

## Research Article

# Determining Neighborhoods of Image Pixels Automatically for Adaptive Image Denoising Using Nonlinear Time Series Analysis

Zhiwu Liao,<sup>1</sup> Shaoxiang Hu,<sup>2</sup> and Wufan Chen<sup>3</sup>

<sup>1</sup> Key Laboratory of Land Resources Evaluation and Monitoring of Southwest, Sichuan Normal University, Ministry of Education, Chengdu 610068, Sichuan, China

<sup>2</sup> School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 610054, Sichuan, China

<sup>3</sup> Institute of Medical Information and Technology, School of Biomedical Engineering, Southern Medical University, Guangzhou 510515, China

Correspondence should be addressed to Zhiwu Liao, [liaoziwu@163.com](mailto:liaoziwu@163.com)

Received 30 January 2010; Accepted 20 March 2010

Academic Editor: Ming Li

Copyright © 2010 Zhiwu Liao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a method determining neighborhoods of the image pixels automatically in adaptive denoising. The neighborhood is named stationary neighborhood (SN). In this method, the noisy image is considered as an observation of a nonlinear time series (NTS). Image denoising must recover the true state of the NTS from the observation. At first, the false neighbors (FNs) in a neighborhood for each pixel are removed according to the context. After moving the FNs, we obtain an SN, where the NTS is stationary and the real state can be estimated using the theory of stationary time series (STS). Since each SN of an image pixel consists of elements with similar context and nearby locations, the method proposed in this paper can not only adaptively find neighbors and determine size of the SN according to the characteristics of a pixel, but also be able to denoise while effectively preserving edges. Finally, in order to show the superiority of this algorithm, we compare this method with the existing universal denoising algorithms.

## 1. Introduction

Image denoising is a very important image preprocessing step. In acquisition, images would be more or less affected by noise. Noise will make the image quality reduction, which will influence the subsequent processing steps. In order to recover a real hidden image from a noisy image, a lot of efforts have been done for a long time.

In 1949, Wiener proposed Wiener filtering using the theory of stationary random process [1]. In theory, Wiener's filter meets the minimum mean-square errors (MMSEs) of

a linear filter. However, Wiener filtering is only applicable to stationary time series, which causes the edges to be blurred in denoising.

The most effective way to address these problems is adaptive denoising [2–5], which assumes that the image gray levels are piecewise constant or piecewise continuous. However, near the singular points, such as edges and textures, the assumption of being piecewise continuous and constant does not hold. It also makes the edges and textures oversmoothing.

An improved form of adaptive denoising is called the bilateral filtering [6–11]. Bilateral filtering integrates range filtering (gray level) and domain filtering (space) together, which preserves edges while denoising. However, in noises, the real gray levels are polluted seriously, which makes the real levels unable to be correctly estimated from the noisy gray levels. In addition, two window parameters, the variances of range filter's kernel and spatial filter's kernel, must be selected by experience. Once it is fixed, it cannot be changed.

Some researchers suggested that the context able to be used for distinguishing the singular points from smooth points [12, 13]. Essentially, the context defined on the local gray level energy is a classifier for image pixels. This makes smooth can be done only among the similar points, which can maintain the singularity in denoising. However, due to context defined in the whole image, it lacks spatial adaptation. Studies in [14–17] propose different methods to improve it.

Some algorithms combine space and context together [14–16]. In these methods, a fixed-size of sliding windows is chosen by experience firstly. Then the true value of each pixel is estimated from the points in the window with the similar context. These methods have better performance than the context. The challenge of these methods is that fixed size sliding windows will make the window too small to obtain reliable estimate for singular points.

Another well-known method is nonlocal denoising algorithm proposed recently [17]. Nonlocal approach determines the similarities through a big and a small window together. The small window is used to determine the nature of local gray energy while the large window is used to look for similarities. As the searching neighborhood is large, nonlocal approach can overcome the default for most of spatial methods in unreliable estimates near singular points, which can more effectively maintain the borders and textures.

We think that nonlocal is the same as context in finding the similar points using local gray level energy. However, the context defined throughout the image lacks spatial adaptation, while nonlocal searches for similarities in a large window with better spatial adaptability. However, on smooth regions, nonlocal also lacks spatial adaption. Besides this, the nonlocal method is with high computational complexity and the sizes of two windows are also fixed and chosen fully by experience.

As can be seen from the above discussion, the neighborhood sizes of existing adaptive image denoising algorithms are selected by experience. Moreover, these neighborhoods can no longer be changed after being selected, which makes the edge-preserving image denoising a very difficult tradeoff problem. That is, denoising needs a large neighborhood to eliminate noises while maintaining edges requires a small neighborhood to keep singularity. A fixed-size neighborhood is impossible to satisfy these two requirements simultaneously. Nonlocal can overcome the shortcomings of the appeal by taking advantage of big and small windows. But its mechanism for the coexistence of two windows makes the computing complexity increase greatly. Besides this, it lacks of spatial adaptation to smooth regions. In this paper, we propose a method to adaptively determine the neighborhood of image pixels in denoising using the theory of NTS analysis.

