Hindawi
Applied Computational Intelligence and Soft Computing
Volume 2017, Article ID 5835020, 11 pages
https://doi.org/10.1155/2017/5835020

Research Article

Color Image Denoising Based on Guided Filter and Adaptive Wavelet Threshold

Xin Sun,[1] Ning He,[1] Yu-Qing Zhang,[2] Xue-Yan Zhen,[2] Ke Lu,[3] and Xiu-Ling Zhou[4]

[1]*Smart City College, Beijing Union University, Beijing 100101, China*
[2]*Beijing Key Laboratory of Information Service Engineering, Beijing Union University, Beijing 100101, China*
[3]*University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100049, China*
[4]*Beijing City University, Beijing 100083, China*

Correspondence should be addressed to Ning He; xxthening@buu.edu.cn

Received 29 May 2017; Revised 20 August 2017; Accepted 21 August 2017; Published 6 November 2017

Academic Editor: Ridha Ejbali

In the process of denoising color images, it is very important to enhance the edge and texture information of the images. Image quality can usually be improved by eliminating noise and enhancing contrast. Based on the adaptive wavelet threshold shrinkage algorithm and considering structural characteristics on the basis of color image denoising, this paper describes a method that further enhances the edge and texture details of the image using guided filtering. The use of guided filtering allows edge details that cannot be discriminated in grayscale images to be preserved. The noisy image is decomposed into low-frequency and high-frequency subbands using discrete wavelets, and the contraction function of threshold shrinkage is selected according to the energy in the vicinity of the wavelet coefficients. Finally, the edge and texture information of the denoised color image are enhanced by guided filtering. When the guiding image is the original noiseless image itself, the guided filter can be used as a smoothing operator for preserving edges, resulting in a better effect than bilateral filtering. The proposed method is compared with the adaptive wavelet threshold shrinkage denoising algorithm and the bilateral filtering algorithm. Experimental results show that the proposed method achieves superior color image denoising compared to these conventional techniques.

1. Introduction

During their acquisition and transmission, images are adversely affected by noise. Color images contain better visual effects than gray image in terms of visual perception, and the edge information of color images is more abundant than in gray images. Ideally, when removing the additive noise from an image, as many of the important features as possible should be retained. The denoising of color images often results in the loss of some edge and texture information, making the image blurred and creating a poor visual effect.

Denoising methods of color image commonly, Wiener filter and Gaussian filter denoising, have edge blurred situation. Bilateral filtering [1] is the most intuitive nonlinear smoothing filter, although it suffers from the gradient inversion effect, which uses a histogram-based approximation to calculate the weight, and it is computationally complex. Recently, Zhang et al. [2] develop an improved bilateral filter based framework which is capable of effectively removing universal noise. Bilateral filter takes spatial information and grayscale similarity into account and achieves both denoising and edge-preserving. Bilateral filter preserves too much high-frequency information but, however, does not denoise the high-frequency noise in color images. Thus, bilateral filter has better denoising effect for low-frequency noise merely. Sometimes, bilateral filter suffers from gradient reverse. The reason is that when an edge pixel has few similar pixels around it, the Gaussian weighted average is unstable. In this paper, proposed method based on local linear model has good edge-preserving smoothing properties like bilateral filter, but it does not suffer from the gradient reverse.

Wavelet threshold denoising [3–8] is a simple and effective denoising method. This technique effectively involves the decomposition of a signal into a set of independent, spatially oriented frequency channels. The discrete wavelet threshold [9] can be used to decompose the original image

into a sequence of images of different spatial resolutions. Dong and Ding [10] proposed a method that can always achieve better performance with lower computation cost and fewer decomposition scales than a high-frequency denoising method. Elyasi and Zermchi [11] proposed several adaptive wavelet denoising methods: Bayes Shrink, Modified Bayes Shrink, and Normal Shrink.

In recent years, many algorithms have improved and studied the wavelet threshold denoising. Like, Bhandari et al. [12] developed an optimized adaptive thresholding function which selects the appropriate threshold values to separate noise from the actual image and preserve edge details. In this paper, a new adaptive wavelet shrink denoising algorithm is proposed. Compared with other threshold algorithms, the proposed approach improves the denoising performance and has lower complexity than existing adjacent pixels methods [13, 14].

In the process of color image denoising, we compared the proposed method with the algorithms mentioned above (bilateral filter, Wiener filter, Gaussian filter, and wavelet threshold methods). Proposed method achieves superior color image denoising to these conventional algorithms. The reason is that classic algorithms could suppress the Gaussian noise effectively, but, at the same time, these methods fail to maintain the quality of denoised color images (like, texture) and may blur edges in the image. To address these short comings, this paper proposes a method based on image structure using adaptive wavelet threshold and guided filter to maintain edges when denoising. It makes edges continuous and the color of image more brightly. Because guided filter using a local linear model to enhance the image, the edge details remain. In particular, the details in color image, like texture, are more abundant and saturation is more greater.

On this basis, this paper presents a new color image denoising method based on the adaptive wavelet threshold shrinkage algorithm combined with image structure-based guided filtering [15]. The method uses the discrete wavelet transform to calculate the energy near the wavelet coefficients and then uses the adaptive threshold shrinkage function to denoise the image. The threshold function depends on the energy of adjacent pixels. Further using guided filter enhances the image after denoising. Experiments show that the proposed technique enables better preservation of edge information during the denoising process.

