Stochastic Differential Equations and Geometric Flows

Gozde Unal, *Student Member, IEEE,* Hamid Krim, *Senior Member, IEEE,* and Anthony Yezzi, *Member, IEEE*

***Abstract*—In recent years, curve evolution, applied to a single contour or to the level sets of an image via partial differential equations, has emerged as an important tool in image processing and computer vision. Curve evolution techniques have been utilized in problems such as image smoothing, segmentation, and shape analysis. We give a local stochastic interpretation of the basic curve smoothing equation, the so called geometric heat equation, and show that this evolution amounts to a tangential diffusion movement of the particles along the contour. Moreover, assuming that a priori information about the shapes of objects in an image is known, we present modifications of the geometric heat equation designed to preserve certain features in these shapes while removing noise. We also show how these new flows may be applied to smooth noisy curves without destroying their larger scale features, in contrast to the original geometric heat flow which tends to circularize any closed curve.**

***Index Terms*—Geometric image and shape flows, stochastic differential equations, nonlinear filtering, shape analysis.**

I. Introduction

IN RECENT YEARS, curve evolution has emerged as an important application of partial differential equations (PDEs) in image processing, computer vision, and computer graphics. Curve evolution techniques have been applied not only to individual curves, for applications such as edge-detection, skeletonization, and shape analysis, but have also been considered for their simultaneous action on the level sets of an image in a number of geometrically based anisotropic smoothing algorithms. Osher and Sethian [1], [2] extended this latter perspective to the treatment of individual curves through a set of algorithms, known as level set methods, which enable the implementation of curve and surface evolution on a fixed grid. These techniques have aided a number of researchers in pushing the application of curve evolution to new limits by providing a simple framework for treating certain types of singularities such as shocks and topological transitions [1], [3].

Much of the research in curve evolution theory has centered around the so called geometric heat equation [4] in which a curve is evolved along the normal direction in proportion to its signed curvature. This flow is well known for its smoothing properties [5]–[7] and the fact that it corresponds to the gradient evolution for arclength (thereby earning the name *curve shortening flow*). Because curvature is a purely geometric quantity (invariant to rotation and translation), curvature-based motion gives rise to a Euclidean invariant scale space [8]–[10], allowing one to trace features in a curve from finer to coarser scales as the evolution proceeds. An affine invariant scale space can be obtained from a related curvature flow which depends upon the cube root of the curvature (see [8], [11], [12]).

When applied to the level sets of an image, these flows have a powerful denoising effect when run for a short amount of time. If run for too long, however, even large scale features will be destroyed. The reason stems from the fact that as the geometric heat flow shrinks any closed curve, the curve becomes more and more circular (elliptical in the case of the affine flow) and will eventually collapse into a single point [4]. It is therefore not always possible to preserve desired features in the shapes of objects (corners for example) if too much evolution is required to remove a significant level of noise. Furthermore, it is not well understood how these curvature-based filters are affected by different noise distributions and when this sort of problem may occur.

To the best of our knowledge, and aside from [13], [14], nonlinear diffusion in the previous literature was discussed from a purely deterministic perspective. In this paper we provide a stochastic formulation of the geometric heat equation and use the resulting insights to develop a new class of curvature-based flows and anisotropic diffusion filters which preserve desired features in the shape of an object. Under these new flows, evolving curves take the limiting form of a polygon (see [15] for evolutions of polygons related to the geometric and affine geometric heat flows, and [16] for evolutions of polygons globally through an electric field concept). The resulting diffusion models may therefore be applied for much longer periods of time without distorting the shapes of polygonal objects in the image, thereby mitigating the tradeoff between noise removal and shape distortion.

Polygonal structures are ubiquitous in images of man-made objects (buildings, roads, vehicles, etc.), which contain many straight lines, often oriented in particular directions (e.g., horizontal and vertical), that come together to form sharp corners. The ability to preserve such distinctive features is not only desirable when filtering an image which contains these types of shapes, but is also important when applying low level smoothing to an extracted shape since such features constitute important and powerful cues for recognizing objects in higher level vision algorithms. We will present both applications in this paper. From

Manuscript received May 29, 2001; revised June 24, 2002. This work was supported in part by AFOSR Grant F49620-98-1-0190, ONR-MURI Grant JHU-72298-S2, and by NCSU School of Engineering. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nasser Kehtarnavaz.

G. Unal was with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA. She is now with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: gozde.unal@ece.gatech.edu)

H. Krim is with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA (e-mail: ahk@eos.ncsu.edu).

A. Yezzi is with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: ayezzi@ece.gatech.edu).

Digital Object Identifier 10.1109/TIP.2002.804568

a dual perspective to our contour-based approach to shape representation, skeletonization approaches may also allow shape analysis without displacement of corners [7], [17]–[21].

In this paper, we develop a new class of curve evolutions, which are obtained by a modification of the geometric heat equation. Given an initial shape in the form of a continuous curve, the class of curve evolution equations we will obtain, deform it into a pre-specified final polygonal shape. The problem of deforming an input shape into a different form has been of interest in various fields such as computer graphics [22].

The contents of this paper are outlined as follows. In Section II, we review some theoretical concepts associated with the curve shortening flow, including its connection to a nonlinear, directional diffusion equation in which image values diffuse locally only along the directions of its edges. In Section III, we provide a stochastic equivalent equation which in turn unveils a new shape/feature-driven flow described in detail in Section IV, which we also believe could offer a variety of applications outside the recognition and classification problems. We conclude with some illustrating and substantiating examples in Section V, and conclusions in Section VI.

II. Background and Formulation

It is known that the low pass Gaussian filter from signal processing can be implemented by evolving the intensities of an image $u_0(x,y)$ via the linear heat equation [10],

$$\begin{aligned} &u(0,x,y) = u_0(x,y), \\ &u_t(t,x,y) = \nabla \cdot (\nabla u(t,x,y)), \quad t > 0 \end{aligned} \tag{1}$$

where the gradient operator, ∇, and the divergence operator, $\nabla\cdot$, involve only the spatial variables x and y. The solution to this equation yields a parameterized family of new images $u(t,x,y)$, where the image at each time $t > 0$ is equivalent to the original image $u_0(x,y) = u(0,x,y)$ convolved with a Gaussian filter of variance $2t$. This equivalence gives rise to a natural generalization of the low pass filter using nonlinear diffusion.

Nonlinear diffusion has a distinct advantage in image processing over linear diffusion in that it may be allowed to handle anisotropies (giving rise to the name *anisotropic diffusion*) in an image. This is particularly important where salient image features are concerned. For example, when the preservation of sharp edges is important, it is natural to consider an anisotropic model which diffuses an image only along the local direction of its edges. One such approach is to consider an image $u(x,y)$ as a collection of iso-intensity contours, or level curves, and to note that at an edge point, the direction of the edge corresponds to the tangent of the iso-intensity contour running through that point. Let η denote the direction normal to the level curve through a given point (the gradient direction), and let ξ denote the tangent direction. We may write these directions in terms of the first derivatives of the image as $\eta = (u_x, u_y)/(\sqrt{{u_x}^2 + {u_y}^2}), \xi = (-u_y, u_x)/(\sqrt{{u_x}^2 + {u_y}^2})$. Since these constitute orthogonal directions, we may exploit the rotational invariance of the Laplacian operator and re-write the linear heat equation in terms of these two variables: $u_t = \nabla \cdot (\nabla u) = u_{\xi\xi} + u_{\eta\eta}$, where $u_{\eta\eta}$ and $u_{\xi\xi}$ denote second-order directional derivatives in the directions of η and ξ respectively. By subtracting the normal diffusion component $u_{\eta\eta}$ from the linear heat equation, which diffuses isotropically, the following anisotropic model, which diffuses along the boundaries of image features but not across them

$$u_t = u_{\xi\xi} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2} \tag{2}$$

is obtained [9]. We may obtain this same equation in a completely different and much more geometric manner by specifying the evolution of each level curve in the image. Let $\mathcal{C}$ denote a particular iso-intensity contour which we will deform over time via the following flow:

$$\mathcal{C}_t = \mathcal{C}_{ss} = \kappa \boldsymbol{N} \tag{3}$$

where s denotes the arclength parameter, κ the Euclidean curvature, and $\boldsymbol{N}$ the inward unit normal. Equation (3), referred to as the *Geometric Heat Equation* (GHE), is well known for its smoothing properties. It has been shown by Grayson [4] that any closed, embedded curve evolving according to (3) will convexify and smoothly shrink to a single point in finite time, becoming more and more circular along the way. This flow is also referred to as the *curve shortening flow* since it corresponds to the gradient (descent) evolution of the arclength functional. See [5]–[7] for a more extensive discussion of the many properties associated with this flow. Because the evolution speed is a function of the curvature at each point on a curve, this flow gives rise to a *Euclidean invariant* scale space (see [8]–[10]) in which finer features are removed first, followed by coarser features, as the curve evolves. A related flow, based upon the affine geometry of the curve, is given by $\mathcal{C}_t = \kappa^{1/3}\boldsymbol{N}$ and shares many of the same properties as the curve shortening flow but gives rise to a more general *affine invariant* scale space (see [8], [11], [12]).

If we apply the geometric heat flow to every single level curve in the image we obtain the same anisotropic diffusion equation that we derived earlier. To see this, note that at time t each level curve $\mathcal{C}^k$ (where the index k distinguishes one level curve from another) is implicitly described by $u(t,x,y) = u^k$ where u^k denotes a particular intensity in the image. Let us choose a parameterization of $\mathcal{C}^k$ so that $\mathcal{C}^k(t,p) = (\mathcal{X}(t,p), \mathcal{Y}(t,p))$ for $p \in [0,1]$ and for all $t \geq 0$. We may then write $u(t, \mathcal{C}^k(t,p)) = u(t, \mathcal{X}(t,p), \mathcal{Y}(t,p)) = u^k$. Differentiating this expression with respect to t yields

$$u_t + \nabla u \cdot \mathcal{C}_t = u_t + \nabla u \cdot (\kappa \boldsymbol{N}) = 0.$$

Note that the inward unit normal and the curvature of each level curve can be expressed as $\boldsymbol{N} = -(\nabla u/||\nabla u||)$ and $\kappa = \nabla \cdot (\nabla u/||\nabla u||)$. This allows us to rewrite the above equation completely in terms of u and its derivatives

$$u_t = \nabla \cdot \left(\frac{\nabla u}{||\nabla u||} \right) ||\nabla u|| = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2} \tag{4}$$

giving us a PDE which is identical to (2).

