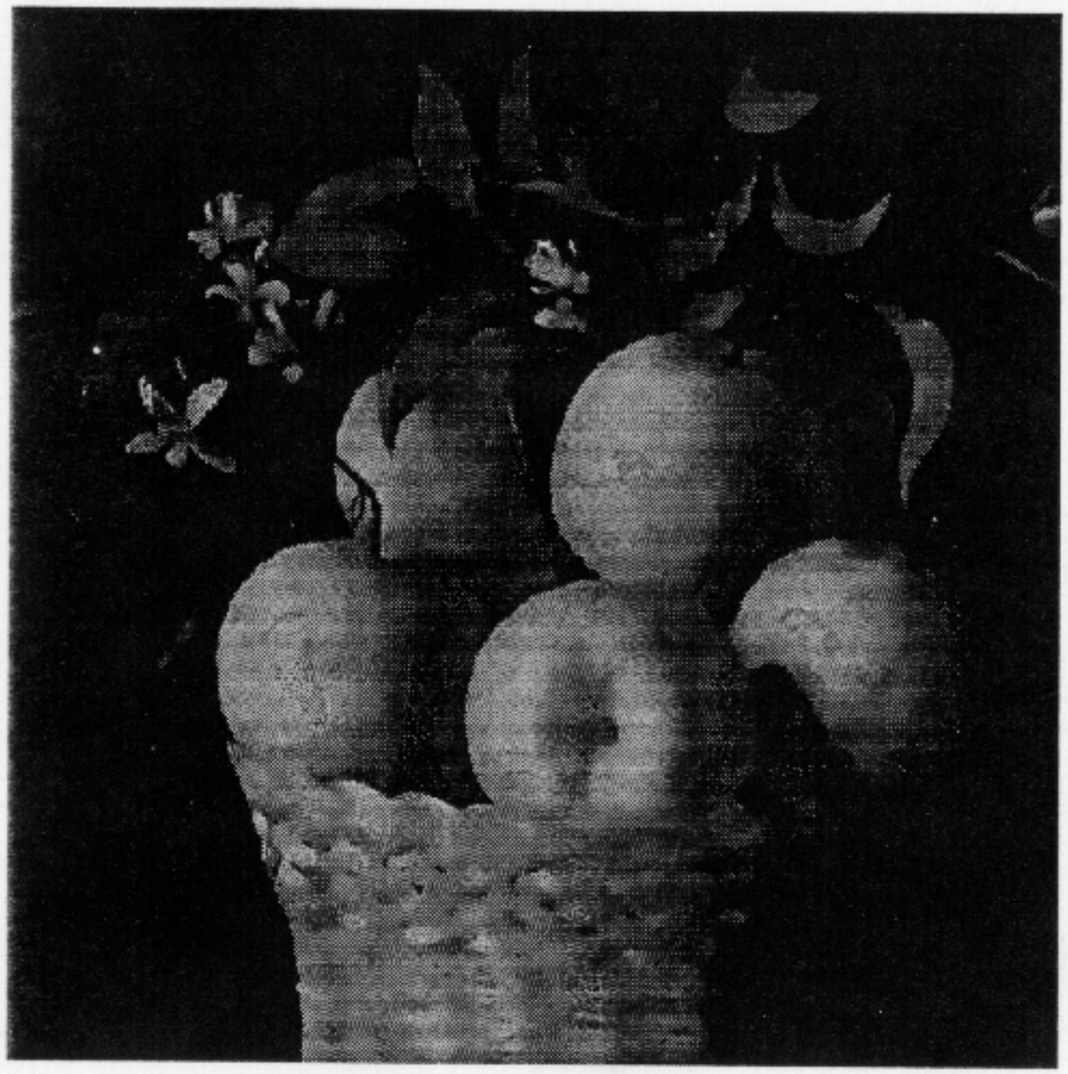

*Figure 2.12.* Anisotropic diffusion of a basket of fruits (implementation: Perona). Detail of a painting by Zurbaran to be found in the Norton-Simon Museum of Pasadena. The diffusion nonlinearity is the exponential, 20 iterations. The value of 'K' was about 10% of the dynamic range. Details of anisotropic diffusion are discused in next chapter.

# ANISOTROPIC DIFFUSION

**Pietro Perona**
*California Institute of Technology*
*Dept. of Electrical Engineering*
*Pasadena, CA 91125, USA*

and

**Takahiro Shiota**
*Kyoto University*
*Math Department, School of Science*
*Sakyo-ku, Kyoto 606-01, Japan*

and

**Jitendra Malik**
*University of California at Berkeley*
*Department of Electrical Engineering and Computer Science*
*Berkeley, CA 94720, USA*

## 1. Introduction

The importance of multi-scale descriptions of images has been recognized from the early days of computer vision e.g. Rosenfeld and Thurston [309]. A clean formalism for this problem is the idea of scale-space filtering introduced by Witkin [386] and further developed in Koenderink[187], Babaud, Duda and Witkin[24], Yuille and Poggio[396], and Hummel[163, 164] and reviewed in the earlier sections of this book.
The essential idea of this approach is quite simple: embed the original image in a family of derived images $I(x, y, t)$ obtained by convolving the original image $I_0(x, y)$ with a Gaussian kernel $G(x, y; t)$ of variance $t$:

$$I(x, y, t) = I_0(x, y) * G(x, y; t) \tag{3.1}$$

Larger values of $t$, the scale-space parameter, correspond to images at coarser resolutions (see Fig. 3.1 and 3.2).

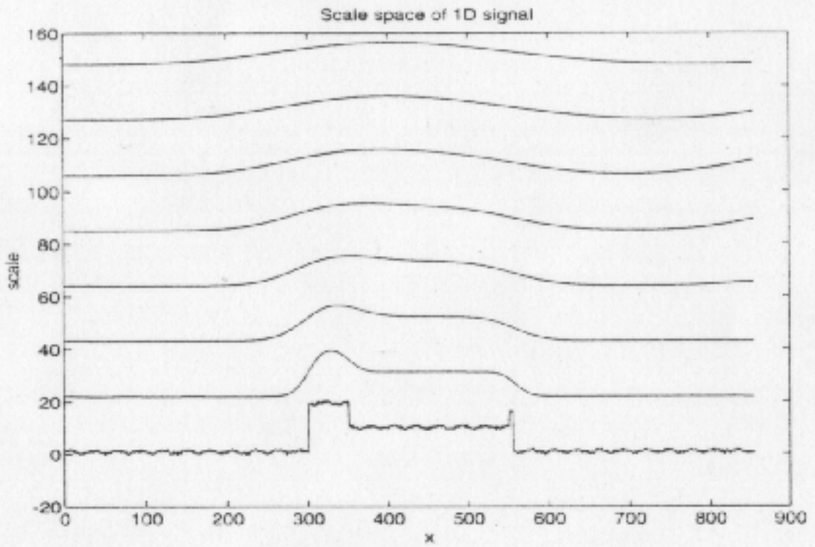


*Figure 3.1.* (Left) A family of 1-D signals, $I(x,t)$, obtained by convolving the original one (lowest one) with Gaussian kernels whose variance increases from bottom to top. (Right) 'Fingerprint' of $I(x,t)$; it is the locus of the zero crossings of $\nabla^2 I(x,t)$. The zero crossings that are associated to strong gradients may be used to represent the boundaries of the main regions at each scale. Notice, however, that their position changes with scale.