Recently, with the development of the theory of time series analysis, NTS analysis becomes the focus in time series analysis [18–31]. Two well-known NTSs include fractal time series and chaotic time series. However, the field of image denoising, to date, almost do not use these NTSs. The reason for this phenomenon is that most of researchers believe that the image noise is random rather than chaos [29, 30].

In this paper, we are concerned about how to convert an NTS to a set of STS. The method firstly removes the false neighbors dynamically using context to obtain a SN. In a SN, since all of the pixels have similar local gray energy, the time series can be considered as stationary. Besides this, the SN is composed with close spatial locations, which guarantees its spatial adaptability.

The motivation for SN is that the observation (the noisy image) of a NTS is the sampled data of an underlying high-dimensional manifold. Some projection points of these sampling points, which are not neighbors in the manifold, become neighbors in one-dimensional projection space. These neighbors are called FNs. To obtain SNs of an image pixel, firstly, we must remove these FNs out. The removing increases the embedding dimension gradually, which makes folding, wrapping, and twisting orbit open. Using this method, FNs can be found and removed [18, 24]. The original neighborhood without FNs becomes a SN. Thus, the real state of a NTS can be estimated from the noisy observation on SNs using the theory of STS.

Note that the different image pixels have different SNs and sometimes SNs are irregular, for example, near the edges. In addition, the proposed method also maintains a nonlinearity for NTS, which is coincident with the same nature for manifolds. That is, local structures are simple while its global structure is very complex [32–34].

The neighborhoods determined by proposed method are fully automatic and reliable in estimate, and they are also able to maintain the image edges with the variable sizes and irregular shapes. It finds a perfect solution to the existing challenges in adaptive image denoising.

Section 2 in this article will discuss the NTS and SN, and Section 3 describes the denoising algorithm presented in this paper. Section 4 presents the experimental results and discussion. Section 5 gives conclusions, and finally the acknowledgment part is given.

## **2. Nonlinear Time Series and Stationary Neighborhood**

In this section, we mainly introduce how to find SNs for the pixels of a noisy image. Thus, the NTS will be converted into the STS in SNs.

### **2.1. Definitions**

From the geometric point of view, an NTS is a projection from a high-dimensional phase space to a one, dimensional space. In this projection, some points, which are not adjacent in the phase space, become neighbors in one-dimensional projection space and are called FNs. To remove FNs, the most direct idea is to increase the embedding dimension for the phase space. As the embedding dimension is increasing, the folding, wrapping, and twisting orbit will gradually be open. Therefore, FNs can easily be removed from the original neighborhood. And then a SN, which is a neighborhood without FNs, is obtained. On this SN, the true state can be restored according to the theory of STS. In order to explain our approach better, the definitions of related terms are given as follows.

*Definition 2.1.* A vector of phase space for a time series is an  $m$ -dimension vector in the phase space, which is composed by a different time delay  $\{0, \tau, \dots, (m-1)\tau\}$  of the time series  $\{z(t)\}$ :

$$\{z(t), z(t + \tau), z(t + 2\tau), \dots, z(t + (m-1)\tau)\}, \quad (2.1)$$

where  $m$  is called *embedding dimension* for the phase space.

The method deleting FNs is from the smallest embedding dimension, such as by  $m = 1$ , and is gradually increasing embedding dimension  $m$ . As  $m$  is increasing, the wrapping and folding orbit of the nonlinear movement will be gradually opened up. When  $m$  increases to a definite value, which the number of FNs no further increases, the correct embedding dimension is found.

*Definition 2.2.* A *manifold* is a topological space that is locally Euclidean (i.e., around every point, there is a neighborhood that is topologically the same as the open unit ball in  $R^n$ ).

Manifold resembles the Euclidean space near each point, and its global structure may be very complicated. The nature of manifold, which is local simple and complex global, coincides with our method. That is, the global complex nonlinear can be parted into local STS.

*Definition 2.3.* The *Neighborhood* of a pixel  $x$  is a collection  $N$  of pixels. The elements of this collection satisfy  $N = \{\xi \mid \|x - \xi\| < d, \xi \neq x\}$ , where  $\|\cdot\|$  is the distance and  $d$  is a predefined constant. The elements in  $N$  are called *neighbors* of  $x$ .

*Definition 2.4.* Two points  $x$  and  $\xi$  on the phase space of NTS are not neighbors but they are neighbors on the one-dimensional orbit, and  $\xi$  is called a *false neighbor* (FN) for  $x$ .