Equation (4) is also referred to as the geometric heat equation since it comes from applying the previous geometric heat (3) to each level curve of an image u. This double meaning of the term *geometric heat equation* is disambiguated by the context in which the flow is applied (i.e., either to an image or to a curve). In this paper, we will be interested in both cases and will

present directional generalizations of the geometric heat flow which are designed to preserve certain types of features either in a curve or in an image. We first however, reformulate the geometric heat flow from a stochastic point of view, giving new insights into the nature and behavior of this nonlinear diffusion model. It was precisely these insights that led us to the generalizations presented in Section IV.

Remark: Another popular approach to anisotropic diffusion is based upon models first introduced by Perona and Malik in [23]. Since then, these models have received a tremendous amount of attention, as have the models based upon curve evolution theory. Perona and Malik extended the linear heat equation by considering diffusion coefficients which vary with the strength of the gradient at different points of an image. This leads to PDEs of the form $u_t = \nabla \cdot (g(||\nabla u||)\nabla u)$, where g: $\mathbb{R} \rightarrow \mathbb{R}$ is typically a monotonically decreasing function which suppresses diffusion where the gradient is high (near an edge). In general, however, these models are not related to curve evolution theory and are only intended for images, not curves (unless the curve has the form of a graph). As such, we will not attempt to relate the curve evolution models developed in this paper to Perona-Malik models which represent a different perspective on the subject of nonlinear diffusion.

III. Stochastic Formulation of a Geometric Heat Equation

A. Introduction to Ito Diffusion

The diffusion of a particle is usually well modeled by a Stochastic Differential Equation (SDE) which, in turn, represents the underlying microscopic process of an evolution of a pixel or a point. The dynamics of this evolution at a macroscopic level are captured by a PDE, also referred to as a generator (infinitesimal) of the diffusion [13], [14], [24]. Suppose we want to describe the motion of a small particle suspended in a moving liquid, subject to random molecular bombardments. If $\boldsymbol{b}(t,\boldsymbol{x}) \in \mathbb{R}^n$ is the velocity of the fluid at a point $\boldsymbol{x} \in \mathbb{R}^n$ and time $t \in \mathbb{R}^+$, then a widely used mathematical model for the position $\boldsymbol{X}(t)$ of the particle at time t is an SDE of the form

$$d\boldsymbol{X}(t) = \boldsymbol{b}(t,\boldsymbol{X}(t))dt + \boldsymbol{\sigma}(t,\boldsymbol{X}(t))d\boldsymbol{B}(t) \tag{5}$$

where $\boldsymbol{X}(t)$ is an n-dimensional stochastic process, $\boldsymbol{\sigma}(t,\boldsymbol{x}) \in \mathbb{R}^{n\times m}$, and $\boldsymbol{B}(t)$ is an m-dimensional Brownian motion. $\boldsymbol{b}(\cdot,\cdot)$ is called the *drift coefficient*, and $\boldsymbol{\sigma}(\cdot,\cdot)$ is called the *diffusion coefficient*. The first term in this equation corresponds to a nonrandom/deterministic motion, whereas the second term models randomness or noise in the motion.

The solution of such an SDE may be thought of as a mathematical description of the motion of a small particle in a moving fluid, and such stochastic processes are called (Ito) diffusions [24]. For many applications, a second order partial differential operator $\mathcal{A}$ can be associated to an Ito diffusion $\boldsymbol{X}(t)$ given by (5). The basic connection between $\mathcal{A}$ and $\boldsymbol{X}(t)$ is that $\mathcal{A}$ is the generator of the process $\boldsymbol{X}(t)$. If $w(\boldsymbol{x}) \in C_0^2(\mathbb{R}^n)$, (i.e., it is continuous with continuous derivatives up to order 2, and has a compact support), then $\mathcal{A}$ is given in the form

$$\mathcal{A}w = \frac{1}{2}\sum_{i,j}(\boldsymbol{\sigma}\boldsymbol{\sigma}^T)_{i,j}(\boldsymbol{x})\frac{\partial^2 w}{\partial x_i \partial x_j} + \sum_i \boldsymbol{b}_i(\boldsymbol{x})\frac{\partial w}{\partial x_i}. \tag{6}$$

In conjunction with this, the so-called Kolmogorov's backward equation [24], gives a probabilistic solution to linear partial differential equations. Kolmogorov's theorem states that given $\boldsymbol{X}(t) = (X^{(1)}(t), X^{(2)}(t))$, where $E^{\boldsymbol{x}}[\cdot]$ is the expectation operator with respect to the probability law of $\boldsymbol{X}(t)$ starting at the point $\boldsymbol{x}$, and defining $\gamma(t,\boldsymbol{x}) = E^{\boldsymbol{x}}[f(\boldsymbol{X}(t))]$, then there exists an operator $\mathcal{A}$ such that

$$\frac{\partial \gamma}{\partial t} = \mathcal{A}\gamma, \quad t > 0, \quad \boldsymbol{x} \in \mathbb{R}^2,$$
$$\gamma(0,\boldsymbol{x}) = f(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{R}^2.$$

SDEs and stochastic processes, most commonly the Brownian motion, have previously been used in curve and image analysis. Mumford [25] used it to model completion curves of occluded edges, the so-called elastica. By taking the curvature function (of arc length) as a Gaussian process, and the tangent direction on the curve then as a Brownian motion, he derived the probability of the curves that link occluded edges. For a more general situation, e.g., curves in $\mathbb{R}^3$, Mumford used other sorts of stochastic processes such as an Uhlenbeck process to find the elastica. Williams and Jacobs [26], later in their "Stochastic Completion Fields" work, define the same SDE as Mumford's, for a particle's position and the orientation, and through this model of diffusion incorporate the prior assumption that the maximum likelihood path followed by a particle between two positions and directions is a curve of least energy, and solve it by a discrete formulation. Similarly, a Kalman filter which produces estimates of a system as it evolves in time and affected by noise, (which is indeed an SDE written for the system and its observations), was used in [27] for grouping of contour segments. Our use of SDEs is along a different line of thought in that our inspiration starts with a desired effect of a nonlinear filter. Specifically, the theory of SDEs provides us with a microscopical interpretation of the well-studied geometric heat equation, and leads to a new macroscopic description of this equation which in turn is used to develop a new class of curve evolutions or filters.

B. Stochastic Formulation of the Geometric Heat Equation

Let us denote by $\theta(t,\boldsymbol{x})$ the angle between the outward normal to the curve and the x-axis at each spatial point $\boldsymbol{x} = (x,y)$. The outward unit normal $\boldsymbol{N}$ can then be expressed as $\boldsymbol{N} = (\cos(\theta(t,\boldsymbol{x}), \sin\theta(t,\boldsymbol{x}))$, which is re-written in terms of $u(\cdot)$ as $\boldsymbol{N}(t,\boldsymbol{x}) = (u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x}))/\sqrt{u_x(t,\boldsymbol{x})^2 + u_y(t,\boldsymbol{x})^2}$. It follows, $\theta(u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x})) = \tan^{-1}((u_y(t,\boldsymbol{x}))/(u_x(t,\boldsymbol{x})))$. Using these equations, and defining an operator $\mathcal{A}_{\mathrm{GHE}}$ of the form

$$\begin{aligned}\mathcal{A}_{\mathrm{GHE}}u(t,\boldsymbol{x}) = {} & \sin^2\theta(u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x}))u_{xx}(t,\boldsymbol{x}) \\ & - 2\sin\theta(u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x})) \\ & \times \cos\theta(u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x}))u_{xy}(t,\boldsymbol{x}) \\ & + \cos^2\theta(u_x(t,\boldsymbol{x}), u_y(t,\boldsymbol{x}))u_{yy}(t,\boldsymbol{x})\end{aligned} \tag{7}$$

the geometric heat equation (2) can be re-written as

$$\begin{aligned}u(0,\boldsymbol{x}) &= u_o(\boldsymbol{x}), \\ u_t(t,\boldsymbol{x}) &= \mathcal{A}_{\mathrm{GHE}}u(t,\boldsymbol{x})\end{aligned} \tag{8}$$

where $u_o(\boldsymbol{x})$ is the initial level set function.

In light of the foregoing development, a natural question which arises is: *given a PDE which governs a curve shortening flow, can we obtain a corresponding SDE associated with the underlying diffusion?*

The nonlinearity of GHE presents a significant challenge to find a global Ito diffusion which explains the overall microscopical behavior of the system. Our approach here for solving such a nonlinear problem is, to explore the short-time behavior by linearizing around a known (nominal) solution. The perturbation equations so obtained will be linear and hence an approximate solution to the nonlinear problem can be obtained as the nominal value plus the perturbation term. Let us denote by $u^n(t,\boldsymbol{x})$ the solution to (8)

$$\frac{\partial u^n}{\partial t} = \sin^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{xx}^n - \sin\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) \times u_{xy}^n + \cos^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{yy}^n$$

and if we write $u(t,\boldsymbol{x})$ as

$$u(t,\boldsymbol{x}) = u^n(t,\boldsymbol{x}) + \epsilon u(t,\boldsymbol{x})$$

and define the corresponding nominal angle $\theta^n(t,\boldsymbol{x}) = \tan^{-1}((u_y^n(t,\boldsymbol{x}))/(u_x^n(t,\boldsymbol{x})))$, we get a linearized version of the geometric heat equation around a nominal value

$$\begin{aligned}\frac{\partial u(t,\boldsymbol{x})}{\partial t} &\approx \mathcal{A}_{\text{GHElin}} u(t,\boldsymbol{x}) \\ &= \sin^2(\theta^n(\boldsymbol{x}))u_{xx}(t,\boldsymbol{x}) - \sin(2\theta^n(\boldsymbol{x}))u_{xy}(t,\boldsymbol{x}) \\ &\quad + \cos^2(\theta^n(\boldsymbol{x}))u_{yy}(t,\boldsymbol{x}) + c(\boldsymbol{x})\left(-u_y^n(\boldsymbol{x})u_x(t,\boldsymbol{x})\right. \\ &\quad \left. + u_x^n(\boldsymbol{x})u_y(t,\boldsymbol{x})\right) \qquad (9)\end{aligned}$$

where $c(\boldsymbol{x}) = (1)/((u_x^n(\boldsymbol{x}))^2 + (u_y^n(\boldsymbol{x}))^2)[\sin(2\theta^n(\boldsymbol{x}))(u_{xx}^n(\boldsymbol{x}) - u_{yy}^n(\boldsymbol{x})) - \cos(2\theta^n(\boldsymbol{x}))2u_{xy}^n(\boldsymbol{x})]$ (see the Appendix for details of this derivation).

