Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2011, Article ID 749456, 15 pages
doi:10.1155/2011/749456

Research Article

Enclosed Laplacian Operator of Nonlinear Anisotropic Diffusion to Preserve Singularities and Delete Isolated Points in Image Smoothing

Zhiwu Liao,[1, 2] Shaoxiang Hu,[3] Dan Sun,[2] and Wufan Chen[4]

[1] *Key Laboratory of Land Resources Evaluation and Monitoring in Southwest (Sichuan Normal University), Ministry of Education, Chengdu 610066, China*
[2] *School of Computer Science, Sichuan Normal University, Chengdu 610066, China*
[3] *School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China*
[4] *Institute of Medical Information and Technology, School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China*

Correspondence should be addressed to Wufan Chen, chenwufan@hotmail.com

Received 19 January 2011; Accepted 13 February 2011

Academic Editor: Ming Li

Existing Nonlinear Anisotropic Diffusion (NAD) methods in image smoothing cannot obtain satisfied results near singularities and isolated points because of the discretization errors. In this paper, we propose a new scheme, named Enclosed Laplacian Operator of Nonlinear Anisotropic Diffusion (ELONAD), which allows us to provide a unified framework for points in flat regions, edge points and corners, even can delete isolated points and spurs. ELONAD extends two diffusion directions of classical NAD to eight or more enclosed directions. Thus it not only performs NAD according to modules of enclosed directions which can reduce the influence of traction errors greatly, but also distinguishes isolated points and small spurs from corners which must be preserved. Smoothing results for test patterns and real images using different discretization schemes are also given to test and verify our discussions.

1. Introduction

Most of real life problems involve nonlinear systems which is a kind of complex system [1–8]. For example, traffic system can be related to the fractal, thus using fractal time series analysis, two results about traffic analysis are presented in [3, 4]. Moreover, [6, 7] also model two problems for DNA analysis and physics using fractal systems. Nonlinearity for signal and image system has been widely discussed for a long time in various literatures both in nonlinear system and image processing [9–17]. These methods include nonlinear times series analysis, fractal, chaos, and fuzzy system.

One important method to model image system is NAD proposed in early 90's [18–21]. Originally, NAD models image system as a nonlinear system, and its smoothing should be carried on in flat regions with linear average while its smoothing should be stopped near edges. Difference between a point in flat region and near edge can be measured by the module of its gradient [18].

Some efforts indicate that stoping diffusion near edges is not a good choice in image smoothing since the artifacts and isolated points should be smoothed either. Thus two new schemes are put forward: diffusing along the tangent line (coherence filter) and backward diffusing (shock filter) [19–21]. However, coherence filter and shock filter also stop at the corners or isolated points.

Recently, some efforts for improving the performance of NAD have been proposed [22–28]. However, these new methods only focus on how to relate NAD to some new tools, such as adaptive smoothing, multiresolution analysis but not to provide new diffusion schemes.

Motivated by the simplified scheme proposed in [18], we propose a new NAD scheme based on high-dimensional vector analysis which extends two diffusion directions to eight or more enclosed diffusion directions. By this way, our proposed method can reduce the influence of discretization errors greatly and provide satisfied smoothing results for points both in flat regions and near singularities.

The remainder of this paper is as follows. Section 2 is devoted to description the motivation of ELONAD. In Section 3 the new method is presented and Section 4 provides behavior analysis. In Section 5 the experimental results are given and discussed. We also give conclusions and acknowledgements finally.

2. Motivation

In image smoothing, nonlinear anisotropic diffusion (NAD), also called Perona-Malik diffusion (PMD), is a technique aiming at reducing image details without removing significant parts of the image contents, typically edges, lines, or textures which are important for the image [18].

With a constant diffusion coefficient, the anisotropic diffusion equations reduce to the heat equation which is equivalent to Gaussian blurring. This is ideal for smoothing details but also blurs edges. When the diffusion coefficient is chosen as an edge seeking function, the resulting equations encourage diffusion (hence smoothing) within regions and stop it near strong edges. Hence the edges can be preserved while smoothing from the image [18].

Formally, NAD is defined as

$$\frac{\partial u(x,y,t)}{\partial t} = \operatorname{div}\left(g(x,y,t)\nabla u(x,y,t)\right), \tag{2.1}$$

where $u(x,y,0)$ is the initial gray scale image, $u(x,y,t)$ is the smooth gray scale image at time t, ∇ denotes the gradient, $\operatorname{div}(\cdot)$ is the divergence operator, and $g(x,y,t)$ is the diffusion coefficient. $g(x, y, t)$ controls the rate of diffusion and is usually chosen as a monotonically decreasing function of the module of the image gradient. Two functions proposed in [18] are

$$g(\|\nabla u(x,y,t)\|) = e^{-(\|\nabla u(x,y,t)\|/\sigma)^2}, \tag{2.2}$$

$$g(\|\nabla u(x,y,t)\|) = \frac{1}{1 + (\|\nabla u(x,y,t)\|/\sigma)^2}, \tag{2.3}$$

where $\|\cdot\|$ is the module of the vector and the constant σ controls the sensitivity to edges.

Although there are many well-posed numerical schemes for (2.1) in the view of mathematics, image smoothing is still an ill-posed problem because of the discreteness and complexity of images. Thus some perfect mathematical schemes, such as diffusing according to tensor, diffusing along tangent line, and diffusing according to the module of gradient, cannot obtain satisfied results in image smoothing because of discretization errors. The discretization errors also make

(1) most of schemes have to stop diffusion near edges and corners which leaves spurs and noise amplification near singularities;

(2) how to select a suitable scale for diffusion becomes a very difficult tradeoff problem.

