Modified Curvature Motion for Image Smoothing and Enhancement

Anthony Yezzi, Jr., *Member, IEEE*

***Abstract*—In this paper, we formulate a general modified mean curvature based equation for image smoothing and enhancement. The key idea is to consider the image as a graph in some $\mathbf{R}^n$, and apply a mean curvature type motion to the graph. We will consider some special cases relevant to grey-scale and color images.**

***Index Terms*—Enhancement, smoothing, mean curvature, partial differential equations.**

I. Introduction

RECENTLY, there have been a number of researchers who have considered the use of nonlinear curvature based equations for various problems in computer vision and image processing. An excellent reference is the volume edited by Romeny [23] to which we refer the interested reader for a large list of references.

In this paper, we consider a twist on the idea of mean curvature smoothing of an image in that we treat the image as the manifold defined by the graph of a function embedded in some Euclidean space. For example, a grey-scale two-dimensional (2-D) image $I : \mathbf{R}^2 \to \mathbf{R}$ may be regarded as the surface $(x, y, I(x, y)) \subset \mathbf{R}^3$. A 2-D color image similarly may be regarded as a surface in $\mathbf{R}^5$. We consider therefore mean curvature motion of these graphs as our underlying model for image smoothing and enhancement. A very attractive feature is that this gives a natural geometric way to treat vector-valued imagery. See [21], [26], [27], and [28] for other approaches. In particular, Sapiro and Ringbach [21] define an image based Riemannian metric in formulating their vector-valued diffusion method. While different from ours, it is nevertheless, closely related in spirit to our philosophy. We should also add that in [11], the authors also consider the image as a graph. However the level set type equations which they derive are different than ours.

The utility of our methods will be demonstrated on some grey-scale and color imagery. See Remark 1.

This work presented in this paper is part of the author's doctoral thesis done at the University of Minnesota and was presented at the author's Ph.D. oral exam in September 1996. After submitting the paper for publication, the author learned that very similar results had been independently obtained by Sochen *et al.* [25].

Manuscript received November 1, 1996; revised March 11, 1997. This work was supported in part by the NSF under Grant ECS-9122106, by the Air Force Office of Scientific Research under Grant F49620-94-1-0058DEF, and by the Army Research Office under Grants DAAH04-94-G-0054 and DAAH04-93-G-0332. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Guillermo Sapiro.

The author is with the Department of Electrical Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: ayezzi@ee.umn.edu).

Publisher Item Identifier S 1057-7149(98)01747-3.

II. 2-D Grey-Scale mage Smoothing

Before considering the general case below, we believe it is instructive to consider the key case of a two dimensional greyscale image. Throughout this paper, we will freely use the basic facts of differential geometry from [7] and [8]. Accordingly, we will consider such an image as the graph of a surface in $\mathbf{R}^3$.

From an inital image $I(x, y)$, construct an initial parameterized surface $S(x, y) = (x, y, I(x, y))$. The unit normal of this surface is given by

$$N(x,y) = \frac{S_x \times S_y}{\|S_x \times S_y\|} = \frac{(-I_x, -I_y, 1)}{\sqrt{1 + I_x^2 + I_y^2}}$$

and the mean curvature by

$$H(x,y) = \frac{I_{xx}\big(1 + I_y^2\big) - 2I_xI_yI_{xy} + I_{yy}\big(1 + I_x^2\big)}{2\big(1 + I_x^2 + I_y^2\big)^{3/2}}.$$

Since S is a graph, it will, under mean curvature motion $S_t = HN$, evolve into a plane without developing singularities [18], [10]. As S evolves in this manner, small scale features of high curvature induced by noise in the image are very quickly removed. However, an undesirable phenomenon occurs from the point of view of image processing, namely, edges become blurred. These effects will be illustrated in the following example.

Consider a 2-D grey-scale image which is constant along the y direction but which is black on the left half and white on the right half so that any horizontal cross section along the x direction yields a common step function. Now add small oscillations to simulate noise in the image and "round off" the corners of the step edge so that our function becomes differentiable. Finally, for the sake of illustration, assume that this modeled noise is constant along the y direction so that mean curvature motion of the surface $(S_t = HN)$ causes the cross sectional curve, C, to evolve according to its curvature $(C_t = \kappa N)$. Fig. 1, which shows a sampling of the mean curvature vectors along the initial curve C, clearly demonstrates that this type of motion will have the desirable

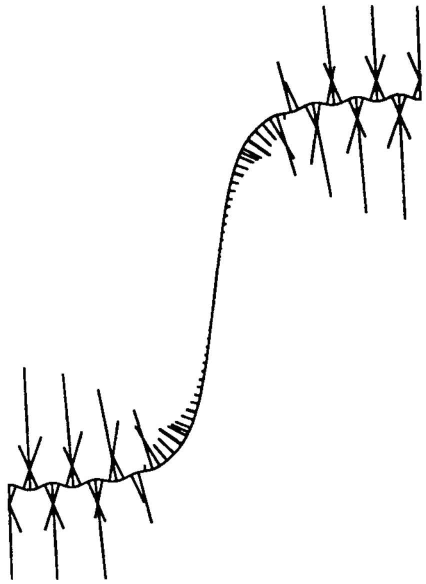


Fig. 1. Mean curvature vectors along a noisy step function.

