An Anisotropic Diffusion PDE for Noise Reduction and Thin Edge Preservation*

Zhouchen Lin and Qingyun Shi
National Key Laboratory of Machine Perception, Peking University
Beijing 100871, P.R. China

Abstract

In this paper, an image denoising operator defined by a non-linear partial differential equation(PDE) is presented. Similar to the operators proposed by Perona, Malik and Catté et al., it can remove noise, enhance step-like edges and keep the locality of the edges. Its extra ability is to keep thin edges. The criterion for stopping time is also investigated. The new operator is capable of removing rather high uniform noise without sacrificing the details of the image. If the noise is Gaussian with not too high standard deviation, the result is also quite good.

1. Introduction

In the field of image processing and analysis, the problem of noise reduction with feature preservation is still pending, though remarkable progress has been made in the past few decades. The primary methods are mainly linear low-pass filtering, using various templates. These approaches, though simple and easily implementable, inherently could not respect local features. The price paid for the removal of noise is the decrease of spatial resolution, which is caused by the flattening of sharp edges. This is a common defect of linear operators. A good operator must be adaptive to local features: ironing out fluctuations at smooth areas and keeping the sharpness of edges. However, with the presence of noise, the discrimination of noise and true edges involves the task of image understanding. Since 1980s, the importance of multiscale description of images is gradually recognized and accepted, e.g., [8][13]. In 1983, Witkin[15] introduced the idea of *scale space* via defining a family of derived images $I(x,y;\sigma)$ obtained by convolving the original image $I_0(x,y)$ with a Gaussian kernel $G_\sigma(x,y) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$, where σ is called the scale parameter, varying from 0 to ∞. Later, Koenderink [6] and Hummel[5] pointed out that computing $I_0 * G_\sigma$ is equivalent to the solution of the following standard heat conduction equation:

$$\frac{\partial I}{\partial t} = \Delta I, \tag{1}$$

at time $t = \frac{1}{2}\sigma^2$, with $I_0(x,y)$ as the initial condition (this will not be repeated in the sequel). The work of Koenderink *et al.* finally made the application of partial differential equations (PDEs) in image processing and analysis stand on its feet. During the following years, many researchers have embarked on finding various PDEs to deal with different problems ([1][2][4][9][10][11][12], to name just a few).

In 1990, Perona and Malik [11] proposed an operator defined by the following anisotropic diffusion equation:

$$\frac{\partial I}{\partial t} = \nabla \cdot (g(\|\nabla I\|)\nabla I). \tag{2}$$

They suggested two choices of $g(x)$:

$$g(x) = e^{-(x/K)^2} \qquad \text{or} \qquad g(x) = \frac{1}{1+\left(\frac{x}{K}\right)^2}.$$

Perona's operator exhibits a dichotomous behavior: fluctuations with gradient smaller than a threshold are gradually wiped out while large gradients are sharpened and enhanced. However, this operator does not work well when applied to very noisy images in that the noise introduces very large oscillations of the gradient, thus all the noise will be kept.

In 1992, Catté *et al.* [4] proposed a modification on (2):

$$\frac{\partial I}{\partial t} = \nabla \cdot (g(\|\nabla(G_\sigma * I)\|)\nabla I). \tag{3}$$

It estimates the gradient on the smoothed image $G_\sigma * I$, therefore promises more stable performance. Though

*This work is supported by the Key Projects of Chinese National Funds of Natural Science (No. 69735020).

Catté's operator is capable of dealing with ramping edges better than Perona's operator, both operators suffer from removing true detail edges. After the removal of noise only large areas remain. Within these areas the greylevel is nearly constant, resulting in the loss of naturalness of images. Therefore, neither operator is quite suitable for the reduction of noise, particularly when noise is considerable, which requires many times of solving (2) or (3).