The rest of the paper is organized as follows. Section 2 reviews the related work. Algorithm analysis and the structure of proposed method are presented in Section 3. Then experimental results and analysis are shown in Section 4. Section 5 concludes the paper.

2. Related Work

Numerous works have been proposed for image denoising. In this part, we review previous and related work about wavelet threshold algorithms and guided filter.

2.1. Wavelet Threshold Shrinkage Algorithm. Wavelet threshold denoising is done by Donoho in 1994, which is based on thresholding the discrete wavelet transform (DWT) of the signal. Hard threshold and soft threshold are traditional threshold algorithm. Donoho [5] proposed wavelet soft threshold denoising and the threshold VisuShrink algorithm. This method for image denoising obtains a series of wavelet coefficients from the wavelet transform and applies a threshold to determine the smaller coefficients (which correspond to noise). Denoising and an inverse wavelet transformation yield the reconstructed image with reduced noise. Hard and soft threshold functions are defined as follows.

The hard threshold function is expressed in

$$Y = \begin{cases} X, & |X| \geq \lambda \\ 0, & |X| < \lambda. \end{cases} \tag{1}$$

The soft threshold function on the other hand is expressed in

$$Y = \begin{cases} \operatorname{sgn}(X)(|X| - \lambda), & |X| \geq \lambda \\ 0, & |X| < \lambda, \end{cases} \tag{2}$$

where λ is a threshold value, X is wavelet coefficients value after the DWT of images, and Y is output value using wavelet threshold shrinkage function.

Normally, hard threshold function can preserve the wavelet coefficients well generated by the useful information from images, but it is discontinuous at $|X| = \lambda$ after reconstruction. An alternative approach to hard threshold is the soft threshold, which has advantages of continuity. Soft threshold function is smooth and continuous relatively at the threshold. But sometimes, there are defects that decreased the wavelet coefficients generated by the effective signal.

An appropriate threshold λ is the most important role of discrete wavelet denoising. In the process of denoising, if the threshold λ is too small, the wavelet coefficients contain too many noise components and cannot denoise effectively. Otherwise, the threshold λ is particularly large resulting in the loss of useful components that causes distortion. Thus, new methods are proposed and some of them have been delivered to real applications.

Adaptive wavelet threshold method first assigns zeroes when the wavelet coefficients are smaller than the given threshold. As the threshold increases, the number of coefficients below the threshold will increase rapidly. When the number of nonzero coefficients reaches a certain value, the threshold is further enlarged and the number of nonzero values slowly decreases; this method can remove most of the noise and improve the compression efficiency. Nasri and Nezamabadi-pour [16] proposed a new thresholding function to be further used in a new subband-adaptive thresholding neural network to improve the efficiency of the denoising procedure. Liu et al. [17] found that a wavelet denoising using neighbor coefficients and level dependency was proposed to separate spikes from background noise.

2.2. Guided Filter. The guided filter is based on a dual integral image architecture VLSI [18] (Very Large Scale Integration). The filtering method computes the output image based on the input guiding image, which preserves the edges of the image well; this is an accurate and fast edge-preserving filtering

algorithm. He et al. [15] proposed a new explicit local linear guided filtering model, which is a fast and nonapproximate linear time algorithm. Their model senses the edges and enhances the texture detail. Gadge and Agrawal [19] used guided filtering to color images.

Guided filter of image is a linear transformable filtering process, where the guidance image I needs to be preset according to the specific application and I could be identical to the input image p. The output value at pixel i is calculated as follows:

$$q_i = \sum_j W_{ij}(I)\, p_j, \tag{3}$$

where i and j are pixel indexes. W_{ij} is filter kernel function, which defined in [15] expressed by

$$W_{ij}(I) = \frac{1}{|\omega|^2} \sum_{k:(i,j)\in\omega_k} \left(1 + \frac{(I_i - \mu_k)(I_j - \mu_k)}{\sigma_k^2 + \varepsilon}\right), \tag{4}$$

where ω_k is the local window that center of k, $|\omega|$ is count of pixels in the local window, μ_k and σ_k^2 are mean and variance of guidance image I in the window, and, finally, ε is a smoothing factor.

A local linear model (3) is used in guided filter. This linear relationship ensures that output q_i has the same edge information with guidance image I. Thus, guided filter has a better edge-preserving smoothing and avoids the gradient reverse. In addition, its algorithm computed efficiently and nonapproximately.

3. Proposed Algorithm

In this paper, we describe an adaptive wavelet transform method to remove noise from a color image and use the inverse discrete wavelet transform to obtain the denoised image. The guided filter is then applied for edge and texture recovery and enhancement, producing a better color image effect.

3.1. Overview of Method Structure. The framework of proposed method contains two main stages (Figure 1). The first step is to obtain preliminary denoised image p using adaptive wavelet threshold shrinkage algorithm, which is based on image structure feature to shrinkage wavelet coefficients. The wavelet coefficients are decomposed by two-level discrete wavelet transform (DWT). The second step is further denoising and enhancing by using guided filter to the previous result p. In guided filter, the guidance image I should be preset. Setting I and p is identical and can perverse edge and texture of image.

In the ideal case, the wavelet threshold shrinkage algorithm subtracts Gaussian noise from the image, and its denoising effect is obvious. Natural image denoising using the wavelet threshold is very effective because it can capture the energy of the converted images.