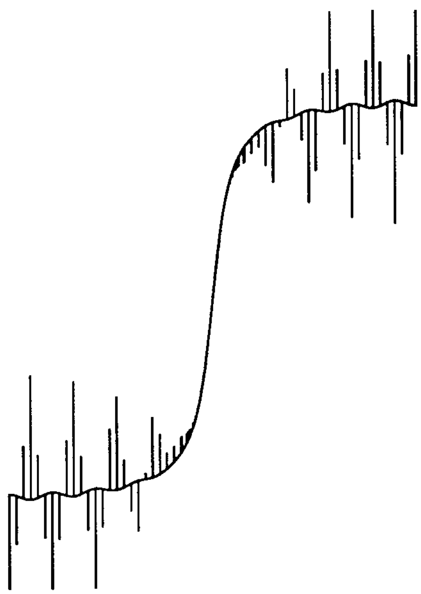


Fig. 2. Vertically projected mean curvature vectors.

effect of flattening out the oscillations but will also have the undesirable effect of widening the step edge.

However, suppose we now constrain the evolution of each point of S to be purely vertical by projecting the mean curvature vectors onto the vertical direction (z-axis) so that no sideways motion can occur. Therefore, instead of $S_t = HN$ we consider $S_t = (HN \cdot Z)Z$ which causes C to evolve according to $C_t = (\kappa N \cdot Z)Z$ where Z represents the unit vector in the vertical direction. Fig. 2, which shows a sampling of the projected mean curvature vectors along the initial curve C, clearly demonstrates that this completely vertical motion will still eliminate the unwanted oscillations but will no longer pull the corners of the step edge further apart. Thus, by vertically projecting regular mean curvature motion of S we obtain an edge preserving, noise removing evolution given by

$$\begin{aligned} S_t &= ((HN) \cdot Z)Z \\ &= \frac{I_{xx}(1+I_y^2) - 2I_xI_yI_{xy} + I_{yy}(1+I_x^2)}{2(1+I_x^2+I_y^2)^2}(0,0,1). \end{aligned}$$

Notice that the evolving surface under this modified form of mean curvature motion takes the form of $S(x,y,t) = (x,y,f(x,y,t))$ and so $I(x,y,t)$ is easily extracted from $S(x,y,t)$ by setting $I(x,y,t) = f(x,y,t)$. This allows us to dispense with S altogether and simply write down the following edge-preserving anisotropic filter for I

$$I_t = \frac{I_{xx}(1+I_y^2) - 2I_xI_yI_{xy} + I_{yy}(1+I_x^2)}{2(1+I_x^2+I_y^2)^2}.$$

III. Scaling Parameter

The filter we have presented can be extended into an entire family of filters by scaling the height of the image. More precisely, if we make the substitution $I \to kI$ into the above equation for some positive constant k we obtain the more general filter

$$I_t = \frac{\triangle I + k^2(I_x^2 I_{yy} - 2I_xI_yI_{xy} + I_y^2 I_{xx})}{(1+k^2\|\nabla I\|^2)^2}.$$

Scaling I by a large value of k amplifies edges in the image and will yield a filter with very strong edge preserving properties. However, choosing a smaller value of k will yield a faster diffusion. This tradeoff between speed and edge preservation will be illustrated on real images in Section VII but can been seen mathematically by observing the limiting cases shown below.

$$\begin{aligned} k \to 0&: \quad I_t \to \triangle I, \\ k \to \infty&: \quad I_t \to \frac{1}{k^2\|\nabla I\|^2} \nabla \cdot \left(\frac{\nabla I}{\|\nabla I\|}\right) \|\nabla I\|. \end{aligned}$$

As k becomes very small the aniosotropic diffusion approaches the isotropic heat equation which implements a rather fast diffusion but does a poor job of preserving edges. As k becomes very large we approach a damped geometric heat equation which does a far superior job of preserving edges. Futhermore, the damping term applied to the limiting geometric heat equation helps to prevent the distortion of shapes caused by the pure geometric heat equation. However, this damping term also makes the diffusion much slower.

IV. Preliminary Lemmas

In this section, we will derive two results which will be useful in analyzing the general formula for projected mean curvature motion.

Lemma 1: If $a_1, \ldots, a_m \in \mathbf{R}^n$ then

$$\pi_N(s) = s^{n-m}\pi_M(s) \quad \text{where} \quad \begin{cases} N = \sum a_i a_i^T \\ M = [a_i \cdot a_j]_{ij} \end{cases}$$

where π_N, π_M denote the characteristic polynomials of M, N.

