

# An Edge Adapting Laplacian Kernel For Nonlinear Diffusion Filters

Mohammad Reza Hajiaboli, *Member, IEEE*, M. Omair Ahmad, *Fellow, IEEE*,  
Chunyan Wang, *Senior Member, IEEE*

**Abstract**—In this paper, first a new Laplacian kernel is developed to integrate into it the anisotropic behavior to control the process of forward diffusion in horizontal and vertical directions. It is shown that even though the new kernel reduces the process of edge distortion, it nonetheless produces artifacts in the processed image. After examining the source of this problem, an analytical scheme is devised to obtain a spatially-varying kernel that adapt itself to the diffusivity function. The proposed spatially-varying Laplacian kernel is then used in various nonlinear diffusion filters starting from the classical Perona and Malik filter to the more recent ones. The effectiveness of the new kernel in terms of quantitative and qualitative measures is demonstrated by applying it to noisy images.

**Index Terms**—Image denoising, nonlinear diffusion, edge preservation, edge adaptive Laplacian kernel.

## I. INTRODUCTION

NONLINEAR diffusion denoising is a powerful denoising technique because of its superior edge preservation capability. This technique has been widely used in medical image denoising [1]–[7], where it is of paramount importance that the denoising process preserves the information contained in high-spatial-frequency components of the images. The nonlinear diffusion filtering introduced by Perona and Malik [8] is a time evolutionary process, in which the denoised image is a solution of a diffusion equation with a spatially varying diffusion coefficient. The diffusion coefficient is a function of the modulus of the gradient of the evolving denoised image. For a region of the image with a low-modulus of the gradient, (i.e. the regions affected by noise), the diffusivity function returns a high diffusion coefficient that results in a forward diffusion and thus in the smoothness of the region. On the other hand, for a region with a high-modulus of the gradient (i.e. the regions with an edge), the diffusivity function returns a small value, which results in a lower forward diffusion of that region or even a backward diffusion leading to edge enhancement.

Since the introduction of the nonlinear diffusion filter by Perona and Malik in 1990, the behavior of the filter in both the continuous and discrete domains and its relation to the functional minimization problem for image restoration [9]

have been extensively studied in the literature [10]–[15]. While the forward-backward diffusion results in smoothing the noise and enhancing the edges, it is known that the backward diffusion is an ill-posed process in the sense that it is very sensitive to perturbations in the initial noisy data [11]. A practical solution to this problem is given by Catte et al. [16] in which the diffusion coefficient is calculated using the modulus of the gradient of the image smoothed by a Gaussian filter.

The edge preservation capability of the nonlinear diffusion filter depends largely on the extent by which the diffusivity function can successfully detect the edges and also control the diffusion process. From the standpoint of improving the edge preservation and denoising performance of nonlinear diffusion filters, most of the studies have focused on introducing new diffusivity functions such as those in [13], [15], [17]–[19] or optimizing the choice of the parameter in the diffusivity function [12], [20]. The basic idea behind all these techniques is to significantly reduce the value of the diffusion coefficient whenever an edge is detected in order to reduce the diffusion rate for the edge. However, as it is reported in [15] and [17], this approach for increasing the edge preservation capability of a filter leads to the formation of pin-hole artifacts in the denoised image, especially when the noise level is high or moderately high. In fact, this drawback of the approach is due to a significant reduction of the diffusion not only for the edges but for high level noise, a process that leaves both the edges and high-level noise almost intact from the diffusion process.

In view of the limitation of the above methods in its capability of removing high-level noise while attempting to aggressively reduce the diffusion process for edges, in this paper, a new kernel based on the anisotropic behavior of the Rosenfeld and Kak Laplacian operator [21] is developed for an efficient preservation of the edges in nonlinear diffusion processes. A study is undertaken first to investigate the impact of the anisotropic behavior of the Laplacian operator on the capability of nonlinear diffusion filters in preserving the edges in different orientations and then, to explore the possibility of designing a new Laplacian operator that is best suited to preserve the edges in certain orientations without affecting the quality of the edges in other directions.

The paper is organized as follows, In Section II, the effect of the anisotropic behavior of the Laplacian kernel of Rosenfeld and Kak on the edge preservation capability of the nonlinear diffusion filter is analyzed. In Section III, a detailed analysis leading to an edge adapting Laplacian kernel is presented. Simulation results demonstrating the effectiveness of the proposed kernel in terms of its edge preservation and noise removal

Manuscript received April 19, 2011; revised July 22, 2011. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada and in part by the Regroupement Stratégique en Microélectronique du Québec (ReSMiQ). The authors are with the Center for Signal Processing and Communications, Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, H3G 1M8 Canada (e-mail:mohammad.hajiaboli@ieee.org; omair@ece.concordia.ca; chunyan@ece.concordia.ca)

capabilities, when it is employed in the different nonlinear diffusion filters are presented in Section IV. Finally, Section V concludes the paper by highlighting the significant features of the proposed scheme.

## II. IMPACT OF THE ANISOTROPIC BEHAVIOR OF THE LAPLACIAN OPERATOR ON THE DIFFUSION RATE OF EDGES

The basic nonlinear diffusion filter is characterized by the partial differential equation (PDE) given by [8]

$$\partial u / \partial t = \text{div} (c(\|\nabla u\|) \nabla u) \quad (1)$$

where  $\|\cdot\|$  and  $\text{div}$  denote, respectively, the  $L_2$ -norm and divergence of the associated quantities,  $c(\cdot)$  is a diffusivity function, also referred to as the diffusion coefficients, and  $\nabla u$  represent the gradient of the diffusing image  $u$ . By carrying out the divergence operation with respect to the independent variables  $x$  and  $y$ , the above equation can be expressed as follows:

$$\partial u / \partial t = c(\cdot) \nabla^2 u + \nabla c(\cdot) \nabla u \quad (2)$$

in which  $\nabla^2 u$  denotes the Laplacian of  $u$ . A commonly used discrete version [20], [22] of (1) (or equivalently of (2)) in which the diffusivity function is approximated by the arithmetic average of the diffusivity coefficients at the adjacent nodes, is given by

$$u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \tau [c_{i,j} \nabla^2 u_{i,j}]^{(n)} + \frac{\tau}{2} \left[ \sum_d \nabla_d c_{i,j} \nabla_d u_{i,j} \right]^{(n)} \quad (3)$$

where  $\nabla_d$ ,  $d \in \{N, S, E, W\}$ , denotes the directional gradient calculated as the difference of two adjacent pixel values or diffusion coefficients in the north (N), south (S), east (E) and west (W) directions,  $\tau$  is the step-size of the independent variable  $t$  used to approximate  $\partial u / \partial t$ , and  $n$  is the discrete time index (or the iteration number). The Laplacian  $\nabla^2 u$  in the approximation given by (3) is actually the discrete convolution of  $u$  with the so-called Rosenfeld-Kak Laplacian kernel  $L_1$  [21]) given by

$$L_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

The Laplacian kernel  $L_1$  has an anisotropic response to step edges [23] in the sense that the convolution result of the constant-contrast edge is orientation dependent. To show this anisotropic behavior, one can consider the four step edges with the same contrast level of  $(u_1 - u_2)$  and the orientation given by  $\theta$ , as shown in Fig. 1

If the central pixel of each of these four intensity profiles is denoted by  $I_5$ , the Laplacian response of  $I_5$  corresponding to each of the four orientation values is given by

$$\nabla^2 I_5 = \begin{matrix} \theta = 90^\circ & \theta = 0^\circ & \theta = +45^\circ & \theta = -45^\circ \\ (u_1 - u_2) & (u_1 - u_2) & 2(u_1 - u_2) & 2(u_1 - u_2) \end{matrix}$$

This simple example shows that the response of  $L_1$  to an edge in the horizontal or vertical direction is different from that to

the diagonal edge thus demonstrating the anisotropic behavior of the discrete Laplacian operator.

We now focus our attention to study the impact of the anisotropic behavior of the Laplacian operator  $L_1$  on the performance of the discrete nonlinear diffusion filters, in as far as their capability in preserving the edges.

In (3), the values of  $c(\|\nabla u\|)$ , calculated based on the central difference operator for  $\|\nabla u\|$  on both sides of a step edge are the same; thus, the directional derivative of  $\nabla_d c(\cdot)$  is zero. Therefore, at step edges, the nonlinear diffusion equation given in (3) becomes

$$u_{i,j}^{(n+1)} = u_{i,j}^{(n)} + \tau [c_{i,j} \nabla^2 u_{i,j}]^{(n)} \quad (4)$$

In fact, the same simplification is noted from the nonlinear diffusion equation of Alvarez et al. [24] given by

$$u_t = c(\|\nabla u\|) u_{\xi\xi} + (c(\|\nabla u\|) + c'(\|\nabla u\|) \|\nabla u\|) u_{\eta\eta} \quad (5)$$

where  $u_{\xi\xi}$  and  $u_{\eta\eta}$  are the second-order derivatives of  $u$  in the direction of the level set (i.e. the direction parallel to the image features) and the direction of the gradient (i.e. the direction across the edge). In this case,  $\nabla^2 u = u_{\eta\eta} + u_{\xi\xi}$  and for step edges  $u_{\xi\xi} = 0$ , this nonlinear diffusion filter at the step edges in the continuous domain assumes the form

$$u_t = c(\|\nabla u\|) \nabla^2 u \quad (6)$$

which can be seen to have the same discrete version as the one given by (4).

