Available online at www.sciencedirect.com
SCIENCE DIRECT®
Applied Numerical Mathematics 52 (2005) 1–11
APPLIED NUMERICAL MATHEMATICS
www.elsevier.com/locate/apnum

Nonlinear diffusion filtering on extended neighborhood

Danny Barash

Genome Diversity Center, Institute of Evolution, University of Haifa, Mount Carmel, Haifa 31905, Israel

Available online 17 September 2004

Abstract

A generalization to the nonlinear diffusion of digital images is proposed, taking into account an extended neighborhood as a prolongation to information derived from local, nearest-neighboring pixels. Traditionally, each iteration in nonlinear diffusion filtering is performed on a 3×3 window. While several iterations can propagate this local information to pixels which are further away from the center of the window, it clearly presents a limitation in some applications as compared to non-PDE based filtering approaches in which the window is larger than 3×3. Here, an extension is presented to describe the nonlinear diffusion equation for larger windows, thus overcoming this limitation.

Keywords: Nonlinear diffusion filtering; Adaptive smoothing; Digital TV filtering; Extended neighborhood

1. Introduction

The nonlinear diffusion equation as a way to filter images has already become a well-established image processing method [25]. Seminal works to demonstrate and formulate this approach at the early 90s [19, 20] laid out the foundation to many applications such as denoising, gap completion and computer aided quality control. Still, when comparing the PDE-based methods to non-PDE-based filters (see [1,22,2,3]) it is often noticed that by itself, each iteration step in the numerical solution of a PDE is intrinsically local when compared to a single iteration step in other filtering approaches. This limitation can be overcome by proposing a simple generalization to describe the nonlinear diffusion equation for extended neighborhoods. Furthermore, for unstructured grids, the digital TV filter [8] has been proposed by using graph connectivities to relate between neighboring pixels. We show a simple extension of nonlinear diffusion

E-mail address: dbarash@research.haifa.ac.il (D. Barash).

0168-9274/$30.00 © 2004 IMACS. Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.apnum.2004.07.002

filtering to treat larger neighborhoods on structured grids, which can be further extended to unstructured grids [8]. The intermediate step that is outlined here, for generalizing the nonlinear diffusion equation to treat $(2S+1 \times 2S+1)$-neighborhood, is important both pedagogically and from an application perspective due to its simplicity in treating structured grids.

In Section 2, the extension of adaptive smoothing and nonlinear diffusion from a 3×3 window to a 5×5 window is demonstrated. This generalization is further carried out to treat $(2S+1 \times 2S+1)$-neighborhood in Section 3. Section 4 shows the merits of this generalization on example images, followed by the Conclusions.

2. Importance of extended neighborhood

The extension of gradient based, edge-preserving smoothing to include information from non-nearest neighboring pixels is natural and has been considered before in various contexts and other approaches than the one proposed in this paper (e.g., [5,8,24,22]). Here, unlike these other approaches, we start from Saint-Marc–Chen–Medioni's adaptive smoothing [21] that was reformulated in [2] for consistency with the diffusion equation, and extend the approach from the original 3×3 window to a window of arbitrary size.

The adaptive smoothing approach is fundamental and intuitive. Given an image $I^{(t)}(\vec{x})$, where $\vec{x} = (x_1, x_2)$ denotes space coordinates, an iteration of adaptive smoothing yields:

$$I^{(t+1)}(\vec{x}) = \frac{\sum_{i=-1}^{+1}\sum_{j=-1}^{+1} I^{(t)}(x_1+i, x_2+j)w^{(t)}}{\sum_{i=-1}^{+1}\sum_{j=-1}^{+1} w^{(t)}}, \tag{1}$$

where the convolution mask $w^{(t)}$ is defined as:

$$w^{(t)}(x_1, x_2) = \exp\left(-\frac{|d^{(t)}(x_1, x_2)|^2}{2k^2}\right), \tag{2}$$

where k is the variance of the Gaussian mask. In [21], $d^{(t)}(x_1, x_2)$ is chosen to depend on the magnitude of the gradient computed in a 3×3 window:

$$d^{(t)}(x_1, x_2) = \sqrt{G_{x_1}^2 + G_{x_2}^2}, \tag{3}$$

where

$$(G_{x_1}, G_{x_2}) = \left(\frac{\partial I^{(t)}(x_1, x_2)}{\partial x_1}, \frac{\partial I^{(t)}(x_1, x_2)}{\partial x_2}\right), \tag{4}$$

noting the similarity (see also [4,14,22] for further analogies) of the convolution mask w with the diffusion coefficient in anisotropic diffusion [19,25], or more specifically, the total variation in Rudin–Osher–Fatemi's original work [20] that demonstrated how edge-preserving smoothing can be achieved from energy minimization.