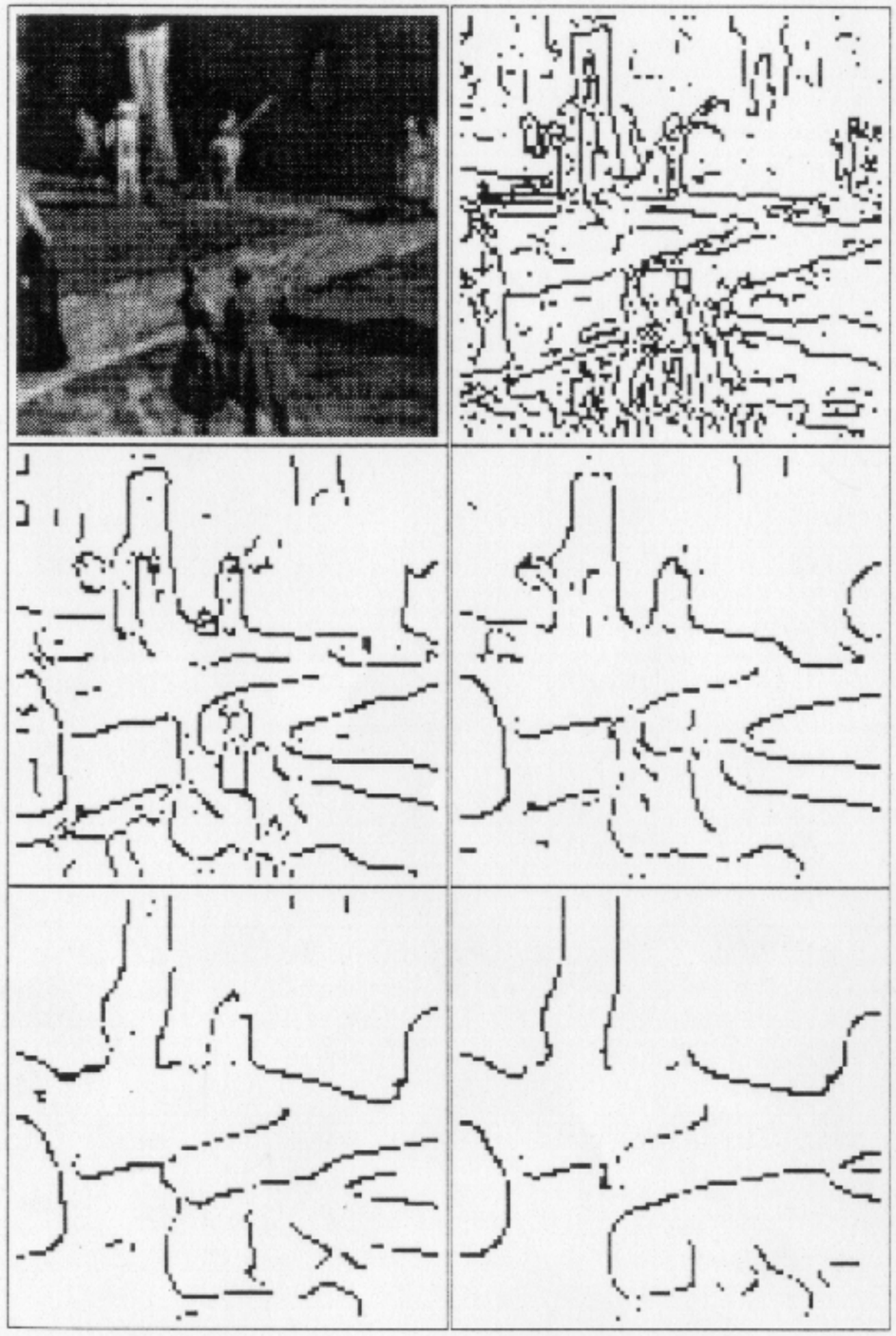

*Figure 3.2.* Sequence of boundary images produced by the Canny detector run at scales 1, 3, 5, 7, 9 pixels. The images are $100 \times 100$ pixels.

As pointed out by Koenderink [187], and Hummel [163] this one parameter

family of derived images may equivalently be viewed as the solution of the heat conduction or diffusion equation

$$I_t = \nabla^2 I = I_{xx} + I_{yy} \tag{3.2}$$

with the initial condition $I(x, y, 0) = I_0(x, y)$, the original image.
Koenderink motivates the diffusion equation formulation by stating two criteria :

1. *Causality :* Any feature at a coarse level of resolution is required to possess a (not necessarily unique) "cause" at a finer level of resolution although the reverse need not be true. In other words, no spurious detailshould be generated when the resolution is diminished.
2. *Homogeneity and Isotropy :* The blurring is required to be space invariant.

These criteria lead naturally to the diffusion equation formulation. It may be noted that the second criterion is only stated for the sake of simplicity; if blurring was allowed to be data-driven and space-varying one would have useful degrees of freedom to play with.
It should also be noted that the causality criterion does not force uniquely the choice of a Gaussian to do the blurring, though this is perhaps the simplest. Hummel [163] has made the important observation that a version of the maximum principle from the theory of parabolic differential equations is equivalent to causality, therefore one would expect that a number of (possibly nonlinear) differential equations would satisfy causality and possibly have useful behaviours for vision applications.
Is the standard scale-space paradigm adequate for vision tasks which need 'semantically meaningful' multiple scale descriptions?
Surfaces in nature usually have a hierarchical organization composed of a small discrete number of levels [247]. At the finest level, a tree is composed of leaves with an intricate structure of veins. At the next level, each leaf is replaced by a single region, and at the highest level there is a single blob corresponding to the treetop. There is a natural range of resolutions (intervals of the scale-space parameter) corresponding to each of these levels of description. Furthermore at each level of description, the regions (leaves, treetops or forests) have well-defined boundaries.
In the linear scale-space paradigm the true location of a boundary at a coarse scale is not directly available at the coarse scale image. This can be seen clearly in the 1-D example in Figure 3.1 and in the 2D example in Figure 3.2. The locations of the edges at the coarse level are shifted from their true locations. In 2-D images there is the additional problem that edge junctions , which contain much of the spatial information of the edge drawing, are destroyed. The only way to obtain the true location of the edges that have been detected at a coarse scale is by tracking them across



the scale-space to their position in the original image. This technique proves to be somewhat complicated and expensive [31, 62].
The reason for this spatial distortion is quite obvious – Gaussian blurring does not 'respect' the natural boundaries of objects. Suppose we have the picture of a treetop with the sky as background. The Gaussian blurring process would result in the green of the leaves getting 'mixed' with the blue of the sky, long before the treetop emerges as a feature (after the leaves have been blurred together). In Figure 3.2 the boundaries in a sequence of coarsening images obtained by Gaussian blurring illustrate this phenomenon.
With this as motivation, we enunciate the criteria which we believe any candidate paradigm for generating multi-scale 'semantically meaningful' descriptions of images must satisfy:

1. **Causality** : As pointed out by Witkin and Koenderink, a scale-space representation should have the property that no 'spurious detail' should be generated passing from finer to coarser scales.
2. **Immediate Localization**: At each resolution, the region boundaries each resolution, the region boundaries should be sharp and coincide with the semantically meaningful boundaries at that resolution.
3. **Piecewise Smoothing**: At all scales, intra-region smoothing should occur preferentially over inter-region smoothing. In the tree example mentioned earlier, the leaf regions should be collapsed to a treetop *before* being merged with the sky background.