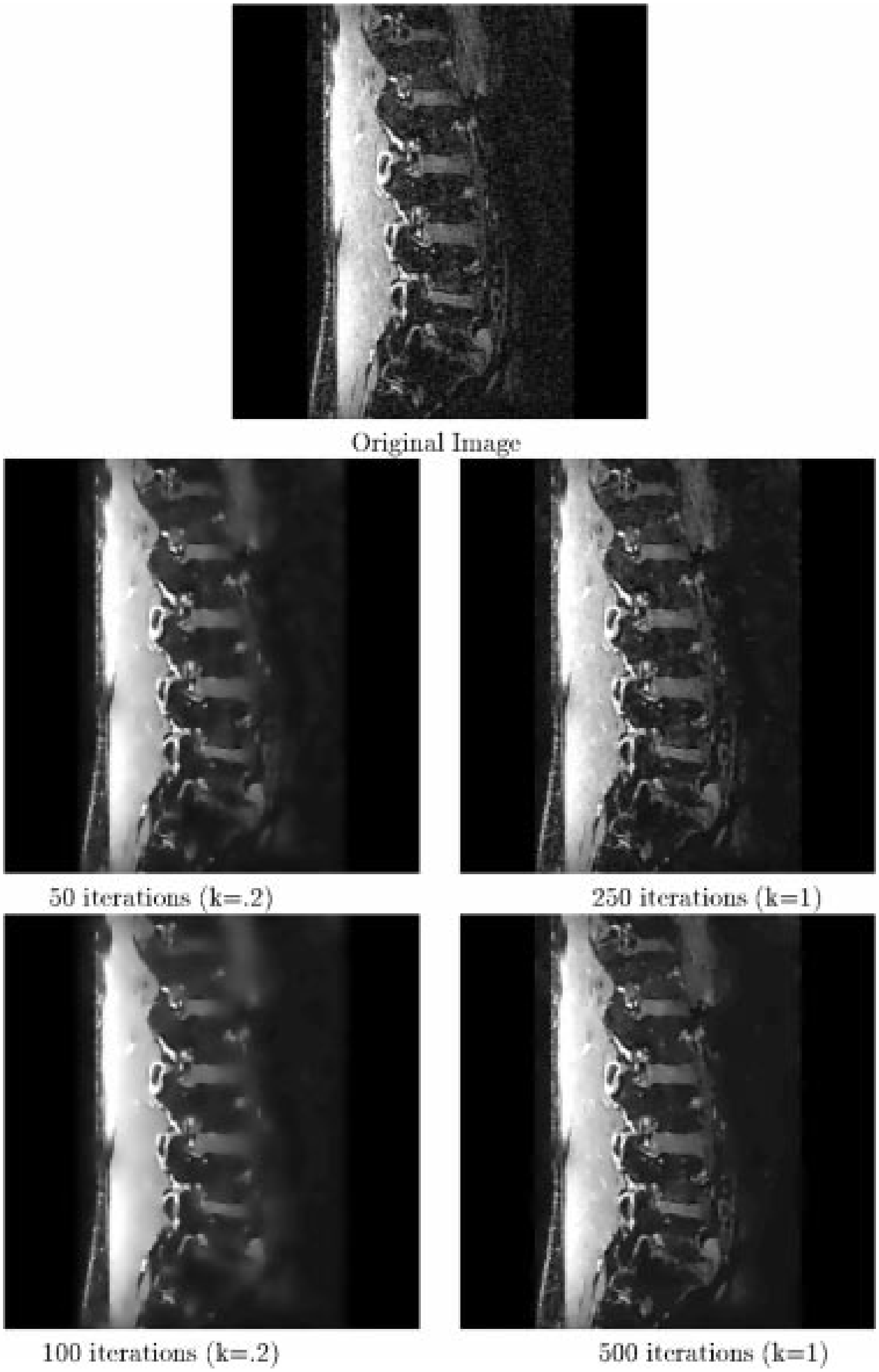


Fig. 3. Projected mean curvature smoothing on a grey-scale MRI image.

Proof: Note that $N = AA^T$ and $M = A^TA$ where $A = [a_1 \cdots a_m]$. If λ is a nonzero eigenvalue of N with the associated nonzero eigenvector v, then since $Nv = AA^Tv = \lambda v \neq 0$ we know that $w = A^Tv \neq 0$. However, since $Mw = A^TAA^Tv = A^T(\lambda v) = \lambda w$ we see that λ is also an eigenvalue of M with the associated nonzero eigenvector w. A similar argument can be used to show that every nonzero eigenvalue of M is also an eigenvalue of N. Therefore, since M and N share the same nonzero eigenvalues, their characteristic polynomials must differ only by factors of s. In