Equation (4) governing the diffusion process at edges has another difference operator arising from the diffusion coefficient  $c(\cdot)$ , which also has an anisotropic behavior. In order to study the impact of only the anisotropic behavior of the Laplacian operator (i.e.  $L_1$ ) on the diffusion process of the edges, we use the Scharr kernels [25] to have the rotationally invariant (isotropic) approximation of  $\|\nabla u\|$ . Two synthetic digital images shown in Fig. 2(a) and (b) containing a vertical edge and a diagonal edge, respectively, are chosen. The contrast level of the edge in both the images is 70 thus the variance of either images is approximately 1225. These two images are filtered in accordance with the diffusion process of (4) in which the diffusivity function of Perona and Malik given by

$$c(\|\nabla u\|) = K^2 / (K^2 + \|\nabla u\|^2) \quad (7)$$

where  $K$  is a positive parameter called a contrast parameter, is used. The value of  $K$ , which is manually set, controls the diffusion in the sense that a larger value of  $K$  results in a more intense diffusion. The diffusion process is repeated by varying the values of  $K$  from 5 to 25 in steps of 2.5. For each run,  $\tau$  is chosen as 0.1 and the diffusion process stopped after  $n = 30$  iterations. The variance of the diffused image is calculated. The result of the variance of the diffused image as a function of  $K$  is plotted in Fig. 2(c). It is seen from this figure that the image containing the vertical edge is less diffused than the one with the diagonal edge. Thus, the same diffusion process can preserve a vertical edge better than a diagonal one.

It is important to note that the commonly used approximation of  $\|\nabla u\|$  is based on the central difference operator

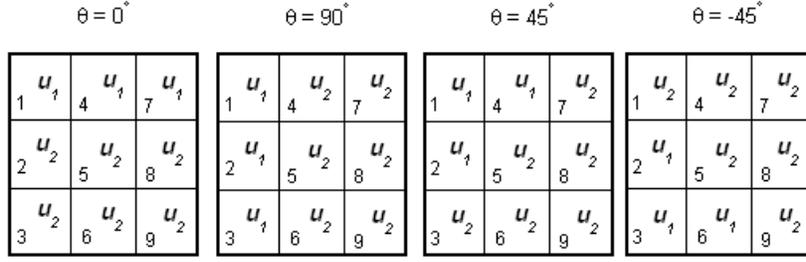


Fig. 1. Step edges with orientations of 0, 90, 45, and  $-45$  degrees and a contrast level of  $u_1 - u_2$ .

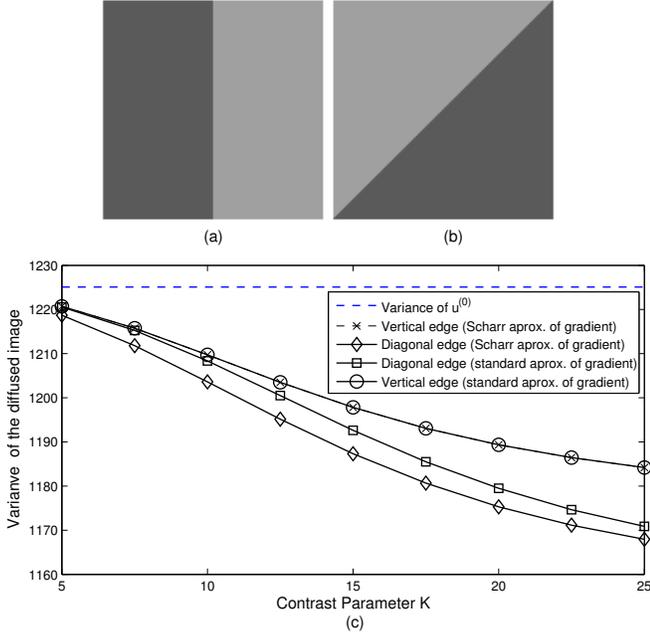


Fig. 2. Impact of the anisotropic behavior of the  $L_1$  kernel, when used in (3), on the vertical and diagonal edges. (a) Image with a vertical edge. (b) Image with a diagonal edge. (c) Variance of the diffused images using isotropic and anisotropic discrete approximations of  $\|\nabla u\|$ .

[17], which is an anisotropic operator. In our experiment, by using this operator to calculate the diffusivity function and  $L_1$  for the Laplacian operator, we obtain results that are also depicted in Fig. 2(c). As seen from this figure, an isotropic or anisotropic approximation of  $\|\nabla u\|$  has a little or no impact on the diffusion of the edges. However, the results of Fig. 2(c) also indicates that there are significant differences between the diffusions of the vertical and diagonal edges. This observation motivates us to develop a new kernel with the capability of providing a better control of the diffusion of edges.

### III. PROPOSED LAPLACIAN KERNEL

Our objective in this section is to explore the possibility of designing a Laplacian kernel from the standpoint of reducing the diffusion process of the edges and make the process less dependent on the orientation of the edges. Hence, in the context of Fig. 2(d), our objective is to raise the level of the curves higher and closer to each other.

#### A. Design of a Laplacian Kernel for Curtailing the Edge Diffusion

A generalized parametric realization of a Laplacian kernel can be written in the form

$$L = \begin{bmatrix} \beta/2 & \alpha & \beta/2 \\ \alpha & -4(\alpha + \beta/2) & \alpha \\ \beta/2 & \alpha & \beta/2 \end{bmatrix} \quad (8)$$

From this generalized Laplacian kernel, one can obtain different kernels depending on the values of the parameters  $\alpha$  and  $\beta$ . For example, the Laplacian kernel  $L_1$  can be obtained by choosing  $\alpha = 1$  and  $\beta = 0$ . Similarly, for  $\alpha = 0$  and  $\beta = 1$ , (8) yields

$$L_2 = \frac{1}{2} \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

which is also a commonly used Laplacian kernel [19]. The generalized parametric kernel given by (8) is a linear combination of two Laplacian kernels of  $L_1$  and  $L_2$  given by

$$L = \alpha L_1 + \beta L_2 \quad (9)$$

In order to study the effect of this Laplacian operator ( $L$ ) on the orthogonal (vertical and horizontal) and diagonal edges shown in Fig. 1, we first obtain the response of  $L$  to each pixel numbered as 1 to 9. For this purpose, we assume that structure of the edges continues beyond the  $3 \times 3$  region of the image shown in this figure. The response to all the 9 pixels in the  $3 \times 3$  region containing the vertical edge (Fig. 1(b)) is given as

$$\nabla^2 I_p = \begin{cases} (\alpha + \beta)u_2 - (\alpha + \beta)u_1, & p = 1, 2, 3 \\ -\nabla^2 I_{p-3}, & p = 4, 5, 6 \\ 0, & p = 7, 8, 9 \end{cases} \quad (10)$$

and that for the pixels of the region containing the diagonal edge with  $\theta = 45^\circ$  (Fig. 1(c)) is given by

$$\nabla^2 I_p = \begin{cases} (-1)^p [(\beta/2)u_1 - (\beta/2)u_2], & p = 1, 6, 8 \\ (-1)^p [(2\alpha + \beta/2)u_2 - (2\alpha + \beta/2)u_1], & p = 2, 3, 4, 5, 7 \\ 0, & p = 9 \end{cases} \quad (11)$$

Our objective is to curtail the diffusion of the edges by making  $\nabla^2 u$  to be zero at all the pixels on the edges and in their immediate neighborhoods. This implies that we must find

the values of the parameters  $\alpha$  and  $\beta$  that reduces  $\nabla^2 I_p$  to be zero for all values of  $p \in \{1, 2, \dots, 9\}$ . It is obvious from (11) that for the diagonal edge with  $\theta = 45^\circ$ , the only values for  $\alpha$  and  $\beta$  that lead to  $\nabla^2 I_p = 0$  are the trivial values of  $\alpha = 0$  and  $\beta = 0$ . The same conclusion can be reached for a diagonal edge with  $\theta = -45^\circ$ . Thus, for a diagonal edge, there does not exist a non-trivial Laplacian operator  $L$  that when operated on the pixels lying on the edges or those in their neighborhoods gives a zero response. However, for a vertical edge, it is seen from (10) that for  $\alpha = -\beta$ ,  $\nabla^2 I_p = 0$  for all  $p = 1, 2, \dots, 9$ . In this case, the Laplacian operator  $L$  as given by (8) can be expressed as

$$L^{(\gamma)} = \gamma L_3 \quad (12)$$

where  $\gamma$  is an arbitrary constant and  $L_3$  is a Laplacian kernel defined as

$$L_3 = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

Note that the Laplacian kernel of  $L_3$  has been used in the literature [26] for estimating of the noise variance of images. It is now clear from the above study as to why  $L_3$  provides a better estimate of the noise variance in images. As a matter of fact, it eliminates the effect of orthogonal edges in the Laplacian map of the images. However, our objective here is to study the effect of  $L^{(\gamma)}$  in the nonlinear diffusion filter given by (3).