2.1. Smoothing on 1D 3-neighborhood

It was suggested in [21] that Eq. (1) is an implementation of anisotropic diffusion. Briefly sketched, let us consider the case of a one-dimensional signal $I^t(x)$ and reformulate the averaging process as follows:

$$I^{t+1}(x) = c_1 I^t(x-1) + c_2 I^t(x) + c_3 I^t(x+1), \tag{5}$$

with

$$c_1 + c_2 + c_3 = 1. \tag{6}$$

Therefore, it is possible to write the above iteration scheme as:

$$I^{t+1}(x) - I^t(x) = c_1\big(I^t(x-1) - I^t(x)\big) + c_3\big(I^t(x+1) - I^t(x)\big). \tag{7}$$

Taking $c_1 = c_3 = c$, this reduces to:

$$I^{t+1}(x) - I^t(x) = c\big(I^t(x-1) - 2I^t(x) + I^t(x+1)\big), \tag{8}$$

which is a discrete approximation of the linear diffusion equation:

$$\frac{\partial I}{\partial t} = c\nabla^2 I. \tag{9}$$

2.2. Adaptive smoothing on 1D 3-neighborhood

When the weights are space-dependent, one should write the weighted averaging scheme (see [2] for more details) as follows:

$$\begin{aligned} I^{t+1}(x) = {} & \frac{c^t(x-1)I^t(x-1) + c^t(x)I^t(x-1)}{2} + c^t(x)I^t(x) \\ & + \frac{c^t(x+1)I^t(x+1) + c^t(x)I^t(x+1)}{2}, \end{aligned} \tag{10}$$

with

$$\frac{c^t(x-1) + c^t(x)}{2} + c^t(x) + \frac{c^t(x+1) + c^t(x)}{2} = 1, \tag{11}$$

noting that the coefficients c_1 and c_3 of (5) are space-dependent, $c_1 = c(x - h/2)$ and $c_3 = c(x + h/2)$ where $h = 1$, approximated by simple averages. Plugging (11) into (10) and rearranging (as in [2]) leads to a discrete approximation to the nonlinear diffusion equation:

$$\frac{\partial I}{\partial t} = \nabla\big(c(x)\nabla I\big). \tag{12}$$

2.3. Smoothing on 1D 5-neighborhood

Back to non-adaptive smoothing, the averaging process can be extended to include second-neighbors:

$$\begin{aligned} I^{t+1}(x) = {} & c_1 I^t(x-2) + c_2 I^t(x-1) + c_3 I^t(x) \\ & + c_4 I^t(x+1) + c_5 I^t(x+2), \end{aligned} \tag{13}$$

with

$$c_1 + c_2 + c_3 + c_4 + c_5 = 1. \tag{14}$$

Taking $c_1 = c_5 = w_2$, $c_2 = c_4 = w_1$, and $c_3 = 1 - 2w_2 - 2w_1$ this reduces to:

$$\begin{aligned} I^{t+1}(x) = {} & w_2\big(I^t(x-2) + I^t(x+2)\big) + (1 - 2w_2 - 2w_1)I^t(x) \\ & + w_1\big(I^t(x-1) + I^t(x+1)\big), \end{aligned} \tag{15}$$

rearrangement of terms leads to:

$$\begin{aligned} I^{t+1}(x) - I^t(x) = {} & w_2\big(I^t(x-2) - 2I^t(x) + I^t(x+2)\big) \\ & + w_1\big(I^t(x-1) - 2I^t(x) + I^t(x+1)\big) \end{aligned} \tag{16}$$

which is a discrete approximation of the linear diffusion equation:

$$\frac{\partial I}{\partial t} = w_1 \nabla_1^2 I + w_2 \nabla_2^2 I, \tag{17}$$

where ∇_1 denotes ∇ over a grid containing only the nearest-neighbors, and ∇_2 denotes ∇ over a grid containing only the second-neighbors. Typically $w_1 > w_2$ since nearest-neighbors have more influence than second-neighbors.

2.4. Adaptive smoothing on 1d 5-neighborhood

Adaptive smoothing can be extended to treat 5-neighborhood in 1D as follows:

$$\begin{aligned} I^{t+1}(x) = {} & \left(\frac{c^t(x-2) + c^t(x)}{2}\right) I^t(x-2) + \left(\frac{c^t(x-1) + c^t(x)}{2}\right) I^t(x-1) \\ & + c^t(x) I^t(x) + \left(\frac{c^t(x+1) + c^t(x)}{2}\right) I^t(x+1) + \left(\frac{c^t(x+2) + c^t(x)}{2}\right) I^t(x+2), \end{aligned} \tag{18}$$

with

$$\begin{aligned} & \frac{c^t(x-2) + c^t(x)}{2} + \frac{c^t(x-1) + c^t(x)}{2} + c^t(x) \\ & \quad + \frac{c^t(x+1) + c^t(x)}{2} + \frac{c^t(x+2) + c^t(x)}{2} = 1. \end{aligned} \tag{19}$$