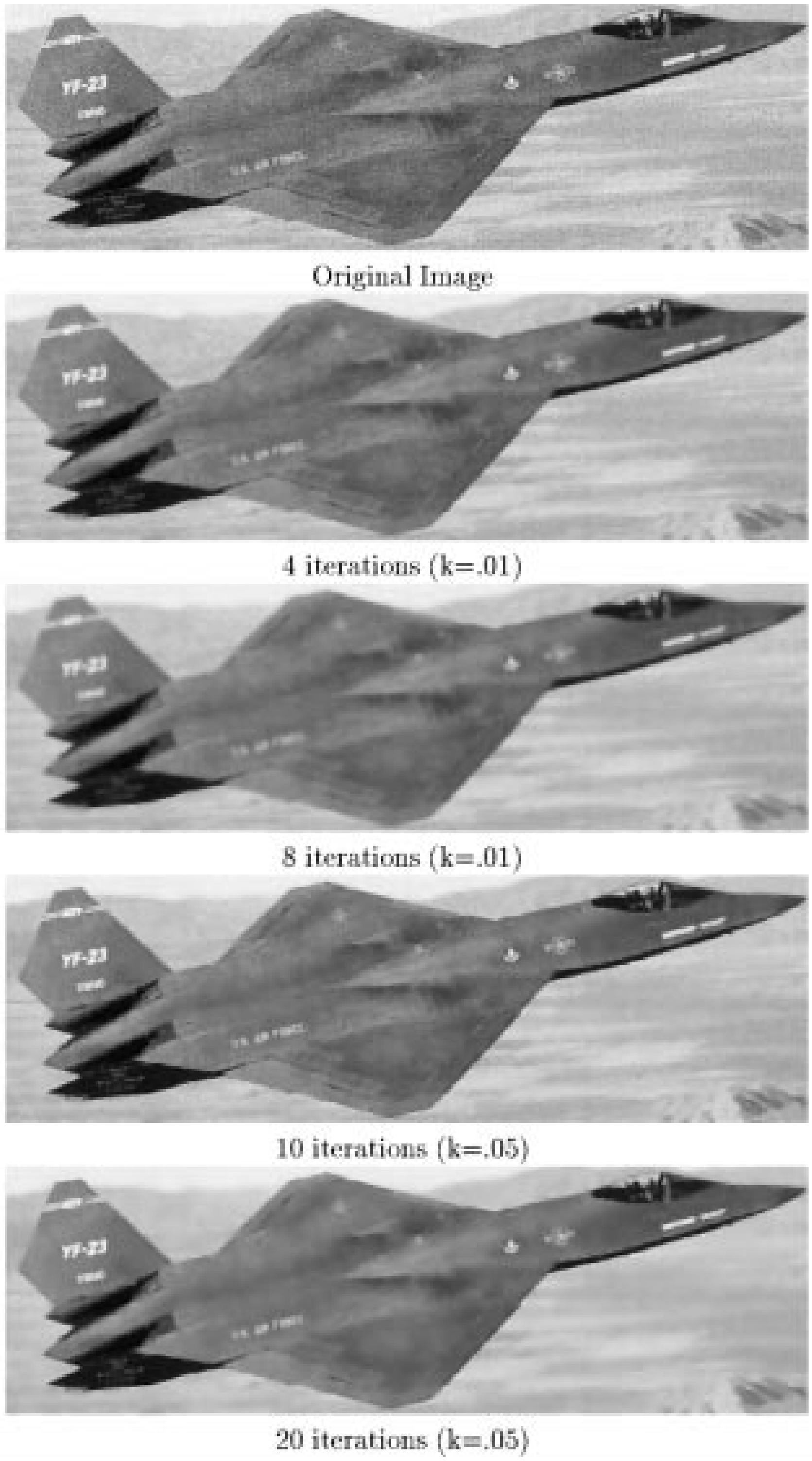


Fig. 4. Projected mean curvature smoothing on a color aircraft image.

particular, since N is $n \times n$ and M is $m \times m$, the characteristic polynomial of N must differ from that of M by a factor of s^{n-m}. □

Lemma 2: If $a_1, \cdots, a_m \in \mathbf{R}^n$ and $k \in \mathbf{R}$ then

$$\det\Big(kI_n + \sum a_i a_i^T\Big) = k^{n-m} \det(kI_m + [a_i \cdot a_j]_{ij})$$

where I_m, I_n denote the $m \times m$ and $n \times n$ identity matrices.

Proof: Using the Lemma above and its notation we have

$$\begin{aligned} \det\Big(kI_n + \sum a_i a_i^T\Big) &= (-1)^n \pi_N(-k) \\ &= (-1)^n (-k)^{n-m} \pi_M(-k) \\ &= (-1)^m k^{n-m} \pi_M(-k) \\ &= k^{n-m} \det(I_m + [a_i \cdot a_j]_{ij}). \end{aligned}$$

V. The General Case

We will now derive the general formula for the vertically projected mean curvature motion of a graph of arbitrary dimension and co-dimension. In what follows below we will use the symbol $\mathcal{I}_p$ for the $p \times p$ identity matrix.

In general, the mean curvature vector H of an m-dimensional surface S in $\mathbf{R}^{m+n}$ (co-dimension n) with coordinates $x_1, \cdots, x_m$ is given by

$$H = \text{Tr}[G^{-1}\,\text{Proj}(\nabla^2 S)], \quad \begin{cases} G = (S_{x_i} \cdot S_{x_j})_{ij} \\ \text{Proj}(\nabla^2 S) = (PS_{x_i x_j})_{ij} \end{cases}$$

where P is the orthogonal projection map which annihilates the component of $S_{x_i x_j}$ in the tangent space of S (span of $S_{x_1}, \cdots, S_{x_m}$). Actually, the H defined here is m times the true mean curvature vector, but by abuse of notation we will continue to refer to H as the mean curvature vector. Since P is linear and $\text{Tr}[G^{-1}\text{Proj}(\nabla^2 S)]$ yields a linear combination of the elements of $\text{Proj}(\nabla^2 S)$ we can pull P outside and write