The nonlinear filter described by (3) has two diffusion terms, namely  $(\tau c(\cdot) \nabla^2 u)$  and  $(\tau \nabla_d c(\cdot) \nabla_d u)$ , which are linearly scaled by the step-size  $\tau$ . Although in practice both these terms are linearly scaled by the same step-size  $\tau$ , the possibility of scaling each term by a different scale factor has been pointed out in [27] in order to provide some controlling mechanism to the forward and backward diffusion components of the filtering process. In (3), when  $L^{(\gamma)} = \gamma L_3$  is used, the first diffusion term becomes  $(\tau' c(\cdot) (L_3 * u))$ , where  $\tau' = \tau \gamma$  and  $*$  represents the convolution operation. Thus, the approximation of the Laplacian operator by  $L^{(\gamma)}$  effectively provides two different scaling factors,  $\tau'$  and  $\tau$ , to the two diffusion terms, respectively and the Laplacian operator is replaced by  $L_3$ . Now, we again consider the images shown in Fig. 2 containing the orthogonal and diagonal edges to examine the effect of  $L^{(\gamma)}$  on the nonlinear diffusion of images in the presence of edges. For this purpose, we chose  $\gamma = 1$ .

The image of Fig. 2(a) is filtered using (3) with the step-size  $\tau = 0.1$  which employs the diffusivity function of (7) with  $K = 10$ . Fig. 3(a) and Fig. 3(b) show the filtered images after 25 and 100 iterations, respectively, resulting from the use of the Laplacian operator  $L_3$ . As expected,  $L_3$  is capable of preserving the vertical edge of the image, since  $\nabla_d c(\cdot)$  and  $u * L_3$  are exactly zero. Thus, the diffusion process is completely halted at the edge. The same result could be observed in the presence of a horizontal edge in the image.

Next, the filter given by (3) with the same parameter settings as used for the orthogonal edge is applied to the image of Fig. 2(b) i.e. an image with a diagonal edge ( $\theta = 45^\circ$ ). The results after 25 and 100 iterations are shown in Fig. 3(c) and Fig. 3(d), respectively. Even though  $L^{(\gamma)}$  has been designed

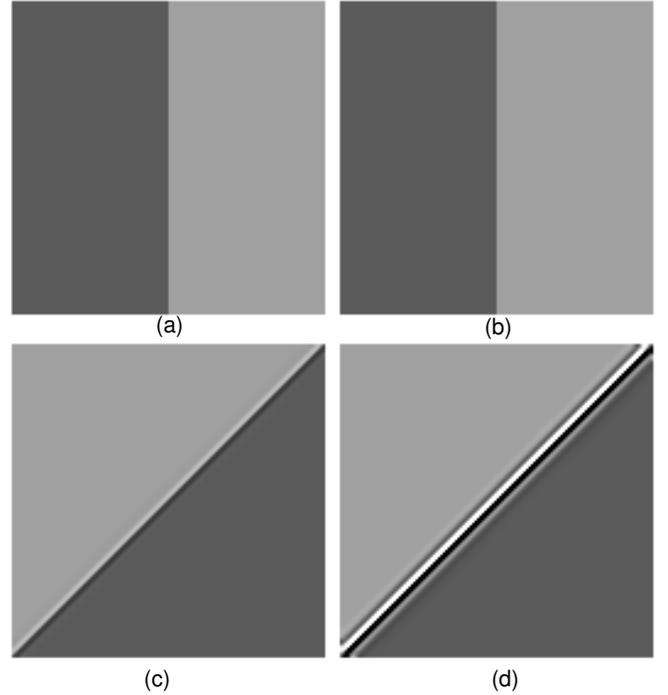


Fig. 3. Diffusion of images containing vertical and diagonal step edges by using the Laplacian kernel  $L^{(\gamma)}$  in (3). The vertical edge after (a) 25 iterations, (b) 100 iterations, and the diagonal edge after (c) 25 iterations, and (d) 100 iterations.

to preserve orthogonal edges, it is seen from these figures that the use of the Laplacian kernel does not smooth the diagonal edge. On the contrary, it gets enhanced as a result of an inverse diffusion process taking place. However, as the number of iterations is increased, ringing artifacts are formed resulting from the up lifting and down lifting of the pixel intensities on both sides of the edge. It also seen that the phenomenon of these lifting of the pixel intensities violates the extremum principle [28], [29]:

$$\min(u^{(0)}) \leq u_{(i,j)}^n \leq \max(u^{(0)}) \quad \forall n \in [0, +\infty] \quad (13)$$

It is known that the nonlinear diffusion filter (3) using the diffusivity function of (7) and the Laplacian kernel  $L_1$  supports the extremum principle [13] even though it performs inverse diffusion on regions where  $\|\nabla u\| > K$ . In the next subsection, we will determine the condition under which the nonlinear diffusion filter does not violate the extremum principle so that a scheme overcoming this problem could be devised.

### B. An Analysis of $L^{(\gamma)}$ in Context of Extremum Principle

In this subsection, we analyze the problem to determine the condition for observing the extremum principle by the nonlinear diffusion filter when it uses  $L^{(\gamma)}$  as the Laplacian kernel. For this analysis, we utilize an expression of the nonlinear diffusion filter in the form of a system of ordinary differential equations given by [13]

$$\begin{aligned} \partial \mathbf{u} / \partial t &= A(\mathbf{u}) \mathbf{u} \\ \mathbf{u}(t=0) &= \mathbf{f} \end{aligned} \quad (14)$$

in which  $\mathbf{u} = (u_1, \dots, u_q, \dots, u_N)$  is a vector consisting of the pixel values of the image taken column-wise such that  $u_q$  represents the pixel value at position  $(i, j)$ ,  $N$  denotes the total number of pixels in the image, and  $A$  is an  $N \times N$  matrix with elements  $a_{k,l}(\mathbf{u})$ . When (14) employs the Laplacian kernel  $L^{(\gamma)}$ , the element of  $A$  can be expressed as

$$a_{k,l} = \begin{cases} (c_k + c_l)/2 + (2\gamma - 1)c_k, & l \in N_o(k) \\ -\gamma c_k, & l \in N_d(k) \\ -\sum_{l \in N_d(k)} \{(c_k + c_l)/2 + (2\gamma - 1)c_k\} + \sum_{l \in N_e(k)} \gamma c_k, & l = k \\ 0, & \text{else} \end{cases} \quad (15)$$

where  $N_o(k)$  and  $N_d(k)$  denote the sets of the indices of the immediate neighbors of  $u_k$  in the 2-D image in the orthogonal and diagonal directions, respectively, and  $c_q$  is the diffusion coefficient at pixel position  $q$ . In [30], it is shown that if matrix  $A$  satisfies the conditions given by

$$\begin{aligned} \text{(S1)} \quad & \sum_{l=1}^N a_{k,l} = 0 \\ \text{(S2)} \quad & a_{k,l} \geq 0 \text{ for all } l = 1, \dots, N \text{ (} l \neq k \text{) for all } k = 1, \dots, N \text{ for which } u_k \text{ is an extremum} \end{aligned}$$

then the diffusion filter (14) satisfies the extremum principle. It is seen from (15) that the filter employing the Laplacian kernel  $L^{(\gamma)}$  always satisfies the condition (S1); however, the condition (S2) is not satisfied because of the presence of the negative entries in  $A$  for indices  $l \in N_d(k)$  and possible negative terms in  $A$  for indices  $l \in N_o(k)$  whenever  $\gamma < 1/4$ .

In order for the filter given by (14) to satisfy the extremum principle, we now propose to modify the matrix  $A$  by using a function  $f(u_k)$  as follows.

$$a_{k,l} = \begin{cases} (c_k + c_l)/2 + (2\gamma - 1)c_k, & l \in N_o(k) \\ -\gamma f(u_k)c_k, & l \in N_d(k) \\ -\sum_{l \in N_d(k)} \{(c_k + c_l)/2 + (2\gamma - 1)c_k\} + \sum_{l \in N_e(k)} \gamma f(u_k)c_k, & l = k \\ 0, & \text{else} \end{cases} \quad (16)$$

It is seen from the above equation that with the proposed modification of  $A$ , the condition (S1) is still readily satisfied. However, in order to satisfy the condition (S2),  $f(u_k)$  should be capable of making the entries of  $A$  for  $l \in N_d(k)$  to vanish whenever  $u_k$  is an extremum. In the next subsection, we explain how this can be accomplished by designing a data-dependent kernel and discuss its implication on the performance of the filter.