Plugging (19) into (18) and rearranging finally leads to:

$$\begin{aligned} & I^{t+1}(x) - I^t(x) \\ & \quad = \frac{c^t(x+2) + c^t(x)}{2}\big[I^t(x+2) - I^t(x)\big] - \frac{c^t(x-2) + c^t(x)}{2}\big[I^t(x) - I^t(x-2)\big] \\ & \qquad + \frac{c^t(x+1) + c^t(x)}{2}\big[I^t(x+1) - I^t(x)\big] - \frac{c^t(x-1) + c^t(x)}{2}\big[I^t(x) - I^t(x-1)\big], \end{aligned} \tag{20}$$

which is a consistent implementation of the nonlinear diffusion equation:

$$\frac{\partial I}{\partial t} = \nabla_1\big(w_1(x_1, x_2)\nabla_1 I\big) + \nabla_2\big(w_2(x_1, x_2)\nabla_2 I\big), \tag{21}$$

where we have used w instead of c, since the variable w was adopted in (17) instead of c in (9). Thus, the weights $w_1(x_1, x_2)$, $w_2(x_1, x_2)$ are the nonlinear diffusion coefficients in the nearest-neighbors grid or second-neighbor grid, respectively, typically taken as:

$$w_{1,2}(x_1, x_2) = g\left(\left\|\nabla_{1,2} I(x_1, x_2)\right\|\right), \tag{22}$$

where $\|\nabla_{1,2} I\|$ is the gradient magnitude on either the nearest-neighbors grid or the second-neighbors grid, respectively, and $g(\|\nabla_{1,2} I\|)$ is an "edge-stopping" function. This function is chosen to satisfy $g(x) \to 0$ when $x \to \infty$ so that the diffusion is stopped across edges. Thus, a fundamental link between the nonlinear diffusion equation and edge-preserving smoothing filters is noticed (in addition to the interesting relationships between nonlinear PDEs and morphological filters that have been explored in [6,18,15], for example). This link will be extended in the next section to a general $(2S + 1 \times 2S + 1)$-neighborhood.

3. Generalized adaptive smoothing and nonlinear diffusion on extended neighborhood

Adaptive smoothing was introduced in [21] as a local process applying a 3×3 window at each iteration, as defined in (1). However, it is natural to extend this definition to an arbitrary, $(2S + 1 \times 2S + 1)$ window

$$I^{(t+1)}(\vec{x}) = \frac{\sum_{i=-S}^{+S} \sum_{j=-S}^{+S} I^{(t)}(x_1 + i, x_2 + j) w^{(t)}}{\sum_{i=-S}^{+S} \sum_{j=-S}^{+S} w^{(t)}}, \tag{23}$$

where i is the index corresponding to the x-direction, j is the index corresponding to the y-direction, and there is no coupling between them. Stability and the extremum principle hold since (22) is a low pass filter, regardless of its nonlinear dependence on the data itself, with positive coefficients that are normalized. A discussion of adaptive smoothing and related filtering approaches without the restriction to nearest-neighbors in the weights is also available in [22,8]. In the rest of this Section, the *extended nonlinear diffusion* is derived using the generalization of adaptive smoothing outlined in (22).

First, it is instructive to derive the *extended nonlinear diffusion* in one-dimension. Considering adaptive smoothing on 1D $(2S + 1)$-neighborhood, it is possible to generalize (18) by using vector notation to:

$$I^{t+1}(x) = \left(\frac{\vec{c}_{(-)} + c(x)\hat{1}}{2}\right) \cdot \vec{I}^{t}_{(-)} + c(x) I^t(x) + \left(\frac{\vec{c}_{(+)} + c(x)\hat{1}}{2}\right) \cdot \vec{I}^{t}_{(+)}, \tag{24}$$

where $\hat{1} = [1, 1, \ldots, 1]$ is the unity vector, "$\cdot$" denotes the dot product, $\vec{c}_{(-)} = [c(x - S), c(x - S + 1), \ldots, c(x - 1)]$, $\vec{c}_{(+)} = [c(x + 1), c(x + 2), \ldots, c(x + S)]$, $\vec{I}_{(-)} = [I(x - S), I(x - S + 1), \ldots, I(x - 1)]$, $\vec{I}_{(+)} = [I(x + 1), I(x + 2), \ldots, I(x + S)]$, $c(x)$ and $I(x)$ are scalars, $I(x)$ being the gray-level intensity at the point of interest x. By analogy to (19), or adaptive smoothing in 1D 3-neighborhood outlined in [2], normalization of the weights can be written in vector notation as:

$$\left(\frac{\vec{c}_{(-)} + c(x)\hat{1}}{2}\right) \cdot \hat{1} + c(x) + \left(\frac{\vec{c}_{(+)} + c(x)\hat{1}}{2}\right) \cdot \hat{1} = 1, \tag{25}$$

and by analogy to (20), or (13) of [2], we obtain:

$$I^{t+1}(x) - I^t(x) = \left(\frac{\vec{c}_{(+)} + c(x)\hat{1}}{2}\right) \cdot \left[\vec{I}^{t}_{(+)} - I^t(x)\hat{1}\right] - \left(\frac{\vec{c}_{(-)} + c(x)\hat{1}}{2}\right) \cdot \left[I^t(x)\hat{1} - \vec{I}^{t}_{(-)}\right], \tag{26}$$