In light of this, we can proceed to state the following.

Proposition 1: The right hand side of the linear PDE in (9) is the generator of the following Ito diffusion satisfying the SDE

$$\begin{pmatrix} d\boldsymbol{X}^{(1)}(t) \\ d\boldsymbol{X}^{(2)}(t)\end{pmatrix} = c(\boldsymbol{X}(t))\begin{pmatrix} -u_y^n(\boldsymbol{X}(t)) \\ u_x^n(\boldsymbol{X}(t))\end{pmatrix} dt + \sqrt{2}\begin{pmatrix} -\sin\theta^n(\boldsymbol{X}(t)) \\ \cos\theta^n(\boldsymbol{X}(t))\end{pmatrix} dB(t). \qquad (10)$$

Proof: The operator $\mathcal{A}_{\text{GHElin}}$ in (9) is first re-written as (see equation at the bottom of the page). where $\mathbf{H}$ is a Hessian operator and $\odot$ is a Hadamard product. The factorization of $(1/2)\boldsymbol{\sigma}\boldsymbol{\sigma}^T$ leads to

$$\boldsymbol{\sigma}(\boldsymbol{X}) = \sqrt{2}\begin{pmatrix} -\sin\theta^n(\boldsymbol{X}) \\ \cos\theta^n(\boldsymbol{X})\end{pmatrix} \quad \text{and by identification,}$$

$$\boldsymbol{b}(\boldsymbol{X}) = c(\boldsymbol{X})\begin{pmatrix} -u_y^n(\boldsymbol{X}) \\ u_x^n(\boldsymbol{X})\end{pmatrix}.$$

Given the functions $\boldsymbol{b}(\boldsymbol{X}(t))$, and $\boldsymbol{\sigma}(\boldsymbol{X}(t))$, we come up with a pair of processes $(\boldsymbol{X}(t), B(t))$ such that the SDE in (10) holds.

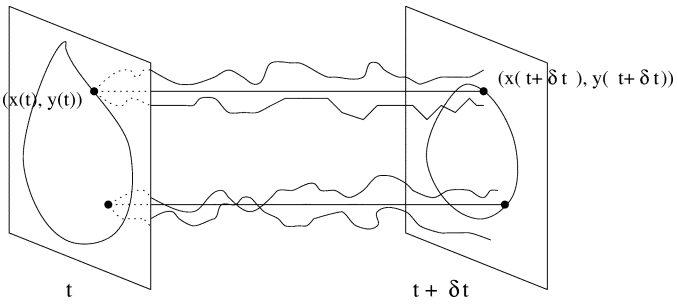


Fig. 1. Points of the zero-level set, i.e., initial contour $(\mathcal{X}(t), \mathcal{Y}(t))$, at time t, is shown on the left. Those points whose sample realizations result in an average value of zero at time $t+\delta t (u(t+\delta t, \boldsymbol{x}) = E^{\boldsymbol{x}}[u(\boldsymbol{X}(t))] = 0)$ form the new contour $(\mathcal{X}(t+\delta t), \mathcal{Y}(t+\delta t))$ (on the right).

In this case, the solution $\boldsymbol{X}(t)$ is called a weak solution, as it does not specify beforehand the explicit representation of the white noise, i.e., the version $B(t)$ of the Brownian motion is not given in advance. ■

Both the drift and diffusion coefficient vectors of this SDE are in the tangent direction of our level curves, which helps us interpret it as a 1-dimensional Ito diffusion on the instantaneous tangent direction $\boldsymbol{T}(u_x^n(t), u_y^n(t))$. A differentiability assumption on $u(t,\boldsymbol{x})$

$$\lim_{\delta t\to 0}\frac{u(t+\delta t,\boldsymbol{x}) - u(t,\boldsymbol{x})}{\delta t} = \frac{\partial u(t,\boldsymbol{x})}{\partial t} \approx \mathcal{A}_{\text{GHElin}} u(t,\boldsymbol{x})$$

is sufficient for a short-time existence of the linearized PDE version of the nonlinear geometric heat equation.

Using Kolmogorov's theorem cited in Section III.A, and assuming that $u(t,\boldsymbol{x})$ and its derivatives are "sufficiently regular" (Lipschitz properties), starting at each time t, the diffusion $\boldsymbol{X}(t)$ in (10) is constructed for each time interval $(t-\delta t, t)$, and may be used to write a Backward Kolmogorov Equation

$$u(t-\delta t,\boldsymbol{x}) = E\{u(t,\boldsymbol{X}(t))/\boldsymbol{X}(t-\delta t) = \boldsymbol{x}\}$$

as a mean value around each pixel dictated by the motion of the constructed diffusion process $\boldsymbol{X}(t)$. This equation can also be written in forward time (since in the small time step δt, the approximate constant-coefficient PDE gives rise to a time-homogeneous diffusion $\boldsymbol{X}(t)$ with $b(\boldsymbol{X}(t))$ and $\boldsymbol{\sigma}(\boldsymbol{X}(t))$ to give way to an averaging process in the tangent direction of a level curve in the course of a forward evolution [i.e., estimate the new pixel value at time t as a mean value of two neighboring pixel values on the tangent at time $(t-\delta t)$ (see Fig. 1)].

This also leads us to infer that locally, we can write a valid diffusion for each time interval $(t-\delta t, t)$

$$d\boldsymbol{X}(t) = \boldsymbol{T}^n(\boldsymbol{X}(t))(c^n(\boldsymbol{X}(t))dt + \sqrt{2}dB(t)) \qquad (11)$$

where $\boldsymbol{T}^n(\boldsymbol{X}(t))$ denotes the known tangent vector at time t, and $c^n(\boldsymbol{X}(t))$ is the known drift coefficient at time t, which is $c^n = (1/\sqrt{(u_x^n)^2 + (u_y^n)^2})[\sin(2\theta^n)(u_{xx}^n - u_{yy}^n) - \cos(2\theta^n)2u_{xy}^n]$. Apart from a drift on $\boldsymbol{T}$, i.e.,

$$d\boldsymbol{X}(t) = \sqrt{2}\boldsymbol{T}^n(\boldsymbol{X}(t))d\tilde{B}(t) \qquad (12)$$

$$\begin{aligned}\mathcal{A}_{\text{GHElin}} &= \boldsymbol{b}^T(\boldsymbol{X})\cdot\nabla + \frac{1}{2}\boldsymbol{\sigma}(\boldsymbol{X})\boldsymbol{\sigma}^T(\boldsymbol{X}) \odot \mathbf{H} \\ &= c(\boldsymbol{X})\begin{pmatrix} -u_y^n(\boldsymbol{X}) \\ u_x^n(\boldsymbol{X})\end{pmatrix}^T \cdot \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y}\end{pmatrix} + \begin{pmatrix} \sin^2\theta^n(\boldsymbol{X}) & -\sin\theta^n(\boldsymbol{X})\cos\theta^n(\boldsymbol{X}) \\ -\sin\theta^n(\boldsymbol{X})\cos\theta^n(\boldsymbol{X}) & \cos^2\theta^n(\boldsymbol{X})\end{pmatrix} \odot \mathbf{H}\end{aligned}$$

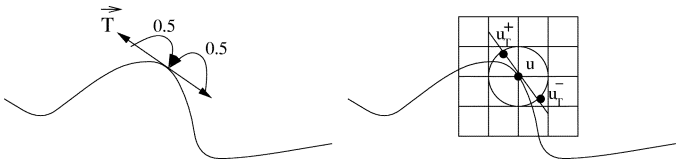


Fig. 2. Symmetric random walk on the tangent direction, and corresponding interpolation on square grid.

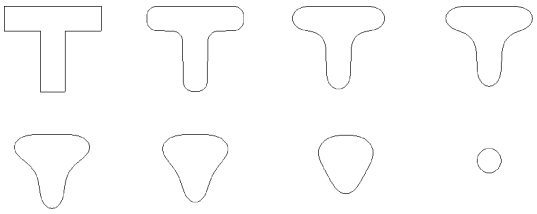


Fig. 3. Generator of symmetric random walk on the tangent direction implemented on the level set function $u(x, y)$. The tangent direction is estimated directly from the level set set function: $\theta_T = \tan^{-1}(-u_x/u_y)$ The level set function is on a 250×250 grid, $\delta t = 0.25$.

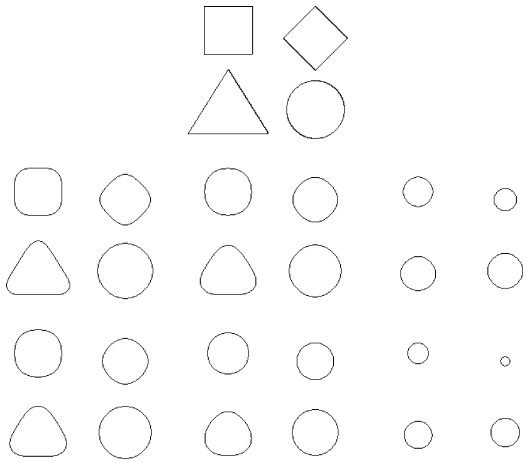


Fig. 4. Middle row: Generator of symmetric random walk on $\boldsymbol{T}$ is shown to produce similar results with those in Bottom Row: Geometric Heat flow. The speeds of the two algorithms are different. The level set function is on a 191×221 grid, $\delta t = 0.25$.

the underlying particle motion can be interpreted as a Brownian Motion (BM) on a local frame in the direction of the tangent. Since BM is an averaging process, this SDE explains that the geometric heat equation smooths iso-intensity contours maximally. On a discrete lattice Brownian motion is captured by a random walk with equally likely (i.e., prob. $1/2$) displacements to u_T^+ and u_T^-. The latters are obtained by a bilinear interpolation around u (in x and y direction and along the tangent, See Fig. 2).

As a result, we can write such an equation as

$$\frac{\partial u}{\partial t} = \frac{1}{2}\Delta_T u \simeq \frac{u_T^+ - u}{2} + \frac{u_T^- - u}{2} \tag{13}$$

where Δ_T is the Laplacian operator in a tangent direction $\boldsymbol{T}$. Summing up, microscopic dynamics of the system captured by the local diffusion $\boldsymbol{X}(t)$ lead to a new macroscopic description of the scenario, i.e., the random walk obtained in terms of the macroscopic variable $u(t, \boldsymbol{x})$.