### C. An Edge Adapting Laplacian Kernel

Using the kernel  $L^{(\gamma)} = \gamma(2L_1 - 2L_2)$  in (3), which is a fully discrete version of (14), we obtain

$$\begin{aligned} (u_{i,j}^{(n+1)} - u_{i,j}^{(n)})/\tau &= 2\gamma [c(\cdot)_{i,j}(u * L_1)_{i,j}]^{(n)} - \\ & 2\gamma [c(\cdot)_{i,j}(u * L_2)_{i,j}]^{(n)} + \frac{1}{2} \left[ \sum_d \nabla_d c(\cdot)_{i,j} \nabla_d u_{i,j} \right]^{(n)} \end{aligned} \quad (17)$$

The backward Laplacian diffusion term  $2\gamma [c(\cdot)_{i,j}(u * L_2)_{i,j}]$  in (17) results from the entries in  $A$  corresponding to  $l \in N_d(k)$  (i.e.  $-\gamma c_k$ ) and  $\sum_{l \in N_e(k)} \gamma c_k$  part of the entries corresponding

to  $l = k$  of  $A$ . It is only these entries that are affected when  $A$  is modified as in (16) using the function  $f(\cdot)$ . Thus, the proposed modification of  $A$  results in modifying (17) as

$$\begin{aligned} (u_{i,j}^{(n+1)} - u_{i,j}^{(n)})/\tau &= 2\gamma [c(\cdot)_{i,j}(u * L_1)_{i,j}]^{(n)} - \\ & 2\gamma [f(\cdot)c(\cdot)_{i,j}(u * L_2)_{i,j}]^{(n)} + \frac{1}{2} \left[ \sum_d \nabla_d c(\cdot)_{i,j} \nabla_d u_{i,j} \right]^{(n)} \end{aligned} \quad (18)$$

In essence, the proposed modification of  $A$  simply affects the backward Laplacian diffusion term by multiplying it with  $f(\cdot)$ . Our objective is to make (18) independent of the backward Laplacian diffusion term at an extremum of  $u$  by letting  $f(\cdot)$  assume a value of zero at such pixel positions of the image.

In [30], a non-standard approximation of the gradient modulus is suggested as

$$\tilde{\nabla} u_{i,j} = [\max(-\nabla_N u_{i,j} \cdot \nabla_S u_{i,j}, 0) + \max(-\nabla_E u_{i,j} \cdot \nabla_W u_{i,j}, 0)]^{1/2} \quad (19)$$

which assumes a value of zero at an extremum of  $u$ . Using this approximation for the gradient,  $\tilde{\nabla} u_{i,j}$ , we propose a function  $f(\cdot)$  to have the following form:

$$f(u_{i,j}) = c(0) - c(\tilde{\nabla} u_{i,j}) \quad (20)$$

With this choice of  $f(u_{i,j})$ , the backward Laplacian diffusion term of the filter (18) is eliminated at extrema of  $u$  and the filter satisfies the extremum principle. In the following, we investigate the behavior of this modified filter given by (18).

Let us consider the value  $u_{r,s}$  that is an edge pixel. Then, it is easy to see that the approximate gradient modulus given by (19) for such a pixel is zero; thus, the function  $f(u_{r,s})$  given by (20) assumes a zero value. Since, as discussed in Section II, the third diffusion term in (18) vanishes for an edge pixel, (18) assumes the following form

$$(u_{r,s}^{(1)} - u_{r,s}^{(0)})/\tau = 2\gamma [c(\cdot)_{r,s}(u * L_1)_{r,s}]^{(0)} \quad (21)$$

for the first iteration of the filtering operation, that is, a forward diffusion is performed on this pixel. Thus, unlike the filtering operation performed by (17) in which the diffusion process on the orthogonal edges is completely halted, in the filtering operation performed by the filter (18) all the edge pixels including the orthogonal edge pixels would undergo the diffusion operation. However, in order to make the filter (18) not to perform more diffusion on the edges than done by (4), one has to choose the value of  $\gamma \leq 1/2$ . A small value of  $\gamma$ , on the other hand, would impact negatively on the noise filtering capability of (18) for the smooth regions of the image.

In order to increase the noise reduction performance of (18) without hindering its edge preservation capability, it is required to devise a mechanism of increasing the value of  $\gamma$  beyond 0.5 without increasing the forward diffusion on the edges. For  $\gamma > 1/2$ , the forward Laplacian diffusion performed by (18) is larger than that performed by the nonlinear diffusion filter (4). In order to eliminate the impact of the extra diffusion on

the edges, we modify the forward diffusion term  $2\gamma c(\cdot)_{i,j}(u * L_1)_{i,j}$  of (18) as

$$c(\cdot)_{i,j}(u * L_1)_{i,j} + f(\cdot)(2\gamma - 1)c(\cdot)_{i,j}(u * L_1)_{i,j} \quad (22)$$

where the first term is exactly same as the forward Laplacian diffusion term of (3) and the second term is the extra diffusion part in (18). Thus, with the modified forward Laplacian diffusion term, its negative effect on the edges would not exceed that of (3) because of the presence of  $f(\cdot)$ , whereas the performance of (18) in smooth regions should improve because of the value of  $\gamma$  being greater than  $1/2$ . The nonlinear diffusion filter (18), after this modification, becomes

$$\begin{aligned} (u_{i,j}^{(n+1)} - u_{i,j}^{(n)})/\tau &= [c(\cdot)_{i,j}(u * L_1)_{i,j}]^{(n)} + \\ \{f(\cdot) [(2\gamma - 1)c(\cdot)_{i,j}(u * L_1)_{i,j} - 2\gamma c(\cdot)_{i,j}(u * L_2)_{i,j}]\}^{(n)} &+ \\ \frac{1}{2} \left[ \sum_d \nabla_d c(\cdot)_{i,j} \nabla_d u_{i,j} \right]^{(n)} &\quad (23) \end{aligned}$$

It is important to note that the filter (23) still satisfies the extremum principle, since the backward Laplacian diffusion term is still intact compared to that in filter (18). The purpose of modifying (18) into (23) has been to improve the noise reduction capability of the resulting filter by using a larger value of the parameter  $\gamma$ . We would now examine as to how this modification affects the performance of (23) in terms of processing orthogonal edges. Since,  $L_3 = 2L_1 - 2L_2$ , the second term on the right side of (23) can be simplified as

$$\begin{aligned} f(\cdot) [(2\gamma - 1)c(\cdot)_{i,j}(u * L_1)_{i,j} - 2\gamma c(\cdot)_{i,j}(u * L_2)_{i,j}] &= \\ f(\cdot) [c(\cdot)_{i,j}\gamma(u * L_3)_{i,j} - c(\cdot)_{i,j}(u * L_1)_{i,j}] &\quad (24) \end{aligned}$$

As at the orthogonal edges  $(u * L_3) = 0$ , the diffusion performed on these edges by (23) is independent of the value of  $\gamma$ , and since it is generally assumed that  $c(0) = 1$ , the Laplacian diffusion process carried out by filter (23) on an orthogonal edge is given by

$$\begin{aligned} [c(\cdot)_{i,j}(u * L_1)_{i,j}] - f(\cdot) [c(\cdot)_{i,j}(u * L_1)_{i,j}] &= \\ c(\|\nabla u_{i,j}\|)_{i,j} c(\tilde{\nabla} u_{i,j})_{i,j} (u * L_1)_{i,j} &\quad (25) \end{aligned}$$

If we compare the diffusion coefficient  $c(\|\nabla u_{i,j}\|)c(\tilde{\nabla} u_{i,j})$  of (23) with the diffusion coefficient  $c(\|\nabla u_{i,j}\|)$  in (4), the diffusion performed on the orthogonal edges by (23) is less than that performed by (4) from iteration 2 onward, since  $c(\|\nabla u_{i,j}\|)^{(n)} > c(\|\nabla u_{i,j}\|)^{(n)} c(\tilde{\nabla} u_{i,j})^{(n)}$  for  $c(\cdot) \in [0,1]$ .

Before closing this section, we make an explicit comparison of the filters given by (3) and (23). By combining the first two terms on the right side of (23), we have a single Laplacian diffusion term in the form of  $c(\cdot)(u * L)_{i,j}$ , where  $L_{i,j}$  is given by

$$L_{i,j} = L_1 - f(u_{i,j})((2\gamma - 1)L_1 - 2\gamma L_2) \quad (26)$$

Recall that in (3), the Laplacian operation is performed by using the kernel  $L_1$  as  $\nabla^2 u = L_1 * u_{i,j}$ . Note that if kernel  $L_1$  in (3) is replaced by  $L_{i,j}$ , then (3) becomes the filter given by (23). In a view of the fact that  $L_{i,j}$  depends on  $u_{i,j}$ , hereafter we refer to it as the edge adapting kernel.

## IV. SIMULATION RESULTS

In this section, we study the effect of the proposed edge adapting kernel on the filtering performance of three nonlinear diffusion filters. The nonlinear diffusion filters considered for this purpose are the classical filter proposed by Perona and Malik [8], the anisotropic filter introduced by Carmona and Zhong [31] and a filter due to Yu and Chua [27]. The performance of these filters are examined when the kernel  $L_1$  of these filters is replaced by the proposed one. Their performances are measured in terms of the SNR representing the noise removal capability of the filters, and a figure of merit (FOM) index introduced by Pratt's [32] that represents the edge preservation capability of the filters. The SNR is defined as

$$SNR = 10 \log_{10} (var(\hat{u})/var(I - \hat{u})) \quad (27)$$

where  $\hat{u}$  and  $I$  denote, respectively, the denoised and noiseless images,  $var(\cdot)$  and  $avg(\cdot)$  denote, respectively, the variance and average of the pixel values of the associated image. The index FOM is defined in [32] as

$$FOM = \frac{1}{max(u_D, u_I)} \sum_{i=1}^{u_D} \frac{1}{1 + \phi d_i^2} \quad (28)$$

where  $u_D$  is the number of detected edge points,  $u_I$  the number of edge points in the image  $I$ ,  $\phi$  is a positive scaling factor often chosen to be  $1/9$ , and  $d_i$  the distance between the  $i^{th}$  detected edge pixel in  $\hat{u}$  and the corresponding pixel in  $I$ . For the evaluation of FOM, the Sobel operator is used for the edge detection. Filters in these experiment are implemented using a MATLAB 2008Ra simulation platform on a Windows-based 64-bit icore5 machine with 4-GB RAM.

