Mean Curvature Evolution and Surface Area Scaling in Image Filtering

Adel I. El-Fallah and Gary E. Ford
CIPIC, Center for Image Processing and Integrated Computing
University of California, Davis
Davis, CA 95616
e mail: elfallah@ece.ucdavis.edu ford@ece.ucdavis.edu
Tel: (916)752-2387 Fax: (916)752-8894

Abstract

We propose a geometrical surface representation for an image, and introduce a nonlinear adaptive filter that diffuses the surface in time at a speed proportional to the mean curvature. Linear variations in intensity (edges) are inclined planes of vanishing mean curvature, and are thus invariant. Noise is characterized by high mean curvature and will be diffused. We show that this diffusion resolves the conflict of removing noise while preserving edges. A novel nonlinear scale space filtering relating surface area to the diffusion speed is introduced resulting in very efficient algorithms. Experiments demonstrating excellent performance and efficiency are presented.

1 Introduction

In recent works [1, 2, 3] we proposed the use of geometry in image processing by representing an intensity image as a surface in 3-space. The linear variations in intensity (edges) were shown to have a nondivergent surface normal. Exploiting this feature we introduced a nonlinear adaptive filter that only averages the divergence in the direction of the surface normal. Employing the inhomogeneous diffusion model introduced by Perona and Malik [4] led to an inhomogeneous diffusion (ID) that averages the mean curvature of the surface, and which has many interesting geometrical properties that are related to minimal surfaces (surfaces of least potential), the Plateau problem, and soap bubbles. In particular, it renders edges invariant while reducing the surface area (thereby removing noise) and imposing regularity everywhere. This mean curvature diffusion (MCD) when applied to an isolated edge embedded in additive Gaussian noise results in complete noise removal and edge enhancement with the edge location left intact. This performance is obtained even for inputs having *negative* SNR (expressed in dB), and the algorithms are data driven with no parameters or thresholds to set other than how far in the future is the surface to evolve.

In this paper we generalize our previous formulation of MCD to include a scaling function dependent on time which can be adaptively adjusted based on changes in surface area. This leads to scaled MCD with strong averaging early in the iterative process, when the averaging is very local, and less averaging later in the process, when the spatial support for the averaging increases. This results in more averaging with less degradation.

2 Theory of Mean Curvature Diffusion

We consider the three-dimensional Euclidean space denoted by $\mathbf{E}^3$, with a point in the space denoted by $\mathbf{p} = (p_1, p_2, p_3)$. The real-valued functions x, y, and z are the *natural coordinate functions* of $\mathbf{E}^3$ defined such that for each point $\mathbf{p}$: $x(\mathbf{p}) = p_1$, $y(\mathbf{p}) = p_2$, $z(\mathbf{p}) = p_3$. An alternative index notation for the natural coordinate functions is $x_1 = x$, $x_2 = y$, $x_3 = z$. The symbol $\sum$ will be used to indicate a sum over $i = 1, 2, 3$. The function $I(x, y)$ is a real-valued function on $\mathbf{E}^2$ representing the grey level of the image, and is assumed differentiable (of at least class C^2).

We characterize the image as a surface $\mathcal{S}$ on $\mathbf{E}^2$

$$\mathcal{S} : g(x, y, z) = z - A\, I(x, y) = B \tag{1}$$

where g is a differentiable real-valued function and A and B are real-valued constants. The differential

$$\mathrm{d}g = \mathrm{d}z - A \frac{\partial I}{\partial x} \mathrm{d}x - A \frac{\partial I}{\partial y} \mathrm{d}y \tag{2}$$

is never zero even at the critical points of I $\left(\frac{\partial I}{\partial x} = \frac{\partial I}{\partial y} = 0\right)$, so from the implicit function theorem, the equation $g(x, y, z) = B$ can be solved for z near a point $\mathbf{p}$. Thus, $z = A\, I(x, y) + B$ is a surface.