## 2. Anisotropic diffusion

There is a simple way of modifying the scale-space paradigm to achieve the objectives that we have put forth in the previous section. In the diffusion equation framework of looking at linear scale-space, the diffusion coefficient $c$ is assumed to be a constant independent of the spatial location. There is no fundamental reason why this must be so. To quote Koenderink [187], (pg. 364, left column, l. 19 from the bottom) "... I do not permit space variant blurring. Clearly this is not essential to the issue, but it simplifies the analysis greatly". We argue that a suitable choice of space- and scale-varying $c(x, y, t)$ ($t$ indicates the scale) will enable us to satisfy the second and third criteria listed in the previous section. Furthermore this can be done without sacrificing the causality criterion.
Consider the "anisotropic diffusion" equation [291, 293]

$$I_t = div(c(x, y, t)\nabla I) = c(x, y, t)\nabla^2 I + \nabla c \cdot \nabla I \tag{3.3}$$

where we indicate with $div$ the divergence operator, and with $\nabla$, and $\nabla^2$ the gradient, and Laplacian operators with respect to the space variables.

It reduces to the isotropic heat diffusion equation $I_t = c\nabla^2 I$ if $c(x,y,t)$ is a constant. Suppose at the 'time' $t$ (the scale is in this chapter sometimes called 'time' as time is the physical evolution parameter of the diffusion), we knew the locations of the region boundaries appropriate for that scale. We would want to encourage smoothing *within* a region in preference to smoothing *across* the boundaries. This could be achieved by setting the conduction coefficientto be 1 in the interior of each region and 0 at the boundaries. The blurring would then take place separately in each region with no interaction between regions. The region boundaries would remain sharp. Of course, we do *not* know in advance the region boundaries at each scale (if we did the problem would already have been solved!). What can be computed is a current best estimate of the location of the boundaries (edges) appropriate to that scale.

Let $\mathbf{E}(x,y,t)$ be such an estimate: a vector-valued function defined on the image which ideally should have the following properties:

1. $\mathbf{E}(x,y,t) = \mathbf{0}$ in the interior of each region.
2. $\mathbf{E}(x,y,t) = K\mathbf{e}(x,y,t)$ at each edge point, where $\mathbf{e}$ is a unit vector normal to the edge at the point, and $K$ is the local contrast (difference in the image intensities on the left and right) of the edge.

Note that the word *edge* as used above has not been formally defined – we mean here the perceptual subjective notion of an edge as a region boundary. A completely satisfactory formal definition is likely to be part of the solution, rather than the problem definition!

If an estimate $\mathbf{E}(x,y,t)$ is available, the conduction coefficient $c(x,y,t)$ can be chosen to be a function $c = g(\|\mathbf{E}\|)$ of the magnitude of E. According to the previously stated strategy $g(\cdot)$ has to be a nonnegative monotonically decreasing function with $g(0) = 1$ (see Fig. 3.3). This way the diffusion process will mainly take place in the interior of regions, and it will not affect the region boundaries where the magnitude of E is large.

It is intuitive that the success of the diffusion process in satisfying the three scale-space goals of section 1 will greatly depend on how accurate the estimate $\mathbf{E}$ is as a "guess" of the edges. Accuracy though is computationally expensive and requires complicated algorithms. Fortunately it turns out that the simplest estimate of the edge positions, the gradient of the brightness function i.e. $\mathbf{E}(x,y,t) = \nabla I(x,y,t)$, gives excellent results in many useful cases

$$c(x,y,t) = g(\|\nabla I(x,y,t)\|) \tag{3.4}$$

There are many possible choices for $g(\cdot)$, the most obvious being a binary valued function. In the next sections we argue that in case we use the edge estimate $\mathbf{E}(x,y,t) = \nabla I(x,y,t)$ the choice of $g(\cdot)$ is restricted to a subclass of the monotonically decreasing functions and that $g$ should be smooth.



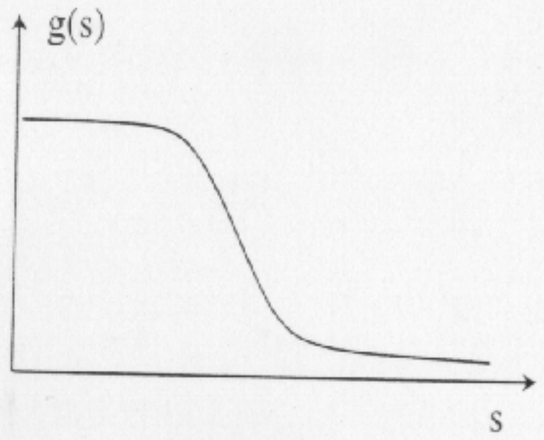


*Figure 3.3.* The qualitative shape of the nonlinearity $g(\cdot)$.

## 3. Implementation and discrete maximum principle

Equation (3.3) may be discretized on a square lattice, with brightness values associated to the vertices, and conduction coefficients to the arcs (see Fig. 3.4). A 4-nearest-neighbours discretization of the Laplacian operator may be used:

$$I_{i,j}^{t+1} = I_{i,j}^{t} + \lambda[c_N \cdot \Delta_N I + c_S \cdot \Delta_S I + c_E \cdot \Delta_E I + c_W \cdot \Delta_W I]_{i,j}^{t} \tag{3.5}$$

where $0 \le \lambda \le \frac{1}{4}$ for the numerical scheme to be stable, $N,S,E,W$ are the mnemonic subscripts for North, South, East, West, the superscript and subscripts on the square bracket are applied to all the terms it encloses, and the symbol $\Delta$ indicates nearest-neighbour differences:

$$\begin{aligned} \Delta_N I_{i,j} &\equiv I_{i-1,j} - I_{i,j} & \Delta_S I_{i,j} &\equiv I_{i+1,j} - I_{i,j} \\ \Delta_E I_{i,j} &\equiv I_{i,j+1} - I_{i,j} & \Delta_W I_{i,j} &\equiv I_{i,j-1} - I_{i,j} \end{aligned} \tag{3.6}$$