### A. The nonlinear diffusion filter of Perona and Malik

The proposed edge adapting kernel used in the filter given by (23) has a parameter  $\gamma$ . Recall that the lower limit of  $1/4$  of this parameter is dictated by the extremum principle. On the other hand, a larger value of this parameter within a limit ensuring the numerical stability of the discrete model of the filter improves the noise reduction capability of the filter. In order to see the impact of the value of  $\gamma$  chosen on the diffusion of orthogonal and diagonal edges, we again consider the synthetic images shown in Figs. 2(a) and (b) containing, respectively, vertical and diagonal edges. These images are diffused by employing the filters given by (23) and (3) using the diffusivity function given by (7) for the value of the contrast parameter  $K$  chosen in the interval  $[5, 25]$  and for values of  $\gamma$  in the range  $[0.5, 1.5]$ . The filtering is performed for  $n = 30$  iterations and the time step size  $\tau = 0.1$ .

The variance of the diffused images as a function of  $K$  for different values of  $\gamma$  is depicted in Fig. 4 for the vertical and diagonal edges. It is seen from this figure that the variance of the images diffused by the proposed filter for both the diagonal and vertical edges is consistently higher than that obtained by using (3) indicating a smaller diffusion of the edges by the proposed filter. Also, as expected, the diffusion performed by (23) on the vertical edge is independent of the choice of the value  $\gamma$  (Fig. 4(a)). Even though the noise reduction capability

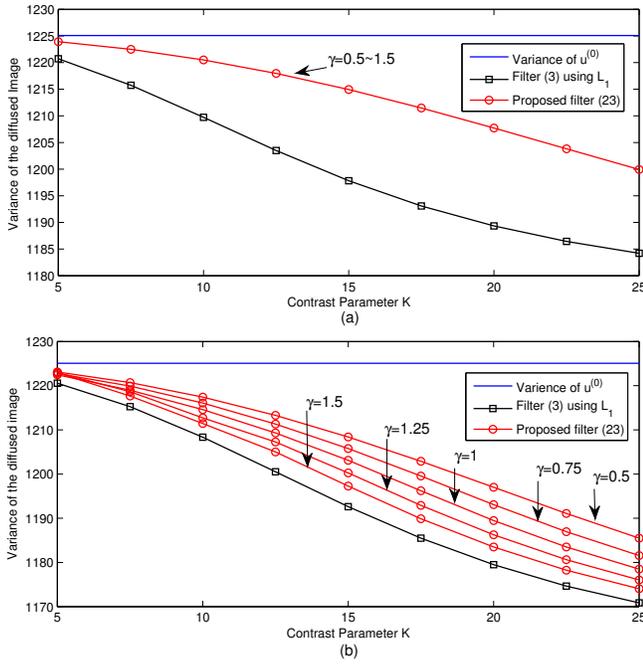


Fig. 4. The variance of the image as a function of the contrast parameter  $K$  and  $\gamma$  illustrating the effects of using the  $L_1$  and  $L_{i,j}$  kernels on the diffusion process of (3) for an image having (a) vertical edge and (b) a diagonal edge.

of the proposed filter is, in general, enhanced at the expense of a reduced edge preservation capability by choosing a larger value of  $\gamma$ , it is seen from Fig. 4(b) that even for  $\gamma = 1.5$ , the diffusion of the diagonal edge by the proposed filter is smaller than that by (3). Therefore, the value of  $\gamma$  for all of the simulation study of this section is chosen to be 1.5.

In order to examine the performance of the filters given by (3) and (23), we use the regularized version [16],  $c(\|\nabla(G_\sigma * u)\|)$ , of the diffusivity function given by (7) in which  $G_\sigma$  denotes the Gaussian kernel with standard deviation  $\sigma$ . The value of  $\sigma$  is chosen to be 1. The value of the contrast parameter  $K$  is set using the so-called threshold freezing scheme given in [33]:

$$K^{(n)} = \frac{1}{\epsilon + \chi \cdot n \cdot \tau} \quad (29)$$

where  $\epsilon = 10^{-10}$  and  $\chi$  is a threshold parameter, which provides a non trivial steady state solution of the nonlinear diffusion filter in a finite number of iterations. The iterative filtering process is stopped once the condition  $\|u^{(n)} - u^{(n-1)}\| / \|u^{(n-1)}\| \leq 10^{-4}$  is satisfied. The time step size  $\tau$  is chosen to be 0.05.

In this simulation study, the synthetic images of *Disk*, and *Flintstone*, the natural image *House*, and an organic superconductor (OSC) image given in [34] are chosen. Table I gives a comparative performance of the two schemes, when  $L_1$  and the proposed kernel are considered. These images are corrupted by a white Gaussian noise (AWGN) with standard deviation  $SD = 15$  and 25. For the purpose of comparison, the SNR and the corresponding FOM values of the filtered images are obtained at the threshold values  $\chi = \chi_{opt}$  that provide the largest possible SNR for the filtered images. This

table also depicts the total number of iterations meeting the stopping condition specified above, and the CPU time. It is seen from this table that the proposed filter yields the SNR and corresponding FOM values that are consistently higher than that provided by the filter using the  $L_1$  kernel. On the average, SNR and FOM values obtained by the filter using the proposed kernel are, respectively, 0.98 dB and 6.98% higher than when the  $L_1$  kernel is used. However, it is noted that this improvement in the performance of the filter is achieved at the expense of an increased computational complexity. The proposed scheme results in a higher number of iterations and a CPU time that is on the average 46% larger.

Fig. 5 depicts the perceptual quality of the image *House* processed by the two filters. The original image and the one degraded by a noise with  $SD = 25$  are shown in Figs. 5(a) and (b), respectively. Figs. 5(c) and (d) show, respectively, the images processed by filters (3) and (23) that employ the optimal values of  $\chi$  of the filters, that is, 0.2 and 0.08, respectively. It is seen from these two images that filter (3) is not as effective in removing the noise and preserving the edges as filter (23). By increasing the value of  $\chi$  to a value larger than  $\chi_{opt}$ , the noise reduction ability of the filters would naturally be reduced. However, the resulting reduction of the diffusion on the edges should increase the FOM value compared to that for  $\chi_{opt}$ . To see the effect of increasing  $\chi$  above  $\chi_{opt}$ , we increase the value of  $\chi$  in filter (3) from its optimal value of 0.2 to 0.35, a value at which the FOM value of the processed image by this filter becomes almost equal to the FOM value of the filter (23) at its optimal  $\chi$ , i.e., 0.7005. It is seen from the the resulting processed image, shown in Fig. 5(e), that the noise reduction ability of filter (3) is considerably reduced, and quantitatively the SNR value gets reduced from its optimal value of 15.46 dB (Fig. 5(c)) to 14.75 dB. Fig. 5(f) shows the image processed by the filter (23) with the value of  $\chi$  increased from its optimal value to that of for filter (3), i.e.,  $\chi = 0.2$ . It is seen from this figure that as expected more noise is left behind compared to that in Fig. 5(d) due to a reduced diffusion, which is particularly visible at the edges. However, the FOM value of the image in Fig. 5(f) is 0.7606, which is larger than the value of 0.7006 for the image in Fig. 5(e).

In order to further compare the edge preservation capability of the two filters, the residual images  $\hat{u} - u^{(0)}$  corresponding to images processed by filters (3) and (23) at their respective  $\chi_{opt}$  are obtained and shown in Fig. 6. It is clear from this figure that the diffusion process of the image edges as carried out by (3) is stronger than that by (23). In order to quantitatively analyze the results of the filtering operations carried out by (3) and (23), the variance of the residual image,  $var(\hat{u} - u^{(0)})$ , and absolute value of its correlation with the estimated image,  $|corr(\hat{u} - u^{(0)}, \hat{u})|$ , are obtained. These values are, respectively, 591 and 0.05 resulting from the filtering operation carried out by (3) and, respectively, 616 and 0.01 resulting from the filtering operation of (23). A higher value of the variance along with a simultaneously lower value of the correlation in the case when the filtering is carried out by (23) indicates the effectiveness of the new kernel in both edge preservation and noise reduction.