Fortunately, Perona and Malik propose a simple method to approach the modules of gradients which is called PM method in the remainder of this paper [18]. Its discretization for the Laplacian operator is

$$\begin{aligned} u(i,j,t+1) = u(i,j,t) + \frac{1}{4}\Big[&c_N \cdot \nabla_N^2 u(i,j,t) + c_S \cdot \nabla_S^2 u(i,j,t) + c_E \cdot \nabla_E^2 u(i,j,t) \\ &+ c_W \cdot \nabla_W^2 u(i,j,t)\Big], \end{aligned} \tag{2.4}$$

where

$$\begin{aligned} \nabla_N^2 u(i,j,t) &= u(i-1,j,t) - u(i,j,t), \\ \nabla_S^2 u(i,j,t) &= u(i+1,j,t) - u(i,j,t), \\ \nabla_E^2 u(i,j,t) &= u(i,j+1,t) - u(i,j,t), \\ \nabla_W^2 u(i,j,t) &= u(i,j-1,t) - u(i,j,t). \end{aligned} \tag{2.5}$$

According to (2.2) and (2.3), the diffusion coefficient is defined as a function of module of the gradient. However, computing a gradient accurately in discrete data is very complex and the module of the gradient is simplified as the absolute values of four directions and diffusion coefficients are

$$\begin{aligned}c_N(i,j,t) &= g\left(\left|\nabla_N^2 u(i,j,t)\right|\right),\\ c_S(i,j,t) &= g\left(\left|\nabla_S^2 u(i,j,t)\right|\right),\\ c_E(i,j,t) &= g\left(\left|\nabla_E^2 u(i,j,t)\right|\right),\\ c_W(i,j,t) &= g\left(\left|\nabla_W^2 u(i,j,t)\right|\right),\end{aligned} \tag{2.6}$$

where $|\cdot|$ is the absolute value of the number and $g(\cdot)$ is defined in (2.2) or (2.3).

Although this scheme is not the exact discretization of (2.1) from the view of mathematics, it is the best way for preserving singularities in existing NAD methods since it can find similar neighbors correctly without computing the gradient. Moreover, unlike other existing methods which find and diffuse corners, edges with different methods, the corners defined in PM methods are no differences from edges and they are diffused with same way-diffusion neighbors with the similar gray levels.

The main default for PM method is that it cannot discriminate spurs and isolated points from the corners and edges since only four directions is not enough to make difference to these cases. Thus one plausible scheme for (2.1) is to extend four approximation directions to much more directions. However, it is not a trivial task for this generalization, we will discuss it from theory to applications in Sections 3 and 4.

3. The Method

Just as above discussion, the PM method proposed in [18] is the best discretization scheme for NAD in existing methods for its fitness for discrete nature of images. However, the number of four discrete directions for the Laplacian operator is too less to find isolated points and small spurs near singularities, thus the four directions should be extended to more directions.

Intuitively, more directions will obtain better performance in image smoothing. In order to extend four directions to the most directions, enclosed directions for $u(i,j)$ which are defined as the differences between $u(i,j)$ and its enclosed nearest neighbors in a scale, is a good choice. For example, the differences between $u(i,j)$ and its eight nearest-neighbors are its first-scale enclosed directions while the differences between $u(i,j)$ and its 24-nearest-neighbors are its second-scale enclosed directions (see Figure 1).

Although the motivation of extending four directions to the enclosed directions is from the simplified form of (2.1), the proposed method is based on different theoretical model. That is, it is based on *the high-dimensional vector analysis for NAD*. The high-dimensional vector used in it has eight or more components, which are half-point differences between the center point $u(i,j)$ and its nearest eight or more half points. The first-scale enclosed gradient vector is defined as

$$\nabla\mathbf{u}(\mathbf{i},\mathbf{j}) = \left(\nabla_0 u(i,j), \nabla_1 u(i,j), \nabla_2 u(i,j), \nabla_3 u(i,j), \nabla_4 u(i,j), \nabla_5 u(i,j), \nabla_6 u(i,j), \nabla_7 u(i,j)\right)^T, \tag{3.1}$$

where T represents the transpose of the vector and $\nabla u_k(i,j)$, $k = 0,\dots,7$ are defined as

$$\begin{aligned}
&\nabla_0 u(i,j) = u(i,j+0.5) - u(i,j), &&\nabla_1 u(i,j) = u(i-0.5,j+0.5) - u(i,j),\\
&\nabla_2 u(i,j) = u(i-0.5,j) - u(i,j), &&\nabla_3 u(i,j) = u(i-0.5,j-0.5) - u(i,j),\\
&\nabla_4 u(i,j) = u(i,j-0.5) - u(i,j), &&\nabla_5 u(i,j) = u(i+0.5,j-0.5) - u(i,j),\\
&\nabla_6 u(i,j) = u(i+0.5,j) - u(i,j), &&\nabla_7 u(i,j) = u(i+0.5,j+0.5) - u(i,j).
\end{aligned} \tag{3.2}$$

Thus the first-scale second-order difference of $u(i,j)$ is

$$\nabla^2 \mathbf{u}(\mathbf{i},\mathbf{j}) = \left(\nabla_0^2 u(i,j), \nabla_1^2 u(i,j), \nabla_2^2 u(i,j), \nabla_3^2 u(i,j), \nabla_4^2 u(i,j), \nabla_5^2 u(i,j), \nabla_6^2 u(i,j), \nabla_7^2 u(i,j)\right)^T, \tag{3.3}$$

where T represents the transpose of the vector. From (3.1) we have

$$\begin{aligned}
&\nabla_0^2 u(i,j) = u(i,j+1) - u(i,j), &&\nabla_1^2 u(i,j) = u(i-1,j+1) - u(i,j),\\
&\nabla_2^2 u(i,j) = u(i-1,j) - u(i,j), &&\nabla_3^2 u(i,j) = u(i-1,j-1) - u(i,j),\\
&\nabla_4^2 u(i,j) = u(i,j-1) - u(i,j), &&\nabla_5^2 u(i,j) = u(i+1,j-1) - u(i,j),\\
&\nabla_6^2 u(i,j) = u(i+1,j) - u(i,j), &&\nabla_7^2 u(i,j) = u(i+1,j+1) - u(i,j).
\end{aligned} \tag{3.4}$$

It can be easily concluded that the first-scale second-order difference of $u(i,j)$ is composed by eight components which are the differences between $u(i,j)$ and its enclosed eight nearest-neighbors. The proof of (3.3) is in the appendix. Since except for the points on border of an image, each point is enclosed by its eight nearest-neighbors, so the first-scale second-order difference of $u(i,j)$ is called enclosed the first-scale second-order difference of $u(i,j)$.