Note that, in our method, the size of the SN is determined by the number of FNs.

*Definition 2.5.* A time series  $\{z(t)\}$  is *stationary* if, for all  $m$ , the joint probability distribution of  $z(t), z(t+1), \dots, z(t+m-1)$  is independent on the time index  $t$ . More specially, the expectation, variance, correlation coefficients of a time series only are functions of time interval and are independent on the origin of the time; hence, the time series is then called *weakly stationary*.

*Definition 2.6.* One neighborhood determined by the method presented in this paper is called a *stationary neighborhood*.

**Theorem 2.7.** *Pixels in a SN form an STS.*

## 2.2. Some Remarks

In this subsection, we will give some remarks on our method.

*Remark 2.8* (The Selection of Time Delay  $\tau$ ). In this paper, the time delay  $\tau$  is set to 1. The reason is that most of adjacent points are very similar in image and this assumption usually is adopted in adaptive denoising. Here, we also follow this assumption.

*Remark 2.9* (Context and Embedding Dimension). Firstly, the definition of context is given.

*Definition 2.10.* The *context* for an image pixel  $x_{ij}$  is defined as a length  $P$  vector  $V_{ij} = [V_{i,j,1}, V_{i,j,2}, \dots, V_{i,j,P}]$  formed as a multidimensional function of observations.

The context commonly used in image processing can be defined as the  $m$ -dimensional vector in phase space of NTS. Since the context in image coding is studied deeply, it can be used directly to construct the SNs. For example, if we follow a specific definition of context, the embedding dimension  $m$  can be determined immediately. From the above discussion, we know the following.

**Theorem 2.11.** *An  $m$ -dimensional vector in phase space is a special form of context.*

*Remark 2.12* (Size of Neighborhood). It should be explained that the neighborhood size and embedding dimension are two different concepts. Generally, neighborhood is a more global concept than the embedding dimension.

It should satisfy two basic criteria simultaneously in choosing neighborhood size. That is, it must be big enough to satisfy the reliability of estimates and be small enough to satisfy spatial adaptation of the singularity detection. As discussed in the previous section, a fixed-size neighborhood cannot satisfy the above two requirements simultaneously.

The method proposed in this paper meets these two requirements simultaneously by building different neighborhoods for different image points. In order to ensure the reliability of estimates, the least number of pixels should be given firstly. Here, we select 48. In other words, a neighborhood after deleting the FNs still has 48 pixels in it; it is a right SN.

The reason why we should give the least number of neighbors is neighborhood on a smooth region is different for that near the singular points. In a smooth region, a  $7 \times 7$  neighborhood is enough, while near a singular point there are few neighbors in  $7 \times 7$  neighborhood, which cannot meet the reliable requirement. Thus, near singular points, it must increase the size of neighborhood in order to increase the number of neighbors. This is the reason why nonlocal method has two different windows.

### 2.3. Determining Stationary Neighborhood

Firstly, we must determine three parameters: time delay  $\tau$ , embedding dimension  $m$ , and size of neighborhood  $d$ . We know that  $\tau = 1$ ,  $m$  could be determined by the context, and neighborhood size  $d$  is determined by 48 and the least number of neighbors automatically. In this way, we can find a SN for a pixel in accordance with the following steps.

*Step 1* (initialization). Give context and set  $\tau = 1$ , the least number of neighbors  $n = 48$ , the threshold  $d_V$  of context  $V_n$ , and initial size of neighborhood  $d = d_n \times d_n = 7 \times 7$ .

*Step 2* (finding FNs in a neighborhood of a pixel). The FN is a pixel  $(i', j')$  in the neighborhood  $d = d_n \times d_n$  and satisfies  $\|V_{ij} - V_{i'j'}\| > d_V$ . Find all FNs in the neighborhood. Then record the index and the number  $f_n$  of FNs.

*Step 3.* If  $d - f_n \leq 48$ , then  $d_n = d_n + 2$ , and repeat Steps 2-3; otherwise, deleting the FNs of the neighborhood, the SN is the remainder of the neighborhood.

### 3. The New Framework

In this section, we will discuss the theory of image denoising [35]. And then the framework will be presented.

#### 3.1. Image Denoising

Image denoising studies on how to recover an original image  $x$  from a noisy observation  $y$ , assuming that the noise is  $n$ .  $X, Y, N$  are  $m \times n$  matrixes, which represent the random fields with  $m \times n$  variables. The relation of their observations can be represented by the following formula:

$$y = x + n, \quad (3.1)$$

where  $y, x$ , and  $n$  are realizations of random fields  $Y, X$ , and  $N$ , respectively.