1058-6393/95 $4.00 © 1995 IEEE

The *natural frame field* on $\mathbf{E}^3$ is defined as the three vector fields U_1, U_2, and U_3 on $\mathbf{E}^3$ such that $U_1(\mathbf{p}) = (1,0,0)$, $U_2(\mathbf{p}) = (0,1,0)$, and $U_3(\mathbf{p}) = (0,0,1)$, for each point $\mathbf{p}$ of $\mathbf{E}^3$. Thus U_i $(i = 1,2,3)$ is the unit vector field in the positive x_i direction.

A *curve* in $\mathbf{E}^3$ is the path taken by a moving point α as a function of the real scalar variable u.

$$\alpha(u) = (\alpha_1(u), \alpha_2(u), \alpha_3(u)) \tag{3}$$

where α is a differentiable function. For each value of u, the *velocity vector* of α at u is defined as the tangent vector

$$\alpha'(u) = \left(\frac{\mathrm{d}\alpha_1(u)}{\mathrm{d}u}, \frac{\mathrm{d}\alpha_2(u)}{\mathrm{d}u}, \frac{\mathrm{d}\alpha_3(u)}{\mathrm{d}u}\right)_{\alpha(u)} \tag{4}$$

at the point $\alpha(u)$ in $\mathbf{E}^3$.

The *gradient* vector field for the surface $\mathcal{S}$

$$\nabla g = \sum \frac{\partial g}{\partial x_i} \mathrm{U}_i \tag{5}$$

is a non-vanishing *normal* vector field on the entire surface. This can be shown by first recognizing that the gradient ∇g is non-vanishing (never zero) on $\mathcal{S}$ since the partial derivatives $\frac{\partial g}{\partial x_i}$ cannot simultaneously be zero at any point of $\mathcal{S}$. If α is a curve in $\mathcal{S}$, then

$$g(\alpha) = g(x_1, x_2, x_3) = g(\alpha_1(u), \alpha_2(u), \alpha_3(u)) \tag{6}$$

has constant value equal to B. Thus by the chain rule

$$\sum \frac{\partial g}{\partial x_i} \frac{\mathrm{d}x_i}{\mathrm{d}u} = 0.$$

Now choose α to have initial velocity $\alpha'(0) = \mathbf{v} = (v_1, v_2, v_3)$ at $\alpha(0) = \mathbf{p}$. Then

$$\sum \frac{\partial g}{\partial x_i}(\mathbf{p})\, v_i = (\nabla g)(\mathbf{p}) \bullet \mathbf{v} = 0$$

Thus, we have shown that $(\nabla g)(\mathbf{p}) \bullet \mathbf{v} = 0$ for every tangent vector $\mathbf{v}$ to $\mathcal{S}$ at $\mathbf{p}$.

2.1 Mean Curvature Diffusion

The diffusion of g is modeled by

$$\frac{\partial g}{\partial t} = \nabla \bullet (C\,\nabla g). \tag{7}$$

We are interested in the properties of inhomogeneous diffusion in which the diffusion coefficient is the inverse of the surface gradient magnitude

$$C = \frac{1}{|\nabla g|} \tag{8}$$

Substitute this in (7) to obtain the diffusion

$$\frac{\partial g}{\partial t} = \nabla \bullet \left(\frac{\nabla g}{|\nabla g|}\right). \tag{9}$$

Since the gradient ∇g is a non-vanishing normal vector field, the quantity

$$\mathcal{N} \triangleq \frac{\nabla g}{|\nabla g|} \tag{10}$$

is a ***unit normal vector field*** on a neighborhood of $\mathbf{p}$ in $\mathcal{S}$. Rewriting (7)

$$\frac{\partial g}{\partial t} = \nabla \bullet \mathcal{N} = \frac{\partial \mathcal{N}_1}{\partial x} + \frac{\partial \mathcal{N}_2}{\partial y}. \tag{11}$$