Definition 3.1. The first-scale second-order-difference of $u(i,j)$ defined in (3.3) is called *the first-scale second-order enclosed difference of* $u(i,j)$.

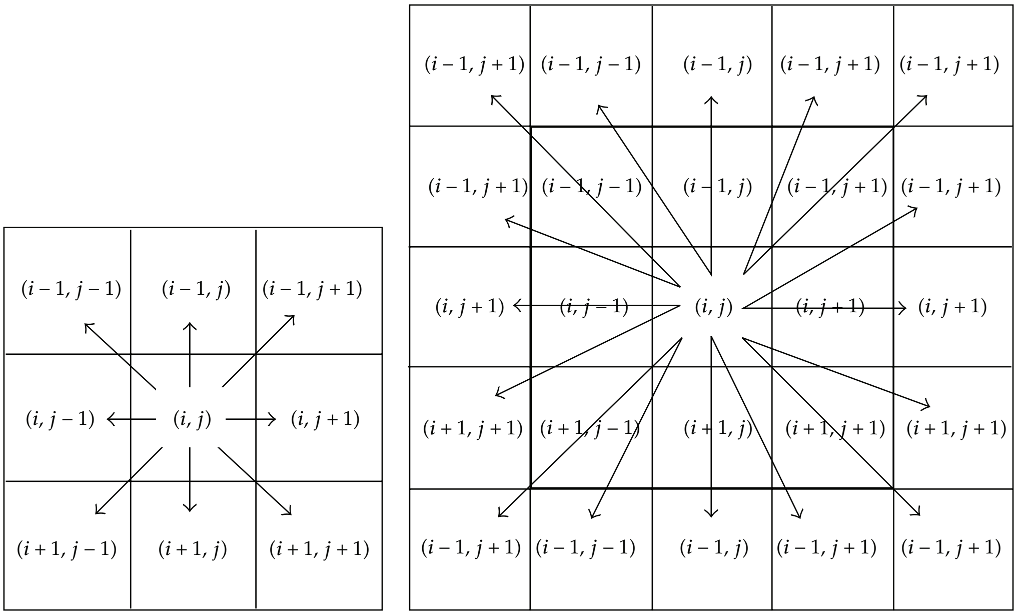
Based on Definition 3.1, we have the following.

Definition 3.2. Let the first-scale second-order-difference of $u(i,j)$ defined in (3.3). The following equation

$$\sum_{k=0}^{7} \nabla_k^2 u(i,j) \tag{3.5}$$

is called *the first-scale enclosed Laplacian operator (ELO) of* $U(i,j)$.

We must indicate that although the discussion in this paper only focuses on the first-scale ELO, it can be put in multiscale framework easily. That is, after finding enclosed points



(a) The first-scale second-order enclosed difference of $u(i,j)$

(b) The second-scale second-order enclosed difference of $u(i,j)$

Figure 1: The second-order enclosed difference of $u(i,j)$.

no less than the scale, the ELO can be computed by the difference between the $u(i,j)$ and these enclosed points.

For example, ELO can be extended to the second-scale by the differences between $u(i,j)$ and its 24 enclosed points (see Figure 1). By this way, the ELONAD can be performed on enough similar points to find noises and small spurs. In order to simplify our explanation and focus on our new theory, the ELONAD is only extended to the first-scale enclosed gradient.

Let

$$\mathbf{g} = \left(g_0, g_1, g_2, g_3, g_4, g_5, g_6, g_7\right)^T, \tag{3.6}$$

where T represents the transpose of the vector and g_k, $k = 0,\ldots,7$ is defined as

$$g_k = \frac{g\left(\left|\nabla_k u(i,j)\right|\right)}{\sum_{n=0}^{7} g\left(\left|\nabla_n u(i,j)\right|\right)}, \quad k = 0,1,\ldots,7, \tag{3.7}$$

where $\nabla_k u(i,j)$, $k = 0,\ldots 7$ defined in (3.1) are the components of vector $\nabla \mathbf{u}(\mathbf{i},\mathbf{j})$ and $\sum_{n=0}^{7} g(|\nabla_n u(i,j)|)$ is the normalized constant, g is the decreasing function of absolute value of $\nabla_k u(i,j)$, $k = 0,\ldots 7$. Following (2.2) and (2.3), $g(|\nabla u_k(x,y,t)|)$ can be defined as

$$g\left(\left|\nabla u_k(x,y,t)\right|\right) = e^{-(|\nabla u_k(x,y,t)|/\sigma)^2}, \quad k = 0,\ldots,7 \tag{3.8}$$

or

$$g\left(\left|\nabla u_k(x,y,t)\right|\right) = \frac{1}{1+\left(\left|\nabla u_k(x,y,t)\right|/\sigma\right)^2}, \quad k = 0,\ldots,7, \tag{3.9}$$

where $|\cdot|$ is the absolute value of the number, the constant σ controls the sensitivity to edges. However, half-point difference for $\nabla \mathbf{u}(\mathbf{i},\mathbf{j})$ defined in (3.1) cannot be computed directly. Thus second-order integral-point difference can be used to approximate the one-order half-point difference. Equations (3.8) and (3.9) become

$$g\left(\left|\nabla u_k(x,y,t)\right|\right) = e^{-(|\nabla u_k(x,y,t)|/\sigma)^2} \approx e^{-(|\nabla^2 u_k(x,y,t)|/\sigma)^2}, \quad k = 0,\ldots,7, \tag{3.10}$$

$$g\left(\left|\nabla u_k(x,y,t)\right|\right) = \frac{1}{1+\left(\left|\nabla u_k(x,y,t)\right|/\sigma\right)^2} \approx \frac{1}{1+\left(\left|\nabla^2 u_k(x,y,t)\right|/\sigma\right)^2}, \quad k = 0,\ldots,7. \tag{3.11}$$

The new NAD based on the enclosed second-order difference is defined as

$$\frac{\partial u(i,j,t)}{\partial t} = \operatorname{div}\begin{pmatrix} g_0\nabla_0 u(i,j,t) \\ g_1\nabla_1 u(i,j,t) \\ g_2\nabla_2 u(i,j,t) \\ g_3\nabla_3 u(i,j,t) \\ g_4\nabla_4 u(i,j,t) \\ g_5\nabla_5 u(i,j,t) \\ g_6\nabla_6 u(i,j,t) \\ g_7\nabla_7 u(i,j,t) \end{pmatrix}, \tag{3.12}$$

where the $\nabla_k u(i,j,t)$, $k = 0,\ldots,7$ are the components of vector $\nabla \mathbf{u}(\mathbf{i},\mathbf{j},\mathbf{t})$ in (3.1) and g_k, $k = 0,\ldots,7$ defined in (3.7) are the components of $\mathbf{g}$ in (3.6). Moreover, the numerator of g_k in (3.7) can be approximated by (3.10) or (3.11).