In 1993, Whitaker and Pizer [14] went a little further. They suggested that the scale parameter should also decrease with time, i.e., their PDE is

$$\frac{\partial I}{\partial t} = \nabla \cdot (g(\|\nabla(G_{\sigma(t)} * I)\|)\nabla I).$$

But they did not provide a method to choose $\sigma(t)$ to achieve optimal results, therefore this idea is still in a fairly primitive form.

In this paper, a new operator defined in an analogous manner is given. It incorporates the second-order derivatives of the image to reduce the conduction coefficient at narrow peaks. As a result, more detail edges are kept after the removal of noise.

2. New Operator and Its Properties

The reason why Perona's and Catté's operators cannot preserve detail edges is that $\|\nabla I\| = 0$ at the narrow peaks, so the conduction coefficient $g(\|\nabla I\|)$ reaches its maximum value 1, leading to the fastest decay of the greylevel at the narrow peaks. On the other hand, the peaks are widened, thus the spatial resolution decreases. Though the sharpness of the edges is increased, the contrast is in fact reduced. Therefore to improve the result of filtering, the preservation of narrow peaks is necessary. The key is to reduce the conduction coefficient at these places. Note that at the peaks, the second-order derivatives of I are in general local maxima, hence taking the values of the second-order derivatives of I into consideration will help to discourage the fast diffusion at the peaks. Regarding the rotational invariance, $\Phi = (I_{xx})^2 + 2(I_{xy})^2 + (I_{yy})^2$ is a good candidate. On the other hand, in order to estimate the derivatives more accurately, the convolution with Gaussian kernel should also be incorporated. In short, the modified PDE is:

$$\begin{aligned}\frac{\partial I}{\partial t} = \;& \nabla \cdot (\tilde{g}(\|\nabla(G_\sigma * I)\|^2 + (G_\sigma * I_{xx})^2 + \\ & +2(G_\sigma * I_{xy})^2 + (G_\sigma * I_{yy})^2)\nabla I),\end{aligned}$$

where $\tilde{g}(x) = g(\sqrt{x})$.

The selective edge-enhancement property is analyzed qualitatively in [7]. The new operator has the following properties:

1. At flat areas, the new operator smoothes the noise as fast as Perona's (or Catté's) operator does, given the same number of iterations.

2. At strong edges, the new operator also enhances them, but not as much as Perona's (or Catté's) operator does. It is more likely to preserve the slope of peaks, avoiding the widening the peaks, therefore maintains the spatial resolution better.

3. The contrast is kept better than Perona's (or Catté's) operator does.

3. Numerical Scheme

We approximate the temporal derivative by forward difference: $\frac{\partial I}{\partial t} = I^{n+1} - I^n$. In order to estimate ∇I^n, I^n_{xx} and I^n_{yy}, we need to compute their convolution with G_σ. Because the convolution and differentiation are commutative, computing the convolution of I with G_σ first and then calculating the corresponding derivatives will call for less computational load. As pointed out in section **1**, computing $I * G_\sigma$ is equivalent to solving the standard heat conduction equation with adiabatic boundary conditions and stopping at $t = \frac{1}{2}\sigma^2$, so it is easy to write down a fast algorithm. We omit it here.

The approximation of I_x and I_y should apply forward or backward difference. Central difference usually will make the value of $\|\nabla(G_\sigma * I)\|^2 + (G_\sigma * I_{xx})^2 + 2(G_\sigma * I_{xy})^2 + (G_\sigma * I_{yy})^2$ at peaks much less than that using forward or backward difference. This will counteract our effort to preserve narrow peaks.