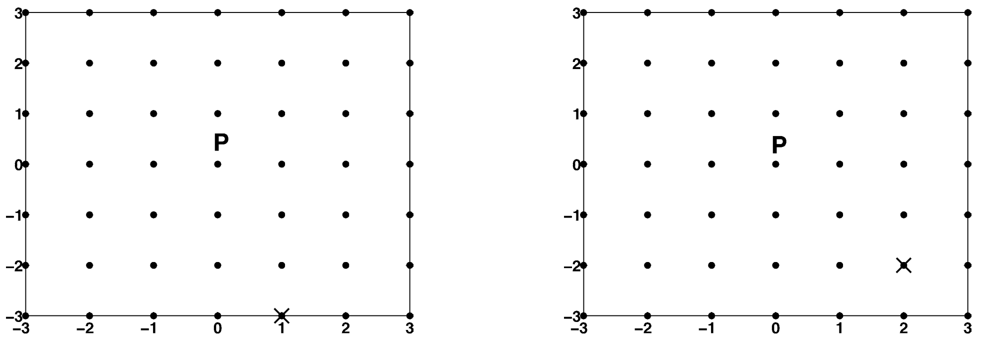

Fig. 1. Two instances for calculating gradients and taking weights with respect to point P, in the case $S = 3$. Marked points correspond to $i = 1$, $j = -3$ of (23), (29) (left figure), and $i = 2$, $j = -2$ of (23), (29) (right figure).

which is an implementation of the nonlinear diffusion equation:

$$\frac{\partial I(x)}{\partial t} = \vec{\nabla} \cdot \big(\vec{w}(x)\vec{\nabla} I(x)\big), \tag{27}$$

where $\vec{\nabla} = [\nabla_{-S}, \nabla_{-S+1}, \ldots, \nabla_S]$ is a vector containing gradients taken at different neighboring configurations (i.e., nearest-neighbors, second-neighbors, etc.) and $\vec{w} = [w_{-S}, w_{-S+1}, \ldots, w_S]$ are the nonlinear diffusion coefficients. It is also possible to write (27) as:

$$\frac{\partial I(x)}{\partial t} = \sum_{j=-S}^{S} \nabla_j \big(w_j(x)\nabla_j I(x)\big), \tag{28}$$

expanding the vector notation used in (27).

Second, the generalization of adaptive smoothing in two-dimensions written in (23) leads to the *extended nonlinear diffusion* in two-dimensions by simple analogy to (27). Taking matrices instead of vectors for ∇, w leads to the *extended nonlinear diffusion*:

$$\frac{\partial I(x_1, x_2)}{\partial t} = \widetilde{\nabla} \cdot \big(\tilde{w}(x_1, x_2)\widetilde{\nabla} I(x_1, x_2)\big), \tag{29}$$

where "$\cdot$" denotes the scalar product between two matrices, and $\widetilde{\nabla}$, $\tilde{w}$ are $(2S + 1) \times (2S + 1)$ matrices that correspond to different neighbor combinations with respect to the center pixel of interest (see Fig. 1 for two illustrative examples). The generalized adaptive smoothing (23) is a discrete approximation of the *extended nonlinear diffusion* (29). Conceptually, further points can be used to obtain higher order accurate schemes for the discretization, although it is not clear how important is the role of accuracy in common image processing tasks. It is noted that in practice, S need not be taken too large (i.e., $S \leqslant 3$), otherwise the generalized adaptive smoothing becomes an inaccurate representation of the extended diffusion equation given in (29).

4. Comparison of denoising with extended vs. standard nonlinear diffusion equation

Following the formulation of the extended nonlinear diffusion in (29), we would like to examine the merits of this generalization with respect to the standard nonlinear diffusion equation. Other numerical

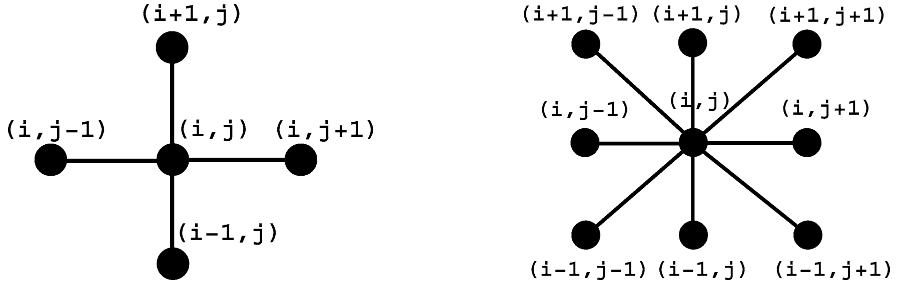

Fig. 2. Two possible stencils for the nearest-neighbors case ($S = 1$): Simple 5-node stencil (left figure), and full 9-node stencil (right figure). The stencils are similar to the ones in [8, Fig. 1].

geometry extensions [23] and nonlinear filtering approaches that are connected to nonlinear diffusion but are not a direct realization of a PDE (e.g., [24,13,12,9–11,1–3]), are discussed elsewhere. Here, a comparison is performed with the purpose of investigating the advantages of using an extended neighborhood over nearest-neighbors in the finite-difference numerical solution of the nonlinear diffusion equation.