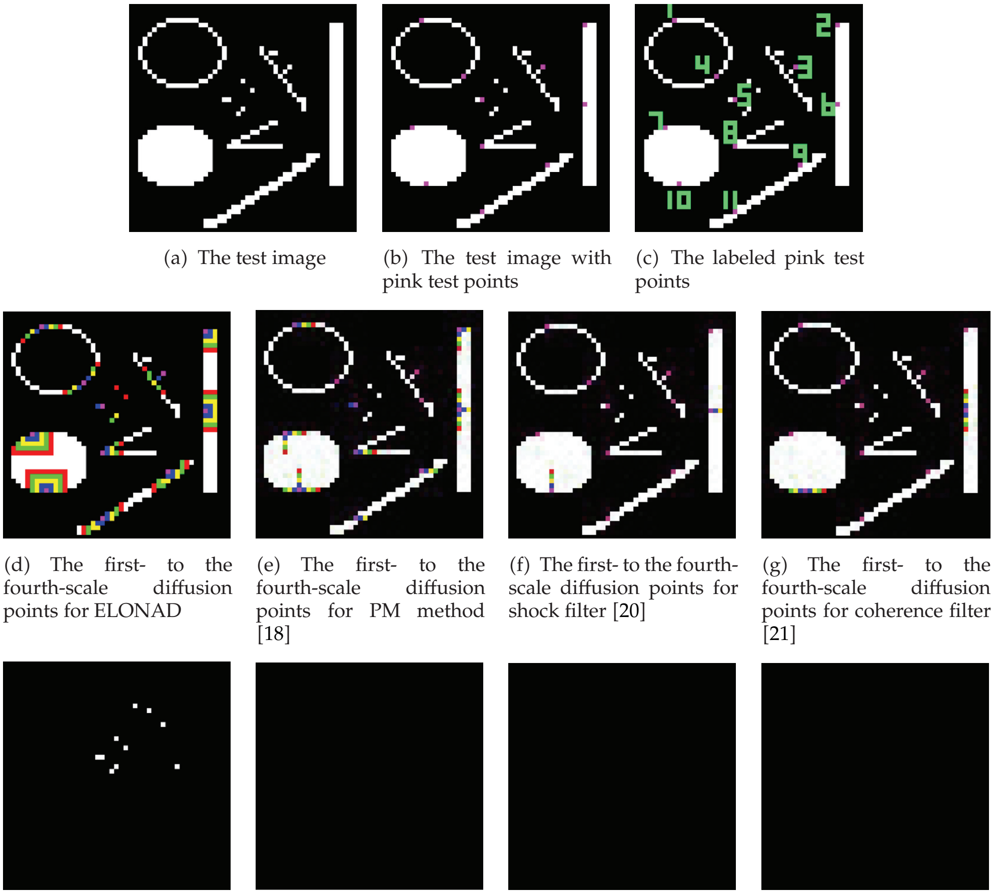
The above equation can be represented as

$$\frac{\partial u(i,j,t)}{\partial t} = \sum_{k=0}^{7} g_k \nabla_k^2 u(i,j,t), \tag{3.13}$$

where $\sum_{k=0}^{7} g_k = 1$ and the r. h. s. of (3.13) is the normalized weight ELO. $\nabla_k^2 u(i,j,t)$ is the second-order difference of the kth components of $u(i,j)$ which can be computed according to (3.3).

Thus the explicit form for solving (3.13) is

$$u(i,j,t+1) = u(i,j,t) + \sum_{k=0}^{7} g_k \nabla_k^2 u(i,j,t), \tag{3.14}$$



(a) The test image

(b) The test image with pink test points

(c) The labeled pink test points

(d) The first- to the fourth-scale diffusion points for ELONAD

(e) The first- to the fourth-scale diffusion points for PM method [18]

(f) The first- to the fourth-scale diffusion points for shock filter [20]

(g) The first- to the fourth-scale diffusion points for coherence filter [21]

(h) The difference image for ELONAD

(i) The difference image for PM method [18]

(j) The difference image for shock filter [20]

(k) The difference image for coherence filter [21]

Figure 2: The test image, diffusion points and difference images between the test image and the smoothing images for different methods.

where $u(i,j,t+1)$ is the gray level of (i,j) at time $t+1$ and g_k, $\nabla_k^2 u(i,j,t)$ are the same as in (3.13).

4. Behavior Analysis for ELONAD

Let the consideration point is $u(i,j)$, since except for isolated points and spurs, ELONAD provides a unified framework for edge points, corners and points in flat regions, the discussion of the behavior analysis focuses on both singularity points and points in flat regions. Besides the smoothing performance for these singularities, we also discuss if the proposed method can find enough close and similar points for smoothing, which is very important in image smoothing. That is, enough smoothing points can help to distinguish different singularities and delete some kinds of them, such as spurs or isolated points. Moreover, it also provides robust estimate in statistical theory.

4.1. Isolated Points and Spurs

Isolated points and spurs should be deleted since most of them are caused by noises.

Definition 4.1. In a digital image, an isolated point is a point $u(i,j)$ whose nearest-eight neighbors are different types of points to it. More specially, the absolute values of differences between $u(i,j)$ and its nearest-eight neighbors are very high.