At last, the computational scheme becomes:

$$\begin{aligned}I^{n+1}_{i,j} = \;& I^n_{i,j} + \lambda(C_N \cdot \nabla_N I^n_{i,j} + C_S \cdot \nabla_S I^n_{i,j} + \\ & +C_E \cdot \nabla_E I^n_{i,j} + C_W \cdot \nabla_W I^n_{i,j}),\end{aligned} \tag{4}$$

where:

$$\nabla_N I^n_{i,j} = I^n_{i-1,j} - I^n_{i,j}, \quad \nabla_S I^n_{i,j} = I^n_{i+1,j} - I^n_{i,j},$$

$$\nabla_E I^n_{i,j} = I^n_{i,j+1} - I^n_{i,j}, \quad \nabla_W I^n_{i,j} = I^n_{i,j-1} - I^n_{i,j},$$

$$C_N = \tilde{g}(\|(\nabla \bar{I}^n)_{i-1,j}\|^2 + \alpha^n_{i,j}),$$

$$C_S = \tilde{g}(\|(\nabla \bar{I}^n)_{i+1,j}\|^2 + \alpha^n_{i,j}),$$

$$C_E = \tilde{g}(\|(\nabla \bar{I}^n)_{i,j+1}\|^2 + \alpha^n_{i,j}),$$

$$C_W = \tilde{g}(\|(\nabla \bar{I}^n)_{i,j-1}\|^2 + \alpha^n_{i,j}),$$

$$\bar{I}^n = G_\sigma * I^n,$$

$$((\bar{I}^n)_{xx})_{i,j} = (\bar{I}^n)_{i,j-1} - 2(\bar{I}^n)_{i,j} + (\bar{I}^n)_{i,j+1},$$

$$((\bar{I}^n)_{yy})_{i,j} = (\bar{I}^n)_{i-1,j} - 2(\bar{I}^n)_{i,j} + (\bar{I}^n)_{i+1,j},$$

$$\begin{aligned}((\bar{I}^n)_{xy})_{i,j} = \;& ((\bar{I}^n)_{i+1,j+1} - (\bar{I}^n)_{i+1,j-1} - \\ & -(\bar{I}^n)_{i-1,j+1} + (\bar{I}^n)_{i-1,j-1})/4,\end{aligned}$$

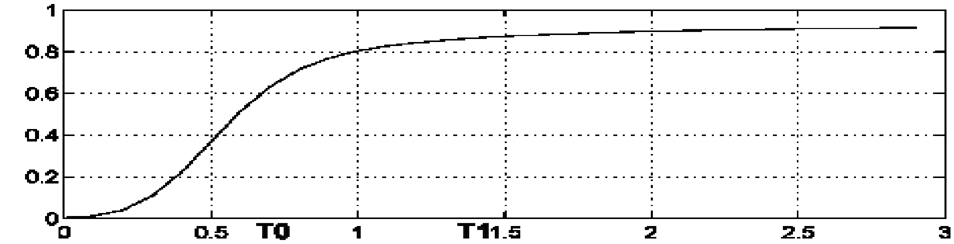

Figure 1. A typical curve C of r as a function of t.

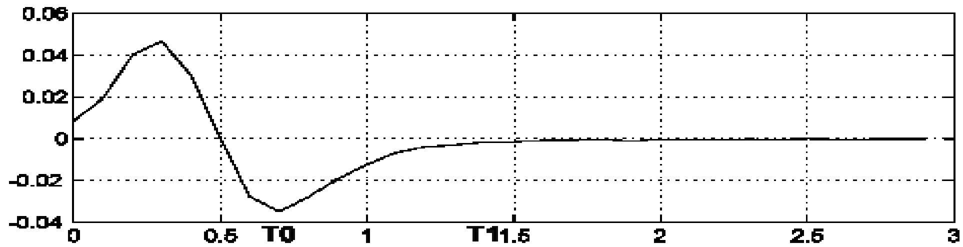

Figure 2. The second-order derivative of the curve C.

$$\alpha^n_{i,j} = [((\bar{I^n})_{xx})_{i,j}]^2 + 2[((\bar{I^n})_{xy})_{i,j}]^2 + [((\bar{I^n})_{yy})_{i,j}]^2.$$

The boundary condition is adiabatic: $\frac{\partial I}{\partial \nu} = 0$. The sufficient condition for the stability of (4) is $0 < \lambda \leq 0.25$.