TABLE I  
 QUANTITATIVE RESULTS OBTAINED BY USING THE LAPLACIAN KERNEL  $L_1$  AND THE PROPOSED KERNEL  $L_{i,j}$  IN FILTER (3)

Noise level	Degraded Image		Kernel	$\chi_{opt}$	Filtered Image		n	Time(s)
	Image	SNR(dB)			SNR(dB)	FOM		
$SD = 15$	<i>House</i>	10.17	$L_1$	0.40	17.60	0.7133	62	1.12
			Proposed	0.23	17.96	0.7688	75	1.71
	<i>OSC</i>	11.47	$L_1$	0.56	17.08	0.7495	67	3.20
			Proposed	0.52	17.34	0.7817	70	4.13
	<i>Disk</i>	15.95	$L_1$	0.28	28.43	0.8491	78	1.35
			Proposed	0.05	31.16	0.8579	90	2.02
	<i>Flintstone</i>	14.13	$L_1$	0.55	17.97	0.8183	51	7.22
			Proposed	0.45	18.09	0.8624	56	9.11
$SD = 25$	<i>House</i>	6.41	$L_1$	0.2	15.46	0.6184	89	1.60
			Proposed	0.08	16.11	0.7005	117	2.81
	<i>OSC</i>	7.52	$L_1$	0.28	14.33	0.6190	60	4.56
			Proposed	0.22	14.52	0.6976	68	6.48
	<i>Disk</i>	11.71	$L_1$	0.17	25.23	0.8161	108	1.9
			Proposed	0.025	28.56	0.8177	119	3.1
	<i>Flintstone</i>	9.97	$L_1$	0.3	15.48	0.7035	69	9.71
			Proposed	0.21	15.65	0.7831	82	13.29

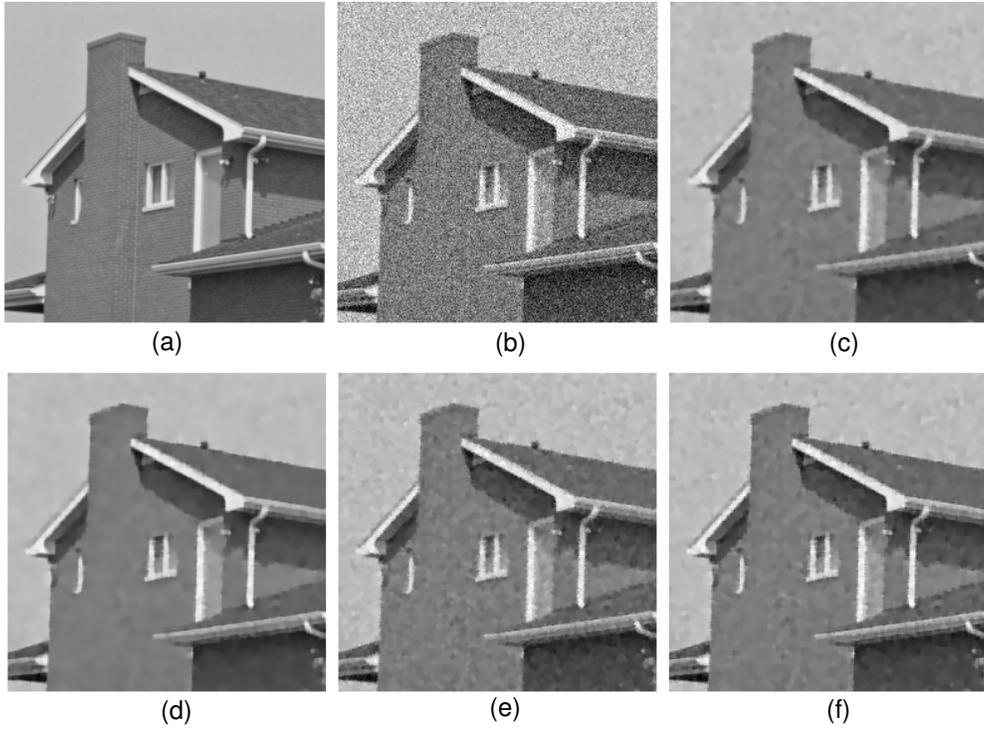


Fig. 5. The perceptual quality of the processed image *House*. (a) Original image. (b) Image contaminated by AWGN with  $SD=25$ . (c) Image denoised using (3) with  $\chi_{opt} = 0.2$ . (d) Image denoised using (23) with  $\chi_{opt} = 0.08$ . (e) Image denoised using (3) with  $\chi = 0.35$ ; (f) Image denoised using (23) with  $\chi = 0.2$ .

For images containing edges with contrast higher than the noise level, such as the image *Disk* shown in Fig. 7, the diffusion coefficients corresponding to the edge pixels is sufficiently small, which provides a good edge preservation capability to both the processing filters. This can easily be seen from FOM values, given in Table I, of the image *Disk* processed by filters (3) and (23). The FOM value of the *Disk* image processed by these two filters are the largest among the images considered in our simulation study without drawing any significant advantage from one filter over the other from standpoint of their edge preservation capability. The good edge preservation capability of both the filters can also be

seen perceptually from the images shown in Fig. 7. However, in terms of noise reduction capability, one clearly sees the superiority of the proposed filter (23) over filter (3) from SNR value in Table I and perceptually from images processed by the two filters shown in Fig. 7.

#### B. The Anisotropic Filter of Carmona and Zhong

As mentioned earlier, the proposed edge adapting kernel  $L_{i,j}$  can also be used in nonlinear diffusion settings other than the one given in (3). In the following, we show how the proposed kernel can be used in the anisotropic filter of

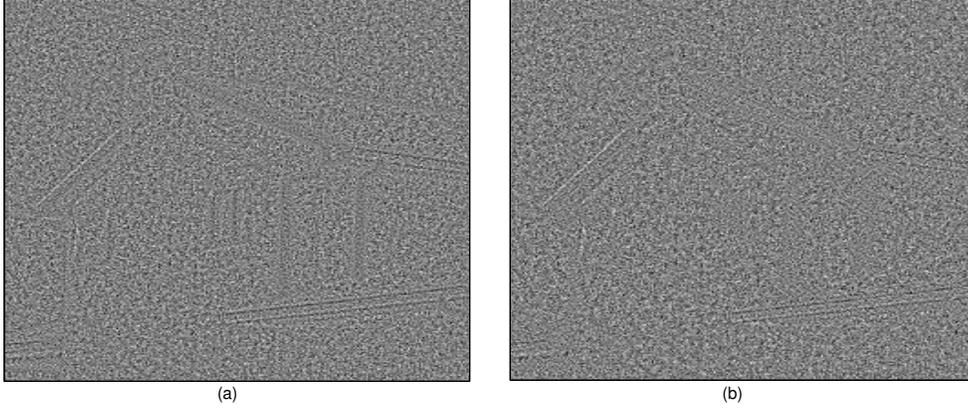


Fig. 6. The quality of edge preservation illustrated by the residual image  $\hat{u} - u^{(0)}$ . The residual of the image processed by (a) filter (3) and (b) filter (23).

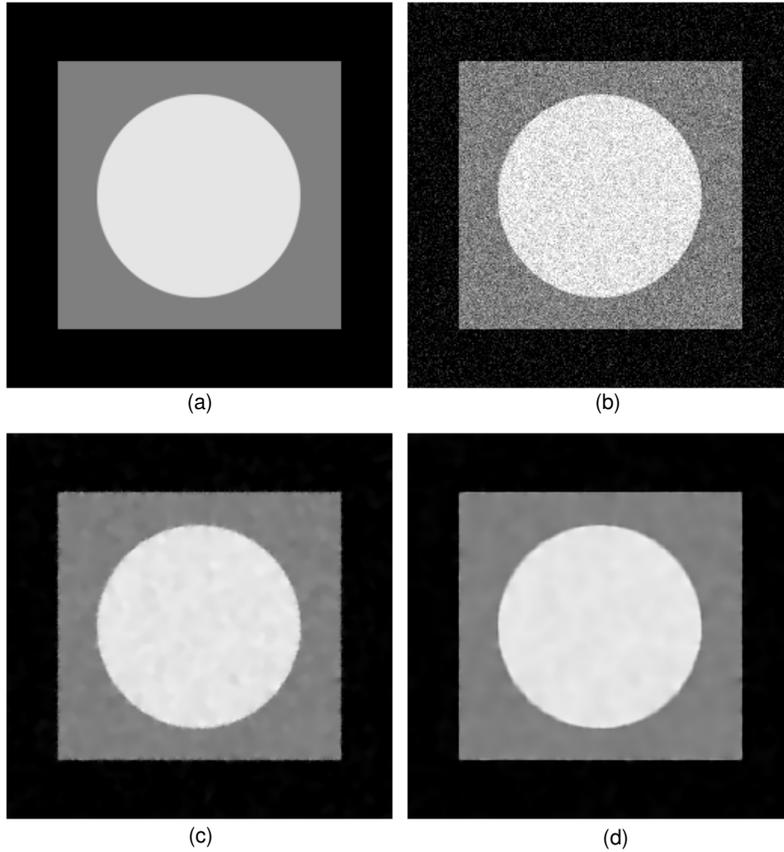


Fig. 7. The perceptual quality of the processed image *Disk*. (a) Original image. (b) Image contaminated by AWGN with  $SD = 25$ . (c) Image denoised by (3). (d) Image denoised by (23).