$$H = P\,\text{Tr}(G^{-1}\nabla^2 S)$$

Note that the matrices P, G, and $\nabla^2 S$ are all functions of $x = (x_1, \cdots, x_m)$. Consider the case where S is a graph of the form $S(x) = (x, I(x)), I : \mathbf{R}^m \to \mathbf{R}^n$. The elements g_{ij} of G, which are just the coefficients of the first fundamental form S, are then $S_{x_i} \cdot S_{x_j} = \delta_{ij} + I_{x_i} \cdot I_{x_j}$, and so $G = \mathcal{I}_m + J(I)^T J(I)$ where $J(I)$ denotes the $n \times m$ Jacobian matrix of I. Also, note that the elements S_{ij} of $\nabla^2 S$ are given by $S_{ij} = (0, I_{x_i x_j})$ where 0 denotes the m-dimensional zero vector and $I_{x_i x_j}$ are the n-dimensional elements of $\nabla^2 I$. Now represent P in block form as

$$P = \begin{bmatrix} A & B \\ C & \hat{P} \end{bmatrix}$$

where A is $m \times m$, B is $m \times n$, C is $n \times m$ and $\hat{P}$ is $n \times n$. Since P annihilates any tangent vector S_{x_i} we have

$$PS_{x_i} = \begin{bmatrix} A & B \\ C & \hat{P} \end{bmatrix} \begin{bmatrix} e_i \\ I_{x_i} \end{bmatrix} = 0$$

where $e_1, \cdots, e_m$ denote the standard orthonormal basis for $\mathbf{R}^m$. From this expression we see that $Ce_i = -\hat{P}I_{x_i} = 0$ for each $i = 1, \cdots, m$ and so

$$C = \begin{bmatrix} -\hat{P}I_{x_1} & \cdots & -\hat{P}I_{x_m} \end{bmatrix} = -\hat{P}J(I).$$

Next, since P preserves any normal vector $N = (u, v)^T$ of S where $u \in \mathbf{R}^m$ and $v \in \mathbf{R}^n$ then,

$$PN = \begin{bmatrix} A & B \\ C & \hat{P} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} Au + Bv \\ Cu + \hat{P}v \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix} = N$$

and so $v = Cu + \hat{P}v$. However, note that $N \cdot S_{x_i} = (u, v) \cdot (e_i, I_{x_i}) = u \cdot e_i + v \cdot I_{x_i} = 0$ for each $i = 1, \cdots, m$ since N is orthogonal to each S_{x_i}. Therefore $u = (-v \cdot I_{x_1}, \cdots, -v \cdot I_{x_m})^T = -J^T(I)v$. Substituting for u and C yields $v = [-\hat{P}J(I)][-J^T(I)v] + \hat{P}v = \hat{P}[J(I)J^T(I) + \mathcal{I}_n]v$. From this expression, it is clear that $\hat{P}[J(I)J^T(I) + \mathcal{I}_n]$ must be the identity map, and so $\hat{P} = [J(I)J^T(I) + \mathcal{I}_n]^{-1}$.

Now consider using projected mean curvature motion of S as a way to smooth the m-dimensional n-vector valued image $I : \mathbf{R}^m \to \mathbf{R}^n$. As before, we project the mean curvature vector H of S onto the n-dimensional subspace of $\mathbf{R}^{m+n}$ orthogonal to the m-dimensional domain of I under the inclusion map via the matrix

$$V = \begin{bmatrix} 0 & \\ & \mathcal{I}_n \end{bmatrix}$$

where zero represents the $m \times m$ zero matrix, and $\mathcal{I}_n$ the $n \times n$ identity matrix. Since the resulting evolution

$$S_t = VH = (VP)\,\text{Tr}(G^{-1}\nabla^2 S) = \begin{bmatrix} 0 & 0 \\ C & \hat{P} \end{bmatrix} \text{Tr}(G^{-1}\nabla^2 S)$$

leaves the first m components of S unchanged and since the last n components of S evolve according to $\hat{P}\,\text{Tr}(G^{-1}\nabla^2 I)$ (this is because the first m components of the elements of $\nabla^2 S$ are zero and the last n components form an element of $\nabla^2 I$), we may dispense with S and evolve the image directly via $I_t = \hat{P}\,\text{Tr}(G^{-1}\nabla^2 I)$. Substituting the values of G and $\hat{P}$ just computed yields

$$I_t = [\mathcal{I}_n + J(I)J^T(I)]^{-1}\,\text{Tr}\{[\mathcal{I}_m + J^T(I)J(I)]^{-1}\nabla^2 I\}.$$

If we make the substitution $I \to kI$ to account for arbitrary scalings of I we obtain the more general equation

$$I_t = [\mathcal{I}_n + k^2 J(I)J^T(I)]^{-1}\,\text{Tr}\{[\mathcal{I}_m + k^2 J^T(I)J(I)]^{-1}\nabla^2 I\}.$$

Note, as seen already in the 2-D grey-scale case, that as k goes to zero, the diffusion approaches the linear heat equation ($I_t \to \text{Tr}\nabla^2 I$).