4. Stopping Criteria

Three parameters are required to make the above scheme workable, namely K, scale parameter σ and stopping time T. $\sigma = 0.8$ [9] will suffice wide categories of images. The choice of K still borrows the framework of Canny's histogram estimation: K is chosen such that for q=85%~90% pixels the value of $\|\nabla(G_\sigma * I)\|$ is less than K [11]. The percentage may vary slightly for different images. There is no need to set K at every timestep, due to the selective edge-enhancement property. Thus, only the stopping time T is left.

It is most preferable to seek an automatic decision strategy on T. A natural idea is: if a noisy area is supposed to be flat and it becomes smooth enough after some iterations, then the algorithm may stop. At a pixel P, if the estimated gradient $\|\nabla(G_\sigma * I)\| < K$ then it is *expected* to be in a flat area. Denote by N_1 the number of such pixels. Next, we use $\Phi = (I_{xx})^2 + 2(I_{xy})^2 + (I_{yy})^2$ to test whether the neighbourhood $\mathcal{N}$ of P is *really* smooth enough. If $\Phi < 30$, then $\mathcal{N}$ is considered smooth; otherwise not smooth enough. The threshold 30 is obtained in a heuristic way. Since we are considering the local fluctuation at P, we may assume that the greylevel of the image is all zero except at $P = (i, j)$. If $I_{i,j} < 2$, then we may regard that it is smooth around P; otherwise not smooth enough. For $I_{i,j} = 2$ on this "conceptual" image, $I_{xx} \approx I_{i+1,j} - 2I_{i,j} + I_{i-1,j} = -4$, $I_{yy} \approx I_{i,j+1} - 2I_{i,j} + I_{i,j-1} = -4$, and $I_{xy} \approx (I_{i+1,j+1} - I_{i-1,j+1} - I_{i+1,j-1} + I_{i-1,j-1})/4 = 0$, therefore the threshold Φ should be around 32. This is how the threshold 30 comes. We have done experiments to show that the stopping time found by the criterion stated below is not sensitive to this threshold.

Note that not all pixels that fulfill $\|\nabla(G_\sigma * I)\| < K$ can satisfy $\Phi < 30$, we have to look at the ratio $r = N_2/N_1$ instead, where N_2 is the number of points that meet both $\|\nabla(G_\sigma * I)\| < K$ and $\Phi < 30$. A naive criterion is: if $r > p$ then stop, otherwise go on, where $p \in (0, 1)$ is some threshold percentage. However, this is not a good criterion. For textured images p should be smaller, while for images with many smooth areas p should be larger. For some images, r increases with t very slowly, thus the appropriate stopping time is very sensitive to the choice of p if such criterion is adopted.

Draw the curve C of r as a function of t (Figure 1), one will notice the time T_0, where the slope of the curve decrease fastest (Figure 2). It is a sign that the noise in flat areas have been smoothed. After T_1, the slope begins to decrease slower and slower, indicating that the edges are being destroyed gradually. Therefore T_1

Figure 3. Comparison between Catte's operator and the new operator. The mechanism of noise addition is pixelwise adding a random number which is uniformly distributed on some interval $[-\beta,\ \beta]$**. (a) Lena with uniform noise on** $[-15,\ 15]$**. (b) a filtered by the new operator,** $T = 1.6$**. (c) a filtered by Catte's operator. (d) Lena with uniform noise on** $[-30,\ 30]$**. (e) d filtered by the new operator,** $T = 3.4$**. (f) d filtered by Catte's operator. (g) Lena with uniform noise on** $[-50,\ 50]$**. (h) g filtered by the new operator,** $T = 4.6$**. (i) g filtered by Catte's operator.**

Table 1. The PSNRs on various images and areas (unit: dB)

	Figure 3a			Figure 3d			Figure 3g		
	on whole image	on area A	on area B	on whole image	on area A	on area B	on whole image	on area A	on area B
before filtering	29.12	29.10	29.10	23.19	23.17	23.11	18.91	18.85	18.91
Catté's operator	33.64	39.84	28.81	30.52	38.49	24.92	27.75	36.85	22.05
new operator	34.82	39.82	30.97	31.58	38.48	26.83	29.39	36.96	24.22

is a good stopping time. However, we cannot obtain the whole curve first in order to find T_1, so we have to estimate it. Experiments on many noisy images show that $T_1 = 2T_0$ is a good choice. As observed in many experiments, C and its first derivative is rather smooth when λ is not too small, thus the determination of T_0 is not ambiguous.