Carmona and Zhong [31]. The PDE of this filter is given by

$$u_t = a(\|\nabla u\|) (b(\|\nabla u\|) u_{\xi\xi} + c(\|\nabla u\|) u_{\eta\eta}) \quad (30)$$

where  $a(\cdot)$ ,  $b(\cdot)$  and  $c(\cdot)$  are the diffusivity functions that provide a great deal of flexibility in controlling the diffusion process of the images. These diffusivity functions can be chosen in a way such that the image edges are significantly diffused in the direction of their level sets than in the direction of their gradient. Introducing the notation  $\nabla^2 u$  in (30) using

the fact that  $\nabla^2 u = u_{\xi\xi} + u_{\eta\eta}$ , we have

$$u_t = a(\|\nabla u\|) (c(\|\nabla u\|) \nabla^2 u + (b(\|\nabla u\|) - c(\|\nabla u\|)) u_{\xi\xi}) \quad (31)$$

Now in our simulation of the filter of Carmona and Zhong given by the above equation, we use the kernel  $L_1$  and the proposed kernel  $L_{i,j}$  to approximate the Laplacian operator. Note that the use of the approximation of the second-order derivatives in the direction of the level set and gradient given

by

$$u_{\xi\xi} = \frac{u_{xx}u_y^2 - 2u_xu_yu_{xy} + u_{yy}u_x^2}{u_x^2 + u_y^2} \quad (32)$$

and

$$u_{\eta\eta} = \frac{u_{xx}u_x^2 + 2u_xu_yu_{xy} + u_{yy}u_y^2}{u_x^2 + u_y^2} \quad (33)$$

ensures that the filter given by (30) or (31) employs the  $L_1$  kernel. In order to use the proposed kernel, we use  $L_{i,j}$  in (31) to approximate the Laplacian operator and (32) for  $u_{\xi\xi}$ .

The parameter settings used for  $L_{i,j}$  to perform the Laplacian operator  $\nabla^2$  are the same as that used in Section IVA except that the contrast parameter  $K$  is now set to unity. The diffusivity function  $a(\|\nabla u\|)$  is chosen to be the same as in [31], i.e.,  $a(\|\nabla u\|) = [K^2 / (K^2 + \|\nabla u\|^2)]^{1/2}$  with  $K$  given by (29),  $b(\|\nabla u\|)$  is set to be unity, and  $c(\|\nabla u\|)$  is given by (7) with  $K$  set to have a value of 10. The time step size is chosen to be 0.05 for the first 50 iterations and 0.25 afterward. The stopping criterion used in the filter simulation is the same as that in Section IV.A.

Table II gives the quantitative results in terms of the SNR and FOM of the four images degraded by AWGN with  $SD = 25$  and then processed by the Carmona and Zhong filter employing the  $L_1$  and  $L_{i,j}$  kernels. It is seen from this table that replacing the kernel  $L_1$  by  $L_{i,j}$  enables the Carmona and Zhong filter to perform consistently better in processing all the four images. On the average, the proposed kernel yields the values of the SNR and FOM that are, respectively, 0.76 dB and 5.3% higher than when the kernel  $L_1$  is employed. However, the convergence rate of the filter using the proposed kernel is lower than that when  $L_1$  used, and this is reflected in making the computational time of the filter with  $L_{i,j}$  kernel to be 24.5% higher.

### C. GVF-based Anisotropic Filter

Nonlinear diffusion filters can be used for simultaneous deblurring and denoising, since they can be set to perform an inverse diffusion on the edges. One of the techniques for performing deblurring-denoising, introduced by Yu and Chua [27], is the gradient vector flow (GVF)-based anisotropic diffusion filter given by

$$u_t = c(\|\nabla u\|) \nabla^2 u - \vec{v} \cdot \nabla u \quad (34)$$

where  $\vec{v}$  is a gradient vector flow field calculated through an evolutionary process given by

$$\begin{aligned} \vec{v}_t &= \mu \nabla^2 \vec{v} - \left( \vec{v} - \nabla E \right) \|\nabla E\|^2 \\ \vec{v}(t=0) &= \nabla E \end{aligned} \quad (35)$$

and  $E$  is the image edge map defined by

$$E = 1 - \frac{1}{\sqrt{2\pi}K} e^{(-\|\nabla u\|^2/2K^2)} \quad (36)$$

In (35),  $\mu$  is a positive blending coefficient used to smoothen the GVF field,  $\vec{v}$ . The deblurring process performed by (34) is due to the inverse diffusion of the edges carried out by the second term of right of (34).

The filter given by (34) has a Laplacian forward diffusion term,  $c(\cdot)\nabla^2 u$ , which in a discrete setting is implemented as  $c(\cdot)_{i,j}(u * L_1)_{i,j}$ . In order to use the proposed edge adapting kernel, we simply replace the Laplacian kernel  $L_1$  in this forward diffusion term by the kernel  $L_{i,j}$ .

In this simulation study, filter (34) using the  $L_1$  and  $L_{i,j}$  kernels is set up as follows:

- 1) The evolutionary processes of (34) and (35) are discretized in the temporal domain using the forward Euler approximation with the time-step size  $\tau$  set to 0.1
- 2) The spatial derivatives in (34) and (35) are based on the central difference operators with a reflective boundary condition.
- 3) The total number of iterations for the discretized diffusion process of (34) is chosen to be 50 with the value of  $\vec{v}$  updated after each 10 iterations using (35). The total number of iterations for the discretized realization of (35) is chosen to be 20.
- 4) The blending coefficient,  $\mu$ , in (35) is set to unity.
- 5) The diffusivity function is computed as  $c(\|\nabla u\|) = e^{(\|\nabla u\|/2K^2)}$  with the value of  $K$  at each iteration set to the 80% value of the integral of the gradient magnitude of the image. The same value of  $K$  is also used in (36) for the evaluation of  $E$ .
- 6) Parameter settings used for the  $L_{i,j}$  kernel are the same as those used in Section IV.B.

In order to see the effect of the proposed kernel in the deblurring-denoising process of (34), the same four images as considered earlier are first blurred using a  $15 \times 15$  Gaussian kernel with the standard deviation of 1.5 and then contaminated with an AWGN having  $SD = 15$ . Table III gives the quantitative results for the four images in terms of the SNR and FOM. It is seen from this table that the use of the proposed kernel in the filtering process of (34) yields the values of SNR and FOM that are consistently better than that in the case when the  $L_1$  kernel is used. On the average, the SNR and FOM values are, respectively, 1.1 dB and 25.73% higher. This improvement in the performance of the filter using the proposed kernel is achieved at the expense of a modest increase of only 10% in the computational time. Fig. 8 depicts the perceptual quality of the image *OSC* processed by (34) using the  $L_1$  and  $L_{i,j}$  kernels. The original test image is shown in Fig. 8(a) and the corresponding degraded image (blurred and noise contaminated) is shown in Fig. 8(b). Figs. 8(c) and (d) show the images resulting from the processing of image of Fig. 8(b) by the filter (34) with the  $L_1$  and  $L_{i,j}$  kernels, respectively. It is seen from these two processed images that the proposed kernel is more effective in deblurring the edges.

## V. CONCLUSION

In this paper, a new scheme to enhance the edge preservation capability of nonlinear diffusion filters has been developed. In contrast with the existing methods for increasing the edge preservation capability of nonlinear diffusion filters in which the focus is on significantly reducing the value of diffusion coefficient at the edges, the new scheme is motivated by the

TABLE II  
 QUANTITATIVE COMPARISON OF THE RESULTS OBTAINED BY USING THE LAPLACIAN KERNEL  $L_1$  AND THE PROPOSED KERNEL  $L_{i,j}$  IN FILTER (31)

Noise level	Degraded Image		Kernel	$\chi_{opt}$	Filtered Image		n	Time(s)	
	Image	SNR(dB)			SNR(dB)	FOM			
$SD = 25$	Proposed		$L_1$	0.7305	85	0.038	3.04		
	House	6.41	$L_1$	0.33		15.14	0.6095	185	2.34
			Proposed	0.21		15.76	0.6682	193	3.04
	OSC	7.52	$L_1$	0.43		14.42	0.6198	258	10.10
			Proposed	0.31		14.74	0.6484	273	12.56
	Disk	11.71	$L_1$	0.2		27.04	0.8336	155	1.93
			Proposed	0.1		28.46	0.8368	156	2.43
	Flintstone	9.97	$L_1$	0.4		15.23	0.6843	172	18.66
			Proposed	0.3		15.73	0.7301	178	19.27

TABLE III  
 QUANTITATIVE COMPARISON OF THE RESULTS OBTAINED BY THE FILTER (34) USING KERNELS  $L_1$  AND THE PROPOSED KERNEL  $L_{i,j}$

Image	Degraded Image		Kernel	Filtered Image		Time(s)
	SNR(dB)	FOM		SNR(dB)	FOM	
House	8.85	0.5039	$L_1$	13.48	0.4748	2.16
			Proposed	14.13	0.5484	2.51
OSC	8.71	0.2816	$L_1$	10.45	0.3043	4.87
			Proposed	11.30	0.3757	5.39
Disk	14.87	0.5804	$L_1$	22.22	0.7975	2.11
			Proposed	22.49	0.8015	2.45
Flintstone	9.79	0.1179	$L_1$	8.34	0.2106	11.37
			Proposed	11.02	0.3442	12.39