If $m > n$, then it may be easier to compute the determinant of the $n \times n$ first fundamental form matrix than it is the $m \times m$ projection matrix in the above equation. We can then use the result of Lemma 2 to avoid the computation of the more difficult determinant. Pulling out a factor of k^{-4} and applying Lemma 2 yields the formulation shown at the bottom of the page, or in case $n > m$, it may be easier to use the formulation shown on the bottom of the next page.

Remark 1: In this paper in co-dimension 1, we are considering the equation

$$I_t = \mathcal{H}(I)$$
$$\mathcal{H}(I) = \text{div}\left(\frac{\nabla I}{(1 + |\nabla I|^2)^{1/2}}\right). \tag{1}$$

$$I_t = k^{2(n-m)-4} \times \frac{\text{Adj}[k^{-2}\mathcal{I}_n + J(I)J^T(I)]\,\text{Tr}\{\text{Adj}[k^{-2}\mathcal{I}_m + J^T(I)J(I)]\nabla^2 I\}}{\det^2[k^{-2}\mathcal{I}_n + J^T(I)J(I)]}$$

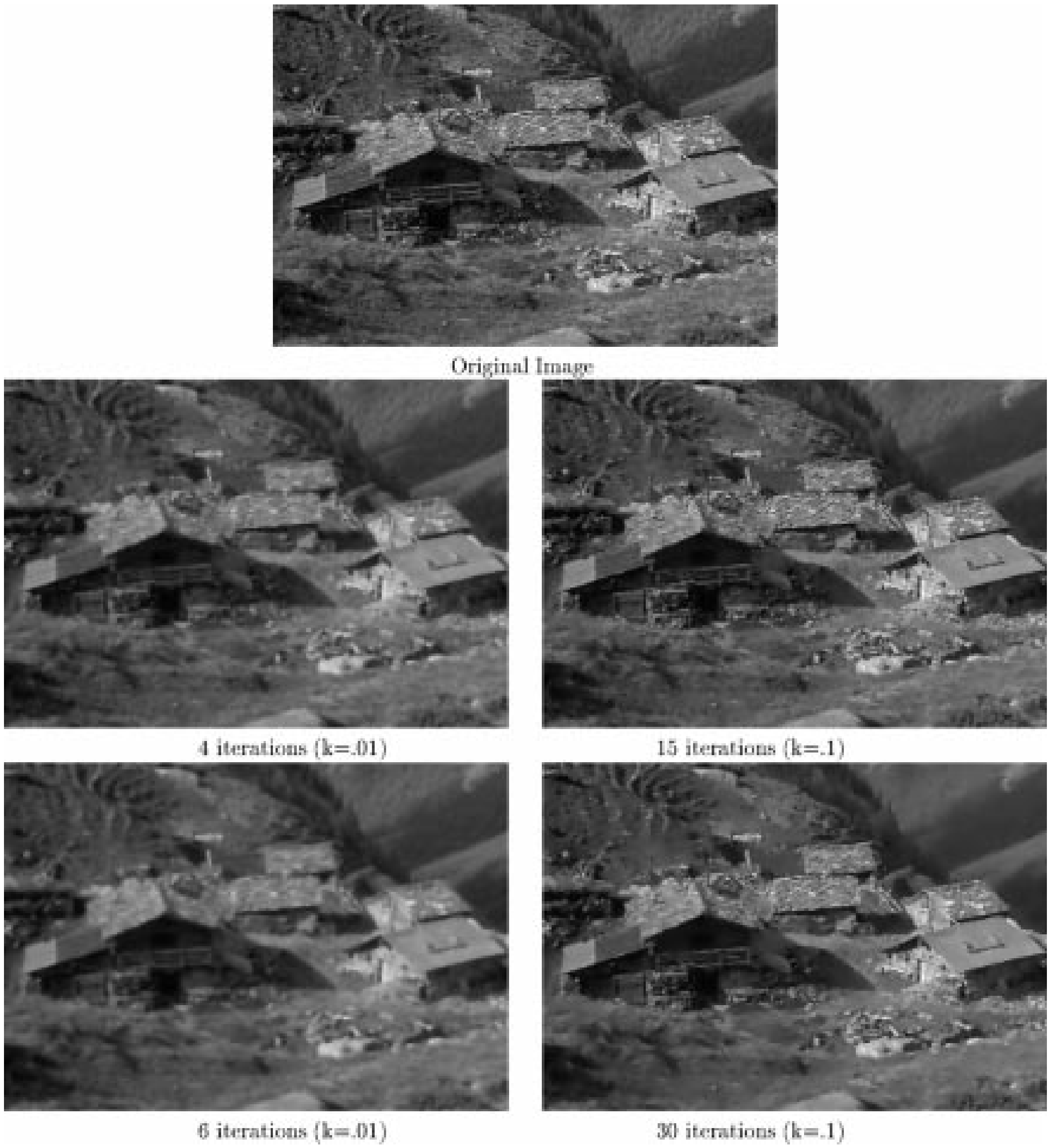


Fig. 5. Projected mean curvature smoothing on a color outdoor image.