The  $X$  can be estimated under Minimum Mean Squared Error (MMSE); that is,

$$\hat{x} = \arg \min_X \|\hat{x} - x\|^2, \quad (3.2)$$

where  $\hat{x}$  represents the estimate value of  $X$ . An uppercase letter represents a random field or a random variable while the lowercase letter represents one realization of the random field or variable.

Thus, the estimate of one pixel  $x_{ij}$  of  $X$  is conditioned on the observation  $y$  optimized by MMSE. If  $N$  is a 0 mean and  $\sigma_N^2$  variance Gaussian white noise (GWN), the optimal estimate of  $x_{ij}$  is

$$\hat{x}_{ij} = \frac{\sigma_X^2}{\sigma_X^2 + \sigma_N^2} EY_{ij}, \quad (3.3)$$

where  $\sigma_X^2$  is variance of the original image  $X$  and  $EY_{ij}$  is the mean of  $Y_{ij}$ .

Here, two parameters should be estimated in denoising:  $\sigma_X^2$  and  $EY_{ij}$ . However, we only have one observation for  $Y$ . Thus we have to share data in a neighborhood, which assumes that the whole data in this neighborhood are independent identical distribution (iid).

#### 3.2. The New Framework

In this subsection, we will give the new framework of our method.

*Step 1.* For each image pixel  $(i, j)$ , determine a SN (see Section 2.3).

*Step 2.* Compute  $\sigma_Y^2$  and  $EY_{ij}$  in the SN using the assumption of iid. And  $\sigma_X^2$  is  $\sigma_Y^2 - \sigma_N^2$ .

*Step 3.* Estimate  $x_{ij}$  using (3.3).

*Step 4.* For all image pixels, repeat Steps 1–3.



Figure 1: Noisy Lena (a) and denoised Lena using Wiener's filter with  $7 \times 7$  mask (b).

#### 4. Experiments and Discussion

In this section, we will compare our method to Wiener's filter, bilateral filter, context, and nonlocal. It should note that these five filters are universal denoising filters. In order to compare them on the same benchmark, the same context is used in our method, nonlocal, and context. The programs are implemented on Matlab with the same designer.

Firstly, we will give some brief comments on these five filters. And then some experimental results will be shown. Finally, we will give discussion.

*Wiener's filter* is proposed in 1949 by Wiener [1]. It uses the natures of STS and the frequency properties to filter noise from the signal. The experiments of Wiener are carried on the function "wiener2" in Matlab. It blurs the edges and texture while denoising. One example is shown in Figure 1. The denoised image using Wiener's filter with  $7 \times 7$  mask is blurred seriously, especially for the mouth of Lena and the decoration on Lena's hat.

*Bilateral Filter.* the formula for bilateral filter is

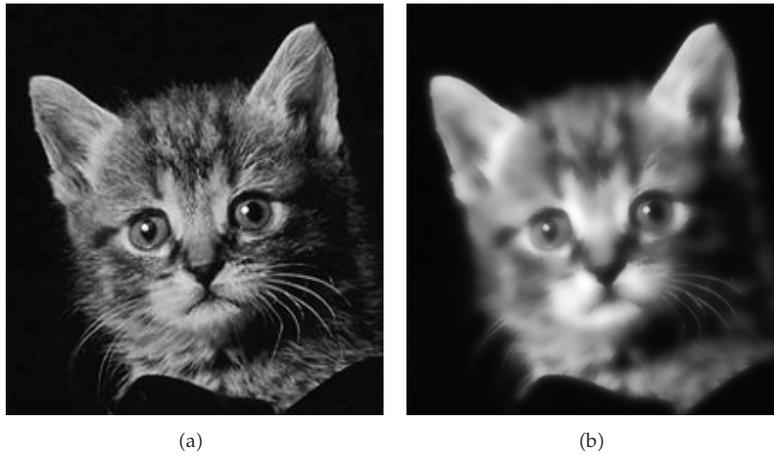
$$h(x) = k^{-1}(x) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} y_{\xi} c(\xi, x) s(y_{\xi}, y_x) d\xi, \quad (4.1)$$

where  $\xi$  and  $x$  are two pixels.  $y_{\xi}$  and  $y_x$  are gray levels of  $\xi$  and  $x$ , respectively.  $k^{-1}(x)$  is a normalized constant for two weighs and is defined as