For simulation purposes, we use a level set methodology [1], which in an Eulerian framework has an advantage of naturally handling topological changes on the level set function. A simulation example where a "T" shape is evolved via a generator of a random walk in a tangent direction, (13), is shown in Fig. 3.

Practical equivalence of GHE and random walk on the tangent direction is tested by several shapes. Another illustrating simulation is shown in Fig. 4.

Our neglecting the drift led to an unbiased random-walk on the tangential direction and is validated by the simulation examples presented above, as the generator of symmetric random-walk implementation results are in agreement with the geometric heat equation implementation. Theoretically, a stronger validation is due to Girsanov theorem (see [24]), which says that if we change the drift coefficient of a given Ito process, then the law of this process does not change dramatically, indeed, the trajectories of the process (distribution) change via the measure change on the trajectories. This theorem involving a change of measure provides us with a means of changing the mean of the process $\boldsymbol{X}(t)$ we obtained in (11), particularly removing the drift and obtaining the process in (12), where only the version of the Brownian motion changes.

This intuitively appealing interpretation of a particle/pixel motion in the process of a diffusion is shown in the next section to be particularly useful and insightful for developing more general and feature/shape adapted flows.

IV. New Class of Flows

The insight gained from the tangential Brownian motion on a curve together with the normal angle $\theta(t, \boldsymbol{x})$, leads to the idea of constraining the Brownian motion at some specific orientation angles at each point $\boldsymbol{x}$. A natural modification of the geometric heat equation, based upon the stochastic framework presented in Section III, is to construct an SDE weighted by a carefully chosen functional $h(\theta^n), (h(\cdot) \in C^\infty(\mathbb{R}^n))$ designed to capture specific features in an image, and we write locally

$$d\boldsymbol{X}(t) = \boldsymbol{T}^n(\boldsymbol{X}(t))(c^n(\boldsymbol{X}(t))h(\theta^n(\boldsymbol{X}(t)))\,dt + \sqrt{2}h(\theta^n(\boldsymbol{X}(t)))dB(t)).$$

Here, again, neglecting the drift motion and concentrating on pure diffusion, the Brownian motion in the tangent direction is being further constrained at some specific orientation values, i.e., at the zeros of the $h(\theta^n)$ function

$$d\boldsymbol{X}(t) \approx \sqrt{2}\boldsymbol{T}^n(\boldsymbol{X}(t))h(\theta^n(\boldsymbol{X}(t)))d\tilde{B}(t)). \tag{14}$$

Constraining the diffusion of particles at points with specified orientations is aimed at extracting desired features of a contour as it is being smoothed. Such models are generated by the following class of PDEs, which directionally modify the geometric heat flow (2), and in this sense, generalize it by making the local generator of the diffusion SDE (14) conceivably arbitrarily selective

$$\frac{\partial u(t, \boldsymbol{x})}{\partial t} = h^2(\theta(t, \boldsymbol{x}))u_{\xi\xi(t, \boldsymbol{x})}. \tag{15}$$

When applied to an image, this flow induces the following curve evolution on each iso-intensity contour $\mathcal{C}$

$$\frac{\partial \mathcal{C}}{\partial t} = h^2(\theta)\kappa \boldsymbol{N}. \tag{16}$$

A. Well-Posedness of the Generalized Model

Proposition 2: The corresponding PDEs (15) are well-posed.

Proof: The geometric heat equation which corresponds to the simplest case of this class with $h^2(\theta(t,\boldsymbol{x}))=1, \forall t, \forall \boldsymbol{x}$, has been shown to be well-posed, and its existence and uniqueness properties may be found in [9], [28], [29]. The operator of the geometric heat equation is given by

$$L[u]=\mathcal{L}[u]-\frac{\partial u}{\partial t}=0 \tag{17}$$

where

$$\begin{aligned}\mathcal{L}[u]&=\sum_{i,j}^{2} a_{ij}\frac{\partial^2 u}{\partial x_i \partial x_j}\\ &=\sin^2\theta u_{xx}-2\sin\theta\cos\theta u_{xy}+\cos^2\theta u_{yy}\end{aligned} \tag{18}$$

is the principal part of the operator L. The matrix of coefficients $[a_{ij}]$ is positive semi-definite with the eigen values 1 and 0. If we multiply this matrix by a positive function, it remains positive semi-definite. Such elliptic-parabolic operators satisfy a maximum principle (see, for example, [30]). In our case, we multiply by a nonnegative function $h^2(\theta)$ which can be made strictly positive by adding a very small number, $\epsilon>0$

$$[h^2(\theta)+\epsilon]\mathcal{L}u>0.$$

This results in a family of nonlinear parabolic equations each of which satisfies a strong maximum principle. Our operator is obtained in the limit as $\epsilon\to 0$. ■

B. Polygon Yielding Diffusions

The geometric heat equation is a rotationally invariant flow which evolves, as mentioned earlier, any shape into a circle [4]. It is the only rotationally invariant shape evolution in Euclidean space. If we wish to capture more general shapes (triangles, squares, etc., ...) it is only then natural to consider flows which are not rotationally invariant. Such a class is given by the form (16) when $h(\theta)$ is chosen to be other than a constant. If we are particularly interested in polygons, we may consider periodic functions (whose periodicity is dictated by the desired number of vertices) such as

$$h(\theta)=\begin{cases}\cos(n\theta)\\ \sin(n\theta)\end{cases} \tag{19}$$

leading to curve evolution equations of the form

$$\frac{\partial\mathcal{C}}{\partial t}=\cos^2(n\theta)\kappa\boldsymbol{N}\quad\text{or}\quad\frac{\partial\mathcal{C}}{\partial t}=\sin^2(n\theta)\kappa\boldsymbol{N}. \tag{20}$$

If we apply (20) to a convex shape, there will be $2n$ points on the curve which do not diffuse (corresponding to the zeros of $\cos(n\theta)$ or $\sin(n\theta)$) at equally separated rotations of the unit normal $\boldsymbol{N}$. As the unit normal moves further and further away from these angles, the diffusion increases. It hence makes sense to expect a curve to develop vertices (points of maximal curvature) at these points.

Lemma 1: The angle of a unit normal does not change at points where the chosen function $h^2(\theta)$ vanishes. Those points, in turn, remain fixed for a short-time, and their speed remains at zero.

Proof: Assume that a family of curves $\mathcal{C}(t,p)$, where p is any parameter along the curve, evolves according to the evolution equation

$$\frac{\partial\mathcal{C}}{\partial t}=\alpha(t,p)\boldsymbol{T}+\beta(t,p)\boldsymbol{N}. \tag{21}$$

The evolution equation for the angle of the unit normal is given in [5] as

$$\frac{\partial\theta}{\partial t}=\frac{-1}{g}[\beta_p-\alpha\kappa g]$$

where $g=\|\mathcal{C}_p\|=\sqrt{\mathcal{X}_p^2+\mathcal{Y}_p^2}$ is the length along the curve (metric). If we consider the case $\alpha=0$ and $\beta=-h^2(\theta)\kappa$ (following the convention used by the authors in [5]), which corresponds to the form of the deformation we proposed, the orientation evolution is governed by

$$\begin{aligned}\frac{\partial\theta}{\partial t}&=\frac{1}{g}[(h^2(\theta)\kappa)_p]\\ &=\frac{1}{g}\{2h(\theta)(h(\theta))_p\kappa+h^2(\theta)\kappa_p\}\\ &=\frac{1}{g}h(\theta)\{2(h(\theta))_p\kappa+h(\theta)\kappa_p\}.\end{aligned} \tag{22}$$

Notice that $(\partial\theta/\partial t)=0$ for those points at which $h(\theta)=0$. ■

We note that in [5], the orientation of a curve is defined as the angle subtended by the tangent and the x-axis, whereas here we define θ as the angle subtended by the normal and the x-axis. There is, however, a complete equivalence in so far as the evolution equation of the angle θ is concerned.

In light of the above development, we can thus state that the zeros of the function $h(\theta)$ lead to fixed end points of curve segments. Fixing two end points, say a_1 and a_2, we examine the evolution of curvature, whose general form is given by (in [5])

$$\frac{\partial\kappa}{\partial t}=-\frac{\partial^2\beta}{\partial s^2}+\alpha\frac{\partial\kappa}{\partial s}-\beta\kappa^2$$

where s is the arc-length parameter along the curve. When substituting $\alpha=0$ and $\beta=-h^2(\theta)\kappa$ into this equation, we have

$$\begin{aligned}\frac{\partial\kappa}{\partial t}&=[h^2(\theta)\kappa]_{ss}+h^2(\theta)\kappa^3\\ \frac{\partial\kappa}{\partial t}&=[(h^2(\theta))_{ss}\kappa+2(h^2(\theta))_s\kappa_s+h^2(\theta)\kappa_{ss}]+h^2(\theta)\kappa^3\\ \frac{\partial\kappa}{\partial t}&=\underbrace{h^2(\theta)\kappa_{ss}}_{\text{diffusion term}}+\underbrace{h^2(\theta)\kappa^3+(h^2(\theta))_{ss}\kappa+2(h^2(\theta))_s\kappa_s}_{\text{reaction term}}.\end{aligned}$$

This clearly demonstrates that a regularizing diffusion takes place, since the multiplicative factor $h^2(\theta)$ never becomes negative (which would result in an ill-posed backward diffusion). In addition, we have the reaction term which is composed of functions of κ,κ^3, and κ_s.

We have hence shown that with fixed end points, a particular curve segment subject to the new evolution equation for the curvature shown above, results in a straight line as a final solution.

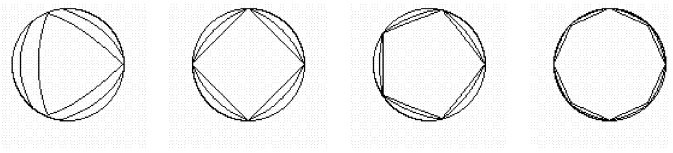
Now, we can state a theorem where we put our argument of convergence to regular polygons.

Theorem 1: A convex curve $\mathcal{C}$ subject to the evolution $\mathcal{C}_t=h^2(\theta)\kappa\boldsymbol{N}$ will converge to an M-sided, regular polygon whose M vertices will be formed at those vanishing points of the function $h^2(\theta)$.