Proposed denoising algorithm has the following steps in detail:

(1) Transform the noisy image into the frequency domain using DWT.

(2) Apply the adaptive wavelet threshold shrinkage algorithm to the local window on each subband and then use inverse DWT to obtain preliminary denoising image p.

(3) Apply guided filter on image p to obtain further denoise image q.

(4) Enhance q and output image.

3.2. Adaptive Wavelet Threshold Algorithm. The discrete wavelet transform (DWT) applied to image processing has two main components: decomposition and reconstruction. We use DWT to decompose the noisy image into a sequence of images of different spatial resolutions. Two-dimensional images can be decomposed in two-degree directions, resulting in different frequency bands: LL (Low-Frequency), LH (Horizontal High-Frequency), HL (Vertical High-Frequency), and HH (Diagonal High-Frequency).

In Figure 2, using DWT to decompose the noisy image into a sequence of images of different frequency bands. Low-Frequency (LL) is decomposed using DWT. And decomposing it produces four different frequency subbands (LL2, HL2, LH2, and HH2) using two-level wavelet decomposition function. On the basis of these frequency subbands, the different wavelet threshold shrinkage algorithm can be used to denoise from the image.

After the two-level wavelet decomposition of the image (Figure 2), an adaptive wavelet transform is used to extract the structure information in the multiresolution image, and the corresponding shrinkage function is applied to the structure features of the image. In fact, the natural image structure of the wavelet coefficients in the resolution scale exhibits a certain similarity, so there is a certain degree of redundancy in the wavelet decomposition scale. For example, the wavelet coefficients of the edge region are usually concentrated together, indicating that there is a certain degree of dependency in the adjacent wavelet coefficients corresponding to the edge region.

The structure information of the image can be obtained by calculating the energy of the local area in the wavelet domain. The smoother the image, the lower the energy. A threshold range is determined based on the local energy calculated by the wavelet decomposition, and a different function is used within the corresponding threshold. The specific algorithm takes the average of the square of each pixel value in the local window to calculate the energy of the center pixel of the window. The appropriate shrinkage factor α, β is then selected according to the corresponding shrink function (6).

In practice, we select the local window $R * R$ (i.e., $R = 5$) using (5) to calculate the local window center pixel energy value $S_{j,k}^2$ and then use (6) to calculate the local shrinkage function:

$$S_{j,k}^2 = \frac{1}{R^2} \sum_{m=-R}^{m=R} \sum_{n=-R}^{n=R} d_{m,n}^2, \tag{5}$$

$$\widehat{d}_{j,k} = \begin{cases} d_{j,k}\left(1 - \alpha * \dfrac{\lambda^2}{S_{j,k}^2}\right), & \text{if } S_{j,k}^2 \geq \beta * \lambda^2 \\ 0 & \text{else,} \end{cases} \tag{6}$$

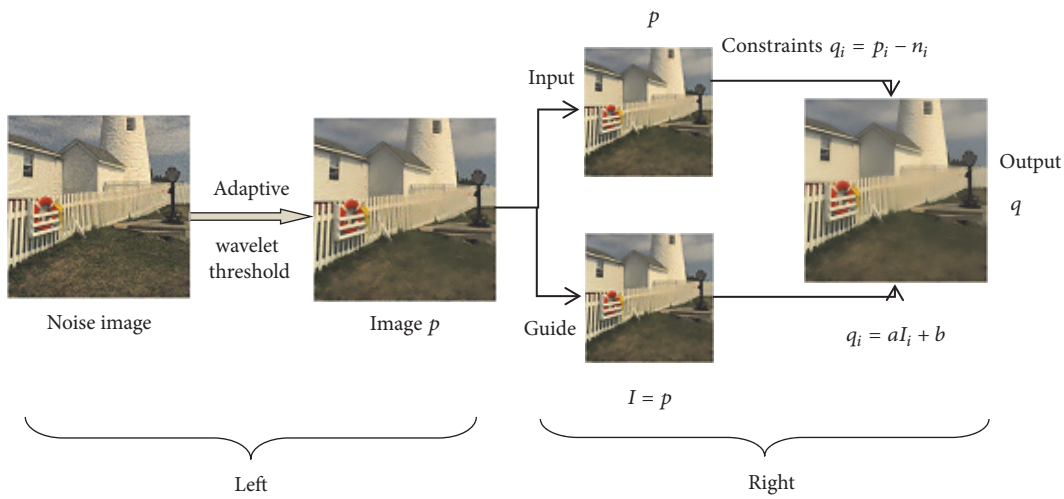


Figure 1: Illustration of the proposed method. It contains two main parts. Left part is denoising by adaptive wavelet threshold algorithm to obtain denoise image p. Right part is using guided filter to enhance edges of image p.