The pseudo-C-codes for the algorithm that are relevant to the stopping time can be written as:

```
num_iteration = 3;
for ( i = 0; i < num_iteration; i ++ )
{
   compute r;
   if ( i < 2 ) { r0 = r1; r1 = r; continue; }
    // stores r;
   r_xx = r0 - 2*r1 + r;
    // second-order derivative of the curve;
   if ( r_xx >= 0 ) num_iteration ++;
   if ( min_r_xx > r_xx )
   { num_iteration = 2*i; min_r_xx = r_xx; }
   r0 = r1; r1 = r; // stores r;
}
```

It is recommended that λ should not be too large, in order to avoid the sensitivity of filtering results to the number of iterations. For large λ (and it should not exceed 0.25!), one more (or less) iteration may cause noticeable difference between two successive iterations. In our experiments, it is set to 0.1.

This criteria is striking when applied to filter nearly noiseless images, such as the standard images Lena, Barbara and Goldhill etc. It stops very quickly, after three to five iterations. Note that three is the least possible number of iterations. When applied to very noisy images, the filtered images have good compromise between smoothing noise and keeping details.

5 Experiments

Figure 3 compares the results of filtering by the new operator and Catté's operator (For better visualization, only the central 4/9 portions are displayed.). The left column are noisy images before filtering, with different levels of uniform noise. The central column are images filtered by the new operator, while the right column are those filtered by Catté's operator with the same stopping time. Throughout the experiments, the percentage q used in the histogram estimation is fixed at 86%.

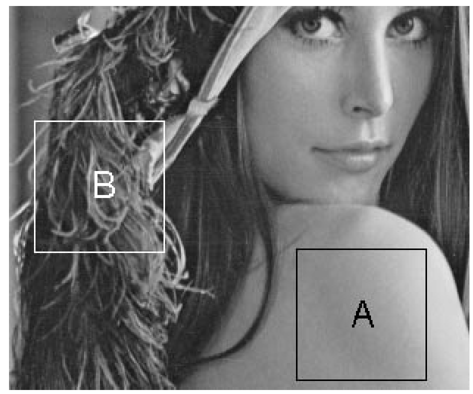

Figure 4. The areas on which the PSNR will be computed.

We see that those visually significant details (sharp features), such as the fringe of the hat and the eye-brow are well preserved by the new operator, even with very high uniform noise. At flat areas, such as the face, shoulder and the mirror, the noise is well removed by both operators. To give a quantitative illustration, we compute the peak signal-to-noise ratio (PSNR) on the whole image, the flat area A (surround by the black box in Figure 4) and the edge-rich area B (surround by the white box in Figure 4) respectively. Table 1 gives the PSNRs in various cases before filtering and after filtering. One can see that even in quantitative aspect, the new operator is also superior to Catté's operator.

We also tested the new operator and Catteé's operator on images contaminated by Gaussian noise. Figure 5 shows the difference. If the standard deviation of the Gaussian noise is not too high, then the new op-