The proof of this theorem can be completed using the arclength evolution equation

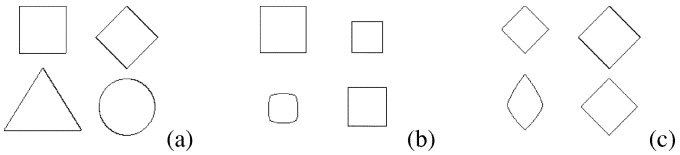
$$\frac{\partial L(\mathcal{C})}{\partial t}=-\int_{a_1}^{a_2}\kappa^2h^2(\theta)\,ds$$

where ds denotes the incremental arclength of $\mathcal{C}$. Since the integrand is strictly positive, we see that a curve will continue to shrink



$h(\theta) = \sin(1.5\theta)$ $h(\theta) = \sin(2\theta)$ $h(\theta) = \sin(2.5\theta)$ $h(\theta) = \sin(4\theta)$

Fig. 5. Morphing of a circle into different shapes by the given flows is demonstrated.



(a) (b) (c)

Fig. 6. (a) Initial set of shapes. (b) Flow $\mathcal{C}_t = \cos^2(2\theta)\kappa \boldsymbol{N}$. (c) Flow $\mathcal{C}_t = \sin^2(2\theta)\kappa N$.

until curvature vanishes, that is the curve segment converges to a straight line between the end points a_1 and a_2. This, in conjunction with the above lemma completes the proof of the theorem.[1]

V. Experimental Results

A. Examples in Polygonization

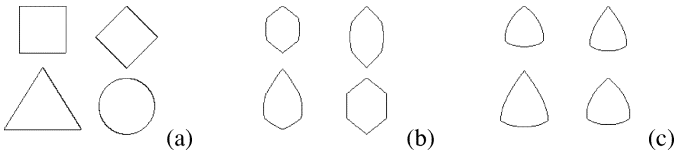
To substantiate the stated theorem, and to intuitively illustrate our flows, we next present simulation results. In our experiments with contours, we use the narrowband implementation of the level set method developed in [32]. The time step is $\delta t = 0.2$. Starting with a circular shape, the flow $\mathcal{C}_t = h^2(\theta)\kappa \boldsymbol{N}$ evolves it toward a specific polygon, i.e., it produces an *n-gone* shape depending on the specific function $h(\theta)$. Several examples on morphing of a circle into different shapes are shown in Fig. 5.

This is one potential application of the proposed flows in computer graphics, where the ability to morph a shape into a known other shape with an efficient algorithm is required for numerous applications. In addition to illustrating the propagation of the proposed flows in several snapshots, Fig. 5 also provides a quantitative and an objective means for characterizing the performance of these algorithms in preserving corners. It can be observed that the position of the desired feature locations, i.e., the orientations at which the vertices of the final polygon are to be formed, are well preserved. There may inevitably be one-to-two pixel displacements due to numerical implementation effects, on account of the finite precision of the computations and the finite resolution of the grid (which affects almost all image processing algorithms).

The flows are also applied to a variety of convex shapes shown in part (a) of each figure: Figs. 6 and 7. In Fig. 6, the shapes in part (b) were obtained by using $h^2(\theta) = \cos^2(2\theta)$ via the following curve evolution:

$$\mathcal{C}_t = \cos^2(2\theta)\kappa \boldsymbol{N} \tag{23}$$

while the shapes in part (c) were obtained using $h^2(\theta) = \sin^2(2\theta)$. In both cases, we expect to obtain four-sided and regular polygons. The zeros of $\cos(2\theta)$ and the zeros of $\sin(2\theta)$ are however 45 degrees out of phase. As such, we see the evolved shapes in part (b) taking the form of a square, whereas the evolved shapes in part (c) take the form of a diamond, corresponding to a 45 degree rotation of the shapes in part (b). In Fig. 7, we see the effect of using different periods. The shapes in part (b) are obtained using $\mathcal{C}_t = \cos^2(3\theta)\kappa \boldsymbol{N}$, while the shapes in part (c) are obtained using $\mathcal{C}_t = \sin^2(1.5(\theta - \pi/2))\kappa \boldsymbol{N}$. In the first case, we expect 6 vertices, and in the second case we expect 3 vertices. Our expectations match the results shown in part (b) and (c), where we observe hexagonal and triangular shapes, respectively.



(a) (b) (c)

Fig. 7. (a) Initial set of shapes. (b) Flow $\mathcal{C}_t = \cos^2(3\theta)\kappa N$, which tends to produce hexagons. (c) Flow $\mathcal{C}_t = \sin^2(1.5(\theta - \pi/2))\kappa N$, which tends to produce triangle-like shapes.

These two figures also suggest an important potential application of the proposed flows, namely shape recognition. A typical scenario, would consist of an unknown shape, which is evolved with one of the polygonizing flows whose parameters are known a priori. The evolution which subsequently results in the least change in the input shape reveals the closest shape category the test shape may belong to.

Recall that we may also apply these flows to the level sets of an image in the same manner that the geometric heat equation may be applied. This gives rise to a family of anisotropic smoothing filters which, unlike the geometric heat equation, are not rotationally invariant. This feature can be useful in smoothing noisy images where corners and edges are priorly known to have certain orientations. These diffusions are effected by PDEs of the following form:

$$u_t = h^2(\theta)\nabla \cdot \left(\frac{\nabla u}{||\nabla u||}\right) ||\nabla u||. \tag{24}$$

Note that the trigonometric expressions we have considered for $h^2(\theta)$ can be written in terms of the first derivatives of u, for example

$$\cos^2(2\theta) = \frac{\left(u_x^2 - u_y^2\right)^2}{\left(u_x^2 + u_y^2\right)^2} \quad \text{and} \quad \sin^2(2\theta) = \frac{(2u_x u_y)^2}{\left(u_x^2 + u_y^2\right)^2}$$

allowing one to implement the PDE without having to compute the orientation of the unit normal to each level curve. Note that these expressions involve only first order derivatives and therefore do not alter the quasilinear structure of these second order flows.

The intended application of the proposed flows in this paper, i.e., the smoothing of structures along the orientation of salient lines in both curves and images will be illustrated in the next two subsections, respectively.

B. Examples in Feature-Preservation

Feature-preserving properties as well as polygonal approximation properties of the proposed flows will be demonstrated in this section. We illustrate the idea of capturing different polygonal features of shapes by our proposed flows on the following examples.

[1]Note added in Proof: We recently found out per Dr. Osher at UCLA that Peng *et al.* [31], have also independently proposed flows similar to those described in this paper, albeit from a totally different perspective, and with a convective trend.

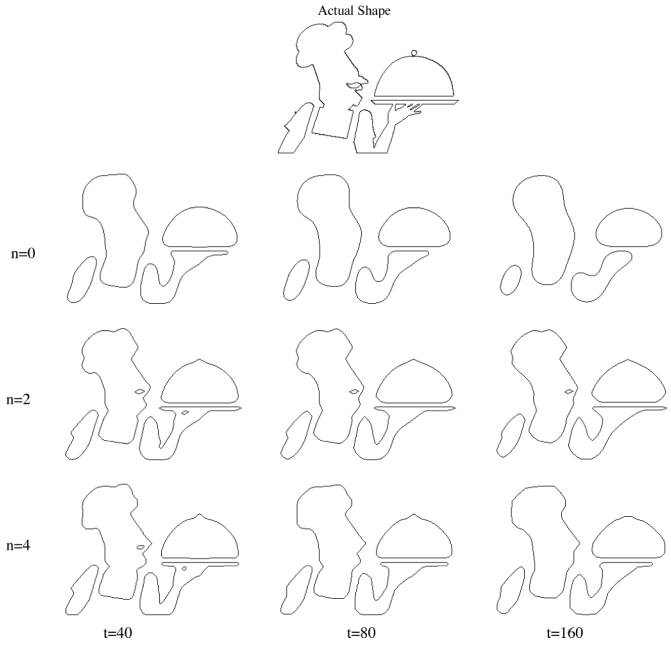


Fig. 8. Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \boldsymbol{N}$, 2nd row: $\mathcal{C}_t = \sin^2(2\theta)\kappa \boldsymbol{N}$, 3rd row: $\mathcal{C}_t = \cos^2(4\theta)\kappa \boldsymbol{N}$.

The first example is a "chef" shape with both round and polygonal features as shown in Fig. 8. The geometric heat flow $\mathcal{C}_t = \kappa \boldsymbol{N}, (n = 0)$, evolves these features into circles as shown in the second row of Fig. 8 for time points $t = 40, 80, 160$. Particularly, at $t = 160$, most parts of the shape turns into incomprehensible blob-like structures. In contrast to this, polygonal features of the chef like his nose, and tray, are preserved by the flow $\mathcal{C}_t = \sin^2(2\theta)\kappa \boldsymbol{N}, (n = 2)$, which favors diamond-like structures (see third row of Fig. 8 for $t = 40, 80, 160$). Similarly, the flow $\mathcal{C}_t = \cos^2(4\theta)\kappa \boldsymbol{N}, (n = 4)$, favors octagonal features as shown on the fourth row of Fig. 8, which is observed at chef's hat at all time points $t = 40, 80, 160$. The regularity of these flows is readily observed through the smoothness of the resulting shapes. When we view each row from left to right, we observe a progression from finer to coarser scale. The scale-spaces produced by our modified flows in the last two rows are visually more pleasing since corners are preserved, whereas in the row above we see them smoothed away by the pure geometric heat flow.

The second shape example is a fish which contains some fine detail structures as well as coarse features (Fig. 9). The second row shows the result of the geometric heat flow $\mathcal{C}_t = \kappa \boldsymbol{N}, (n = 0)$, which smoothes away not only fine features but some coarse features as well (the fins for example). The results of the flow, $\mathcal{C}_t = \cos^2(2\theta)\kappa \boldsymbol{N}, (n = 2)$, are shown in the third row of Fig. 9. In this case, rectangular features are preserved for longer periods throughout the evolution. Finally, the flow, $\mathcal{C}_t = \sin^2(4\theta)\kappa \boldsymbol{N}, (n = 4)$, is depicted in the last row, preserving octagonal features as shown in the nose and the dorsal fins.