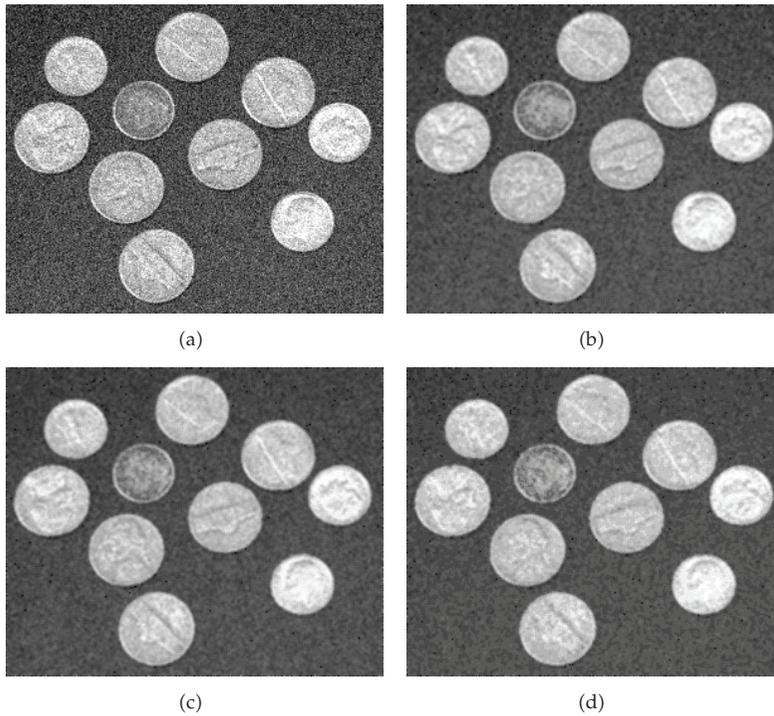
$$k(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} c(\xi, x) s(y_{\xi}, y_x) d\xi, \quad (4.2)$$

where  $c(\xi, x)$  and  $s(y_{\xi}, y_x)$  are measures of the spatial and range closeness between the center pixel  $x$  and its neighbor  $\xi$ , respectively. Usually, these two measures can be defined as two Gaussian Kernel functions:

$$\begin{aligned} c(\xi, x) &= e^{-(1/2)(\|\xi-x\|/\sigma_d)^2}, \\ s(y_{\xi}, y_x) &= e^{-(1/2)(\|y_{\xi}-y_x\|/\sigma_r)^2}. \end{aligned} \quad (4.3)$$



**Figure 2:** An image before (a) and after (b) using bilateral filter.



**Figure 3:** A noisy image (a) and denoised images using context when  $d_V$  (see Section 2.3) is 5 (b), 10 (c), and 20 (d).

Bilateral filter integrates domain filter and range filter together. It also defines a space neighborhood using the variance of domain filter  $\sigma_d$ . The range filter is used for selecting the points with similar gray levels to  $x$ . However, in denoising, the real gray level is hidden in the noisy data. Therefore, the range filter cannot work well. Besides this, two neighborhood sizes of bilateral filter also are fixed after defining the two variances  $\sigma_d$  and  $\sigma_r$ . The program



**Figure 4:** A noisy image (a) and denoised images using nonlocal when  $d_V$  (see Section 2.3) is 10 (b), 20 (c), and 30 (d).

of bilateral filter is designed on Matlab. Figure 2 gives one image and the image after bilateral filtering.

*Context.* For an image, pixel  $X_{i,j}$  is defined as a length  $P$  vector  $V_{i,j} = [V_{i,j,1}, V_{i,j,2}, \dots, V_{i,j,P}]$  formed as a function of observations. In order to ensure that the comparison is on the same benchmark, the context in our method, nonlocal, and context is defined as

$$v_{i,j} = \frac{\sum_{k=i-1}^{k=i+1} \sum_{t=j-1}^{t=j+1} y_{k,t}}{9}, \quad (4.4)$$

where  $y_{k,t}$  is the gray level of pixel  $(k, t)$ .

Using the context defined by (4.4), the image pixels can be classified to several groups according to their local energy. In this paper, we use a parameter  $d_V$  (see Section 2.3) to control the difference for each of group. Figure 3 gives denoising results for different  $d_V$ 's for context. Although context has better denoising results than these Wiener's filter and bilateral filter, it also lacks spatial adaptivity. We also design context denoising on Matlab.