The first term on the right is the normal curvature in the x direction and the second term is the normal curvature in the y direction. The ***mean curvature*** H is the average value of the normal curvature in ***any*** two orthogonal directions, and since the directions x and y are orthogonal, we obtain

$$\frac{\partial g}{\partial t} = 2\mathrm{H} \tag{12}$$

2.2 MCD vs. Homogeneous Diffusion

In homogeneous diffusion, the diffusion coefficient C in (7) is constant and the diffusion equation becomes

$$\frac{\partial g}{\partial t} = C\,\nabla^2 g \tag{13}$$

where the Laplacian $\nabla^2 g$ is the divergence of the gradient of g: $\nabla^2 g = \nabla \bullet (\nabla g)$. The gradient of g at $\mathbf{p}$ can be written as $\nabla g(\mathbf{p}) = |\nabla g|(\mathbf{p})\,\mathcal{N}(\mathbf{p})$, with $\mathcal{N}(\mathbf{p})$ being the normal unit vector. In what follows we will omit the point of application $\mathbf{p}$, with the understanding that the formulas are evaluated at $\mathbf{p}$ *on* the surface. Decomposing the Laplacian using the identity $\nabla \bullet (f\mathbf{F}) = \nabla f \bullet \mathbf{F} + f(\nabla \bullet \mathbf{F})$, with $f = |\nabla g|$ and $\mathbf{F} = \mathcal{N}$, we obtain

$$\begin{aligned} \nabla^2 g &= \nabla(|\nabla g|) \bullet \mathcal{N} + |\nabla g|(\nabla \bullet \mathcal{N}) \\ &= \nabla(|\nabla g|) \bullet \mathcal{N} + |\nabla g|(2\mathrm{H}) \end{aligned} \tag{14}$$

where 2H is substituted for $\nabla \bullet \mathcal{N}$ (Section 2.1). Letting the variable n be the coordinate in the direction of the surface normal then

$$\nabla(|\nabla g|) \bullet \mathcal{N} = \nabla\left(\frac{\partial g}{\partial n}\right) \bullet \mathcal{N} = \frac{\partial^2 g}{\partial n^2}, \tag{15}$$

and using this equation in (13) and (14) we obtain

$$\frac{\partial g}{\partial t} = C\left\{\frac{\partial^2 g}{\partial n^2} + |\nabla g|(2\mathrm{H})\right\}. \tag{16}$$

Homogeneous diffusion is driven by the Laplacian which is the divergence of the gradient. This involves *two types of divergences,* magnitude and direction. The coupling of the two types is given in (16) which also indicates that the magnitude type divergence is represented by: $|\nabla g| = \frac{\partial g}{\partial n}$ and $\frac{\partial^2 g}{\partial n^2}$ whereas the direction type divergence is represented by 2H (Section 2.1). This is in contrast to the inhomogeneous diffusion of (12), which is solely dependent on the divergence in direction, 2H.

3 Properties of MCD

In this section we briefly discuss mean curvature diffusion properties and their relationship to image filtering. The complete details of the algorithm and further properties can be found in [1, 2].

3.1 Stability

Applying the gradient operator to the equation characterizing the image as a surface (1), we have

$$\nabla g = \nabla z - A\,\nabla I, \tag{17}$$

Applying (5) and taking the magnitude, we find that the diffusion coefficient of (8) becomes

$$C = \frac{1}{|\nabla g|} = \frac{1}{\sqrt{1 + A^2\,|\nabla I|^2}}. \tag{18}$$

In addition to having an important geometric interpretation, this coefficient leads to a smooth stable filter as its value lies in the interval [0,1].

3.2 The Scaling Parameter

From (1) the parameter A is a scaling factor for the image intensity, which also becomes a scaling parameter for the coefficient in (18). Edges satisfying $|\nabla I| \gg 1/A$ are considered informative and determine the scale of MCD. The smaller the value of A, the greater the diffusion in each iteration and the faster the surface evolves. For $A = 0$ the diffusion coefficient is constant and the diffusion becomes homogeneous. This is consistent, since from (1), the surface is the plane $z =$ B, a non-informative flat region.

3.3 MCD Evolution of a Noisy Step

The evolution of a noisy version of a ramp is shown in Fig. 1. This MCD of the ramp results in complete noise removal and edge enhancement with the location of the edge left intact [1, 2].

