite limit polygon, $P^{(\infty)}$, which can be directly computed using the initial conditions. We shall next derive this relation. Using the $|||\cdot|||_\infty$ norm, defined as

$$|||A|||_\infty \overset{def}{=} \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}|,$$

we readily have that

$$|||Q|||_\infty = |||M - B|||_\infty = \max_i (|1 - \alpha - \beta_i| + |\alpha|) < 1.$$

Since $0 \le \alpha \le 1$ we need to satisfy

$$\begin{aligned}
|1 - \alpha - \beta_i| &< 1 - \alpha\\
-1 + \alpha < 1 - \alpha - \beta_i &< 1 - \alpha\\
\Rightarrow 0 < \beta_i &< 2(1 - \alpha) \quad \forall i.
\end{aligned}$$

We have thus found a class of B matrices for which the smoothing evolution converges to a steady-state polygon. If there exists a steady-state polygon $P^{(\infty)}$ then by Eq. (22)

$$\begin{aligned}
&P^{(\infty)} = QP^{(\infty)} + BP^{(0)}\\
\Rightarrow\ &P^{(\infty)} = (I - Q)^{-1} BP^{(0)} = (I - M + B)^{-1} BP^{(0)}.
\end{aligned} \tag{25}$$

Thus, $P^{(\infty)}$ can be calculated explicitly using the initial polygon. ■

3.2.2. *Terms of Convergence for the Restrained Linear Exaggeration Law*

For the exaggeration evolution law, terms for convergence and the steady-state polygon can similarly be defined.

THEOREM 2. *Given the polygon exaggeration evolution,*

$$P^{(n+1)} = (2I - M)P^{(n)} + B(P^{(0)} - P^{(n)}), \tag{26}$$

so that $2\alpha < \beta_i < 2, \forall\, 0 \le i \le N-1$, *there exists a steady-state polygon given by*

$$P^{\infty} = \lim_{n\to\infty} P^{(n)} = (M + B - I)^{-1} BP^{(0)}. \tag{27}$$

Proof. Following $P^{(n)}$ evolution, given by Eq. (26), we obtain

$$\begin{aligned}
P^{(1)} &= \overbrace{(2I - M - B)}^{V} P^{(0)} + BP^{(0)} = VP^{(0)} + BP^{(0)}\\
P^{(2)} &= (2I - M - B)P^{(1)} + BP^{(0)} = V^2 P^{(0)} + BVP^{(0)} + BP^{(0)}\\
&\vdots\\
P^{(n)} &= (2I - M - B)P^{(n-1)} + BP^{(0)} = \left[V^n + B\sum_{i=0}^{n-1} V^i \right] P^{(0)}.
\end{aligned} \tag{28}$$

Using the $|||\cdot|||_\infty$ norm, we have

$$\begin{aligned}
|||V|||_\infty &= |||2I - M - B|||_\infty\\
&= \max_i \left(|2 - \beta_i - (1 - \alpha)| + 2\left|\frac{\alpha}{2}\right| \right) < 1.
\end{aligned}$$

Since $0 \le \alpha \le 1$ we need to satisfy

$$\begin{aligned}
|2 - \beta_i - (1 - \alpha)| &< 1 - \alpha\\
-1 + \alpha < 1 + \alpha - \beta_i &< 1 - \alpha\\
\Rightarrow 2\alpha < \beta_i &< 2 \quad \forall i.
\end{aligned}$$

If this sufficient condition is satisfied, then there exists a steady-state polygon, $P^{(\infty)}$, given by Eq. (26):

$$\begin{aligned}
&P^{(\infty)} = VP^{(\infty)} + BP^{(0)}\\
\Rightarrow\ &P^{(\infty)} = (I - V)^{-1} BP^{(0)} = (M + B - I)^{-1} BP^{(0)}.
\end{aligned} \tag{29}$$