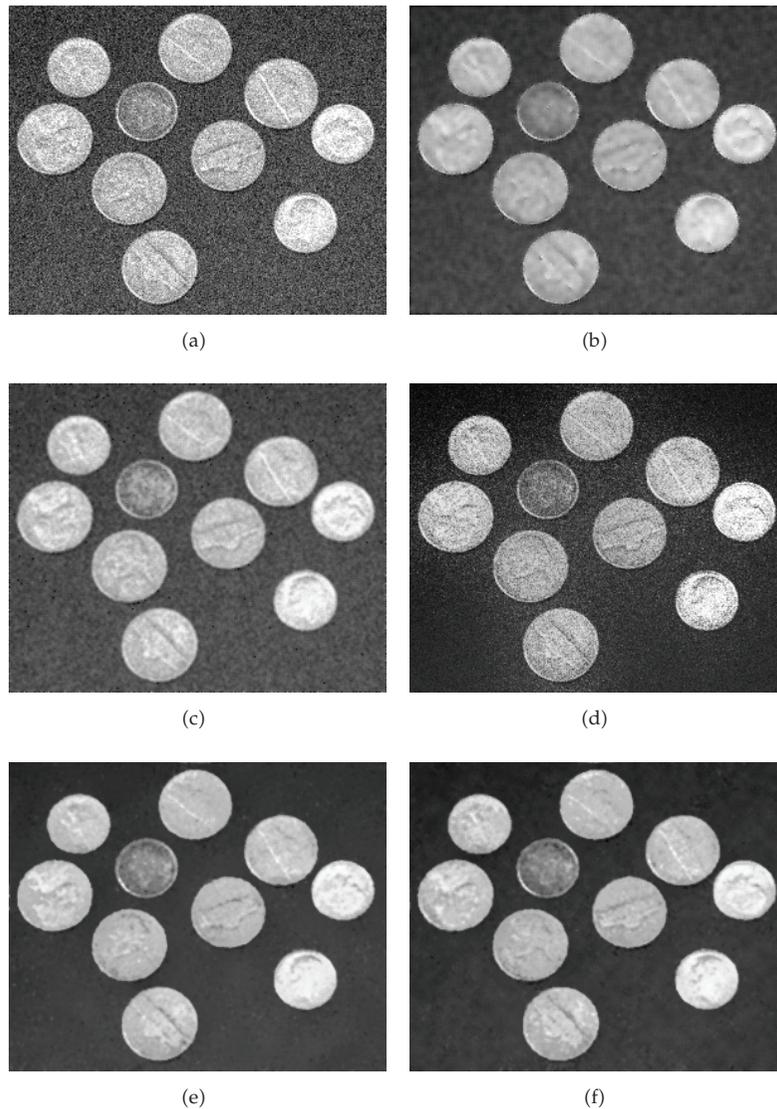
**Figure 5:** A noisy image (a) and denoised images using SN  $d_V$  (see Section 2.3) is 10 (b), 20 (c), and 30 (d).

Note that, designing more complex context or using tools like wavelet, FFT and so forth will undoubtedly obtain better denoising results. However, it is beyond the scope of paper.

*Nonlocal* is a famous good denoising algorithm. The most important mechanism for nonlocal is using two windows simultaneously. At least, it improves the estimate near singular points. However, on smooth region, since it lacks spatial adaptation, it leads over smooth on these regions.

The Nonlocal program is designed on Matlab, in which the size of small window is  $3 \times 3$  (context) while the size of large window is  $21 \times 21$ . The points on the large window with similar context are used for estimating the real gray level of the center point. In Figure 4 denoised images using different  $d_V$ 's (see Section 2.3) are shown. It is obvious that Figure 4(d) is oversmooth on smooth regions but still has good performance in edge preserving.

*Stationary Neighborhood* is finding a stationary neighborhood for each image pixel. The neighborhood has at least 48 pixels after deleting FNs. Since On smooth regions and near singular points, if two requirements of designing a neighborhood are satisfied, it has good performance on both type regions. That is, in theory analysis, the method proposed in this paper has the same performance near singular points while having better performance on smooth regions than the nonlocal. Figure 5 gives us denoised results of SN.



**Figure 6:** A noisy image (a) and denoised images using Wiener's filter (b), context (c), bilateral filter (d), nonlocal (e), and SN (f).

In order to compare the performance of above five filters, we test some images in Matlab and some images in the image databases on the internet. These images include *lena.jpg* ( $256 \times 256$ ) and *coins.png* (in Matlab). For coins, nonlocal and SN overmatch other three methods both on denoising and edge preserving; see Figure 6. Nonlocal also has very similar Visual Effects to SN! It shocks me much since I think that SN should have obviously better performance than nonlocal.

After analysis, I think the reason is that the image (coins) is too simple to find the difference between nonlocal and SN. The most important difference between these two methods should be the different neighborhoods on the smooth regions. Thus, *Lena*, a famous denoising test image with big smooth regions, becomes a test image for comparing nonlocal