The conduction coefficients are updated at every iteration as a function of the brightness gradient:

$$\begin{aligned} c_{N_{i,j}}^{t} &= g(\|(\nabla I)_{i+\frac{1}{2},j}^{t}\|) & c_{S_{i,j}}^{t} &= g(\|(\nabla I)_{i-\frac{1}{2},j}^{t}\|) \\ c_{E_{i,j}}^{t} &= g(\|(\nabla I)_{i,j+\frac{1}{2}}^{t}\|) & c_{W_{i,j}}^{t} &= g(\|(\nabla I)_{i,j-\frac{1}{2}}^{t}\|) \end{aligned} \tag{3.7}$$

The value of the gradient can be computed on different neighbourhood structures achieving different compromises between accuracy and locality. The simplest choice consists in approximating the norm of the gradient at

each arc location with the absolute value of its projection along the direction of the arc:

$$c^t_{N_{i,j}} = g(|\Delta_N I^t_{i,j}|) \qquad c^t_{S_{i,j}} = g(|\Delta_S I^t_{i,j}|)$$
$$c^t_{E_{i,j}} = g(|\Delta_E I^t_{i,j}|) \qquad c^t_{W_{i,j}} = g(|\Delta_W I^t_{i,j}|) \tag{3.8}$$

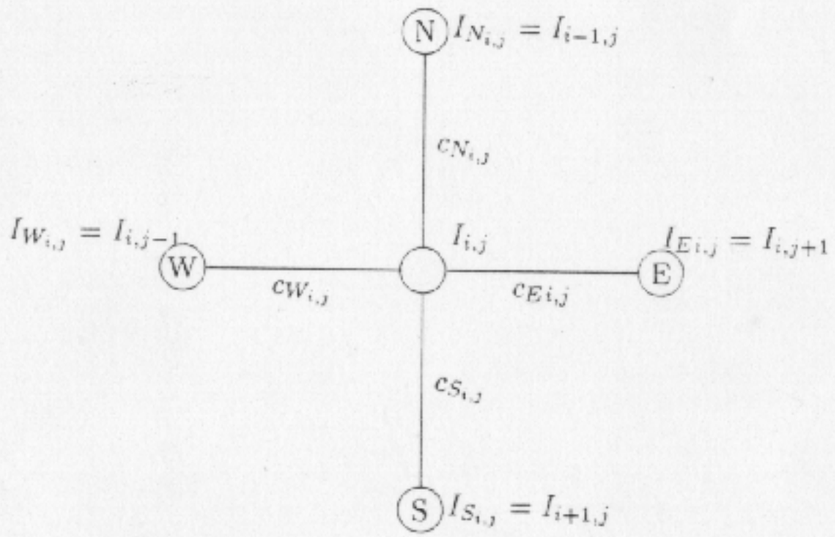


*Figure 3.4.* The structure of the discrete computational scheme for simulating the diffusion equation. The brightness values $I_{i,j}$ are associated to the nodes of a lattice, the conduction coefficients $c$ to the arcs. One node of the lattice and its 4 North, East, West, and South neighbours are shown.

This scheme is not the exact discretization of equation (3.3), but of a similar diffusion equation in which the conduction tensor is diagonal with entries $g(|\ I_x\ |)$ and $g(|\ I_y\ |)$ instead of $g(\|\nabla I\|)$ and $g(\|\nabla I\|)$. This discretization scheme preserves the property of the continuous equation (3.3) that the total amount of brightness in the image is preserved. Additionally the "flux" of brightness through each arc of the lattice only depends on the values of the brightness at the two nodes defining it, which makes the scheme a natural choice for analog VLSI implementations [292, 149].

Less crude approximations of the gradient yielded perceptually similar results at the price of increased computational complexity.

It is possible to show that, whatever the choice of the approximation of the gradient, the discretized scheme still satisfies the maximum (and minimum) principle provided that the function $g$ is bounded between 0 and 1.



```
function [outimage] = anisodiff(inimage,iterations,K)

lambda = 0.25;
outimage = inimage;       [m,n] = size(inimage);

rowC = [1:m];       rowN = [1 1:m-1];       rowS = [2:m m];
colC = [1:n];       colE = [1 1:n-1];       colW = [2:n m];

for i=1:iterations,
deltaN = outimage(rowN,colC) - outimage(rowC,colC);
deltaE = outimage(rowC,colE) - outimage(rowC,colC);

fluxN = deltaN .* exp( - (1/K) * abs(deltaN) );
fluxE = deltaE .* exp( - (1/K) * abs(deltaE) );

outimage = outimage + lambda *
(fluxN - fluxN(rowS,colC) + fluxE - fluxE(rowC,colW));
end;
```

*Figure 3.5.* A Matlab implementation of anisotropic diffusion with adiabatic boundary conditions and exponential nonlinearity.

**Theorem 1 (Discrete maximum-minimum principle)** *If $c \in [0,1]$, the maxima/minima of solutions of (3.5) on a bounded cylinder $(i,j,t) \in (i_0,i_1)\times(j_0,j_1)\times(t_0,t_1)$ belong either to the 'bottom' face $(i,j,t) \in (i_0,i_1)\times(j_0,j_1)\times t_0$ or to the sides ($j = j_0 \times (t_0,t_1)$, $j = j_1 \times (t_0,t_1)$ etc).*

**Proof 1** *We may show this directly from equation (3.5), using the facts $\lambda \in [0,\frac{1}{4}]$, and $c \in [0,1]$, and defining $I^t_{M_{i,j}} \doteq max\{(I, I_N, I_S, I_E, I_W)^t_{i,j}\}$, and $I^t_{m_{i,j}} \doteq min\{(I, I_N, I_S, I_E, I_W)^t_{i,j}\}$, the maximum and minimum of the neighbours of $I_{i,j}$ at iteration $t$. We can see that*

$$(I_m)^t_{i,j} \le I^{t+1}_{i,j} \le (I_M)^t_{i,j} \tag{3.9}$$

*i.e. no (local) maxima and minima are possible in the interior of the discretized scale-space:*

$$\begin{aligned} I^{t+1}_{i,j} &= I^t_{i,j} + \lambda[c_N\cdot\Delta_N I + c_S\cdot\Delta_S I + c_E\cdot\Delta_E I + c_W\cdot\Delta_W I]^t_{i,j} \\ &= I^t_{i,j}(1-\lambda(c_N+c_S+c_E+c_W)^t_{i,j}) + \\ &\quad +\lambda(c_N\cdot I_N + c_S\cdot I_S + c_E\cdot I_E + c_W\cdot I_W)^t_{i,j} \\ &\le I_{M^t_{i,j}}(1-\lambda(c_N+c_S+c_E+c_W)^t_{i,j}) + \\ &\quad +\lambda I_{M^t_{i,j}}(c_N+c_S+c_E+c_W)^t_{i,j} \\ &= I_{M^t_{i,j}} \end{aligned} \tag{3.10}$$

*and, similarly:*

$$I^{t+1}_{i,j} \ge I_{m^t_{i,j}}(1-\lambda(c_N+c_S+c_E+c_W)^t_{i,j}) +$$

$$+\lambda I^t_{m^t_{i,j}}(c_N + c_S + c_E + c_W)^t_{i,j}$$
$$= I^t_{m^t_{i,j}} \tag{3.11}$$