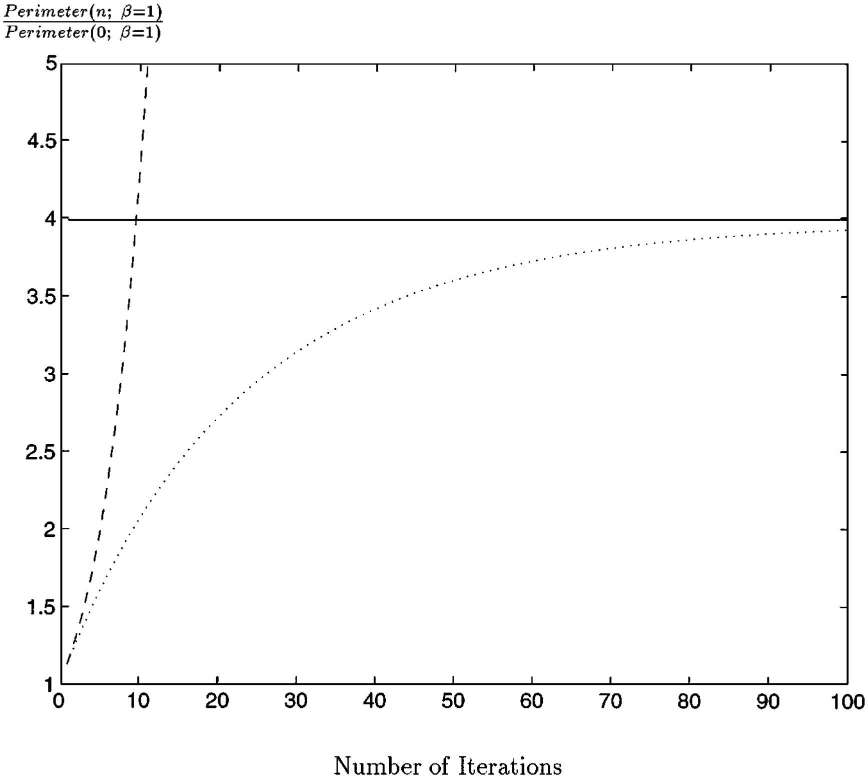

FIG. 14. Example of the restrained exaggeration convergence. The ratio between the perimeter of the evolving polygon and the initial one is shown as a function of the number of iterations. The horizontal line is the perimeter of the steady-state polygon. Using the restrained evolution (dotted), the polygon asymptotically reaches the steady-state perimeter. On the other hand, the unrestrained evolution (dashed) causes the polygon to "explode."

We have thus found sufficient conditions for the existence of a steady-state polygon and expressed it explicitly using the initial polygon. ■

For the linear polygonal case, we have found the terms for convergence to a steady-state polygon, as well as defining the steady-state polygon in terms of the initial polygon and the restraining matrix B. Figures 14 and 15 demonstrate these results. Starting with an initial polygon, we arbitrarily use its perimeter to characterize its evolution in time. When $P^{(\infty)}$ exists, the ratio between the initial polygon's perimeter and that of the steady state characterizes the amount of change the polygon had undergone. If $P^{(\infty)}$ does not exist, the polygon "explodes," sending the perimeter toward infinity. Figure 14 shows the difference between the unrestrained evolution and the restrained one; the latter approaches the steady-state solution as $n \to \infty$. Figure 15 shows the ratio *perimeter* $(P^{(\infty)})/$*perimeter* $(P^{(0)})$ as a function of β resulting from the smoothing evolution with different restraining matrices of the type $B = \beta I$. The continuous line is calculated directly from the steady-state polygon as given by Eq. (27) while the "star"-line use the perimeter of the polygon after 1000 iterations with Eq. (26). For $2\alpha < \beta < 2$, both the explicit evolution and the steady-state polygon yield the same result. Outside the convergence range of β, the evolving polygon explodes and does not reach a steady state. The explicit evolutions yield extremely high perimeter after 1000 iterations. Hence in this case steady state is not reached.

4. CONCLUSIONS

The reverse GHE can be used to enhance features in planar curves. For a given initial curve, known to have been distorted by a smoothing operation (such as blurring), evolution using the reverse GHE for short times can approximately restore it. It is an analog operation to image deblurring [13, 7]. Longer time evolution will further enhance the curve yielding an exaggeration effect. For continuous curves, the level-set Eulerian formulation [19] was utilized and a generalization of the reverse GHE, which enables control over the intensity of exaggeration, was introduced, leading to suppressed or enhanced exaggeration. For polygonal shapes, two different evolution laws were explored, one derived directly from the continuous GHE and the second being a discrete approximation of the GHE given by [2]. The relation between the two was shown and, in the linear case, conditions for convergence to a steady-state polygon were explicitly derived, as well as a closed form formula for the steady-state polygon itself.