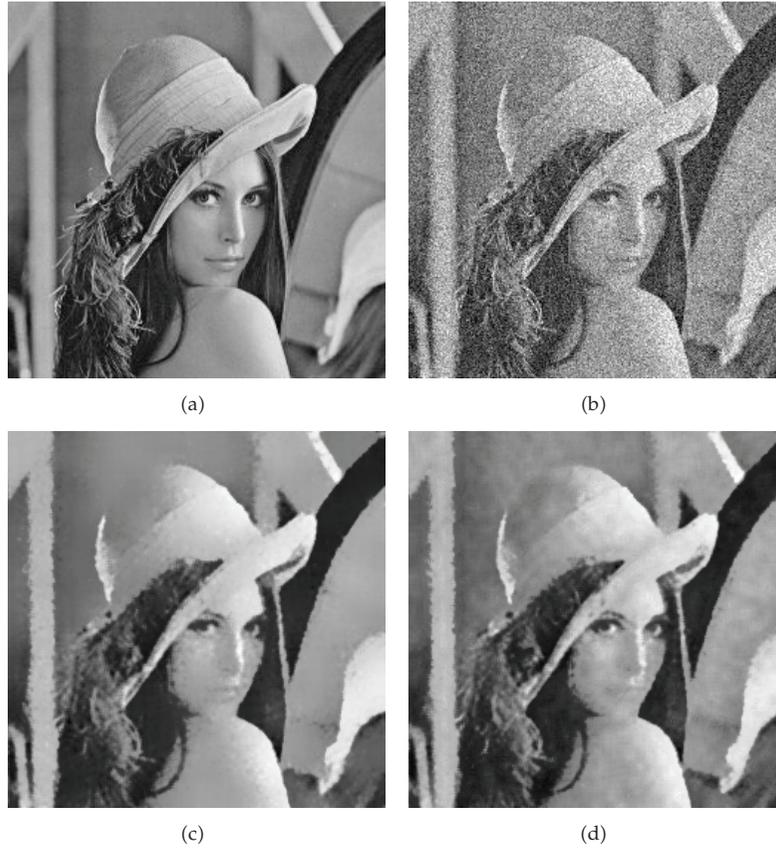


Figure 7: Lena (a), noisy Lena (b), and denoised images of nonlocal (c) and SN (d).

and SN. In theory, SN has the same performance of nonlocal near singular points while having better performance on smooth regions.

From Figure 7, we can see that SN has better performance than nonlocal on smooth regions. That is, SN preserves much more gray levels and details in smooth regions, especially for upper borderline of hat where SN preserves the borderline but nonlocal loses it! In addition, SN also provides us more good visual effects.

Besides these, SN also has relatively low computation complexity. The computation of nonlocal is  $9 \times 441 \times N^2$  [17]. Since most of image pixels (about 90%) are smooth pixels [13], SN reduces the computation complexity greatly. That is, the smooth points only need a  $7 \times 7$  neighborhood for denoising. Thus the computation complexity is about  $9 \times 49 \times N^2 \times 0.9 + 9 \times 441 \times N^2 \times 0.1$ , where 441 is an estimate mean for singular regions according to nonlocal. Its computation complexity is about 2% of nonlocal.

## 5. Conclusions

In this paper, we propose a new method to determine a neighborhood, named SN, for each image pixel in adaptive image denoising. The motivation for finding SN is based on the idea that an NTS can be convert to STS in some overlapped neighborhoods. An SN is a

neighborhood whose false neighbors are deleted and has at least 48 neighbors. SN satisfies two requirements for designing neighborhood on both smooth regions and singularity regions. It also has good performance on two type regions with about 2% computation complexity of nonlocal.

## Acknowledgments

This paper is supported by the National Natural Science Foundation of China (nos. 60873102, 60573125, 60973157, and 60873264), National Key Basic Research Program Project of China (no. 2010CB732501), and Open Foundation of Key Laboratory of Land Resources Evaluation and Monitoring of Southwest, Ministry of Education, (no. KLEM2009001).