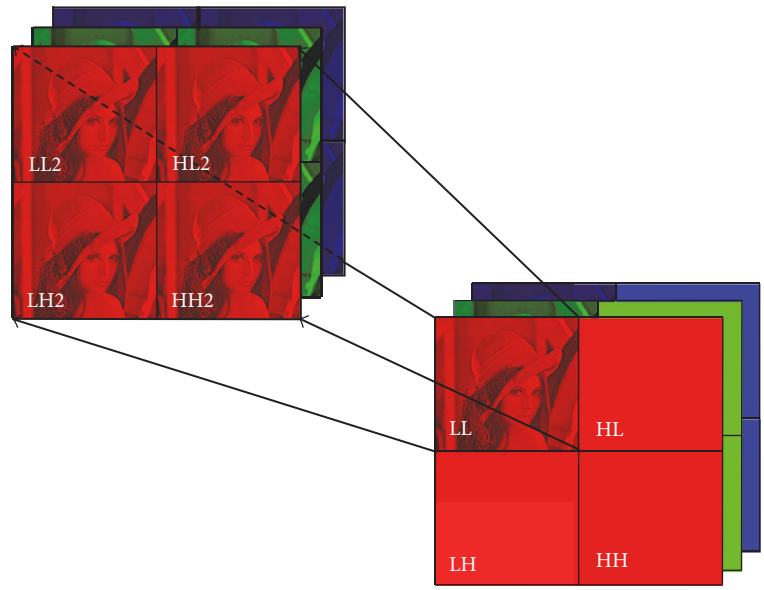


Figure 2: Two-level decomposition of the image. Obtaining Low-Frequency image (LL) from the noise image and decomposing it resulting in subbands LL2, HL2, LH2, and HH2. The adaptive wavelet shrink algorithm is applied to each subband to denoise the image.

$$\sigma^2 = \left[\frac{\text{Median}\left(\left|Y_{i,j}\right|\right)}{0.6745}\right] \quad \left(Y_{i,j} = \text{subband HH}\right), \tag{7}$$

where $\lambda^2 = (4\sigma^2 \log R)$. Equation (7) gives the noise variance σ^2, and $d_{j,k}$ is the central pixel of the local window. If $d_{j,k}$ is at the boundary of the second wavelet coefficients, a boundary condition is required. In the experiments, we set $\alpha = 0.1$, $\beta = 0.3$. Finally, the reconstructed image is denoised.

This denoising algorithm uses the DWT to calculate the energy near the wavelet coefficients and then applies the adaptive wavelet threshold shrink function to denoise the image. In the experiments, we added Gaussian noise with variances of $\sigma^2 = 0.01$ and $\sigma^2 = 0.03$ to the images. Adaptive wavelet threshold shrink function denoising noise and getting denoised image p. From the experimental results, it can be seen that when the noise variance is large, the image p is not very clear after denoising; in particular, the effect of texture is not obvious. Therefore, we need to enhance the image p after this processing step. Thus, the proposed technique uses the guided filter to enhance the image after denoising.

3.3. Guided Filter to Further Processing. Guided filtering is a spatial enhancement technique for the spatial domain, and the filtered output is a linear transformation of the localized image. The filtering algorithm uses a guiding image to process the edges of the noisy image. The guiding image can be the image itself. At this time, their structures are the same; that is, the edges of the original image are the same as the edges of the guiding image. The output pixel values take into account

the statistics of the local spatial neighborhood in the guided image. Hence, using guided filtering, the output image is more structured. This can be used for image dehazing and so on. The guided filter adopts an exact linear algorithm. The algorithm is efficient and fast and is considered to be one of the fastest edge-preserving filters.

For both grayscale and color images, the guided filtering algorithm has $o(N)$ time complexity, regardless of the local window radius (r).

3.3.1. Algorithm of Guided Filter

Guided Filter. The guide image can be a separate image or the original input image; when the guiding image is the original input image, the guided filter retains the edges for image reconstruction.

Assume that the input image is p, the output image is q, and the guiding image is I. q and I have a local linear relationship in the window ω_k centered on pixel k:

$$q_i = a_k I_i + b_k, \quad \forall i \in \omega_k, \tag{8}$$

where a_k, b_k are linear (constant) coefficients in the local window, and the window radius is r. Equation (8) calculates the guidance of the guiding image to be $\nabla q = a\nabla I$. Therefore, the linear equation is only guaranteed if I is in the presence of an edge, and the output image q will contain edges. To determine the coefficients, we require constraints from the input image p. The output q is the input p with a number of noise components n_i subtracted, that is,

$$q_i = p_i - n_i. \tag{9}$$

To minimize the difference between the output q and the input p and ensure the linearity of (8), the function $E(a_k, b_k)$ is minimized in the window ω_k, where ε is the regularization parameter:

$$E(a_k, b_k) = \sum_{i \in \omega_k} \left(\left(a_k I_i + b_k - p_i \right)^2 + \varepsilon a_k^2 \right). \tag{10}$$

In order to minimize the value of the model $E(a_k, b_k)$, we can derive a_k and b_k, respectively, and (10) can be solved using linear regression:

$$\begin{aligned} a_k &= \frac{\left((1/|\omega|) \sum_{i \in \omega_k} I_i p_i \right) - \mu_k \overline{p}_k}{\sigma_k^2 + \varepsilon}, \\ b_k &= \overline{p}_k - a_k \mu_k, \end{aligned} \tag{11}$$

where μ_k and σ_k^2 represent the mean and variance in the local window ω_k, respectively; $|\omega|$ is the number of pixels in the window; and $\overline{p}_k$ represents the mean value in the window ω_k of p. After obtaining a_k and b_k, in Figure 3, the output q_i is

$$q_i = \frac{1}{|\omega|} \sum_{k | i \in \omega_k} \left(a_k I_i + b_k \right) = \overline{a}_i I_i + \overline{b}_i, \tag{12}$$

where $\overline{a}_i = (1/|\omega|) \sum_{k \in \omega_i} a_k$, $\overline{b}_i = (1/|\omega|) \sum_{k \in \omega_k} b_k$. Convert a_k and b_k into weights forms, which is the general form of filtering.