Using our approach, planar curves are exaggerated using only their intrinsic features, without a priori knowledge on their classification and with no need for further information. This is an

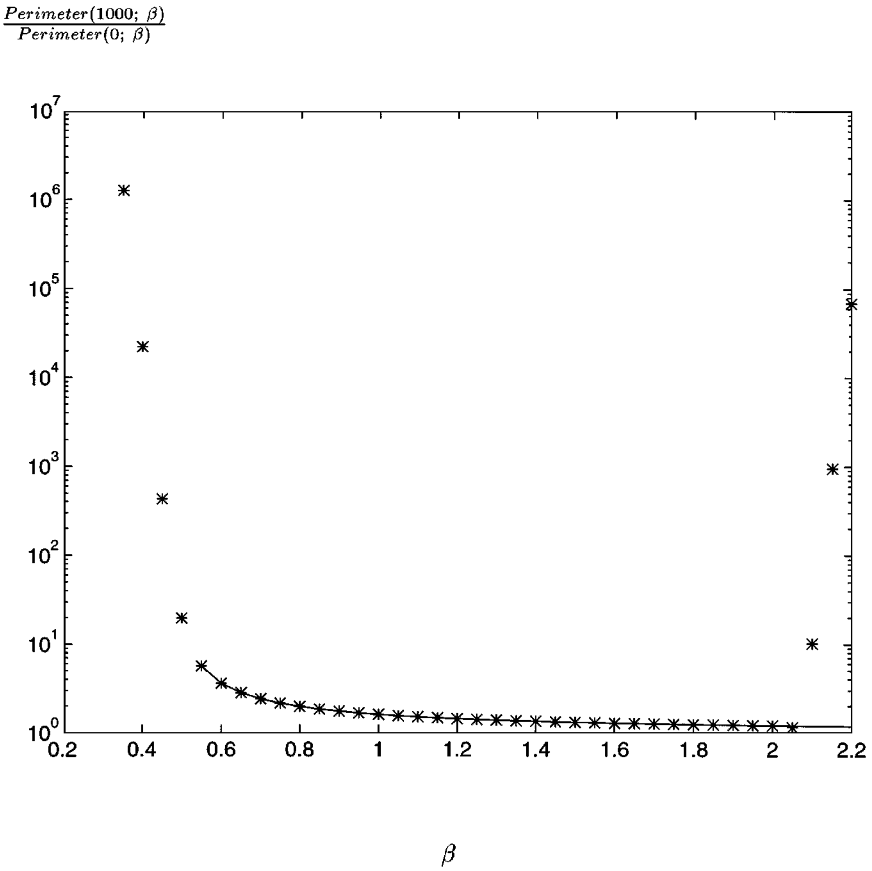

FIG. 15. Restrained exaggeration convergence for different bounding forces. The ratio between the perimeter of the evolving polygon and the initial one is given for different values of $B = \beta I$. The "star" line is a result of 1000 iterations with the restrained evolution law. The other line is the steady-state result. In the convergence range, $2\alpha < \beta < 2\,(\alpha = 0.25)$, the two are identical. Outside the convergence range, the polygon "explodes" to infinite perimeter and does not reach a steady state.

advantage over previously stated exaggeration methods which require such knowledge. We introduced tools to control the parameters of the exaggeration, and in some cases allow the evolution to converge to a well-defined steady-state curve.

ACKNOWLEDGMENTS

We thank Yaacov Farkash (Zeev), a leading Israeli caricaturist, for his helpful caricature demonstrations, Asher Koriat from Haifa University, for his suggestions on caricaturization heuristics, and Michael Elad and Doron Shaked from Israel HP Labs. for suggesting and sharing fruitful ideas with us. This work was supported in part by the Ollendorff Center Research Fund and by the Fund for Promotion of Research at the Technion. RK's work was supported in part by the Applied Mathematics Subprogram of the Office of Energy Research under DE-AC03-76SFOOO98, ONR Grant NOOO14-96-1-0381.