The numerical scheme used to obtain the pictures in this chapter is the one given by equations (3.5), (3.6), (3.8), using the original image as the initial condition, and adiabatic boundary conditions, i.e. setting the conduction coefficient to zero at the boundaries of the image (see in Fig. 3.5 the `Matlab` implementation that was actually used). A constant value for the conduction coefficient $c$ (i.e. $g(\cdot) \equiv 1$) leads to Gaussian blurring.
Different functions may be used for $g(\cdot)$ (see Eq. (3.4)) giving anisotropic diffusion behaviour. The images in this chapter were obtained using

$$g(\nabla I) = e^{-(\frac{\|\nabla I\|}{K})^2} \tag{3.12}$$

however there are other good possibilities, e.g.

$$g(\nabla I) = \frac{1}{1 + (\frac{\|\nabla I\|}{K})^2}$$

The scale-spaces generated by these two functions are different: the first privileges high-contrast edges over low-contrast ones, the second privileges wide regions over smaller ones. The role of the nonlinearity $g$ in determining the edge-enhancing behaviour of the diffusion is studied in a later section. It may be worth noticing that the discrete and the continuous model may have different limit behaviours unless the nonlinearity $g$ is modified in one of the two cases. Since the initial condition is bounded (images are bounded below by 0 – negative brightness does not exist – and above by the saturation of the sensors) and this, in turn, bounds the scale-space, the maximum value for the discrete gradient is bounded above by the dynamic range of the image divided by the spacing of the sampling lattice. Therefore the conduction coefficient will never be zero if the nonlinearities $g$ defined above are used. This will not be the case for the continuous model, where the magnitude of the brightness gradient may diverge and conduction may therefore stop completely at some location.
The constant $K$ may either be set by hand at some fixed value, or set using using the "noise estimator" described by Canny [51]: a histogram of the absolute values of the gradient throughout the image is computed, and $K$ is set equal to the, say, 90% value of its integral at every iteration.
A model of the primary visual cortex due to Cohen and Grossberg [67] aims at achieving the same no-diffusion-across boundaries behaviour. The model contains an explicit representation of the boundaries and is much more ambitious and sophisticated than the simple PDE scheme that we discuss in this chapter.



The computational scheme described in this section has been chosen for its simplicity. Other numerical solutions of the diffusion equation, and multiscale algorithms may be considered for efficient software implementations.

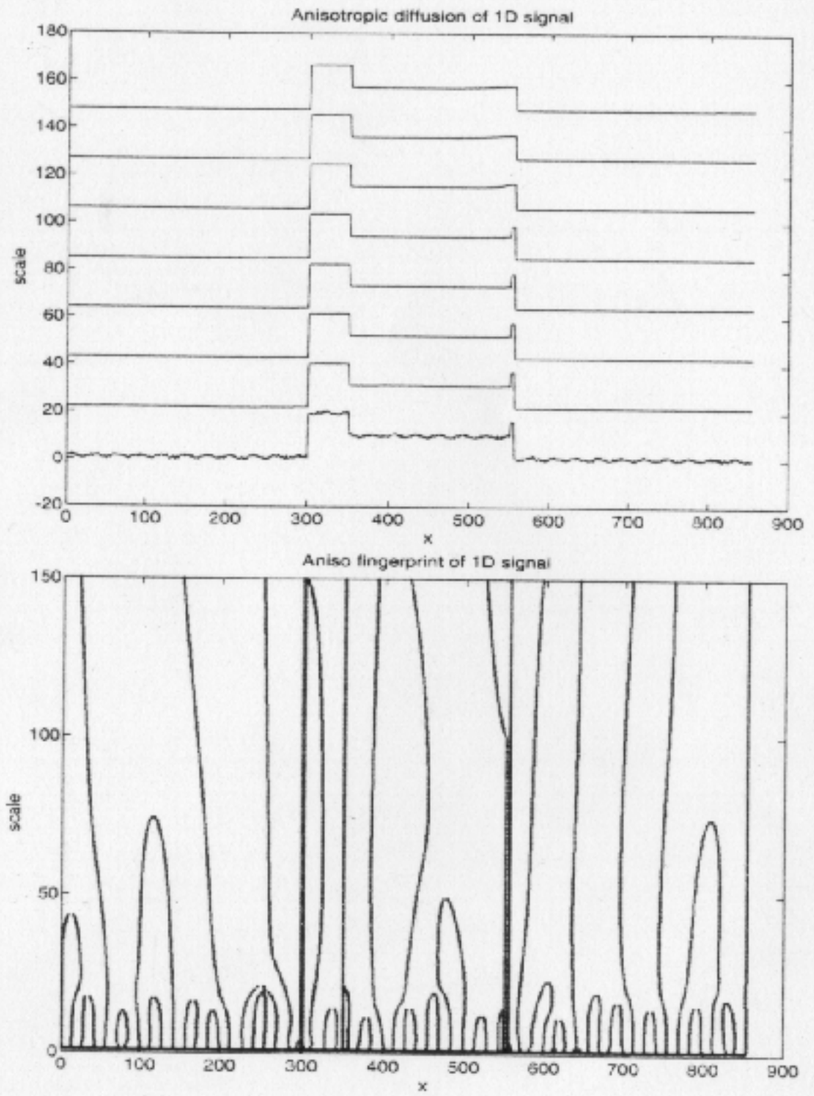


*Figure 3.6.* (Top) A family of 1-D signals, $I(x,t)$, obtained by running anisotropic diffusion (code in figure 3.5) on the original one (lowest one). (Bottom) Fingerprint of the anisotropic diffusion family. It is the locus of the zero crossings of $\nabla^2 I(x,t)$. Notice that the zero-crossings corresponding to high brightness gradients (i.e. edges) have no drift in scale (they are located at positions 300, 350, 550, 555). Around scale 100 the zero-crossing at position 550 undergoes a transition: it ceases having edge value and its position starts drifting in scale-space. Compare with figure 3.1

*Figure 3.7.* Sequence of images produced by anisotropic diffusion. The code presented in figure 3.5 was run on the image at the top-left corner for 10, 20, 30, 60, 100 iterations. The original image has pixel values between 0 (black) and 255 (white) and had a size of $100 \times 100$ pixels. The coefficient K was set equal to $K = 10$.