The first-scale second-order enclosed difference of $u(i,j)$ can distinguish isolated points from other kinds of singularities by counting the number of points with similar gray levels to $u(i,j)$ in its eight nearest-neighborhood.

Theorem 4.2 (distinguishing theorem for an isolated point). *A point*

$$\begin{gathered} u(i,j) \textit{ is an isolated point} : \textit{if } u(i,j)'s \textit{ num-similar} = 0, \\ u(i,j) \textit{ is not an isolated point} : \textit{otherwise}, \end{gathered} \tag{4.1}$$

where "num-similar" is the number of similar points in $u(i,j)$'s eight nearest-neighborhood.

Proof. Since the first-scale second-order enclosed difference of $u(i,j)$ is the differences between $u(i,j)$ and its eight nearest-neighbors, if all eight nearest-neighbors are different points to $u(i,j)$, $u(i,j)$ is an isolated point. □

In addition, considering the influence of noises, the number num-similar can be relaxed to a bigger threshold.

Theorem 4.3 (relaxed distinguishing theorem for an isolated point). *A point*

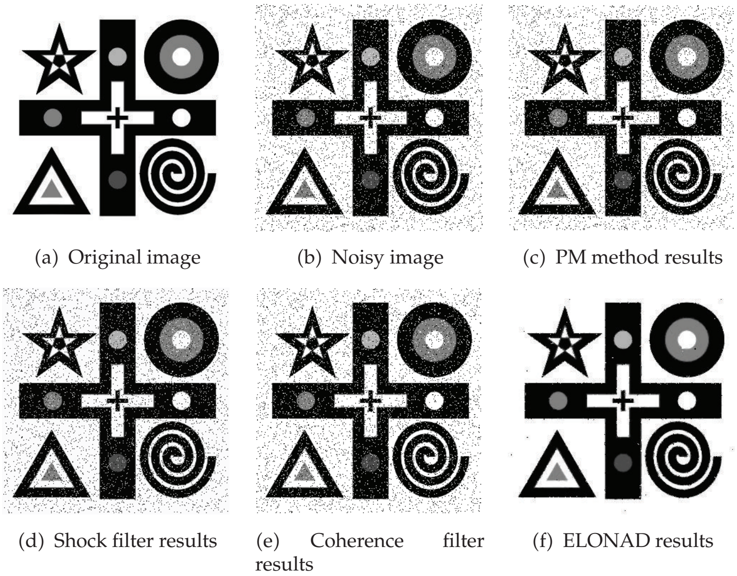
$$\begin{gathered} u(i,j) \textit{ is an isolated point} : \textit{if } u(i,j)'s \textit{ num-similar} \le T_s, \\ u(i,j) \textit{ is not an isolated point} : \textit{otherwise}, \end{gathered} \tag{4.2}$$

where $0 \le T_s \le 8$ is a predefined threshold and "num-similar" is the number of points with similar gray levels to $u(i,j)$ in its eight nearest-neighbors.

For example, in Figure 2(c), two white points at top right corner of the label 5 are two isolated points. If relaxing T_s to 1, points with labels 5 and 3 are isolated points either. The isolated points can also be handled with a multiscale manner. For example, we can define an isolated point is that its num-similar in the first scale is 1 and its num-similar in the second-scale is 2. This multiscale scheme has better performance in edge preserving than the one-scale scheme.

Definition 4.4. A spur is a very thinning and short unwanted branch in an image.

Most of spurs are caused by noises. One example for spurs in Figure 2(c) is the small branch near the point labeled by 3 and the point with label 3 is the top of the spur where is the end point of the spur. The top of a spur can be found by one-scale or multiscale second-order enclosed difference of $u(i,j)$ using Theorem 4.3 or its multiscale version, the most of spur points can be deleted by recursively smoothing using ELONAD (see Figure 2(h)).



(a) Original image (b) Noisy image (c) PM method results

(d) Shock filter results (e) Coherence filter results (f) ELONAD results

Figure 3: Smoothing results for different schemes.

4.2. Corners, Edges, and Points in Flat Regions

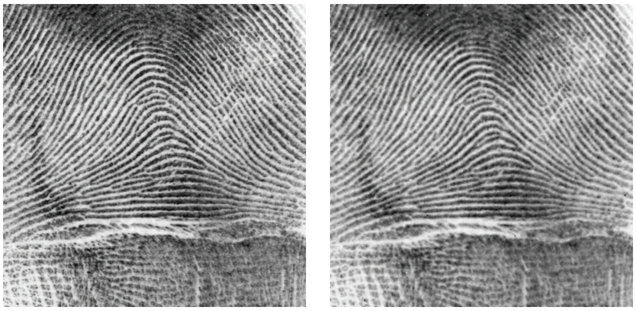
Except for isolated points and spurs which should be deleted, ELONAD puts corners, edges and points in flat regions, which must be preserved, into a unified framework. According to Theorem 4.3, if $u(i,j)$'s "num-similar" $> T_s$, it is neither an isolated point nor a top of a spur. In fact, it is a corner point or an edge point or a point in a flat region. Thus we have the following.

Theorem 4.5 (distinguishing theorem for different points). *A point*

$$\begin{aligned} &u(i,j) \textit{ is an isolated point or the top of a spur} : \textit{if } u(i,j)'s \textit{ num-similar} \le T_s, \\ &u(i,j) \textit{ is a corner or an edge point or a smooth point} : \textit{otherwise}, \end{aligned} \tag{4.3}$$

where $0 \le T_s \le 8$ *is a predefined threshold and "num-similar" is the number of points with similar gray levels to* $u(i,j)$ *in its eight nearest-neighborhood.*

Following Theorem 4.5, the points in a digital image can be parted into two groups: one is isolated points and spur points which should be deleted while the other is corners, points in flat regions and edges which should be preserved. Thus these two types of points can be performed separately. Unlike the existing methods which regard isolated points and the tops of spurs as corners, and then stop diffusion at the corners and diffusion edges along tangent lines, the behavior of ELONAD for the latter-corners, edges and smooth points is the same: diffusion with the neighboring points having similar gray levels to $u(i,j)$'s.



(a) Original image (b) ELONAP results

Figure 4: Smoothing results for ELONAP.