REFERENCES

1. A. M. Bruckstein, G. Sapiro, and D. Shaked, *Evolutions of Planar Polygons*, *Int. J. Pattern Recognition* **9**(6), 1995, 991-1014.
2. A. M. Bruckstein and D. Shaked, *On Projective Invariant Smoothing and Evolutions of Planar Curves and Polygons*, Technion-CIS Report #9328, November 1993. [*J. Math. Imaging Vision* **7**(3), 1997, 225–240.]
3. A. M. Bruckstein, Analyzing and synthesizing images by evolving curves, *Proc. of ICIP'94, Austin Texas, Nov. 1994.*
4. S. E. Brennan, *Caricature Generator*, M.Sc. thesis, MIT, 1982.
5. S. E. Brennan, *Caricature Generator: The Dynamic Exaggeration of Faces by Computer*, Vol. 18, No. 3, pp. 170–178, Leonardo, 1985.
6. M. G. Darboux, Sur un probleme de geometrie elementaire, *Bull. Sci. Math* **2**, 1878, 298–304.
7. D. Gabor, Information theory in electron microscopy, *Lab. Investigat.* **14**(6), 1965, 801–807.
8. M. Gage and R. S. Hamilton, The heat equation shrinking convex plane curves, *J. Differential Geometry* **23**, 1986, 69–96.
9. E. Goldstein and C. Gotsman, Polygon morphing using a multiresolution representation, *Proceedings of Graphics Interface, Quebec City, May 1995*, pp. 247–254.
10. M. Grayson, The heat equation shrinks embedded plane curves to round points, *J. Differential Geometry* **26**, 1987, 285–314.
11. M. Grayson, Shortening embedded curves, *Ann. Math.* **129**, 1989, 71–111.
12. R. A. Horn and C. J. Johnson, *Matrix Analysis*, Cambridge Univ. Press, Cambridge, UK, 1985.
13. R. Hummel, B. B. Kimia, and S. W. Zucker, Deblurring gaussian blur, *Graphical Models Image Process.* **33**, 1997, 66–80.
14. M. Kass, A. Witkin, and D. Torsopolos, Snakes: active contour models, *Int. J. Comput. Vision* 1988, 321–331.

15. R. Kimmel and A. M. Bruckstein, *Shape from Shading via Level-Sets*, CIS Report #9209. Center for Intelligent systems, Technion, June 1992.

16. R. Kimmel and A. M. Bruckstein, Shape offsets via level sets, *CAD* **25**(5), 1993, 154–162.

17. R. J. LeVeque, *Numerical Methods for Conservation Laws*, 2nd ed., Lectures in Mathematics, ETH Zurich, Birkhäuser Verlag, Basel, 1992.

18. R. Malladi and J. A. Sethian, *A Unified Framework for Shape Segmentation, Representation, and Recognition*, LBL-36039, Lawrence Berkely Laboratory, UC-Berkely, August 1994.

19. S. J. Osher and J. A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *J. Comp. Phys.* **79**, 1988, 12–49.

20. S. J. Osher and L. Rudin, Feature-oriented image enhancement using shock filters, *SIAM J. Numerical Anal.* **27**, 1990, 919–940.

21. E. Rouy and A. Tourin, A viscosity solutions approach to shape-from-shading, *SIAM J. Numerical Anal.* **29**(3), 1992, 867–884.

22. J. A. Sethian, Curvature and the evolution of fronts, *Commun. Math. Phys.* **101**, 1985, 487–499.

23. J. A. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*, Cambridge Univ. Press, Cambridge, UK, 1996.

24. J. A. Sethian and J. Strain, Crystal growth and dendritic solidification, *J. Comp. Phys.* **98**, 1991.

25. G. Sapiro and A. Tannenbaum, Affine invariant scale-spaces, *Int. J. Comput. Vision* **11**(1), 1993, 25–44.

26. M. Sussman, P. Smereka, and S. Osher, *A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow*, UCLA Comp. and Applied Mathematics, CAM Report 93-18, June 1993.

27. A. Yuille, D. Cohen, and P. Hallinan, Feature extraction from faces using deformable templates, *CVPR* 1989, 104–109.