As an illustrative example, the Additive Operator Splitting (AOS) numerical scheme (along with a pre-smoothing to ensure the well-posedness of the PDE approach [7]), as described in [26], will be used for the comparison. It should be noted, however, that the formulation outlined in this paper is applicable to any numerical scheme that solves the continuous equation (29). Moreover, it should be stressed that the AOS uses the simple 5-node stencil as in Fig. 2(left), without taking into account diagonal neighboring pixels when calculating gradients, whereas the generalized adaptive smoothing (23) and extended nonlinear diffusion (29) support a full 9-node stencil. The full stencil is shown in Fig. 2(right) and may possess some advantages (also drawn in [8, Fig. 1], although actual experiments in [8] were performed with the 5-node stencil as here, see [8, remark to caption of Fig. 1]). Nevertheless, for simplicity and given the fact that the AOS scheme is an efficient implementation of nonlinear diffusion filtering, we will perform the comparison with the 5-point stencil in Fig. 2(left) extending it to treat second-neighbors ($S = 2$). Thus, the comparison here is performed using the AOS scheme between the cases of nearest-neighbors ($S = 1$) and second-neighbors ($S = 2$), in denoising synthetic images. In our experiments the diagonal elements in the window will not influence the filtering because of the splitting imposed by the AOS scheme. The AOS was devised in [26] for an efficient implementation of nonlinear diffusion image filtering, and has been suggested in the past for solving the Navier–Stokes equations in [16,17].

In Figs. 3 and 4 two noisy synthetic images are assessed for the comparison. Fig. 5 presents the same example on a natural image. The number of arithmetic operations when using AOS with $S = 1$, Figs. 3(B), 4(B) and 5(B), along with 40 iterations is similar to the number of arithmetic operations when using AOS with $S = 2$, Figs. 3(C), 4(C) and 5(C), along with 20 iterations. Intentionally, a large time step of $\Delta t = 4.0$ (which is amenable with the AOS scheme from the point of view of stability and accuracy [26]) was chosen for the comparison in order to accentuate the differences between $S = 1$ and $S = 2$. In these examples (Figs. 3–5) we observe the merits of using an extended neighborhood: for roughly the same complexity, better denoising was achieved, albeit a tradeoff in sharpness. Thus, in some image processing applications, there may be an advantage for solving the nonlinear diffusion equation with an extended neighborhood.