In the third example, we start with a noisy shape at time $t = 0$ shown in Fig. 10. This shape is evolved with the geometric heat flow $\mathcal{C}_t = \kappa \boldsymbol{N}, (n = 0)$, the flow $\mathcal{C}_t = \sin^2(1.5(\theta - \pi/2))\kappa \boldsymbol{N}, (n = 1.5)$, the flow $\mathcal{C}_t = \sin^2(2\theta)\kappa \boldsymbol{N}, (n = 2)$, and the flow $\mathcal{C}_t = \cos^2(2.5(\theta - \pi/2))\kappa \boldsymbol{N}, (n = 2.5)$, as shown in

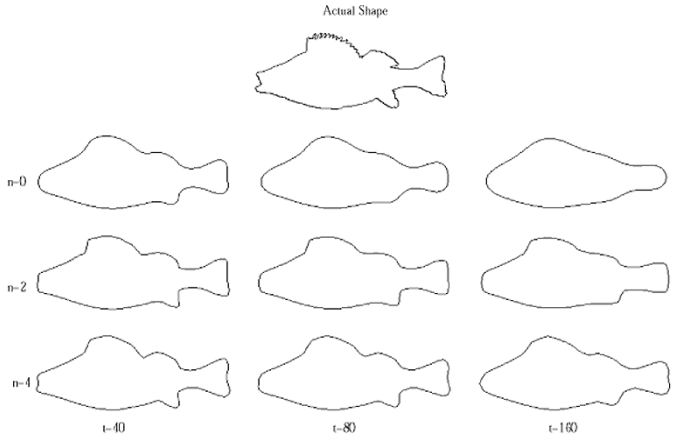


Fig. 9. Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \boldsymbol{N}$, 2nd row: $\mathcal{C}_t = \cos^2(2\theta)\kappa N$, 3rd row: $\mathcal{C}_t = \sin^2(4\theta)\kappa N$.

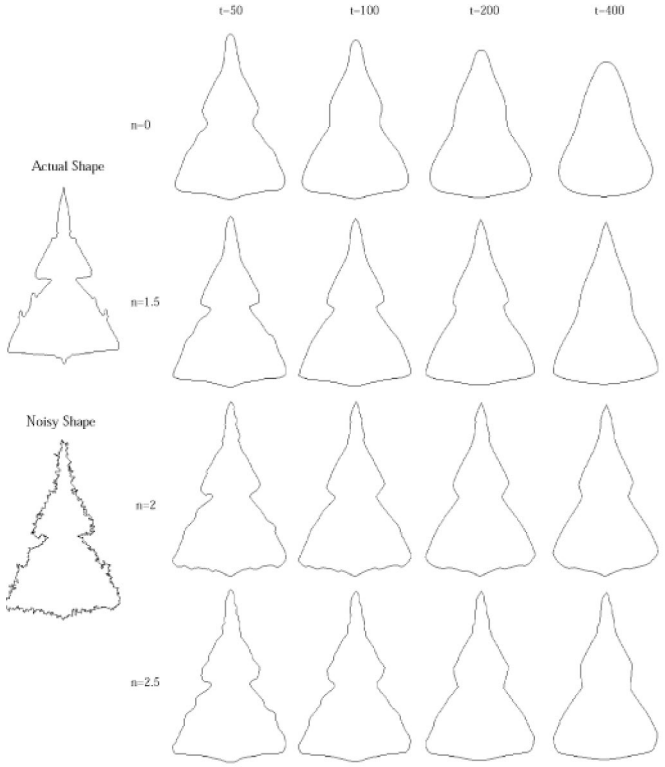


Fig. 10. Each row corresponds to a curve evolution method with different n, 1st row: $\mathcal{C}_t = \kappa \boldsymbol{N}$, 2nd row: $\mathcal{C}_t = \sin^2(1.5(\theta - \pi/2))\kappa \boldsymbol{N}$, 3rd row: $\mathcal{C}_t = \sin^2(2\theta)\kappa \boldsymbol{N}$, 4th row: $\mathcal{C}_t = \cos^2(2.5(\theta - \pi/2))\kappa N$.

Fig. 10. The geometric heat flow at the top row quickly smoothes corners of the shape out, and at coarser scales, the shape loses all of its features. The initial shape converges to a circle in spite of the global feature of the plane being a polygonal shape. This motivates the application of the geometric heat flow with a $\sin^2(n\theta)$ factor, where $n = 1.5$, and whose weak limiting shape is a triangle which intuitively matches the coarser form of the given plane shape. Similarly, for $n = 2$ and $n = 2.5$, different features of the shape are preserved, and persist over a much longer time period as can be observed from the column of shapes at $t = 400$. Note that the geometric heat flow result at the top quickly washes out any similarity to the actual shape, whereas the results of the other three flows preserve the global shape as well as some finer details on the wings, the tail, and the head part.

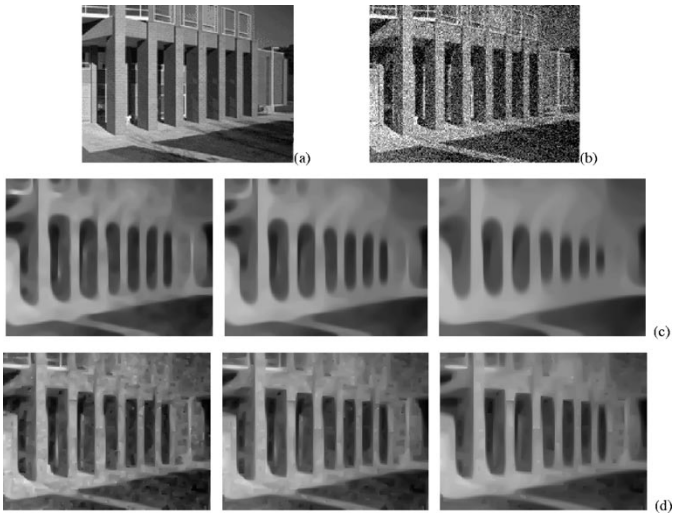


Fig. 11. (a) Clean building image, (b) noisy building image, (c) Geometric heat flow $u_t = u_{\xi\xi}$ (left-right) $t = 10, 20, 40$, (d) Flow $u_t = \cos^2(2\theta)u_{\xi\xi}$ (left-right) $t = 10, 20, 40$.

C. Examples With Grayscale Images

The proposed flows may also be applied to images in a straightforward fashion. For the case $h^2(\theta) = \cos^2(2\theta)$, all level sets of the image are driven to rectangles, thereby enhancing those features in an image. Such features can be found in contemporary buildings where one example in NCSU, Centennial Campus is shown in Fig. 11(a). The part of the building image with an additive Gaussian noise is shown in Fig. 11(b). The 2nd row shows the results of the geometric heat flow $u_t = u_{\xi\xi}$ at $t = 10, 20, 50$. The noisy image at $t = 0$ is smoothed out very quickly at the expense of rounding off all the corners because the level sets of the image converge to circles. The 3rd row shows the $u_t = \cos^2(2\theta)u_{\xi\xi}$ flow results at the same time points $t = 10, 20, 50$. Since the diffusion is constrained in order to drive image level sets to rectangles, the removal of noise is slower. However, the rectangular features still nicely appear after noise removal (see the image on the right), making it worthwhile to slow down the denoising effect of the geometric heat flow as deemed appropriate.

In Fig. 12, an experiment involving diamond-like shapes in the image taken from a poster on a wall is shown. In the middle row, rounding effects on diamond shapes performed by the geometric heat flow are clearly observed during the evolution of this image. The proposed flow, shown in the bottom row, takes the form $u_t = \sin^2(2\theta)u_{\xi\xi}$ for this particular shape, and particularly adapted to carrying out a shape-based smoothing which takes place at the boundaries of the diamonds. The slight blurring effect on the picture at continued application however is due to the interaction between consecutive level curves.

A photograph taken by pathfinder in mars, shown in Fig. 13, is argued to be a hexagon-shaped structure on mars' surface. The particular flow adapted to this shape is given by $u_t = \cos^2(3\theta)u_{\xi\xi}$, and the resulting images at the second column of the figure demonstrate a better smoothing performance at the boundaries of the hexagon when compared to the images in the first column processed by the geometric heat equation. From low scales to very large scales, the hexagon-adapted flow enhances and keeps on highlighting the related structure.

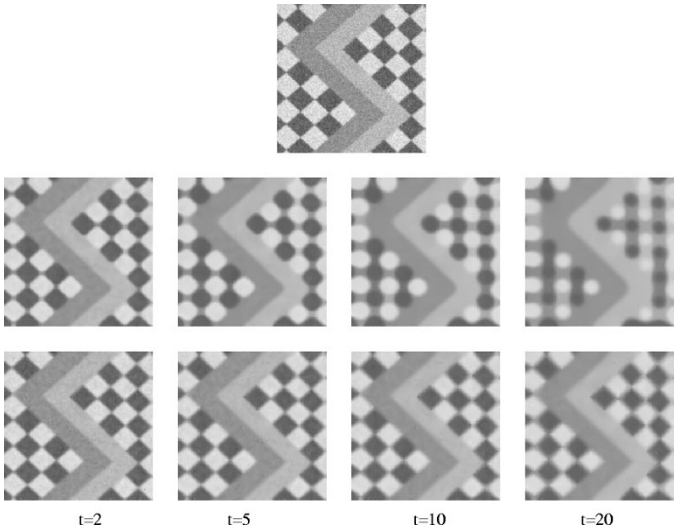


Fig. 12. (Top) Diamonds image (Middle row) Geometric heat flow $u_t = u_{\xi\xi}$ (Bottom row) Flow $u_t = \sin^2(2\theta)u_{\xi\xi}$.

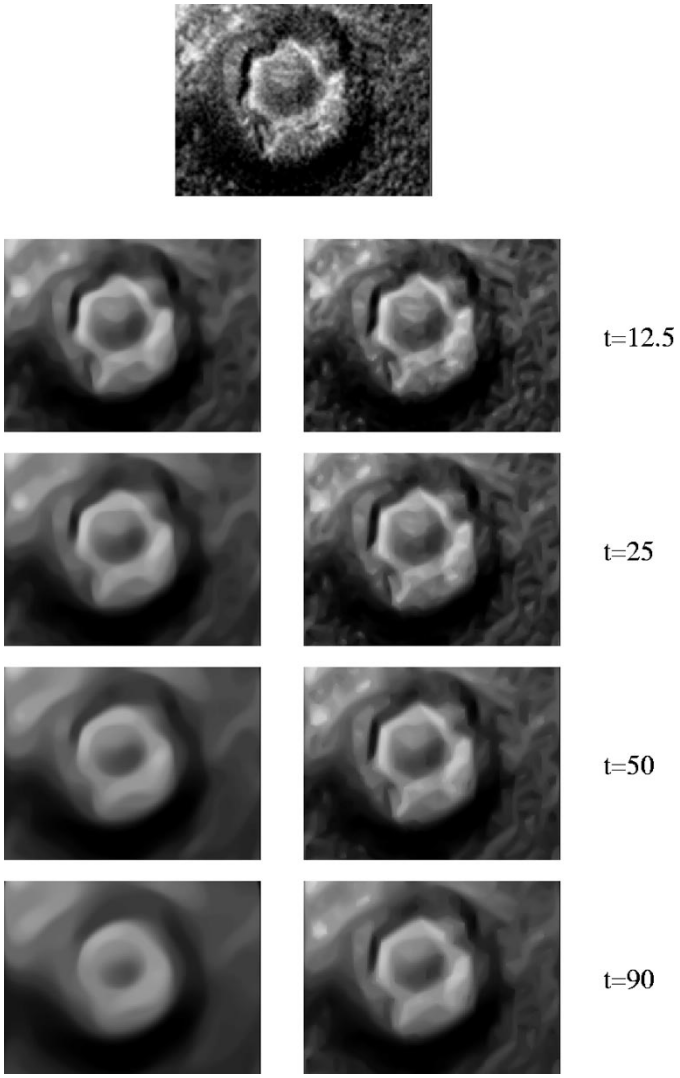


Fig. 13. (Top) An image from mars pathfinder, (First column) Geometric heat flow $u_t = u_{\xi\xi}$, (Second column) Flow $u_t = \cos^2(3\theta)u_{\xi\xi}$. (Image: Origin NASA, exposed by and courtesy of N. Coombs & NPAAG 1998).