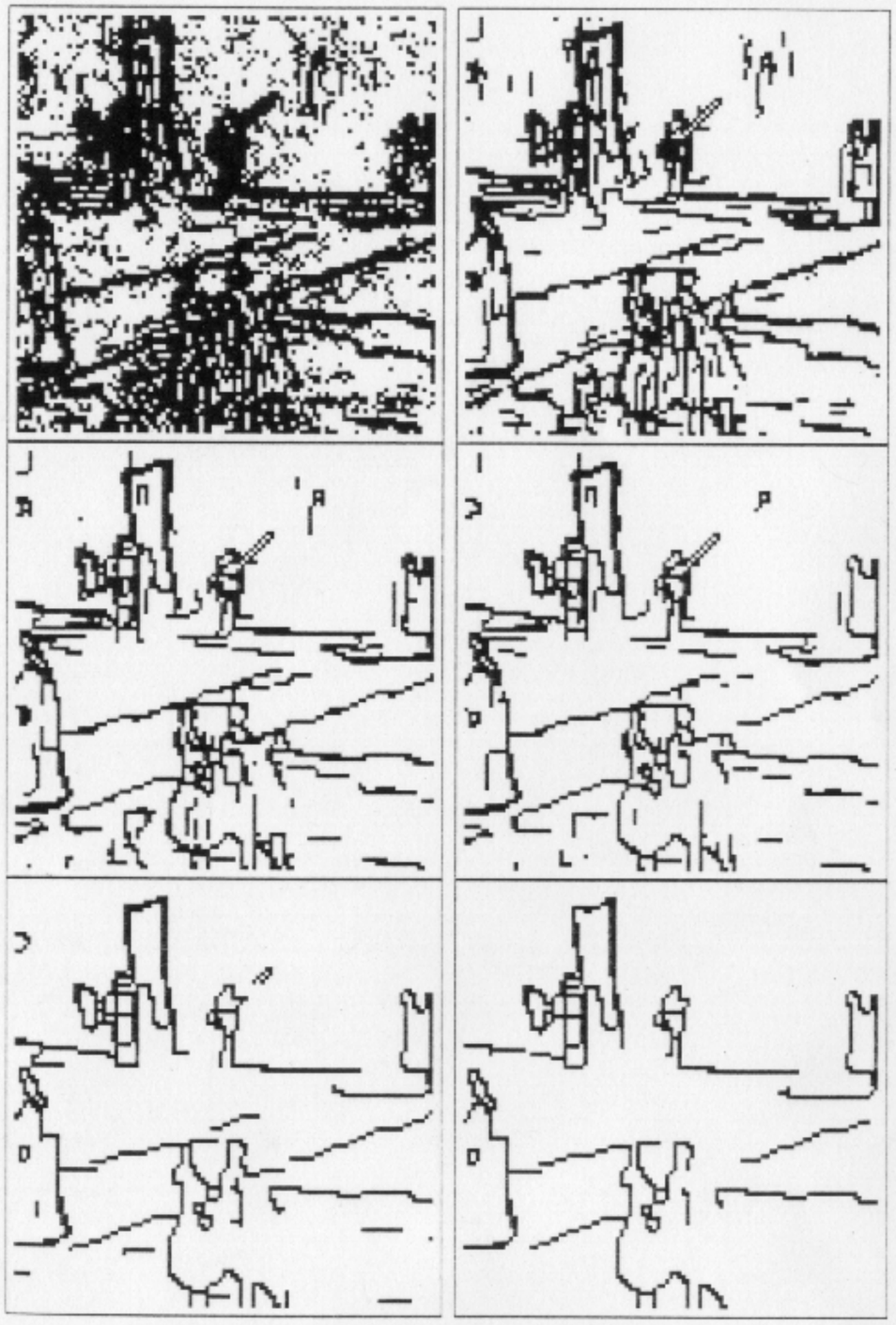

*Figure 3.8.* Sequence of boundary images obtaining by thresholding the gradient of the images in figure 3.7. The threshold was equal to $K$. Compare with figure 3.2.

## 4. Edge Enhancement

With conventional low-pass filtering and diffusion the price paid for eliminating the noise, and for performing scale-space, is the blurring of edges. This causes their detection and localization to be difficult. An analysis of this problem is presented in [51].
Edge enhancement and reconstruction of blurry images can be achieved by high-pass filtering or running the diffusion equation backwards in time. This is an ill-posed problem, and gives rise to numerically unstable computational methods, unless the problem is appropriately constrained or reformulated [165].
If the conduction coefficient is chosen to be an appropriate function of the image gradient we can make the anisotropic diffusion enhance edges while running *forward* in time, thus enjoying the stability of diffusions which is guaranteed by the seen in the previous section.
We may study the problem in 1D and model an edge as a step function convolved with a Gaussian. This corresponds to a straight 2D edge that is aligned with the $y$ axis (for curved edges the notation becomes more complicated).
The expression for the divergence operator simplifies to:

$$div(c(x,t)\nabla I) = \frac{\partial}{\partial x}(c(x,t)I_x)$$

We choose $c$ to be a function of the gradient of $I$: $c(x,t) = g(I_x(x,t))$ as in the previous sections. Let $\phi(I_x) \doteq g(I_x) \cdot I_x$ denote the flux $c \cdot I_x$.
Then the 1-D version of the diffusion equation (3.3) becomes

$$I_t = \frac{\partial}{\partial x}\phi(I_x) = \dot{\phi}(I_x) \cdot I_{xx} \tag{3.13}$$

We are interested in looking at the variation in time of the slope of the edge: $\frac{\partial}{\partial t}(I_x)$. If the function $I(\cdot)$ is regular enough the order of differentiation may be inverted:

$$\frac{\partial}{\partial t}(I_x) = \frac{\partial}{\partial x}(I_t) = \frac{\partial}{\partial x}(\frac{\partial}{\partial x}\phi(I_x)) = \ddot{\phi} \cdot {I_{xx}}^2 + \dot{\phi} \cdot I_{xxx} \tag{3.14}$$

Suppose the edge is oriented in such a way that $I_x > 0$ (see Fig.3.9). At the point of inflection $I_{xx} = 0$, and $I_{xxx} \ll 0$ since the point of inflection corresponds, to the point with maximum slope (see figure 3.9).
Then in a neighbourhood of the point of inflection $\frac{\partial}{\partial t}(I_x)$ has sign opposite to $\dot{\phi}(I_x)$. If $\dot{\phi}(I_x) > 0$ the slope of the edge will decrease with time; if, on the contrary $\dot{\phi}(I_x) < 0$ the slope will increase with time.



*Figure 3.9.* (Left to right) A mollified step edge and its $1^{st}$, $2^{nd}$, and $3^{rd}$ derivatives.

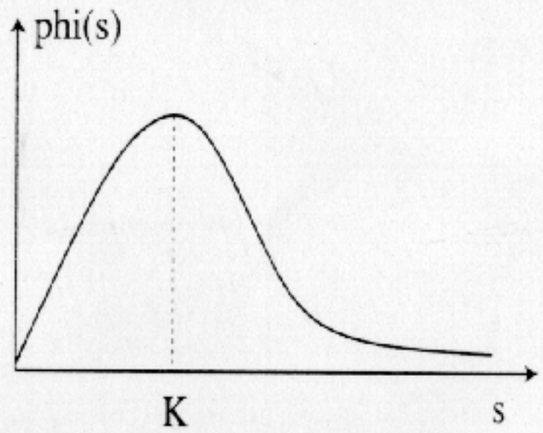


*Figure 3.10.* A choice of the function $\phi(\cdot)$ that leads to edge enhancement. See sec. 4.