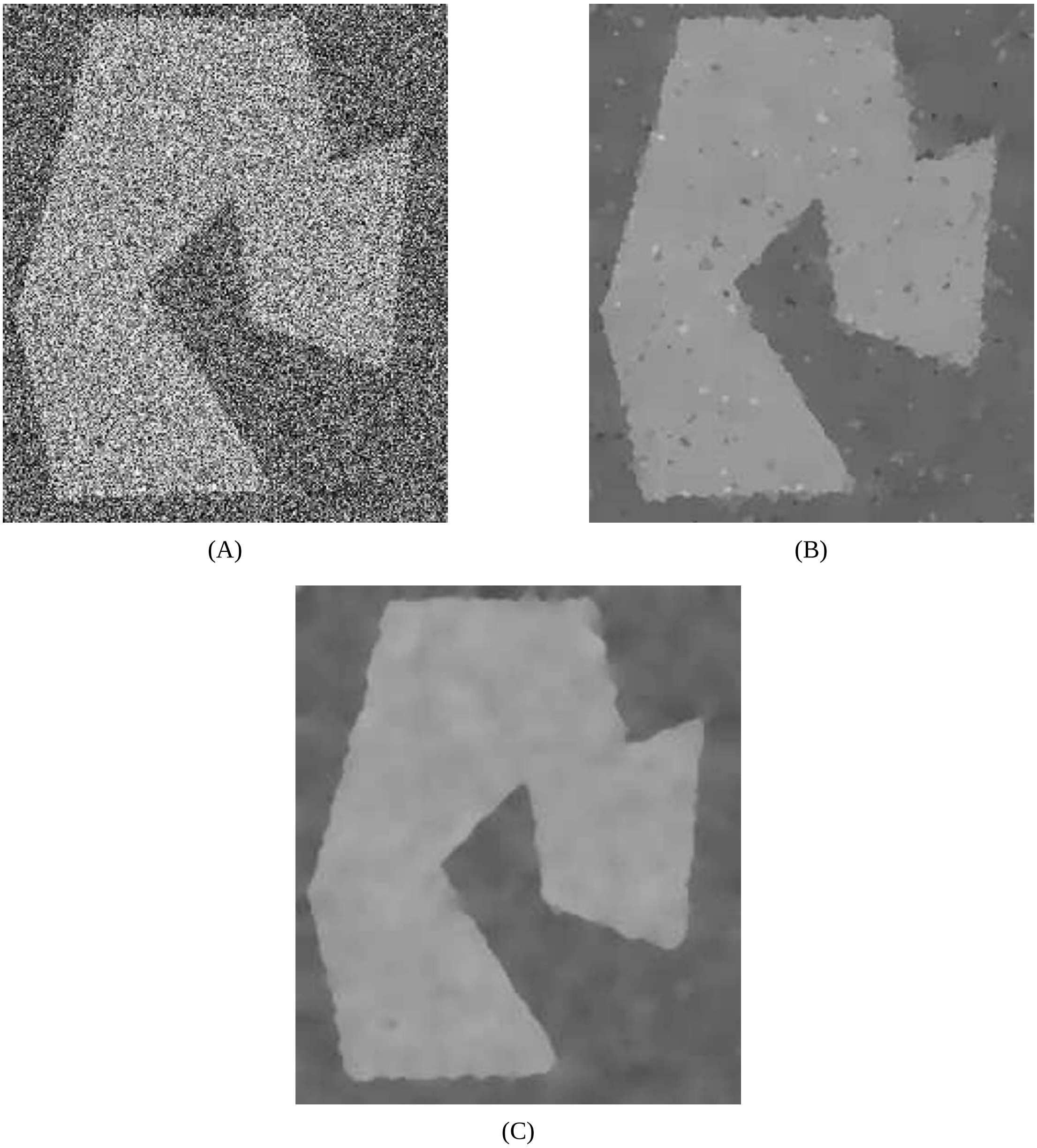

Fig. 3. Comparison Example I (Pants): the additive operator splitting (AOS) [26] with a fixed pre-smoothing of $\sigma = 0.1$ is applied to denoise the original image in (A), with a time step of $\Delta t = 4.0$. Nearest-neighbors ($S = 1$) window was used for 40 iterations in (B), whereas second-neighbors ($S = 2$) window was used for 20 iterations in (C).

5. Conclusions

The formulation outlined in this paper is directly applicable to images on structured grids. It generalizes the nonlinear diffusion equation, based on the adaptive smoothing approach, to treat extended

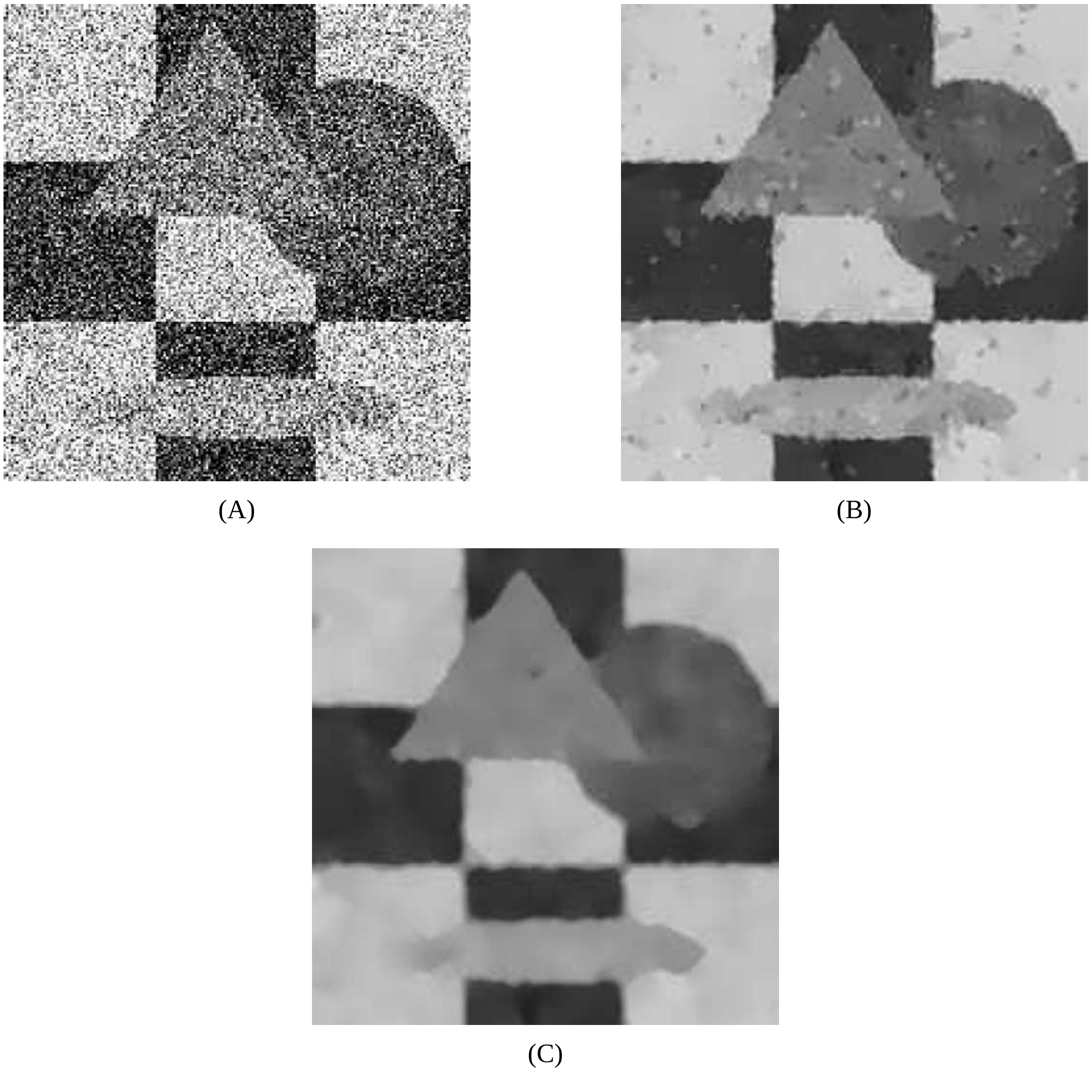

Fig. 4. Comparison Example II (Shapes): the additive operator splitting (AOS) [26] with a fixed pre-smoothing of $\sigma = 0.1$ is applied to denoise the original image in (A), with a time step of $\Delta t = 4.0$. Nearest-neighbors ($S = 1$) window was used for 40 iterations in (B), whereas second-neighbors ($S = 2$) window was used for 20 iterations in (C).