The main advantage for this behavior is the unnecessary for determining the type of $u(i, j)$ before diffusion after deleting the isolated points and spurs. Thus all three types of points can be unified into a diffusion framework to reduce influences of the discretization errors.

In order to show above discussion more clearly, an image is designed composed by: two white circles-one is one-pixel width white circle and the other is a circle filled in white points; one white vertical line with three-pixel width; two declining white lines-one is three-pixel width and the other is one-pixel width; a corner composed by a horizontal white line and a declining white line; some isolated points and two spurs near the one-width declining line (see Figure 2(a)). 11 test points are selected and labeled by the pink points in Figure 2(b) while their labels are shown on Figure 2(c).

Two points labeled by 3 and 5 should be deleted where point 3 is the top of the spur and 5 is an isolated point. The method to find the isolated points and the tops of the spurs is based on Theorem 4.3 while deleted isolated points and the tops of spurs using ELONAD are show in Figure 2(h) with white points.

However, since isolated points and the tops of spurs also have large modules both for the gradients and the tangent lines, existing methods cannot distinguish isolated points and the tops of spurs from corners. Therefore, the diffusion will be stopped at these three types of points (see Figures 2(e)–2(g)). That is, the isolated points and the tops of spurs are preserved at existing methods (see Figures 2(f)–2(g) and 2(i)–2(k)).

The 11 test points, except for points 3 and 5, are smoothed using the same diffusion scheme-diffusing among similar points in eight or more nearest neighborhood. The diffusion points for the first- to the fourth-scale are shown using different colors: blue, yellow, green and red for the first-scale to the fourth-scale, respectively, (see Figures 2(d)–2(g)). We can observe that these 9 test points are surrounded with different color diffusion points which shows us the diffusion will be carried on multiscale similar points (see Figure 2(d)).

However, for methods proposed in [20, 21], only two test points labeled by 6 and 10 are surrounded by different color points while other 7 points are isolated pink points indicating the diffusion is stopped at these 7 points (see Figures 2(f)–2(g)). The reason for this situation is only points labeled as 6 and 10 are considered as edge points for the shock filter and coherence filter while other points are considered as corners and the diffusion has to be stopped at there.

The method proposed in [18] is can provide better performance in finding diffusion points comparing to the methods proposed in [20, 21] (see Figure 2(e)). However, its small number for diffusion points compared to ELONAD indicates that four directions for discretization is not enough to obtain satisfied smoothing results.

5. Experiments and Discussion

In order to show the performance of ELONAD, two images are used: one is a 320 × 320 gray level image with "salt and pepper" noise whose density is 0.1; the other is a real 512 × 512 fingerprint image with some artifacts.

The main objective for smoothing these two images is to delete the noise in the first image and deleting the artifacts in the fingerprint image while to preserve main patterns for the images.

In order to compare ELONAD with some state-of-art discretization methods, three schemes proposed in [18, 20, 21] are used. These three methods are called PM method in [18], shock filter in [20] and coherence filter in [21].

Except for PM method, the main behavior for shock filter and coherence filter is diffusion along edges and stop at corners. However, since corners cannot be distinguished from the isolated points and the tops of spurs, the shock filter and coherence filter have to stop at these three kinds of points. Thus both of them cannot delete the "salt and pepper" noise (see Figures 3(b) and 3(c)).

Although PM method has very similar smoothing results to the shock filter, their behavior has two differences: one is the PM method will not stop diffusion at corner while shock filter will stop at corners; the other is PM only stops at the isolated points while shock filter will stop at corners, isolated points and the tops of the spurs (see Figure 3(a)).

Since proposed method can distinguish the isolated points from the corners, the isolated points can be deleted easily (see Figure 3(d)).

Moreover, in order to show the smoothing performance of ELONAD, it also is carried on a real fingerprint image. Just as above discussion, PM, shock filter and coherence filter have very similar manner to the original image. Thus in Figure 4, we only present the original fingerprint image and smoothing image by ELONAD.

Observing fingerprint image, we can find irregular gray levels are smoothed and some artifacts are also smoothed out while important patterns are preserved. It demonstrates ELONAD can obtain satisfied smoothing results both in deleting isolated points and spurs, and in preserving important patterns.

6. Conclusions

In this paper, we propose a new discretization scheme for NAD based on high-dimensional vector analysis. Unlike existing NAD schemes only have two directions in their discretization whose traction errors make they cannot distinguish isolated points and spurs from corners, our method provides the maximum directions by enclosed Laplacian operator (ELO) to reduce the discretization errors and distinguish different types of points. Moreover, ELO parts image points into two groups: one is isolated points and the top of spurs which should be deleted; the other is corners, edges and smooth points which should be preserved.

Since these two groups related to two different behavior, respectively, the algorithm can be designed according to different groups. Thus it can obtain satisfied smoothing results in digital images.

Appendix

Proof of the First-Scale Second-Order Difference of $u(i,j)$

The second-order difference of $u(i,j)$ is

$$\nabla^2\mathbf{u}(\mathbf{i},\mathbf{j}) = \left(\nabla_0^2 u(i,j), \nabla_1^2 u(i,j), \nabla_2^2 u(i,j), \nabla_3^2 u(i,j), \nabla_4^2 u(i,j), \nabla_5^2 u(i,j), \nabla_6^2 u(i,j), \nabla_7^2 u(i,j)\right)^T, \tag{A.1}$$

where T represents the transpose of the vector $\nabla^2\mathbf{u}(\mathbf{i},\mathbf{j})$. From (3.1) we have