Figure 3: Illustration of (12). When the window ω is sliding in the image, a pixel is involved in many windows that covers pixel. For instance, windows ω_1 and ω_2 are different local windows that cover the same pixel i. Therefore, the output q_i should average all the values of pixel i in different windows.

Therefore, the guide filter algorithm proceeds as follows. Traverse the entire image via each local window, implementing the following calculation, where f is the mean filter with local window radius r; input image is p; guidance image is I; corr is the correlation coefficient; var is the variance; ε is smoothing factor; and cov is the covariance:

$$\begin{aligned} \text{mean}_I &= f_{\text{mean}}(I, r) \\ \text{mean}_p &= f_{\text{mean}}(p, r) \\ \text{corr}_I &= f_{\text{mean}}(I. * I, r) \\ \text{corr}_{Ip} &= f_{\text{mean}}(I. * p, r) \\ \text{var}_I &= \text{corr}_I - \text{mean}_I. * \text{mean}_I \\ \text{cov}_{Ip} &= \text{cov}_{Ip} - \text{mean}_I. * \text{mean}_p \\ a &= \frac{\text{cov}_{Ip}.}{(\text{var}_I + \varepsilon)} \\ b &= \text{mean}_p - a. * \text{mean}_I \\ \text{mean}_a &= f_{\text{mean}}(a, r) \\ \text{mean}_b &= f_{\text{mean}}(b, r) \\ q &= \text{mean}_a. * I + \text{mean}_b. \end{aligned} \tag{13}$$

3.3.2. Edge Preservation. When $I = p$, the guided filter becomes an edge-preserving filter. At this point, we have

$$\begin{aligned} a_k &= \frac{\sigma_k^2}{(\sigma_k^2 + \varepsilon)}, \\ b_k &= (1 - a_k) \mu_k. \end{aligned} \tag{14}$$

Therefore, when the local window variance is large, the center pixel value remains unchanged. In smoother areas, the average value of the neighboring pixels is used as the center pixel value.

3.3.3. Guided Filter for Color Image. When the guided filter is applied independently to the three color channels of the color image, (8) can be rewritten as

$$q_i = a_k^T I_i + b_k, \quad \forall i \in \omega_k, \tag{15}$$

where I_i is a 3×1 color vector, a_k is a 3×1 coefficient vector, and q_i, b_k are scalars. Thus, the color image of the guided filter is

$$\begin{aligned} a_k &= \left(\sum_k + \varepsilon U \right)^{-1} \left(\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i p_i - \mu_k \overline{p}_k \right), \\ b_k &= \overline{p}_k - a_k^T \mu_k, \\ q_i &= \overline{a}_i^T I_i + \overline{b}_i, \end{aligned} \tag{16}$$

where $\sum_k$ is the 3×3 covariance matrix of I in ω_k and U is the 3×3 identity matrix.

Because the local linear model is more effective in the color space, the edges of gray images cannot be identified but through the color image of the guided filter it can be well-preserved. Thus, the image edges have a significant effect.

3.4. Image Enhancement. For images with more texture (i.e., Image 5), the above denoising method may cause some regions to appear too smooth. Therefore, it is necessary to further enhance the image texture detail. Using (17), the denoised image q is subtracted from the original noiseless image to yield the details of the loss in q. Positive number c is to control and balance the degree of its stacking:

$$q\_\text{enhanced} = (I - q) * c + q. \tag{17}$$

The proposed algorithm uses the characteristics of the image structure. In the wavelet domain, threshold shrinkage is used to denoise the image, and then the guided filter enhances the edges of the denoised image to better reflect the texture details of the image.

4. Experimental Results and Analysis

We conducted a series of experiments using MATLAB R2015b and images in Figure 4. Images in Figure 4, Image 9, Image 10, Image 11, and Image 12, are rich in texture. First, variance of 0.01 and 0.03 Gaussian noise was added to the color images in Figure 4. This produced the noisy images shown in Figures 5(a), 6(a), 7(a), 8(a), 9(a), 10(a), 11(a), and 12(a). The proposed method based on image characteristics and the adaptive wavelet threshold shrinkage algorithm was then applied to obtain the denoised images p shown in Figures 5(c), 6(c), 7(c), 8(c), 9(c), 10(c), 11(c), and 12(c). We used the "sym4" two-level wavelet decomposition function to decompose the noisy images, with a local window $R = 5 * 5$, $\beta = 0.3$, and $\alpha = 0.1$ to control the degree of shrinkage in the wavelet coefficients.

The guided filter was used to denoise the image p with the guiding image I set to the original image without noise. This was intended to give the image a better edge effect after denoising. The local window radius was set to $r = 8$ and $\varepsilon = 0.02^2$. The image p was denoised after the adaptive wavelet threshold algorithm. In image p, the r, g, b channels were individually subjected to the guide filter. The texture and edge information of the images were enhanced and we obtained the output q. The parameter c in (17) was found to give better texture effects and undistorted images when $c = 1.5$. The final results q_enhanced images are presented in Figures 5(f), 6(f), 7(f), 8(f), 9(f), 10(f), 11(f), and 12(f). From Figures 9(f), 10(f), 11(f), and 12(f), it can be concluded that the texture and the edge of images had a good performance in proposed method.