4 Surface Area

Observing the evolution of the noisy step in Fig. 1, it is apparent that changes in surface area with time indicate the state of the evolution. In this section we exploit this observation and introduce a new measure

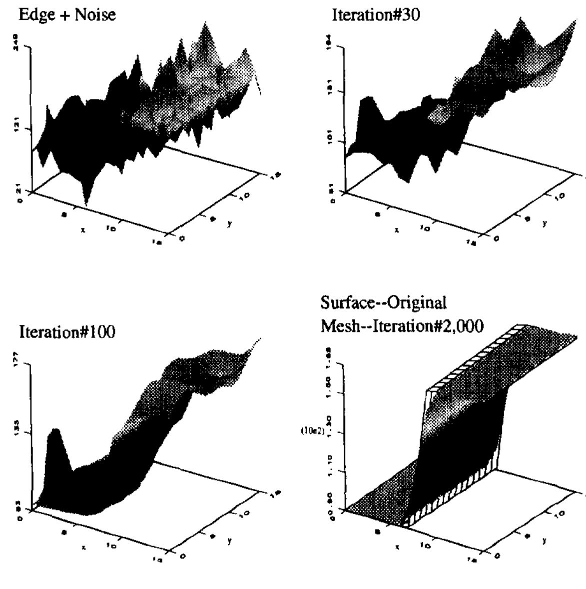

Figure 1: The *evolution* of the noisy edge surface.

relating these changes to the adaptive adjustment of the scaling parameter A of (1). This leads to an efficient algorithm in which the diffusion speed is adaptively controlled.

4.1 Changes in Surface Area

Since edges are preserved and enhanced by MCD, then any surface area reduction must be due to the noise removal produced by the algorithm and is *not* due to the averaging of an edge. Similarly any increase in the surface area is due to the enhancement of edges and is *not* due noise enhancement. The two processes of averaging and enhancement occur simultaneously and thus the measure of the state of the evolution will be based on the ***absolute*** value of changes in surface area. The total absolute percentage change in surface area produced while evolving from time $(t-1)$ to time (t) will be denoted by $\mathcal{L}(t)$.

The theory in Section 2 is still valid if the constant A is replaced by a function $A(t)$, leading to a modified diffusion coefficient

$$C = \frac{1}{|\nabla g|} = \frac{1}{\sqrt{1 + A^2(t)\;|\nabla I|^2}}. \tag{19}$$

If a diffusion iteration results in a large change in surface area, we propose that the algorithm be accelerated and the averaging and enhancement be increased. This is achieved by choosing $A^2(t)$ in (19) to be a monotonic decreasing function of $\mathcal{L}(t)$. This strategy is very effective as it leads to strong averaging and enhancement early in time when the spatial scale is small, resulting in less degradation than might develop later in time when the spatial scale becomes large. It is also effective later in the evolution, when considerable averaging and enhancement have already occurred, resulting in a less noisy and more enhanced image, making $\mathcal{L}(t)$ small and $A^2(t)$ large, which in turn leads to less averaging and enhancement and less degradation, as desired.

4.2 Determining the Scaling Function

A point $\mathbf{p}$ lying on the image surface can be represented by the tip of the position vector

$$\mathbf{r}(x, y) = x(\mathbf{p})\, U_1(\mathbf{p}) + y(\mathbf{p})\, U_2(\mathbf{p}) + z(\mathbf{p})\, U_3(\mathbf{p}) \quad (20)$$

where z is a linear function of the intensity (1). The x-parameter curve for constant y is shown in Fig. 2. For a small increment Δx the *curve length* from P_1

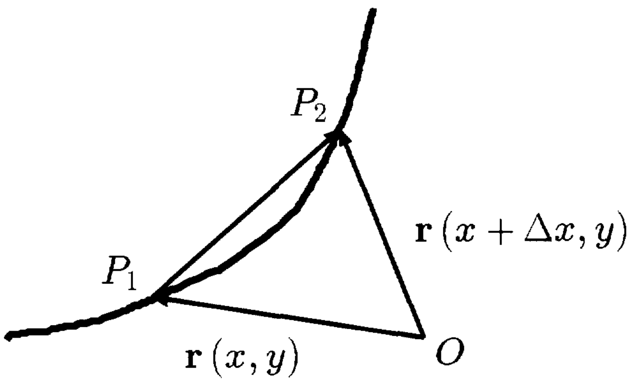

Figure 2: The x-parameter curve: y = constant, lying on the image surface.