$$\begin{aligned}
\nabla_0^2 u(i,j) &= \nabla_0[u(i,j+0.5) - u(i,j)] \\
&= u(i,j+1) - u(i,j+0.5) + u(i,j+0.5) - u(i,j) \\
&= u(i,j+1) - u(i,j), \\
\nabla_1^2 u(i,j) &= \nabla_1[u(i-0.5,j+0.5) - u(i,j)] \\
&= u(i-1,j+1) - u(i-0.5,j+0.5) + u(i-0.5,j+0.5) - u(i,j) \\
&= u(i-1,j+1) - u(i,j), \\
\nabla_2^2 u(i,j) &= \nabla_2[u(i-0.5,j) - u(i,j)] \\
&= u(i-1,j) - u(i-0.5,j) + u(i-0.5,j) - u(i,j) \\
&= u(i-1,j) - u(i,j), \\
\nabla_3^2 u(i,j) &= \nabla_3[u(i-0.5,j-0.5) - u(i,j)] \\
&= u(i-1,j-1) - u(i-0.5,j-0.5) + u(i-0.5,j-0.5) - u(i,j) \\
&= u(i-1,j-1) - u(i,j), \\
\nabla_4^2 u(i,j) &= \nabla_4[u(i,j-0.5) - u(i,j)] \\
&= u(i,j-1) - u(i,j-0.5) + u(i,j-0.5) - u(i,j) \\
&= u(i,j-1) - u(i,j), \\
\nabla_5^2 u(i,j) &= \nabla_5[u(i+0.5,j-0.5) - u(i,j)] \\
&= u(i+1,j-1) - u(i+0.5,j-0.5) + u(i+0.5,j-0.5) - u(i,j) \\
&= u(i+1,j-1) - u(i,j),
\end{aligned}$$

$$\begin{aligned}
\nabla_6^2 u(i,j) &= \nabla_6\left[u(i+0.5,j)-u(i,j)\right] \\
&= u(i+1,j)-u(i+0.5,j)+u(i+0.5,j)-u(i,j) \\
&= u(i+1,j)-u(i,j), \\
\nabla_7^2 u(i,j) &= \nabla_7\left[u(i+0.5,j+0.5)-u(i,j)\right] \\
&= u(i+1,j+1)-u(i+0.5,j+0.5)+u(i+0.5,j+0.5)-u(i,j) \\
&= u(i+1,j+1)-u(i,j).
\end{aligned} \tag{A.2}$$

Acknowledgments

This paper is supported by the National Natural Science Foundation of China (no. 60873102), Major State Basic Research Development Program (no. 2010CB732501), and Open Foundation of Visual Computing and Virtual Reality Key Laboratory Of Sichuan Province (no. J2010N03). This work was supported by a grant from the National High Technology Research and Development Program of China (no. 2009AA12Z140) and Open foundation of Key Laboratory of Land Resources Evaluation and Monitoring of SouthwestSichuan Normal University, Ministry of Education.

References

[1] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, New York, NY, USA, 2001.

[2] M. Li, “Fractal time series—a tutorial review,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 157264, 26 pages, 2010.

[3] M. Li, “Modeling autocorrelation functions of long-range dependent teletraffic series based on optimal approximation in Hilbert space-A further study,” *Applied Mathematical Modelling*, vol. 31, no. 3, pp. 625–631, 2007.

[4] M. Li and W. Zhao, “Representation of a stochastic traffic bound,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1368–1372, 2010.

[5] K. S. Miller and B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations*, A Wiley-Interscience Publication, John Wiley & Sons, New York, NY, USA, 1993.

[6] C. Cattani, “Fractals and hidden symmetries in DNA,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 507056, 31 pages, 2010.

[7] E. G. Bakhoum and C. Toma, “Dynamical aspects of macroscopic and quantum transitions due to coherence function and time series events,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 428903, 2010.

[8] J. L. Casti, *Nonlinear System Theory*, vol. 175 of *Mathematics in Science and Engineering*, Academic Press, Orlando, Fla, USA, 1985.

[9] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*, John Wiley & Sons, New York, NY, USA, 1980.

[10] C. L. Nikias and A. P. Petropuop, *Higher-Order Spectra Analysis: A Non-linear Signal Processing Framework*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[11] M. Li, M. Scalia, and C. Toma, “Nonlinear time series: computations and applications,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 101523, 5 pages, 2010.

[12] Z. Liao, S. Hu, and W. Chen, “Determining neighborhoods of image pixels automatically for adaptive image denoising using nonlinear time series analysis,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 914564, 14 pages, 2010.

[13] S. Y. Chen, H. Tong, Z. Wang, S. Liu, M. Li, and B. Zhang, “Improved generalized belief propagation for vision processing,” *Mathematical Problems in Engineering*, vol. 2011, Article ID 416963, 12 pages, 2011.

[14] J. Zhang and Z. Wei, “A class of fractional-order multi-scale variational models and alternating projection algorithm for image denoising,” *Applied Mathematical Modelling*, vol. 35, no. 5, pp. 2516–2528, 2011.

[15] S. Y. Chen and Y. F. Li, “Determination of stripe edge blurring for depth sensing,” *IEEE Sensors Journal*, vol. 11, no. 2, pp. 389–390, 2011.

[16] S. Y. Chen, Y. F. Li, and J. Zhang, “Vision processing for realtime 3-D data acquisition based on coded structured light,” *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 167–176, 2008.

[17] C. Chaux, L. Duval, A. Benazza-Benyahia, and J.-C. Pesquet, “A nonlinear Stein-based estimator for multichannel image denoising,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, part 2, pp. 3855–3870, 2008.

[18] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[19] L. Alvarez, P.-L. Lions, and J.-M. Morel, “Image selective smoothing and edge detection by nonlinear diffusion. II,” *SIAM Journal on Numerical Analysis*, vol. 29, no. 3, pp. 845–866, 1992.

[20] L. Alvarez and L. Mazorra, “Signal and image restoration using shock filters and anisotropic diffusion,” *SIAM Journal on Numerical Analysis*, vol. 31, no. 2, pp. 590–605, 1994.

[21] J. Weickert, “Coherence-enhancing diffusion filtering,” *International Journal of Computer Vision*, vol. 31, no. 2, pp. 111–127, 1999.

[22] N. Sochen, G. Gilboa, and Y. Y. Zeevi, “Color image enhancement by a forward and backwardadaptive Beltrami flow,” in *Proceedings of the International Workshop on Algebraic Frames for the Perception-Action Cycle (AFPAC ’00)*, G. Sommer and Y. Y. Zeevi, Eds., vol. 1888 of *Lecture Notes in Computer Science*, pp. 319–328, Springer, 2000.