The peak signal-to-noise ratio *(PSNR)* of the proposed method is presented in Table 1. These *PSNRs* indicate that the method proposed in this paper is superior to bilateral filtering, the adaptive wavelet threshold algorithm, nonlocal means [20] algorithm, and BM3D [21] method. When the noise level increases, the denoising effect of all five methods decreases, but the proposed method gives better performance than the other four.

The *PSNR* is calculated as follows:

$$\begin{aligned} \text{PSNR} &= 10 * \log_{10} \left(\frac{255^2}{\text{MSE}} \right), \\ \text{MSE} &= \frac{1}{3} \left(\text{mse}_r + \text{mse}_g + \text{mse}_b \right), \\ \text{mse}_k &= \frac{1}{m * n} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(I(i,j) - q(i,j) \right)^2, \\ & \qquad k = r, g, b, \end{aligned} \tag{18}$$

where the input image is I and the output image is q_enhanced. The mean square error *(MSE)* is the average mean square error of the three channels (r, g, b) in the color image.

Wells [22] proposed a quality factor to evaluate the edges of an image after denoising. This quality factor (Pratt's figure of merit) takes into account three kinds of error: the loss of the effective edge, the edge of the positioning error, and noise misjudged as the edge. Pratt's figure of merit provides a quantitative evaluation of image edges and can be used to compare the edges in the denoised image and the original image. From Figures 13 and 14, it is obvious that there are differences between the texture parts. Pratt's figure of merit was measured in four images. The results in Figure 15 confirm that the edge quality factor (red curve) of the proposed method is higher than that of the other four methods for different degrees of noise variance. Thus, our method achieves better edge effects.

From a subjective point of view, we find that not only can the proposed method preserve edge better, it also reduces noise well when compared with the other four methods. From Figures 5–12, the result images (f) using proposed method look clearer than the others because we use a guided filter to filter the image and eliminate noise. Moreover, in Figures 13 and 14, the edge of the result image that uses the proposed method is smooth and coherent, which greatly helps to enhance edge.

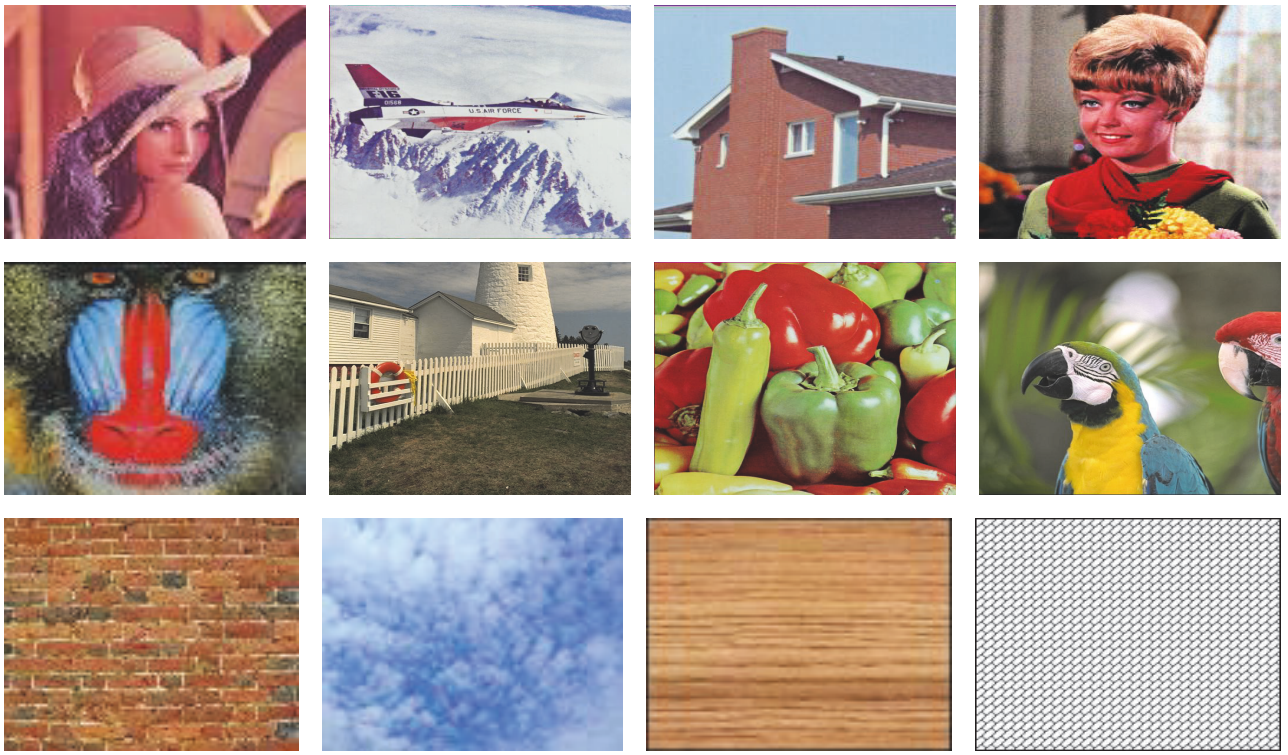
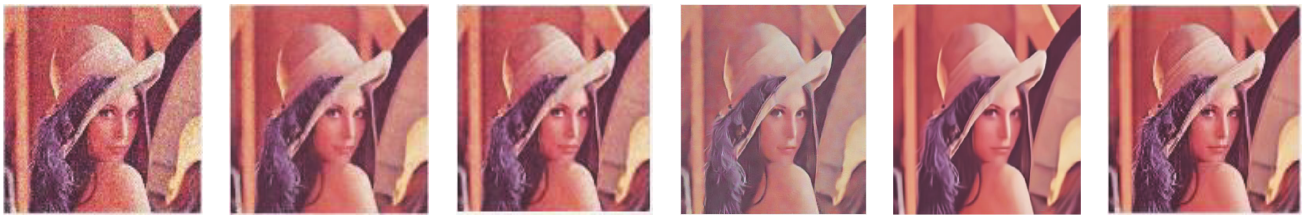