to P_2 is approximately the length of the *line segment* P_1P_2. This line segment has a length of $|\mathbf{r}_x\, \Delta x|$, where $\mathbf{r}_x \equiv \frac{\partial \mathbf{r}}{\partial x}$, the velocity (tangent) vector to the x-parameter curve with y constant. This follows directly from the definition of a derivative since the vector $\overrightarrow{P_1P_2} = \mathbf{r}(x + \Delta x, y) - \mathbf{r}(x, y)$ approaches $\mathbf{r}_x(x, y)\, \Delta x$ as $\Delta x \longrightarrow 0$. Similarly, a change Δy is approximately mapped by the surface into $\mathbf{r}_y \Delta y$, where $\mathbf{r}_y$ is the velocity (tangent) vector to the y-parameter curve, with x constant. A small rectangle with sides Δx and Δy is mapped by the surface $\mathbf{r}(x, y)$ into a small *curved* region which can be linearly approximated by a parallelogram with sides $\mathbf{r}_x \Delta x$ and $\mathbf{r}_y \Delta y$ having area

$$\Delta \mathcal{A} = |\mathbf{r}_x \Delta x \times \mathbf{r}_y \Delta y| = |\mathbf{r}_x \times \mathbf{r}_y|\, \Delta x \Delta y. \quad (21)$$

The cross product $\mathbf{r}_x \times \mathbf{r}_y$ is a vector orthogonal to both $\mathbf{r}_x$ and $\mathbf{r}_y$ and hence normal to the surface, implying that it can be replaced by the surface gradient, shown in Section 2.1 to be a normal vector field on the entire surface. The change in area becomes

$$\Delta \mathcal{A} = |\nabla g|\, \Delta x \Delta y. \quad (22)$$

In evolving from $(t-1)$ to t, the total absolute change in the image surface area is

$$\int\!\!\int \Big| |\nabla g(x, y; t)| - |\nabla g(x, y; t-1)| \Big|\, dx dy \quad (23)$$

and the measure $\mathcal{L}(t)$ is taken to be the percentage change in this quantity in one iteration. We have found experimentally that a good choice for an adaptive scaling function is

$$A^2(t) = \frac{1}{\mathcal{L}(t)}. \quad (24)$$

We initially set $A^2(0)$ to unity, which produces small averaging and enhancement, and assumes a non noisy input. This is a conservative approach, and if the assumption is invalid, then in time the algorithm will adaptively alter $A(t)$ to a lower value.

The number of iterations at which to terminate the diffusion process is the one parameter that is left to the user. For most applications the number of iterations needed is between four and ten. There is little risk of degradation if the algorithm is run for longer periods, as $A(t)$ soon becomes large, effectively halting the diffusion.

4.3 Evolution Speed

The function $\mathcal{L}(t)$ defined above can be regarded as the evolution speed. We use the term evolution rather than diffusion since as we previously discussed averaging as well as enhancement is produced. We return to the noisy edge example to demonstrate the increase in algorithm speed resulting from the adaptive scaling function.

Application of MCD with the adaptive scaling function of (24) to the noisy edge of Section 5.1 is shown in Fig. 3. The increased speed in surface area evolution is observed by comparison with Fig. 1. The scaling function for this example is shown in Fig. 4. The inverse of this function is the evolution speed, which is initially set to 1, rises to 8.73 at time 5, and then smoothly decreases, reaching 0.24 at time 200. This is in contrast with the fixed speed of 1 for the example in Section 3.