Notice that this increase in slope cannot be caused by a scaling of the edge, because this would violate the maximum principle. The edge becomes sharper.
There are several possible choices for $\phi(\cdot)$, for example, $g(I_x) = C(1 + (I_x/K))^{-(1+\alpha)}$ with $\alpha > 0$ (see fig. 3.10). Then there exists a certain threshold value related to $K$, and $\alpha$, below which $\phi(\cdot)$ is monotonically increasing, and beyond which $\phi(\cdot)$ is monotonically decreasing, giving the desirable result of blurring small discontinuities *and* sharpening edges. Notice also that in a neighbourhood of the steepest region of an edge the diffusion appears to run 'backwards' since $\dot{\phi}(I_x)$ in equation (3.13) is negative. This may be a source of concern since it is known that constant-coefficient diffusions running backwards are unstable and amplify noise generating ripples. In our case this concern is unwarranted: the maximum principle guarantees that ripples are not produced. Experimentally one observes that the areas where $\dot{\phi}(I_x) < 0$ quickly shrink, and the process keeps stable.
Notice that while no ripples are created in $I$, this is not true for $I_x$. As discussed above the derivative of the image *does* diverge at some points. This is necessary for the edge sharpening behaviour to happen (contrary to the intuition that informs the initial discussion in [57]). If the regions where $\dot{\phi}(I_x) < 0$ are broad these will tend to segment into portions where $I_x$ diverges, separated by portions where $I_x \to 0$. We call this behaviour

*staircasing*; it is rather rare in common images, however one may notice it on very blurry images where the brightness gradient varies slowly. Staircasing arises when at the steepest point of the edge the curvature does not have maximum rate of variation; in this case sharpening of the edge may stop after a while because $I_{xxx}$ might cease to be negative at the inflection point. At that point $I_x$ starts decreasing. Of course in this case $I_{xx}$ will have other zero-crossings, and $|I_x|$ starts increasing at those points. The edge will not disappear, but it can be split into multiple edges.

One may reproduce the staircasing behaviour using a Gaussian-smoothed step edge ($\sigma = 100$ pixels, say) as a starting condition (better if some white noise is superimposed to it) and setting the parameter $K$ of the diffusion to a fraction (e.g. 10%) of the maximum of the gradient. The blurry edge slowly converges to a sequence of steps, rather than a single step. A way to prevent this from happening is to use smoothed versions of the gradient, rather than the gradient itself, in Eq. (3.4), this solution was independently proposed by Catté et al [57], and by Nitzberg and Shiota [271] (see also chapter by Niessen et al. on discrete implementations in this book).

## 5. Continuous model and well-posedness question

In this section we consider issues of stability, existence and uniqueness of solutions of the anisotropic diffusion scheme that uses the magnitude of the gradient as an estimate of the local 'edginess', i.e. equation (3.3) with conduction (3.4).

We will study a continuous model, although a discrete one gives a better approximation of the algorithm. The analysis of discretized schemes poses additional problems since discretization introduces another scale parameter (the lattice spacing) and directionality (e.g., vertical and horizontal edges may be preferred to slanted edges). Such effects may be reduced by making the grid size fine enough, *if* the continuous model is well-posed and stable. Ill-posed problems often appear in applied mathematics, e.g., as inverse problems in control theory. For example, the initial value problem, instead of the boundary value problem, for the laplacian:

$$\begin{aligned}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)u &= 0, \quad 0<x<a, -\infty<y<\infty,\\ u|_{x=0} &= u_0\\ \left.\frac{\partial u}{\partial x}\right|_{x=0} &= u_1\end{aligned}$$

is an ill-posed problem. The Euler scheme and other usual methods for initial value problem quickly accumulate errors, and do not give sensible results. But more careful method of constructing a solution using Poisson's kernel



does work, at least for a small enough 'time' $a$ (which may be the thickness of the wall of a pipe, etc.). This example shows that the choice of algorithm is very important when solving an ill-posed problem, even if the problem does allow a solution. See [147] and literature therein.

Some ill-posed problems are more ill-posed than others, and may not have solutions at all. In such a case the behavior of the solution of the algorithm is highly dependent on the algorithm, and the model is not very helpful to conceptually understand the algorithm. A typical example of this may be the problem of solving the heat equation backward. One must limit the frequency range to do the task, and the behavior of the solution is very sensitive to the frequency limit. In the fixed grid approximation, this means the result is highly dependent on the grid size.

The nature of the equation (3.3) with conduction (3.4) (continuous case) is not well-understood (we will call it Perona-Malik equation in the following). At the time of writing we don't know whether the equation is well-posed, or, if not, how bad the ill-posedness is. One possible way to study this is to have a regularization of the equation to make it well-posed, and then reduce the amount of regularization to observe the behavior of the solution. If the solution converges, the limit would solve the original equation. If the solution is very sensitive to the amount of regularization and fails to converge, there may be some sort of ill-posedness.

Catté and collaborators [57] have showed existence of solutions to (3.3) with conduction coefficient given by $c(x,y,t) = g(|\nabla(G_\sigma * u)|)$, i.e. a variation of (3.4) where the gradient of the image is substituted with the gradient of the image smoothed by a Gaussian of variance $\sigma$ (recall the 'physical derivatives' from chapter 1). J.-M. Morel (personal communication) has suggested that one may take the limit of the solution as a function of sigma as a definition in the weak sense of the solution to the Perona-Malik equation. Since the a norm of the solution (defined as the integral of $|\nabla u|$) is bounded above (e.g. by the norm of the initial condition) the sequence must have an accumulation point when $\sigma \to 0$. This accumulation point is a good candidate for a weak solution of the equation.

In a private communication to Mumford, P.-L. Lions showed the well-posedness of the modified Perona-Malik equation in one space dimension with time delay regularization

$$\frac{\partial I}{\partial t} = \left(\frac{I_x}{1+v/\kappa^2}\right)_x, \tag{3.15}$$

$$\frac{\partial v}{\partial t} = \omega(|I_x|^2 - v) \tag{3.16}$$

(larger $\omega$ gives a smaller time delay). This suggests, as also noted by Perona

elsewhere in this chapter, that it may be physically more natural for the equation to have some time delay in the heat conduction coefficient. Note also that the equations are still local (i.e., differential equations rather than integral or difference equations) with (3.16) giving time delay, like every law of physics. It is not local if, e.g., the time delay is instead given by

$$v(x,t) = |I_x(x,t-1/\omega)|^2. \tag{3.17}$$

Actually, this regularization has an advantage that the solution of (3.15) with (3.17) is smooth, as long as the initial data is smooth, while the solution may be much less regular with the regularization (3.16).