Figure 4: Test image: Images 1–12 (from left to right, top to bottom).



(a) $\sigma^2 = 0.01$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

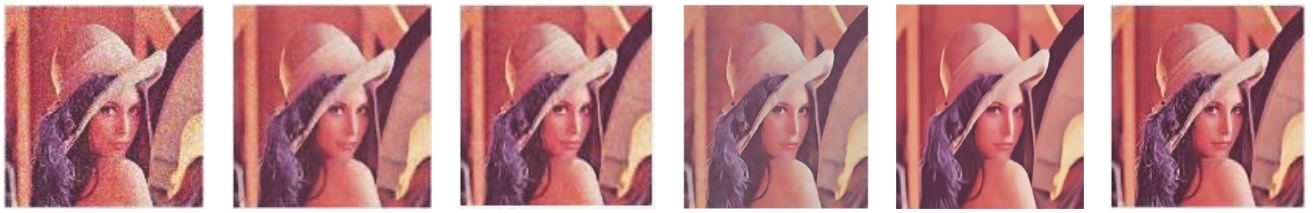
Figure 5: Image 1: comparison of various experimental results under noise variance $\sigma^2 = 0.01$.

To objectively evaluate the results, we use the Pratt's figure of merit and PSNR for image quality evaluation. Pratt's figure of merit results are listed in Figure 15. y-axis is the Pratt factor results and x-axis is noise variance. The curves showed that Pratt factor decreases when the noise variance increases, but proposed method (red curves) is much higher than others. Curves results suggested that edge of image using proposed method is effective and abundant. Table 1 is result of *PSNR*. The proposed method is superior to the other four methods with respect to image processing.

From the data in Table 1 and the quality factor curves in Figure 15, we can conclude that the proposed method is superior to bilateral filtering, the adaptive wavelet threshold denoising algorithm, nonlocal means algorithm, and BM3D algorithm on color image denoising. From a visual point of view, the proposed method not only gives a good denoising effect but also achieves better edge retention.

5. Conclusion

In this paper, a new denoising method based on the adaptive wavelet threshold denoising algorithm and edge-guided filtering has been proposed. The image denoising is performed according to the local structure of the image. In comparative experiments against bilateral filtering, the adaptive wavelet denoising method, nonlocal means algorithm, and BM3D algorithm, the proposed method exhibited the best denoising and edge preservation performance. The proposed approach removes Gaussian noise in the frequency domain and then uses linear guided filtering to further enhance the image recovery, resulting in better denoising and edge effects. As color images display more detailed textures, this filtering overcomes the gradient inversion effect in the edge regions. As the linear model (see (8)) is a block-unsupervised learning method, it can be combined with other models to obtain new

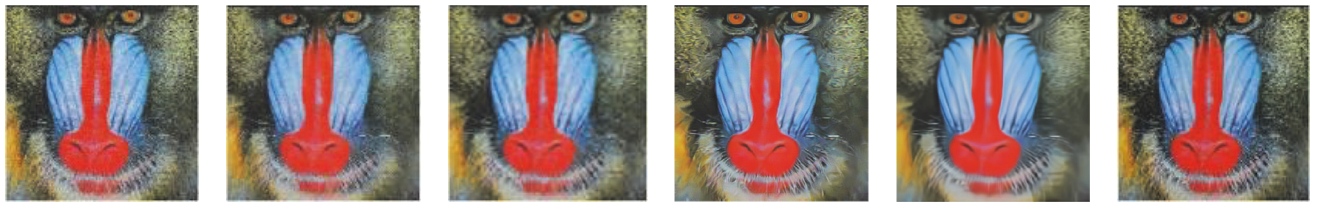


(a) $\sigma^2 = 0.03$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 6: Image 1: comparison of various experimental results under noise variance $\sigma^2 = 0.03$.

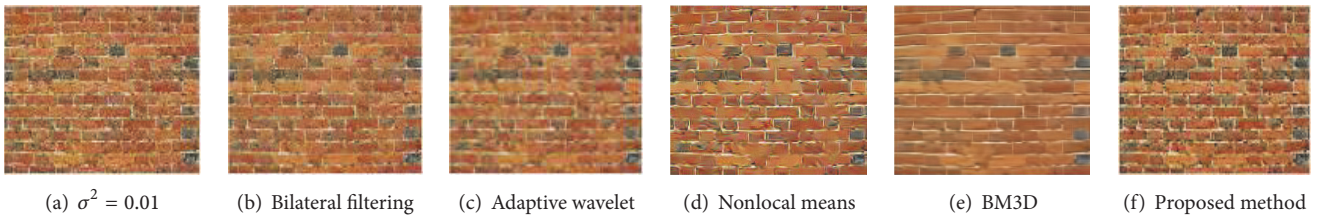
(a) $\sigma^2 = 0.01$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 7: Image 5: comparison of various experimental results under noise variance $\sigma^2 = 0.01$.