It is different from standard mean curvature flow

$$I_t = (1 + |\nabla I|^2)^{1/2} \mathcal{H}(I) \tag{2}$$

in which the normal speed equals the mean curvature of the current surface. There is of course the important question of uniqueness and existence of solutions of the evolution of graphs via such equations of the type we are proposing. For (1), there are existence and uniqueness results which prescribed initial and boundary data but in a weak sense, via a variational formulation; see [16], [17], and [18].

In the variational formulation, the boundary data need not be taken in a classical sense so that the surface does not necessarily support on the given curve. For (2), the situation seems more difficult because the problem is not in divergence form. The surface will support on the specified curve (the

$$I_t = k^{2(m-n)-4} \times \frac{\operatorname{Adj}[k^{-2}\mathcal{I}_n + J(I)J^T(I)] \operatorname{Tr}\{\operatorname{Adj}[k^{-2}\mathcal{I}_m + J^T(I)J(I)]\nabla^2 I\}}{\det^2[k^{-2}\mathcal{I}_m + J(I)J^T(I)]}$$

$$I_t = \frac{k^{-2}\nabla^2 I + \left(I_y^2 + I_z^2\right)I_{xx} + \left(I_x^2 + I_z^2\right)I_{yy} + \left(I_x^2 + I_y^2\right)I_{zz} - 2(I_xI_yI_{xy} + I_xI_zI_{xz} + I_yI_zI_{yz})}{(k^{-2} + \|\nabla I\|^2)^2}$$

boundary data will be taken in a classical sense) essentially if the boundary is convex. If not vertical walls may appear. This is extensively discussed in [22]. The problem for hypersurfaces in $\mathbf{R}^n$ was considered [18], [10]. For periodic boundary conditions, the regularity of the flow has been proven in these works. For higher co-dimension, such problems have recently been studied by [5].

VI. 2-D Color and 3-D Grey-Scale Imagery

In this section, we compute from the general equation, the projected mean curvature diffusions for the important special cases of 2-D color and 3-D grey-scale images.

A 2-D color image amounts to a surface in $\mathbf{R}^5$, given by $S(x,y) = (x,y,I(x,y))$. Solving the general equation for $m = 2$ and $n = 3$ with the scaling factor k yields

$$I_t = k^{-2}\mathrm{Adj}\left(k^{-2}\mathcal{I}_3 + I_xI_x^T + I_yI_y^T\right) \times \frac{(k^{-2} + I_y\cdot I_y)I_{xx} - 2(I_x\cdot I_y)I_{xy} + (k^{-2} + I_x\cdot I_x)I_{yy}}{[(k^{-2} + I_x\cdot I_x)(k^{-2} + I_y\cdot I_y) - (I_x\cdot I_y)^2]^2}$$

In fact, by merely replacing $\mathcal{I}_3$ in the above equation with $\mathcal{I}_n$, we obtain the projected mean curvature diffusion equation for an n-vector valued 2-D image.

A 3-D grey scale image amounts to a 3-D hypersurface in $\mathbf{R}^4$ given by $S(x,y,z) = (x,y,z,I(x,y,z))$. Solving the general equation for $m = 3$ and $n = 1$ with the scaling factor k yields the formulation shown at the top of the page.

VII. Numerical Experiments

Figs. 3–5 illustrate the use of projected mean curvature smoothing on 2-D grey-scale and color data. The color images here will be shown in grey scale. The original color images can be found at the author's website, www.ece.umn.edu/users/ayezzi. A time step of 0.1 was consistently applied in each example so that the number of iterations performed in different cases could be compared in a meaningful manner. In all three figures, the uppermost image displays the original unfiltered data and the lower four images display the results of our filter under two different scaling factors. In Fig. 3, a grey-scale magnetic resonance image (MRI) of the spinal cord lumbar region, the two images of the left exhibit the effect of 50 and 100 iterations using the scaling factor $k = 0.2$. Alongside these images, on the right-hand side, are the results of 250 and 500 iterations using a larger scaling factor $k = 1.0$. The larger scaling factor has done a superior job of preserving edges; the images on the left-hand side are clearly more blurred. However, this improvement in performance has come at the cost of a slower diffusion as seen by comparing the number of iterations required in these two cases. Fig. 4, a rather grainy aircraft image, reveals this same type of behavior on a color image. The bottom two images show the results of 10 and 20 iterations using a scale factor $k = 0.05$ while the preceding two images show the results of four and eight iterations using a scale factor $k = 0.01$. Although a larger number of iterations were required in the bottom two images to attain an equivalent level of smoothing, the improvement in edge preservation can be seen very clearly by looking at the letters "YF-23" on the tail of the airplane in the two sets of images. Finally, Fig. 5, a color outdoor image chosen for its fine features and significant color contrast, very clearly exhibits this trade-off in choosing scaling factors. The two images on the left show the effect of 4 and 6 iterations using a scaling factor $k = 0.01$ while the two images on the right show the effect of 15 and 20 iterations using a scaling factor $k = 0.1$, an order of magnitude larger. By comparing the tiles on the roofs of the houses in each set of images, one can easily see that the larger scaling factor has again done a superior job of preserving edges.