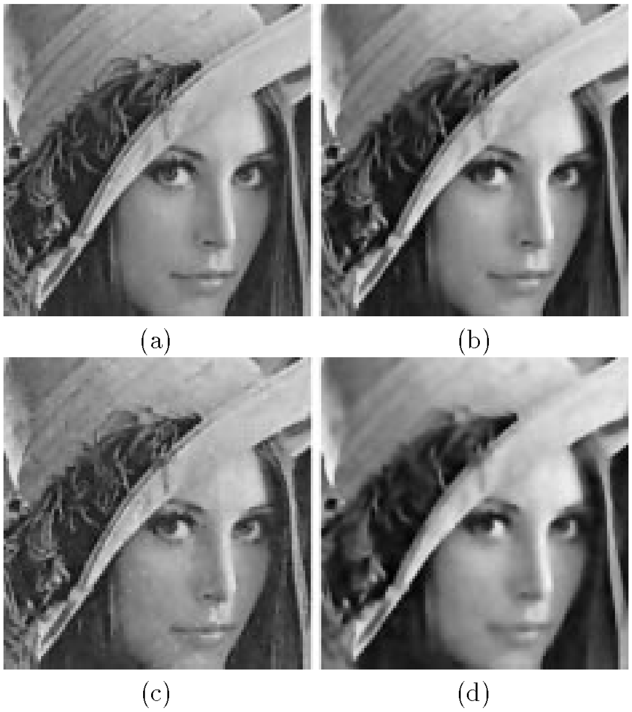

Figure 5. Filtering results on images with Gaussian noise. (a) Lena with Gaussian noise $N(0,100)$ filtered by the new operator; (b) filtered by Catte's operator. (c) Lena with Gaussian noise $N(0,400)$ filtered by the new operator; (d) filtered by Catte's operator.

erator still performs well (Figure 5a). However, if the standard deviation becomes large, many isolated spots will appear on the images filtered by the new operator (Figure 5c). At this time, the images filtered by Catté's operator are blurry (Figure 5d).

6. Conclusions

In this work, a new image operator defined by an anisotropic diffusion PDE is presented. Since it uses the estimated gradient as well as the second-order derivatives of the image to compute the conduction coefficient at every pixel, thin edges are better preserved than Perona's or Catté's operator does. The removal of noise in flat areas is also fast. The criterion for stopping time works well in our algorithm. The new operator is very powerful when the noise is uniform. For Gaussian noise, if the standard deviation is not too high (the amount may depend on the characters of the image) then the result is also very satisfactory.

References

[1] L. Alvarez, P.L. Lions and J.M. Morel. Image selective smoothing and edge detection by nonlinear diffusion II. *SIAM J. Numer. Anal.*, 29(3): 845–866, 1992.

[2] L. Alvarez and L. Mazorra. Signal and image restoration using shock filters and anisotropic diffusion. *SIAM J. Numer. Anal.*, 31(2):590–605, 1994.

[3] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Machine Intell.*, 8(6):679–698, 1986.

[4] F. Catté *et al.*. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182–193, 1992.

[5] A. Hummel. Representations based on zero-crossings in scale-space. *Proc. IEEE Computer Vision and Pattern Recog. Conf.*:204–209, 1986.

[6] J. Koenderink. The structure of images, *Biol. Cybern.*, 50:363–370, 1984.

[7] Z.C. Lin and Q.Y. Shi. An anisotropic diffusion equation that can remove noise and keep naturalness (in Chinese). *Chinese Journal of Computers*, 1998, accepted.

[8] D. Marr. *Vision*. San Francisco, CA:Freeman, 1982.

[9] W. Niessen *et al.*. A general framework for geometry-driven evolution equations. *Int. J. Computer Vision*, 21(3):187–205, 1997.

[10] S. Osher and L. I. Rudin. Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.*, 27(4):919–940, 1990.

[11] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(7):629–639, 1990.

[12] Bart M. ter Haar Romeny (eds.). *Geometry-Driven Diffusion in Computer Vision*. Kluwer Academic Publishers, Netherlands, 1994.

[13] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Trans. Comput.*, C-20:562–569, 1971.

[14] R. T. Whitaker and S. M. Pizer. A multi-scale approach to nonuniform diffusion. *CVGIP: Image Understanding*, 57(1):99–110, 1993.

[15] A. Witkin. Scale-space filtering. *Int. Joint Conf. Artif. Intell.*, pp.1019–1021, Karlsruhe, west Germany, 1983.