5 Image Noise Removal

In Table 1, the noise variance and PSNR are given for the original and noisy images and for the images produced by four and ten iterations of the adaptively

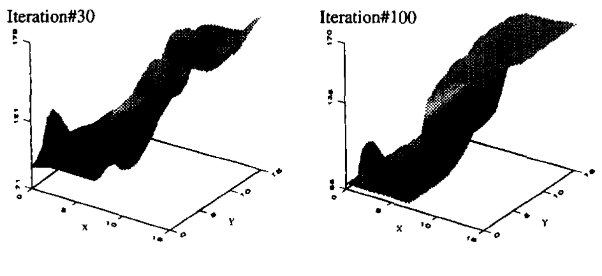

Figure 3: The Result of Scaled MCD

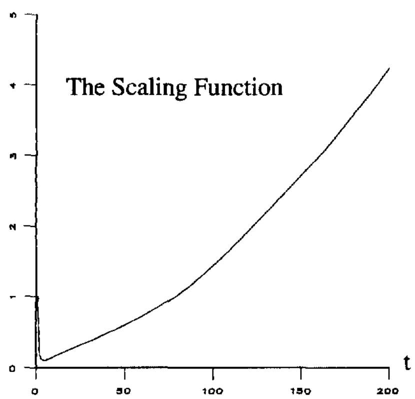

Figure 4: The Scaling Function for the Noisy Step

scaled MCD algorithm. The PSNR dropped by 12 dB after noise addition, but was nearly 10 dB above that of the original image after only four iterations and nearly 20 dB above after ten iterations.

Image	Variance	PSNR
Original	1.2757	45.99
Noisy Input	24.0227	34.01
Iteration 4	0.1867	54.97
Iteration 10	0.0173	65.22

Table 1: Quantitative performance of the adaptively scaled MCD algorithm

The differences in the gray-scale renditions of the images were difficult to detect, so edge maps having a fixed threshold are shown in Fig. 5. Note the excellent preservation of edges in the processed images.

6 Conclusions

The interpretation of an image as a surface provides the basis for the development of a new formulation for inhomogeneous diffusion, in which the diffusion coefficient is the inverse of the magnitude of the surface normal and the image surface evolves with a speed proportional to the mean curvature. This averaging preserves image edges, as their mean curvature is zero, while noise is averaged since it has high mean curvature. A nonlinear scaling function was introduced that allows for algorithms with more averaging power and less degradation.

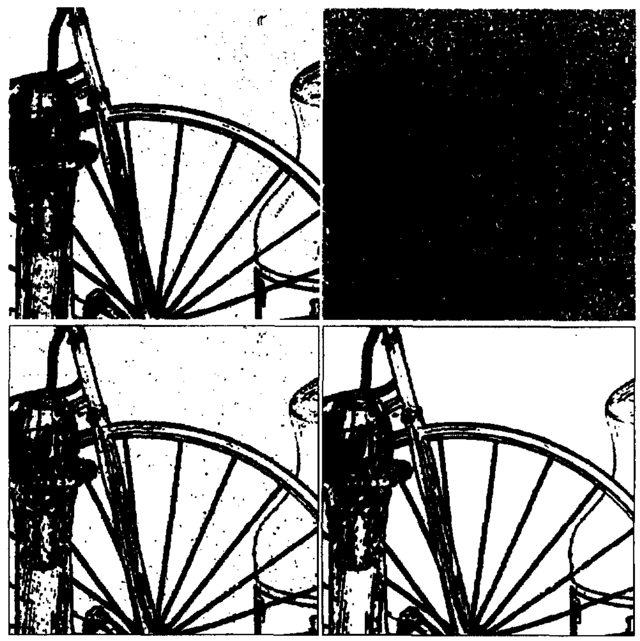

Figure 5: Edge maps of the original image, noisy input image, and adaptively scaled MCD filtered output images for iterations four and ten.

References

[1] A. I. El-Fallah and G. E. Ford, "Structure preserving inhomogeneous diffusion in image filtering," in *27th Asilomar Conf. Signals, Syst., Comput.*, (Pacific Grove, CA), pp. 563–567, Nov. 1993.

[2] A. I. El-Fallah and G. E. Ford, "Nonlinear adaptive image filtering based on inhomogeneous diffusion and differential geometry," in *SPIE Proceedings*, vol. 2182, pp. 49–63, 1994.

[3] A. I. El-Fallah and G. E. Ford, "Mean curvature evolution and surface area scaling in image filtering," *Submitted to IEEE Trans. Image Processing*, 1994.

[4] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, pp. 629–639, 1990.