neighborhoods that are larger than a 3×3 window. This formulation can be used in two distinct directions: the first is to construct a robust iterative procedure that performs broad kernel filtering based on (23) with a dynamic window [3], and the second is to extend the PDE solution approach for the diffusion process. Here, we have experimented with the latter direction, extending the PDE solution approach. The next level of complexity for generalizing the diffusion process to unstructured grids, based on graph connectivities, has already been devised in [8] for the case of total variation diffusivities and merits further investigation.

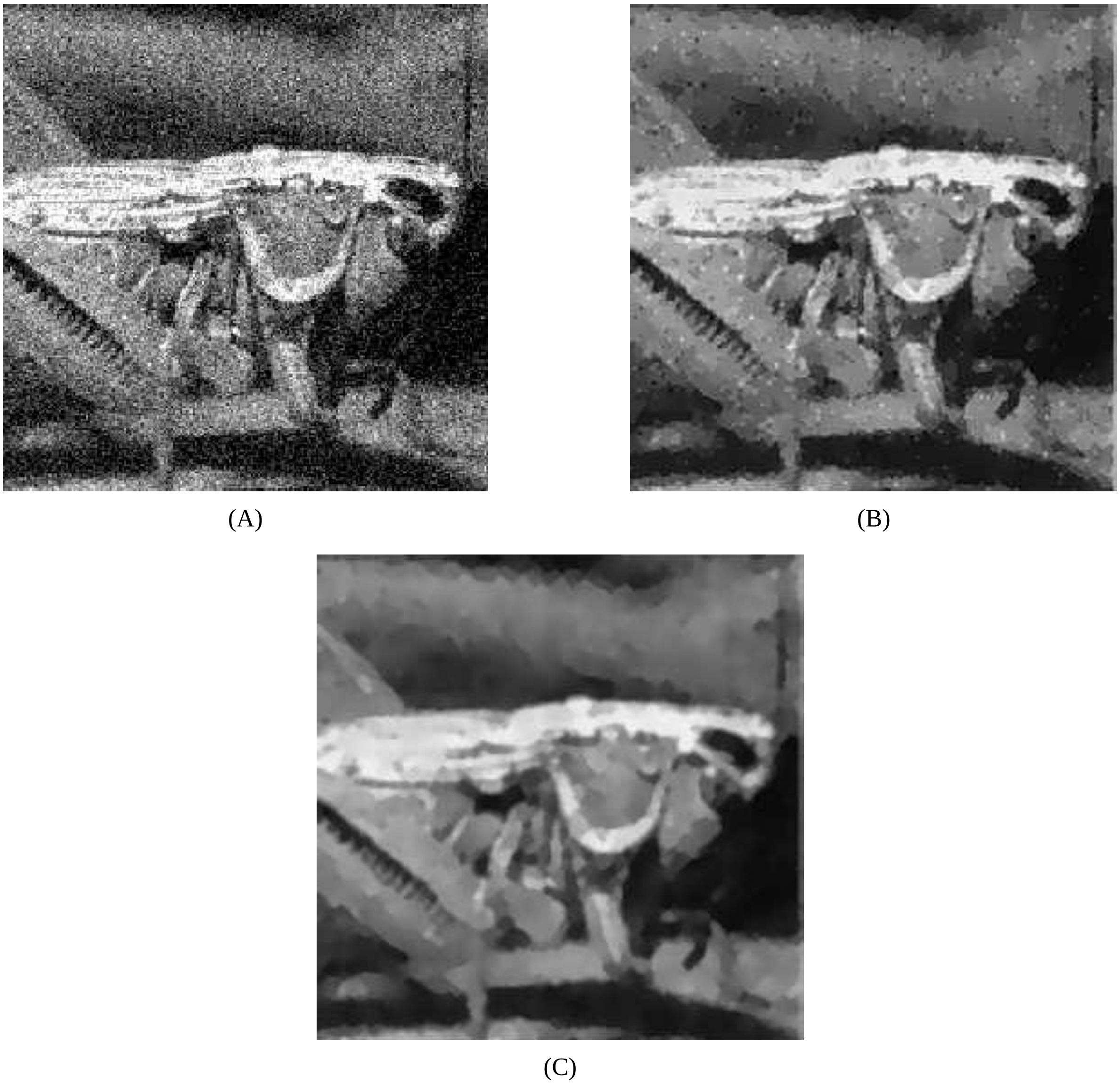

Fig. 5. Comparison Example III (Grasshopper): the additive operator splitting (AOS) [26] with a fixed pre-smoothing of $\sigma = 0.1$ is applied to denoise the original image in (A), with a time step of $\Delta t = 4.0$. Nearest-neighbors ($S = 1$) window was used for 40 iterations in (B), whereas second-neighbors ($S = 2$) window was used for 20 iterations in (C).