A noise contaminated Aerial image is shown in Fig. 14(a). The geometric heat equation (see 2nd row, $t = 20, 40, 80$)

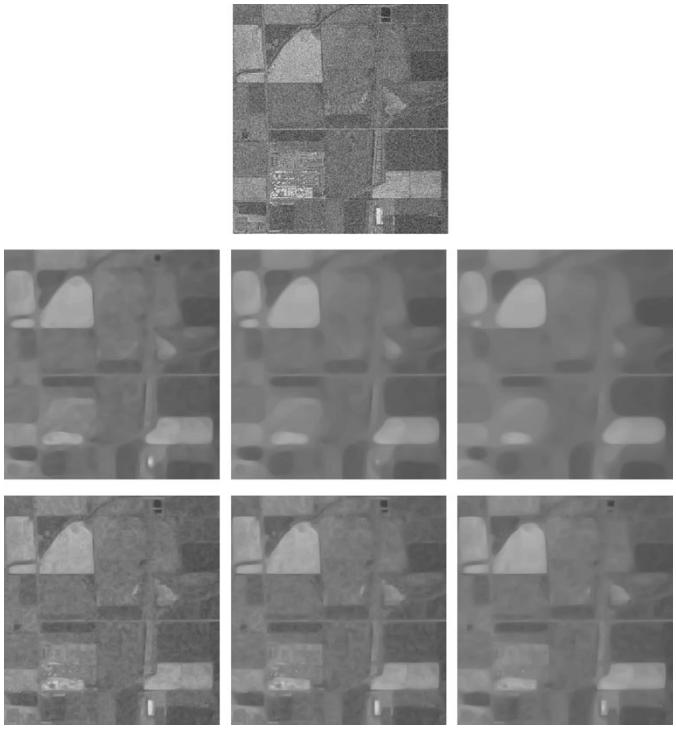


Fig. 14. Top: Aerial image; 2nd row: geometric heat flow $u_t = u_{\xi\xi}$, (left to right) $t = 20, 40, 80$; 3rd row: flow $u_t = \cos^2(2\theta)u_{\xi\xi}$, (left to right) $t = 20, 40, 80$.

sweeps away the shape information of the important details such as the city on the left bottom, the white bright rectangle on the right bottom, and the black feature at the top. The three images resulting from the $u_t = \cos^2(2\theta)u_{\xi\xi}$ flow, are quite sharp at the edges between both low and high contrast fields, therefore more useful in recognition of details as well as removing noise.

A last example is shown in Fig. 15, where windows and a roof of a section of a house are seen. On the left column, the results of the geometric heat flow $u_t = u_{\xi\xi}$ at times $t = 40, 80, 160$, are shown, and on the right those of $u_t = \cos^2(2\theta)u_{\xi\xi}$ at similar time steps. The noise is successfully removed by the geometric heat equation whose smearing effect on different regions into one another is also slow, at a cost of a problematic rounding off of corners. At time $t = 160$ for the result on the right bottom, approximately the same amount of noise as that of geometric heat equation at $t = 40$ is removed, and in addition to that the corners are still well-preserved.

VI. Conclusion

In this paper, we have formulated a local stochastic view of a nonlinear filtering technique, namely the geometric heat equation. The theory of stochastic differential equations provides a microscopic view of a system, and through a local linearization of the nonlinear geometric heat equation we have provided an alternative macroscopic view of this equation. We then modified this macroscopic description to propose new flows that vanish at pre-defined directions. We showed that these flows, although rotationally noninvariant, are capable of smoothing along priorly known orientations of salient lines in both curves and images, leading to preservation of polygonal structures. In the context of curve evolutions, curves evolved with the new flows are morphed into a limiting polygonal shape, this approach may hence be relevant to shape-morphing applications in computer graphics. Another application of these polygonizing flows is in classification of a shape after its filtering via a certain set of these flows, and its identification according to the outputs of these different filters.

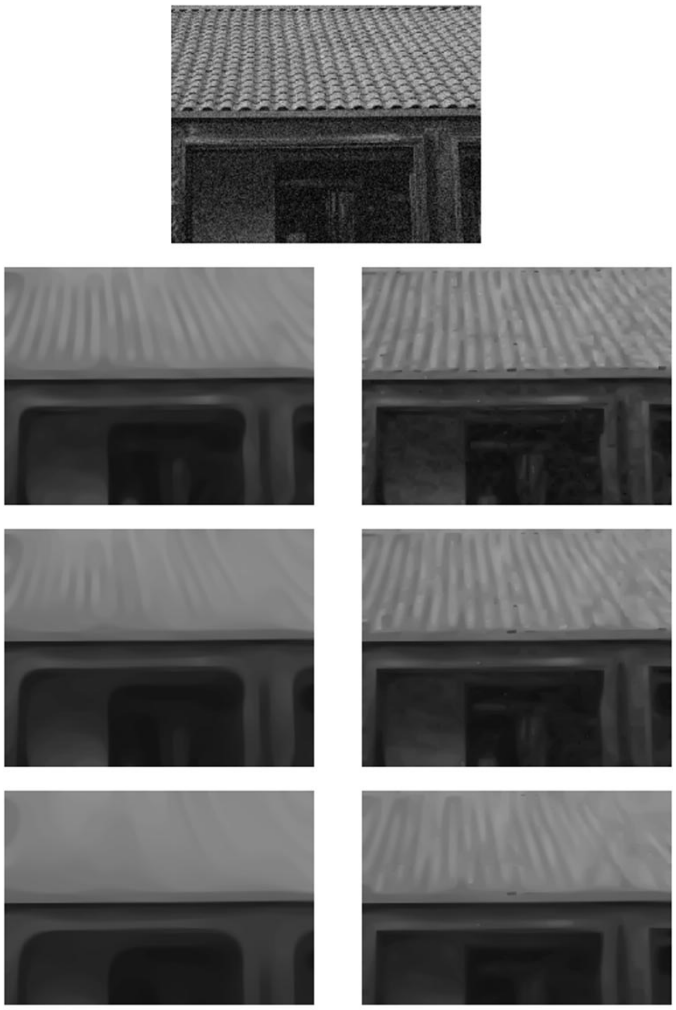


Fig. 15. Top: House; Left: Geometric heat flow $u_t = u_{\xi\xi}$, (top to bottom) $t = 40, 80, 160$; Right: Flow $u_t = \cos^2(2\theta)u_{\xi\xi}$, (top to bottom) $t = 40, 80, 160$.

Appendix

Let us denote by $u^n(t, \boldsymbol{x})$ the solution to (8)

$$\frac{\partial u^n}{\partial t} = \sin^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{xx}^n - \sin\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{xy}^n + \cos^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{yy}^n$$

and define

$$f_1\left(u_x^n, u_y^n, u_{xx}^n\right) = \sin^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{xx}^n,$$

$$f_2\left(u_x^n, u_y^n, u_{xy}^n\right) = \sin\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{xy}^n,$$

$$f_3\left(u_x^n, u_y^n, u_{yy}^n\right) = \cos^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) u_{yy}^n.$$

If we write $u(t, \boldsymbol{x})$ as

$$u(t, \boldsymbol{x}) = u^n(t, \boldsymbol{x}) + \epsilon u(t, \boldsymbol{x})$$

$$\frac{\partial \epsilon u}{\partial t} \approx \epsilon \boldsymbol{x_1} \cdot \nabla f_1\left(\boldsymbol{x_1^n}\right) - \epsilon \boldsymbol{x_2} \cdot \nabla f_2\left(\boldsymbol{x_2^n}\right) + \epsilon \boldsymbol{x_3} \cdot \nabla f_3\left(\boldsymbol{x_3^n}\right) \tag{26}$$

$$\begin{aligned}
\frac{\partial \epsilon u}{\partial t} \approx\ & \epsilon (u_x \quad u_y \quad u_{xx}) \cdot \begin{pmatrix} 2\sin\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\cos\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{-u_y^n}{(u_x^n)^2}u_{xx}^n \\ 2\sin\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\cos\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{1}{u_x^n}u_{xx}^n \\ \sin^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) \end{pmatrix} \\
& - \epsilon (u_x \quad u_y \quad u_{xy}) \cdot \begin{pmatrix} 2\cos\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{-u_y^n}{(u_x^n)^2}u_{xy}^n \\ 2\cos\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{1}{u_x^n}u_{xy}^n \\ \sin\left(2\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) \end{pmatrix} \\
& + \epsilon (u_x \quad u_y \quad u_{yy}) \cdot \begin{pmatrix} 2\cos\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\left(-\sin\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{-u_y^n}{(u_x^n)^2}u_{yy}^n \\ 2\cos\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\left(-\sin\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right)\right)\frac{1}{1+\left(\frac{u_y^n}{u_x^n}\right)^2}\frac{1}{u_x^n}u_{yy}^n \\ \cos^2\left(\tan^{-1}\left(\frac{u_y^n}{u_x^n}\right)\right) \end{pmatrix}
\end{aligned} \tag{27}$$

then $\epsilon u(t, \boldsymbol{x})$ satisfies

$$\begin{aligned}
\frac{\partial \epsilon u}{\partial t} =\ & f_1\left(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{xx}^n + \epsilon u_{xx}\right) \\
& - f_1\left(u_x^n, u_y^n, u_{xx}^n\right) \\
& - \left\{f_2\left(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{xy}^n + \epsilon u_{xy}\right)\right. \\
& \left. - f_2\left(u_x^n, u_y^n, u_{xy}^n\right)\right\} \\
& + f_3\left(u_x^n + \epsilon u_x, u_y^n + \epsilon u_y, u_{yy}^n + \epsilon u_{yy}\right) \\
& - f_3\left(u_x^n, u_y^n, u_{yy}^n\right).
\end{aligned} \tag{25}$$