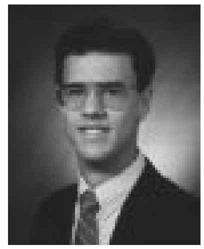
Acknowledgment

The author would like to thank Prof. A. Tannenbaum for a number of very helpful conversations about the material in this paper, and the referee for his/her comments. We are grateful to the reviewer for pointing out a number of the references and key issues in Remark 1.

References

[1] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel, "Axiomes et equations fondamentales du traitement d'images," *C. R. Acad. Sci.*, vol. 315, pp. 135–138, 1992.
[2] ——, "Axioms and fundamental equations of image processing," *Arch. Rational Mechan. Anal.*, vol. 123, 1993.
[3] ——, "Axiomatization et nouveaux operateurs de la morphologie mathematique," *C. R. Acad. Sci.*, vol. 315, pp. 265–268, 1992.
[4] L. Alvarez, P. L. Lions, and J. M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, vol. 29, pp. 845–866, 1992.
[5] L. Ambrosio and H. M. Soner, "Level set approach to mean curvature flow in arbitrary codimension," Tech. Rep., Preprints di Matematica 34, Scuola Normale Superiore, Pisa, Italy, 1994.
[6] B. Chow, "Deforming convex hypersurfaces by the nth root of the Gaussian curvature," *J. Diff. Geom.*, vol. 22, pp. 117–138, 1985.
[7] M. P. Do Carmo, *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice-Hall, 1976.
[8] M. P. Do Carmo, *Riemannian Geometry*. Englewood Cliffs, NJ: Prentice-Hall, 1992.
[9] K. Ecker and G. Huisken, "Mean curvature flow of entire graphs," *Ann. Math.*, vol. 130, pp. 453–471, 1989.
[10] L. C. Evans and J. Spruck, "Motion of level curves by mean curvature, III," *J. Geom. Anal.*, vol. 2, pp. 121–150, 1992.
[11] A. El-Fallah and G. Ford, "The evolution of mean curvature in image filtering," in *Proc. ICIP*, 1994, pp. 298–303.
[12] G. Huisken, "Flow by mean curvature of convex surfaces into spheres," *J. Differ. Geom.*, vol. 20, pp. 237–266, 1984.
[13] M. 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, pp. 1–67, 1992.
[14] 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. Applic.*, vol. 163, pp. 438–458, 1992.

[15] R. J. LeVeque, *Numerical Methods for Conservation Laws*. Boston, MA: Birkhäuser, 1992.
[16] A. Lichnewski and R. Temam, "Pseudosolutions of the time-dependent minimal surface problem," *J. Differ. Equ.*, vol. 30, pp. 340–364, 1978.
[17] C. Gerhardt, "Evolutionary surfaces of prescribed mean curvature," *J. Differ. Equ.*, vol. 36, pp. 139–172, 1980.
[18] K. Ecker, "Estimates for evolutionary surfaces of prescribed mean curvature," *Math. Zeitschr.*, vol. 180, pp. 179–192, 1982.
[19] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 629–639, 1990.
[20] M. H. Protter and H. F. Weinberger, *Maximum Principles in Differential Equations*. New York: Springer-Verlag, 1984.
[21] G. Sapiro and D. Ringbach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Trans. Image Processing*, vol. 5, pp. 1582–1585, 1996.
[22] V. I. Oliker and N. N. Uraltseva, "Evolution of nonparametric surfaces with speed depending on curvature II. The mean curvature case," *Commun. Pure Appl. Math.*, vol. 56, pp. 97–135, 1993.
[23] B. ter Haar Romeny, Ed., *Geometry-Driven Diffusion in Computer Vision*. Boston, MA: Kluwer, 1994.
[24] J. Smoller, *Shock Waves and Reaction-Diffusion Equations*. New York: Springer-Verlag, 1983.
[25] N. Sochen, R. Kimmel, and R. Malladi, "From high energy physics to low level vision," this issue, pp. 310–318.
[26] J. Weickert, "Scale-space properties of nonlinear diffusion filtering with a diffusion tensor," Tech. Rep.110, Lab. Technomath., Univ. Kaiserslautern, Germany, 1994.
[27] R. Whitaker and G. Gerig, "Vector-valued diffusion," in *Geometry-Driven Diffusion in Computer Vision*, B. ter Haar Romeny, Ed. Boston, MA: Kluwer, 1994, pp. 93–133.
[28] Y. L. You, W. Zhu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion," *IEEE Trans. Image Processing*, vol. 5, pp. 1539–1553, 1996.



Anthony Yezzi, Jr. (S'71–M'72) was born in Florida in 1972. He received the B.S. degree (summa cum laude) and the Ph.D. degree in electrical engineering in 1994 and 1998, respectively, both from the University of Minnesota, Minneapolis. He began a post-doctoral position at the Massachusetts Institute of Technology, Cambridge, in January 1998.

His research interests are in computer vision, image processing, control, and applied differential geometry.

He was awarded the AT&T Graduate Fellowship Award.