Acknowledgement

This work was conducted at the Genome Diversity Center and supported by the Institute of Evolution, University of Haifa, Israel.

References

[1] D. Barash, Bilateral filtering and anisotropic diffusion: Towards a unified viewpoint, Hewlett-Packard Laboratories Technical Report, HPL-2000-18 (R.1), 2000.

[2] D. Barash, A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation, IEEE Trans. PAMI 24 (2002) 844–847.

[3] D. Barash, D. Comaniciu, A common viewpoint on broad kernel filtering and nonlinear diffusion, in: Proc. Scale-Space ’03, Lecture Notes in Comput. Sci., vol. 2695, Springer, Berlin, 2003, pp. 683–698.

[4] M.J. Black, G. Sapiro, D. Marimont, D. Heeger, Robust anisotropic diffusion, IEEE Trans. Imag. Proc. 7 (1998) 421–432.

[5] T. Boult, R.A. Melter, F. Skorina, I. Stojmenovic, G-neighbors, in: S. Chattopadhyay, P.P. Das, D.G. Dastidar (Eds.), Reconstruction of Lines from Real Vision Geometry II, Proc. SPIE, vol. 2060, 1993, pp. 96–109.

[6] V. Caselles, G. Sapiro, D.H. Chung, Vector median filters, inf-sup operations, and coupled PDE’s: Theoretical connections, J. Math. Imag. Vision 8 (2000) 109–119.

[7] F. Catté, P.L. Lions, J.M. Morel, T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, SIAM J. Numer. Anal. 29 (1992) 182–193.

[8] T.F. Chan, S. Osher, J. Shen, The digital TV filter and nonlinear denoising, IEEE Trans. Imag. Proc. 10 (2001) 231–241.

[9] D. Comaniciu, P. Meer, Mean shift analysis and applications, in: Proc. ICCV ’99, IEEE Computer Society, 1999, pp. 1197–1203.

[10] D. Comaniciu, P. Meer, Mean shift: A robust approach towards feature space, IEEE Trans. PAMI 24 (2002) 603–619.

[11] D. Comaniciu, An algorithm for data-driven bandwidth selection, IEEE Trans. PAMI 25 (2003) 281–288.

[12] F. Durand, J. Dorsey, Fast bilateral filtering for the display of high-dynamic range images, ACM Trans. Graph. 21 (2002) 257–266.

[13] M. Elad, On the bilateral filter and ways to improve it, IEEE Trans. Imag. Proc. 11 (2002) 1141–1151.

[14] B. Fischl, E. Schwartz, Adaptive nonlocal filtering: A fast alternative to anisotropic diffusion for image segmentation, IEEE Trans. PAMI 22 (1999) 42–48.

[15] F. Guichard, J.M. Morel, Image Iterative Smoothing and PDE’s, online book, CVonline, 2000.

[16] T. Lu, P. Neittaanmäki, X.-C. Tai, A parallel splitting up method and its application to Navier–Stokes equations, Appl. Math. Lett. 4 (1991) 25–29.

[17] T. Lu, P. Neittaanmäki, X.-C. Tai, A parallel splitting up method for partial differential equations and its application to Navier–Stokes equations, RAIRO Math. Model. Numer. Anal. 26 (1992) 673–708.

[18] P. Maragos, F. Meyer, Nonlinear PDEs and numerical algorithms for modeling levelings and reconstruction filters, in: Proc. 4th Int. Conf. Scale-Space ’03, Lecture Notes in Comput. Sci., vol. 1682, Springer, Berlin, 1999, pp. 363–374.

[19] P. Perona, J. Malik, Scale-Space and edge detection using anisotropic diffusion, IEEE Trans. PAMI 12 (1990) 629–639.

[20] L.I. Rudin, S. Osher, F. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.

[21] P. Saint-Marc, J.S. Chen, G. Medioni, Adaptive smoothing: A general tool for early vision, IEEE Trans. PAMI 13 (1991) 514–529.

[22] N. Sochen, R. Kimmel, A.M. Bruckstein, Diffusions and confusions in signal and image processing, J. Math. Imag. Vision 14 (2001) 195–209.

[23] N. Sochen, R. Kimmel, R. Malladi, A geometric framework for low-level vision, IEEE Trans. Imag. Proc. 7 (1998) 310–318.

[24] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images, in: Proc. ICCV ’98, Narosa Publ. House, New Delhi, 1998, pp. 839–846.

[25] J. Weickert, Anisotropic Diffusion in Image Processing, Teübner, Stuttgart, 1998.

[26] J. Weickert, B.M. ter Haar Romeny, M. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, IEEE Trans. Imag. Proc. 7 (1998) 398–410.