Assuming $f_1(\cdot,\cdot,\cdot), f_2(\cdot,\cdot,\cdot)$, and $f_3(\cdot,\cdot,\cdot)$ are differentiable in their arguments, we can expand $f_1(\cdot,\cdot,\cdot)$ in Taylor series about $(u_x^n, u_y^n, u_{xx}^n), f_2(\cdot,\cdot,\cdot)$ about (u_x^n, u_y^n, u_{xy}^n), and $f_3(\cdot,\cdot,\cdot)$ about (u_x^n, u_y^n, u_{yy}^n). For notational simplicity, let us denote by $\boldsymbol{x_1} = (u_x, u_y, u_{xx})$ and $\boldsymbol{x_1^n} = (u_x^n, u_y^n, u_{xx}^n), \boldsymbol{x_2} = (u_x, u_y, u_{xy})$ and $\boldsymbol{x_2^n} = (u_x^n, u_y^n, u_{xy}^n)$, and $\boldsymbol{x_3} = (u_x, u_y, u_{yy})$ and $\boldsymbol{x_3^n} = (u_x^n, u_y^n, u_{yy}^n)$. If we assume that $\epsilon u(t, \boldsymbol{x})$ is small enough, we can neglect higher order terms and write a linear approximation as (see the equation at the top of the page). Defining the corresponding nominal angle $\theta^n(t, \boldsymbol{x}) = \tan^{-1}((u_y^n(t, \boldsymbol{x}))/(u_x^n(t, \boldsymbol{x})))$, and re-arranging the terms of (27), we get the linearized version of the geometric heat equation around a nominal value

$$\begin{aligned}
& \frac{\partial u(t, \boldsymbol{x})}{\partial t} \\
& \approx \mathcal{A}_{\mathrm{GHElin}} u(t, \boldsymbol{x}) \\
& = \sin^2(\theta^n(t, \boldsymbol{x}))u_{xx}(t, \boldsymbol{x}) - \sin(2\theta^n(t, \boldsymbol{x}))u_{xy}(t, \boldsymbol{x}) \\
& \quad + \cos^2(\theta^n(t, \boldsymbol{x}))u_{yy}(t, \boldsymbol{x}) + c\left(-u_y^n(t, \boldsymbol{x})u_x(t, \boldsymbol{x})\right. \\
& \quad \left. + u_x^n(t, \boldsymbol{x})u_y(t, \boldsymbol{x})\right)
\end{aligned} \tag{28}$$

where $c = (1)/((u_x^n)^2 + (u_y^n)^2)[\sin(2\theta^n)(u_{xx}^n - u_{yy}^n) - \cos(2\theta^n)2u_{xy}^n]$.

ACKNOWLEDGMENT

The authors appreciate fruitful discussions with Dr. Y. Bao of VISSTA, NCSU, and Dr. J. P. Fouque, and Dr. K. Ito of the Mathematics Dept. at NCSU.

REFERENCES

[1] S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on the Hamilton-Jacobi formulation," *J. Comput. Phys.*, vol. 49, pp. 12–49, 1988.

[2] J. A. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*. Cambridge, U.K.: Cambridge University Press, 1966.

[3] S. Osher, "Riemann solvers, the entropy condition and difference approximations," *SIAM J. Num. Anal.*, vol. 21, pp. 217–235, 1984.

[4] M. A. Grayson, "The heat equation shrinks embedded plane curves to round points," *J. Diff. Geom.*, vol. 26, pp. 285–314, 1987.

[5] B. B. Kimia, A. Tannenbaum, and S. W. Zucker, "On the evolution of curves via a function of curvature, I the classical case," *J. Math. Anal. Applicat.*, vol. 163, pp. 438–458, 1992.

[6] B. B. Kimia and K. Siddiqi, "Geometric heat equation and nonlinear diffusion of shapes and images," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1994.

[7] B. B. Kimia, A. Tannenbaum, and S. Zucker, "Shapes, shocks, and deformations I," *Int. J. Comput. Vis.*, no. 15, pp. 189–224, 1995.

[8] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel, "Axioms and fundamental equations of image processing," *Arch. Rational Mech. Anal.*, vol. 123, pp. 199–257, 1993.

[9] L. Alvarez, P. L. Lions, and J. M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Num. Anal.*, vol. 29, no. 3, pp. 845–866, 1992.

[10] A. P. Witkin, "Scale space filtering," in *Proc. Int. Conf. on AI*, Karlsruhe, Germany, 1983, pp. 1019–1023.

[11] G. Sapiro and A. Tannenbaum, "Affine invariant scale space," *Int. J. Comput. Vis.*, vol. 11, pp. 25–44, 1993.

[12] ——, "On invariant curve evolution and iamge analysis," *Indiana Univ. J. Math.*, vol. 93, 1993.

[13] H. Krim and Y. Bao, "A stochastic approach to signal denoising," presented at the ICASSP, Phoenix, AZ, 1999.

[14] ——, "Smart nonlinear diffusion; a probabilistic approach," , 2001, submitted for publication.

[15] A. Bruckstein, G. Sapiro, and D. Shaked, "Evolutions of planar polygons," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 9, no. 6, pp. 991–1014, 1995.

[16] G. Unal, A. Yezzi, and H. Krim, "Unsupervised texture segmentation by information-theoretic active polygons," , 2002, submitted for publication.
[17] H. Blum, "Biological shape and visual science," *J. Theor. Biol.*, pp. 205–287, 1973.
[18] K. Siddiqi and B. B. Kimia, "Parts of visual form: Computational aspects," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1993.
[19] R. L. Ogniewicz and M. Ilg, "Voronoi skeletons: Theory and applications," *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 63–69, 1992.
[20] L. Da, F. Costa, and L. F. Estrozi, "Multiresolution shape representation without border shifting," *Electron. Lett.*, vol. 35, no. 21, pp. 1829–1830, 1999.
[21] S. M. Pizer, A. L. Thall, and D. T. Chen, "M-Reps: A New Object Representation for Graphics," Univ. North Carolina, Chapel Hill, vol. 17, Tech. Rep. TR99-030, 1999.
[22] J. Gomes, "Warping and morphing of graphical objects, course notes," in *Proc. SIGGRAPH'95*, 1995.
[23] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 7, pp. 629–639, 1990.
[24] B. Oksendal, *Stochastic Differential Equations, an Introduction with Applications*. Berlin, Germany: Springer, 1998.
[25] D. Mumford, "Elastica and computer vision," in *Algebraic Geometry and Its Applications*, C. Bajaj, Ed. New York: Springer Verlag, 1994, pp. 491–506.
[26] L. R. Williams and D. W. Jacobs, "Stochastic completion fields: A neural model of illusory contour shape and salience," *Neural Comput.*, vol. 9, pp. 837–858, 1997.
[27] I. J. Cox, J. M. Rehg, and S. Hingorani, "A Bayesian multiple hypothesis approach to edge grouping and contour segmentation," *Int. J. Comput. Vis.*, vol. 11, pp. 5–24, 1993.
[28] M. G. Crandall, H. Ishii, and P. L. Lions, "User's guide to viscosity solutions of second order partial differential equations," *Bull. Amer. Math. Soc.*, vol. 27, no. 1, pp. 1–67, 1992.
[29] Y.-G. Chen, Y. Giga, and S. Goto, "Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations," *J. Diff. Geom.*, vol. 33, pp. 749–786, 1991.
[30] J. Rauch, *Partial Differential Equations*. New York: Springer-Verlag, 1991.
[31] D. Peng, S. Osher, B. Merriman, and H.-K. Zhao, "The geometry of Wulff crystal shapes and its relations with Riemann problems," *Contemp. Math.*, vol. 238, pp. 251–303, 1999.
[32] D. Adalsteinsson and J. A. Sethian, "A fast level set method for propagating interfaces," *J. Comput. Phys.*, vol. 118, pp. 269–277, 1995.

Gozde Unal (S'96) received the B.S. degree in electrical engineering from Middle East Technical University, Ankara, Turkey, in 1996, the M.S. degree in electrical engineering from Bilkent University, Ankara, in 1998, and the Ph.D. degree from the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, in 2002.

She is now a Postdoctoral Fellow with the School of Electrical Engineering, Georgia Institute of Technology, Atlanta. Her research interests include curve evolution theory with connections to information theory and probability theory, application of curve evolution techniques to various image and video processing problems such as image smoothing, image and texture segmentation, object tracking, and computer vision problems such as object recognition.

Hamid Krim (SM'98) received his degrees in electrical engineering.

As a Member of Technical Staff at AT&T Bell Labs, he has worked in the area of telephony and digital communication systems/subsystems. In 1991, he became a NSF Postdoctoral Scholar at Foreign Centers of Excellence (LSS Supelec/University of Orsay, Paris, France). He subsequently joined the Laboratory for Information and Decision Systems, MIT, Cambridge, MA, as a Research Scientist performing/supervising research in his area of interest,and later as a faculty in the ECE dept. at North Car. State Univ. in Raleigh, N.C. in 1998. He is an original contributor and now an affiliate of the Center for Imaging Science sponsored by the Army.

Dr. Krim is a recipient of the NSF Career Young Investigator Award. He is on the editorial board of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and regularly contributes to the society in a variety of ways. His research interests are in statistical estimation and detection and mathematical modeling with a keen emphasis on applications.

Anthony Yezzi (M'99) received the Ph.D. degree in 1997 from the Department of Electrical Engineering, University of Minnesota, Minneapolis.

After completing a postdoctoral research position in the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, he began his current position at Georgia Institute of Technology as an Assistant Professor in 1999. He has consulted for a number of medical imaging companies including GE, Picker, and VTI. His research lies primarily within the fields of image processing and computer vision. His work within these fields includes anisotropic diffusion for image smoothing and adaptive thresholding, edge-detection, active contours, segmentation, multiframe shape from shading and stereo matching, and shape analysis. His work in anisotropic smoothing and segmentation has been largely motivated and directed toward problems in medical imaging applied to MRI, ultrasound, CT, and OCT modalities. Two central themes of his research in general are curve/surface evolution theory from differential geometry and partial differential equations.