[23] F. Zhang, Y. M. Yoo, L. M. Koh, and Y. Kim, “Nonlinear diffusion in laplacian pyramid domain for ultrasonic speckle reduction,” *IEEE Transactions on Medical Imaging*, vol. 26, no. 2, pp. 200–211, 2007.

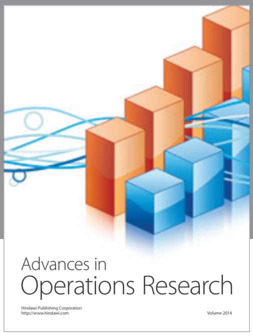
[24] M. Ceccarelli, V. De Simone, and A. Murli, “Well-posed anisotropic diffusion for image denoising,” *IEE Proceedings: Vision, Image and Signal Processing*, vol. 149, no. 4, pp. 244–252, 2002.

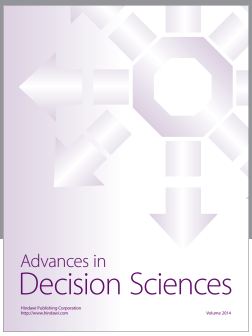
[25] V. B. Surya Prasath and A. Singh, “Well-posed inhomogeneous nonlinear diffusion scheme for digital image denoising,” *Journal of Applied Mathematics*, vol. 2010, Article ID 763847, 14 pages, 2010.

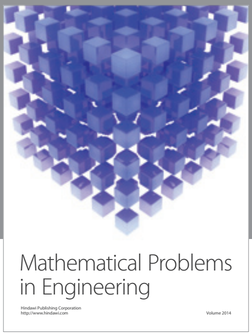
[26] J. Ling and A. C. Bovik, “Smoothing low-SNR molecular images via anisotropic median-diffusion,” *IEEE Transactions on Medical Imaging*, vol. 21, no. 4, pp. 377–384, 2002.

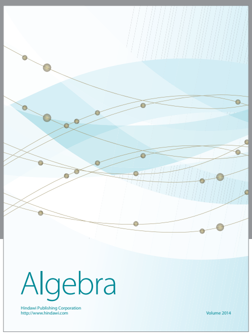
[27] J. Yu, Y. Yang, and A. Campo, “Approximate solution of the nonlinear heat conduction equation in a semi-infinite domain,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 421657, 24 pages, 2010.

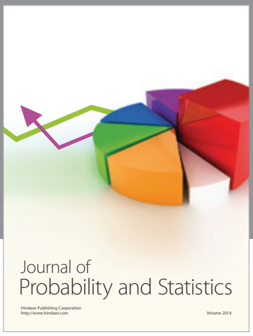
[28] G. Gilboa, N. Sochen, and Y. Y. Zeevi, “Complex diffusion processes for image filtering,” in *Proceedings of the 3rd International Conference on Scale-Space and Morphology in Computer Vision*, vol. 2106 of *Lecture Notes in Computer Science*, pp. 299–307, Springer, 2001.


Advances in
Operations Research

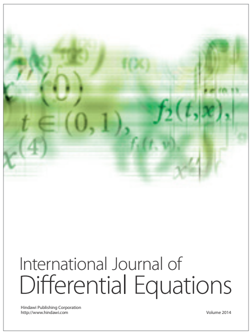

Advances in
Decision Sciences

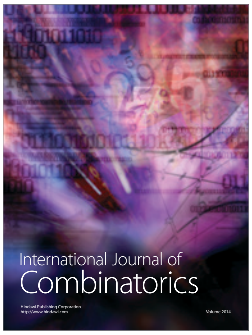

Mathematical Problems
in Engineering

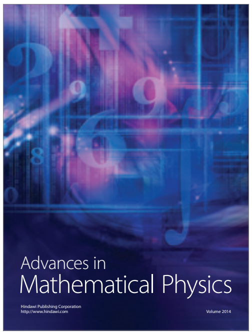

Algebra

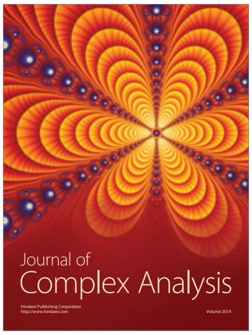

Journal of
Probability and Statistics

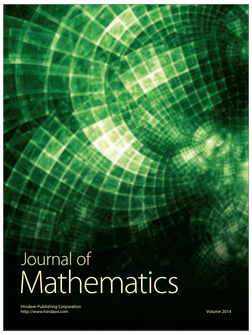
The Scientific
World Journal

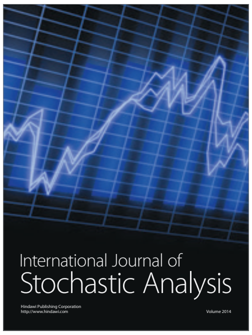

International Journal of
Differential Equations

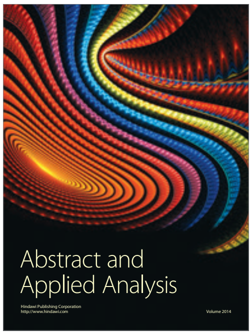

International Journal of
Combinatorics

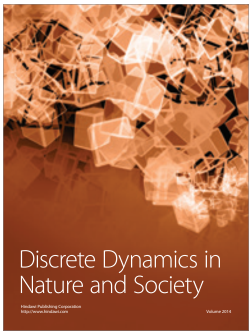

Advances in
Mathematical Physics

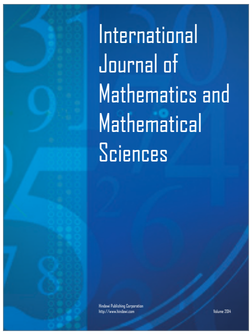

Journal of
Complex Analysis


Journal of
Mathematics


International Journal of
Stochastic Analysis

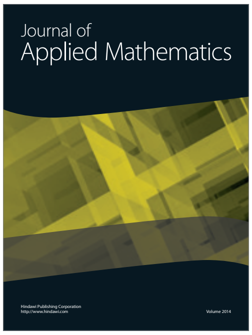

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


Journal of
Applied Mathematics

Journal of
Optimization