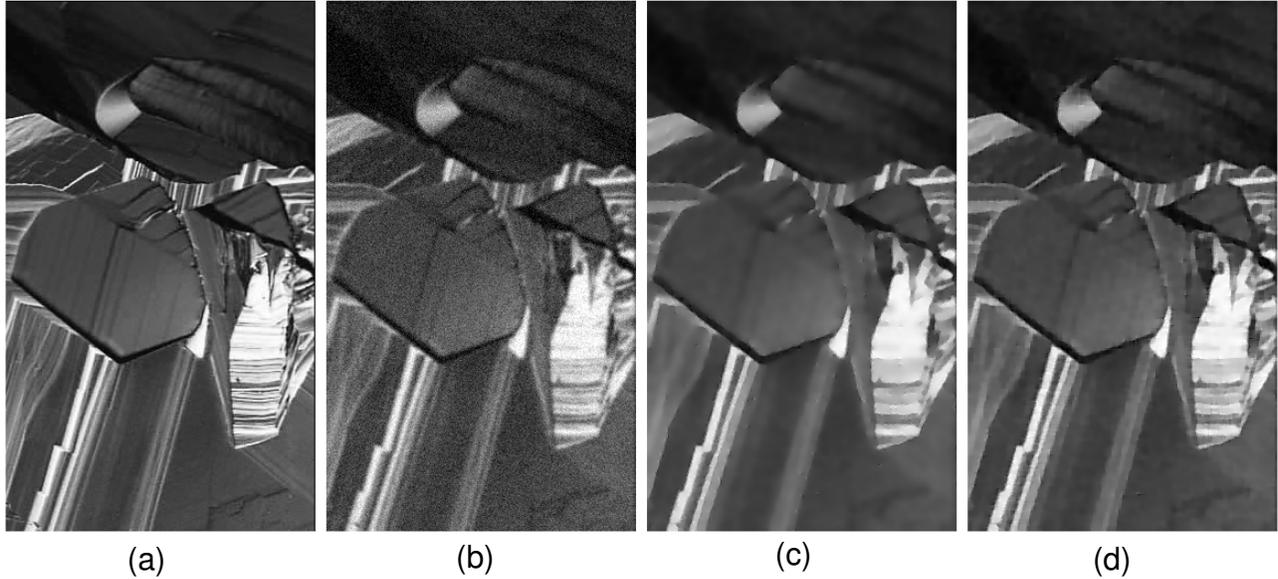


Fig. 8. The perceptual quality of an image restored by (34) using the  $L_1$  and  $L_{i,j}$  kernels. (a) The original image OSC. (b) Degraded image blurred by the  $15 \times 15$  Gaussian kernel with  $SD = 1.5$  and contaminated by AWGN with  $SD=15$ . (c) Image processed by (34) using  $L_1$ . (d) Image processed by (34) using  $L_{i,j}$ .

result of a study on the impact of the anisotropic behavior inherent in the discrete approximations of the Laplacian operator on the diffusion of the edges. The study has shown that the Laplacian kernel when it operates on pixels of the orthogonal edges results in a smaller diffusion of these pixels compared to the case of diagonal edges. Motivated by this result, a new Laplacian kernel has been designed, which when operates on pixels of the orthogonal edges yields a zero response. It has been shown that even though the filter employing this kernel completely halts the diffusion process of the orthogonal edges, the filter itself in this case does not satisfy the extremum

principle. This problem has been analytically investigated to drive a sufficient condition for observance of the extremum principle and eventually to propose an edge adapting Laplacian kernel. To demonstrate the effectiveness of the new kernel, extensive simulations have been carried out by employing this kernel in some well known nonlinear diffusion filters. It has been shown that at the expense of a little or modest increase in the processing time, the use of the proposed kernel in the nonlinear diffusion filters not only provides improved FOM values but also enhances their noise reduction capability.

## REFERENCES

- [1] G. Guido, O. Kabler, R. Kikinis, and F. A. Jolesz. Nonlinear anisotropic filtering of mri data. *IEEE Transactions on Medical Imaging*, 11(2):221–232, 1992.
- [2] J. S. Jin, Y. Wang, and J. Hiller. An adaptive nonlinear diffusion algorithm for filtering medical images. *IEEE Transactions on Information Technology in Biomedicine*, 4(4):298–305, 2000.
- [3] W. Wang. Anisotropic diffusion filtering for reconstruction of poisson noisy sinograms. *Journal of Communication and Computer*, 2(11), 2005.
- [4] H. Y. Kim, J. Giacomantone, and Z. H. Cho. Robust anisotropic diffusion to produce enhanced statistical parametric map from noisy fmri. *Computer Vision and Image Understanding*, 99(3):435–452, 2005.
- [5] B. Li, N. Sang, Z. Cao, and T. Zhang. Enhancement of x-ray angiogram images by adaptive anisotropic diffusion. *Electronics Letters*, 41(20):1107–1109, 2005. 0013-5194.
- [6] H. M. Salinas and D. C. Fernández. Comparison of PDE-Based Nonlinear Diffusion Approaches for Image Enhancement and Denoising in Optical Coherence Tomography. *IEEE Transactions on medical imaging*, 26(6), June 2007.
- [7] M. Schaap, Arnold M. R. Schilham, K. J. Zuiderveld, M. Prokop, E.-J. Vonken, and W. J. Niessen. Fast noise reduction in computed tomography for improved 3-d visualization. *IEEE Transactions on Medical Imaging*, 27(8):1120 – 1129, 2008.
- [8] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–39, 1990.
- [9] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [10] Y-L You and M. Kaveh. Differences in the behaviors of continuous and discrete anisotropic diffusion equations for image processing. In *International Conference on Image Processing, (ICIP)*, volume vol.3, pages 249–253, 1998.
- [11] J. Weickert. A review of nonlinear diffusion filtering. In *Scale-Space Theory in Computer Vision*, volume 1252 of *First International Conference, Scale-Space '97. Proceedings*, pages 3–28, Utrecht, Netherlands, 1997. Springer-Verlag.
- [12] S. Kichenassamy. Perona-malik paradox. *SIAM Journal on Applied Mathematics*, 57(5):1328 – 1342, 1997.
- [13] J. Weickert. *Anisotropic Diffusion in Image processing*. B. G. Teubner, Stuttgart, 1998.
- [14] O. Scherzer and J. Weickert. Relations between regularization and diffusion filtering. *Journal of Mathematical Imaging and Vision*, 12(1):43 – 63, 2000.
- [15] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Transactions on Image Processing*, 7(3):421–32, 1998.
- [16] F. Catte, P. Lions, J. M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182 –193, 1992.
- [17] Z. Lin and Q. Shi. An anisotropic diffusion pde for noise reduction and thin edge preservation. In *Proceedings of the 10th International Conference on Image Analysis and Processing*, pages 102 –107, Venice, Italy, 1999.
- [18] S. L. Keeling and R. Stollberger. Nonlinear anisotropic diffusion filters for wide range edge sharpening. In *Proceedings of the SPIE - The International Society for Optical Engineering*, Proc. SPIE - Int. Soc. Opt. Eng. (USA), pages 1309–22, San Diego, CA, USA, 2000.
- [19] S. L. Keeling and R. Stollberger. Nonlinear anisotropic diffusion filtering for multiscale edge enhancement. *Inverse Problems*, 18(1):175 – 190, 2002.
- [20] S. K. Weeratunga and C. Kamath. PDE-based nonlinear diffusion techniques for denoising scientific and industrial images: an empirical study. In Edward R. Dougherty; Jaakko T. Astola; Karen O. Egiazarian, editor, *Image Processing: Algorithms and Systems*, volume 4667, pages 279–290, 2002.
- [21] A. Rosenfeld and A. C. Kak. *Digital Image Processing*, volume 2. Academic, New York, 2nd edition, 1982.
- [22] F. Voci, S. Eiho, N. Sugimoto, and H. Sekiguchi. Estimating the gradient threshold in the perona-malik equation. *IEEE Signal Processing Magazine*, 21(3):39 – 46, 2004.
- [23] James S. J. Lee, R. M. Haralick, and L. G. Shapiro. Morphologic edge detection. *IEEE Journal of Robotics and Automation*, RA-3(2):142–156, 1987.
- [24] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations of image processing. *Archive for Rational Mechanics and Analysis*, 123(3):199 – 257, 1993.
- [25] J. Weickert and H. Schar. A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance. *Journal of Visual Communication and Image Representation*, 13(1-2):103 – 118, 2002.
- [26] J. Immerkar. Fast noise variance estimation. *Computer Vision and Image Understanding*, 64:300–302, Sept. 1996 1996.
- [27] H. Yu and C.-S. Chua. GVF-based anisotropic diffusion models. *IEEE Transaction on image processing*, 15(6):1517–24, 2006.
- [28] T. Lindeberg. Scale-space for discrete signals. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(3):234–254, 1990.
- [29] T. Lindeberg. Discrete derivative approximations with scale-space properties: a basis for low-level feature extraction. *Journal of Mathematical Imaging and Vision*, 3(4):349 – 76, 1993.
- [30] M. Welk, G. Gilboa, and J. Weickert. Theoretical foundations for discrete forward-and-backward diffusion filtering. In *Scale Space and Variational Methods in Computer Vision*, 2009.
- [31] R. A. Carmona and S. Zhong. Adaptive smoothing respecting feature directions. *IEEE Transactions on Image Processing*, 7(3):353 – 358, 1998.
- [32] W.K. Pratt. *Digital Image Processing*. Wiley, New York, 1977.
- [33] G. Gilboa, Y.Y. Zeevi, and N. Sochen. Image enhancement segmentation and denoising by time dependent nonlinear diffusion processes. In *Proceedings of International Conference on Image Processing.*, volume 3, pages 134 –137 vol.3, 2001.
- [34] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, Upper Saddle River, NJ, 2002.