Shiota, in collaboration with Mumford and Nitzberg, has attempted some numerical experiments to see if the solution of (3.15), with (3.16) and/or the regularization given by the spatial smoothing of $v$, converges as the amount of regularization is decreased. These experiments seem to suggest a negative result: with the amount of regularization decreased, the graph of the solution became more and more jagged instead of developing a small number of sharp edges, and the numerical scheme failed more and more quickly as excessive grid subdivision led to floating point exception. These experiments are not conclusive; the jaggedness of solutions and other hints of instability that were observed on the numerical results might be artefacts of the numerical scheme, since as the amount of regularization decreased and the solution became more and more jagged the numerical errors may have become significant.

Useful properties of the Perona-Malik equation may be proved once the existence of a smooth solution is guaranteed (in a sense, our discussions in section 3 is a special case of it, where the regularization is given by discretization). For example, Nirenberg's theorem on maximum principle [269, Sect. 3, theorem 4] may be applied to prove maximum principle on a solution of the regularized equation, and any $C^2$ solution of the original equation. We conjecture that a maximum principle applies to all solutions of the Perona-Malik equation as well, however we have been unable to prove this.

A physical analogy may help in understanding this conjecture. The Perona-Malik equation models a physical diffusion process where the conduction coefficient is *never* negative. Although the conduction coefficient is coupled to the brightness/temperature distribution $u$ during its evolution in time, one could think of the problem in the following way. Consider the set of all possible (not necessarily continuous) choices for time-varying conduction coefficients with values between zero and one. To each one of these, given the same initial condition $u_0$, would correspond a solution to the diffusion equation (3.3). None of these would be ill-behaved, e.g. no divergence of $u$ would be possible, all of these, we conjecture, satisfy a maximum principle. One or more of these solutions would correspond to the solution to the Perona-Malik equation. Therefore the solution to the Perona-Malik equation is bounded and probably satisfies a maximum principle.

Some aspects of the Perona-Malik equation which make the analytic study of the equation difficult:

- *Lack of comparison theorem*: Because of nonlinearity, the maximum principle discussed above does *not* imply the comparison theorem like

  $$I_1(x,0) > I_2(x,0)\ (\forall x) \Rightarrow I_1(x,t) > I_2(x,t)\ (\forall x).$$

  It is easy to make a counterexample, using the way the heat conductivity depends on the gradient. This means estimation by an inequality, a basic technique in analysis, may fail to apply.
- *Difficulty of applying variational formalism*: For the Perona-Malik equation, the corresponding energy functional (i.e., the one for which the equation gives the gradient descent) has no dependence on the input image, and thus the set of minima of the functional does not reflect the information of the input image. For the Nordstrom equation [274, 273], the infimum of the energy functional is never achieved (at the limit of any 'minimizing sequence' of the energy functional, the energy value jumps up) unless the input image itself is a step function.
- *Possible instability*: Given a solution $I$ of the Perona-Malik equation, the linearization (infinitesimal perturbation) of the equation at $I$ gives a backward linear heat flow where $|\nabla I| > K$ (or some other function of $K$, this depends on the nonlinearity $g$ used). This means there is no nontrivial smooth family of solutions through $I$. Because of nonlinearity, this infinitesimal instability cannot logically imply finite instability (a small difference of initial data may become a huge difference of the solution), but it is likely to be the case. It thus suggests that errors of numerical scheme may be very difficult to control, and that, with the lack of approximability, many useful tools of analysis would fail to apply.

Although there is no general theory of nonlinear differential equations, a certain kind of nonlinear evolution equations are relatively well-understood. This theory is not applicable to the Perona-Malik equation, but it may shed some light on it.

A nonlinear operator $A : H \to H$, where $H$ is a Hilbert space, is said to be *monotone* if $\langle Au - Av, u - v\rangle \geq 0$ for any $u, v \in H$. This notion is a nonlinear analogue of positive operator. If $A$ is monotone and cannot be extended further as a monotone operator, it is called *maximal monotone*. If $A$ is maximal monotone, the equation $Au = f$ is the nonlinear analogue of an elliptic equation, and $du/dt = -Au + f$ is the nonlinear analogue of

a parabolic equation. Those equations are very well understood; e.g., the unique solvability is known. See, e.g., [43]. One can check by integration by part that the operator of the form $A := -\nabla \cdot (C(\|\nabla I\|)\nabla I)$ (resp. $-A$) on a bounded domain with Dirichlet boundary condition is monotone if $C(y)y$ is a monotone increasing function (resp. monotone decreasing function) of $y$. Although this is a functional analytic notion and thus cannot be 'localized', the above observation suggests that the Perona-Malik equation might be 'locally well-posed' in a region where $\|\nabla I\| < \kappa$; while the *backward* Perona-Malik equation might be well-posed where $\|\nabla I\| > \kappa$. If this can be made rigorous, it may prove the nonexistence of $C^1$ solution to the forward equation with initial data which satisfies $\|\nabla I\| > \kappa$ in a nonempty region. (This would not exclude the existence of less regular solution: appearance of jagged solution in numerical experiments suggests that the situation is far more complicated as the "unstable" region may instantly break up into many thin pieces separated by "stable" regions.)

To conclude, here is a summary of our current understanding of the behaviour of the solutions to the equation:

**Existence** – Yes, for many regularized versions, but unknown for the original equation. It appears to be possible to prove the existence of weak solutions (see discussion above).

**Uniqueness** – There is none. Is is easy to construct examples of bifurcations using either blurry edges with arbitrarily small sinusoids superimposed, or by studying a blurry edge which is at the limit between the enhancing and the diffusing behavior of the equation. In all these cases arbitrarily close initial conditions will generate different solutions.

**Maximum principle** – Easy to prove for $C^2$ solutions. Conjectured but not proven for discontinuous solutions.

# VECTOR-VALUED DIFFUSION

**Ross Whitaker**
*University of North Carolina*
*Department of Computer Science*
*Chapel Hill, NC 27514, USA*

and

**Guido Gerig**
*Communication Technology Laboratory*
*Image Science Division, ETH-Zentrum*
*Gloriastr.35, CH-8092 Zürich, Switzerland*

## 1. Introduction

In this chapter we generalize the ideas of previous chapters in order to construct variable-conductance diffusion processes that are sensitive to higher-order image structure. To do this we consider *anisotropic diffusion* as one instance of a whole class of processes that blur images while preserving certain kinds of interesting structure.

In particular the anisotropic diffusion process uses the *homogeneity* of neighborhoods (measured at some scale) in order to to determine the amount of blurring that should be applied to those neighborhoods. Previous chapters have dealt with equations that define local structure in terms of the properties of a single input image. In these cases structural inhomogeneity is quantified by the gradient magnitude. Such processes seem to be well suited to finding "edges" (discontinuities in luminance) in digital images. In this chapter we extend the definition of an "image" to include vector-valued or multi-valued images and we describe a general framework for quantifying the homogeneity of local image structure for such functions.

We will begin by developing a general framework for variable-conductance diffusion which incorporates vector-valued functions. We will then show how this framework can apply to the specific case of vector-valued images which result from imaging devices that measure multiple properties at each point in space. We will apply the same framework to vector-valued images consisting of sets of *geometric features* that derive from a single scalar image. We will examine two strategies that operate within this framework. The first uses