## References

- [1] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, John Wiley & Sons, New York, NY, USA, 1949.
- [2] L. Şendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 438–441, 2002.
- [3] M. Mignotte, "Image denoising by averaging of piecewise constant simulations of image partitions," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 523–533, 2007.
- [4] Z. Dengwen and C. Wengang, "Image denoising with an optimal threshold and neighbouring window," *Pattern Recognition Letters*, vol. 29, no. 11, pp. 1694–1697, 2008.
- [5] D. Coupier, A. Desolneux, and B. Ycart, "Image denoising by statistical area thresholding," *Journal of Mathematical Imaging and Vision*, vol. 22, no. 2, pp. 183–197, 2005.
- [6] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 839–846, Bombay, India, 1998.
- [7] M. Zhang and B. K. Gunturk, "Multiresolution bilateral filtering for image denoising," *IEEE Transactions on Image Processing*, vol. 17, no. 12, pp. 2324–2333, 2008.
- [8] H. Yu, L. Zhao, and H. Wang, "Image denoising using trivariate shrinkage filter in the wavelet domain and joint bilateral filter in the spatial domain," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2364–2369, 2009.
- [9] Y.-L. Liu, J. Wang, X. Chen, Y.-W. Guo, and Q.-S. Peng, "A robust and fast non-local means algorithm for image denoising," *Journal of Computer Science and Technology*, vol. 23, no. 2, pp. 270–279, 2008.
- [10] Z. Ji, Q. Chen, Q.-S. Sun, and D.-S. Xia, "A moment-based nonlocal-means algorithm for image denoising," *Information Processing Letters*, vol. 109, no. 23–24, pp. 1238–1244, 2009.
- [11] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola, "From local kernel to nonlocal multiple-model image denoising," *International Journal of Computer Vision*, vol. 86, no. 1, pp. 1–32, 2010.
- [12] X. Wu and N. Memon, "Context-based, adaptive, lossless image code," *IEEE Transactions on Communications*, vol. 45, no. 4, pp. 437–444, 1997.
- [13] M. S. Crouse and R. G. Baraniuk, "Contextual hidden Markov models for wavelet-domain signal processing," in *Proceedings of the 31st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, Calif, USA, November 1997.
- [14] G. Fan and X.-G. Xia, "Image denoising using a local contextual hidden Markov model in the wavelet domain," *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 125–128, 2001.
- [15] S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Transactions on Image Processing*, vol. 9, no. 9, pp. 1522–1531, 2000.
- [16] Z. Liao and Y. Y. Tang, "Signal denoising using wavelets and block hidden Markov model," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 5, pp. 681–700, 2005.
- [17] A. Buades, B. Coll, and J.-M. Morel, "Nonlocal image and movie denoising," *International Journal of Computer Vision*, vol. 76, no. 2, pp. 123–139, 2008, special section: selection of papers for CVPR 2005, Guest Editors: C. Schmid, S. Soatto and C. Tomasi.

- [18] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*, Cambridge University Press, Cambridge, UK, 2nd edition, 2004.
- [19] M. Li, W.-S. Chen, and L. Han, "Correlation matching method for the weak stationarity test of LRD traffic," *Telecommunication Systems*, vol. 43, no. 3-4, pp. 181–195, 2010.
- [20] M. Li and J.-Y. Li, "On the predictability of long-range dependent series," *Mathematical Problems in Engineering*, vol. 2010, Article ID 397454, 9 pages, 2010.
- [21] M. Li and W. Zhao, "Representation of a stochastic traffic bound," *IEEE Transactions on Parallel and Distributed Systems*. In press.
- [22] C. Cattani and A. Kudreyko, "Application of periodized harmonic wavelets towards solution of eigenvalue problems for integral equations," *Mathematical Problems in Engineering*, vol. 2010, Article ID 570136, 8 pages, 2010.
- [23] E. G. Bakhoun and C. Toma, "Mathematical transform of traveling-wave equations and phase aspects of quantum interaction," *Mathematical Problems in Engineering*, vol. 2010, Article ID 695208, 15 pages, 2010.
- [24] C. Keng and H. Bo-Tang, "A survey of state space reconstruction of chaotic time series analysis," *Computer Science*, vol. 32, no. 4, pp. 67–70, 2005.
- [25] S. Lin, J. Qiao, G. Wang, S. Zhang, and L. Zhi, "Phase space reconstruction of nonlinear time series based on Kernel method," in *Proceedings of the 6th World Congress on Intelligent Control and Automation (WCICA '06)*, vol. 1, pp. 4364–4368, Dalian, China, 2006.
- [26] M. Casdagli, S. Eubank, J. D. Farmer, and J. Gibson, "State space reconstruction in the presence of noise," *Physica D*, vol. 51, no. 1–3, pp. 52–98, 1991.
- [27] L. Wang, H. Zhang, H. Meng, and X. Wang, "Nonlinear analysis of individual vehicle behavior in car following," in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC '08)*, pp. 265–268, Beijing, China, October 2008.
- [28] J. B. Dingwell and J. P. Cusumano, "Nonlinear time series analysis of normal and pathological human walking," *Chaos*, vol. 10, no. 4, pp. 848–863, 2000.
- [29] B. Dennis, R. A. Desharnais, J. M. Cushing, S. M. Henson, and R. F. Costantino, "Can noise induce chaos?" *Oikos*, vol. 102, no. 2, pp. 329–339, 2003.
- [30] R. Poole, "Is it chaos, or is it just noise," *Science*, vol. 243, no. 4887, pp. 25–28, 1989.
- [31] M. Li, "Fractal time series—a tutorial review," *Mathematical Problems in Engineering*, vol. 2010, Article ID 157264, 26 pages, 2010.
- [32] <http://www.hudong.com/wiki/>.
- [33] V. Guillemin and A. Pollack, *Differential Topology*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1974.
- [34] R. C. Kirby and L. C. Siebenmann, *Foundational Essays on Topological Manifolds, Smoothings, and Triangulations*, Princeton University Press, Princeton, NJ, USA, 1977.
- [35] Z. Liao, *Image Denoising Based on Wavelet Domain Hidden Markov Models*, UESTC Press, Chengdu, China, 2006.