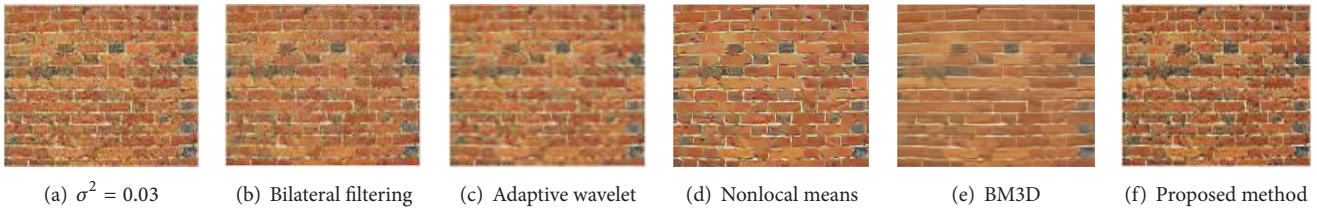
(a) $\sigma^2 = 0.03$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 8: Image 5: comparison of various experimental results under noise variance $\sigma^2 = 0.03$.



(a) $\sigma^2 = 0.01$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 9: Image 9: comparison of various experimental results under noise variance $\sigma^2 = 0.01$.

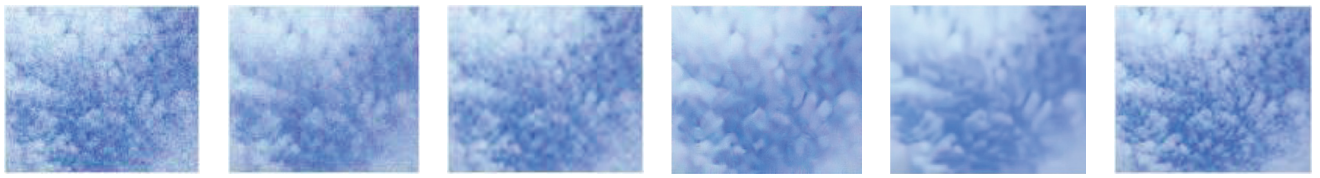


(a) $\sigma^2 = 0.03$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 10: Image 9: comparison of various experimental results under noise variance $\sigma^2 = 0.03$.

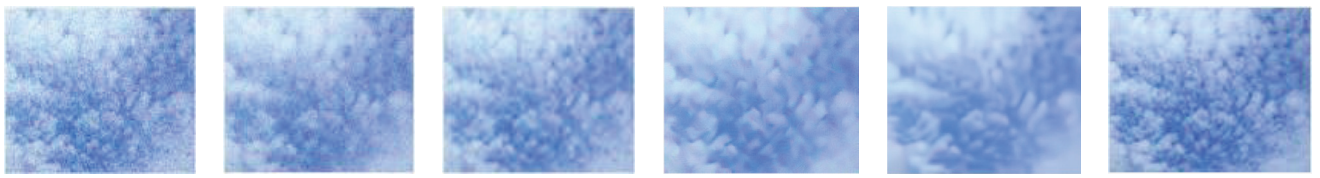
Table 1: PSNR of the proposed method, bilateral filtering, and adaptive wavelet.

Noise variance	0.01	0.03	0.01	0.03	0.01	0.03
	Image 1 (256 ∗ 256)		Image 2 (512 ∗ 512)		Image 3 (256 ∗ 256)	
Adaptive wavelet	26.7048	25.4507	28.0801	26.4821	27.7967	26.1699
Bilateral filtering	22.5751	22.0167	23.1551	22.6118	22.7560	22.1802
Nonlocal means	29.9765	27.4871	30.1415	27.6777	30.1910	27.6823
BM3D	28.9235	26.9276	30.4215	27.8901	30.9457	28.0957
Proposed method	**39.5634**	**35.3129**	**41.5647**	**36.0267**	**40.8101**	**35.5930**
	Image 4 (256 ∗ 256)		Image 5 (256 ∗ 256)		Image 6 (512 ∗ 512)	
Adaptive wavelet	25.2375	24.2724	23.4663	22.7332	25.6576	24.5766
Bilateral filtering	22.6074	21.9415	22.1076	21.5494	22.3016	21.7254
Nonlocal means	26.6512	25.3287	25.3588	24.3120	27.8813	26.2105
BM3D	25.5157	24.4968	24.0394	23.2556	28.0621	26.3334
Proposed method	**37.0265**	**34.1344**	**33.8535**	**32.1364**	**36.8294**	**33.9588**
	Image 7 (512 ∗ 512)		Image 8 (512 ∗ 512)		Image 9 (200 ∗ 150)	
Adaptive wavelet	27.5706	25.9130	29.0039	26.9908	21.4746	20.8466
Bilateral filtering	23.2116	22.4080	23.1373	22.4268	21.4834	21.0023
Nonlocal means	29.4155	27.0736	31.3181	28.2534	23.7273	23.0137
BM3D	29.1443	26.9120	32.0590	28.6197	21.7645	21.2712
Proposed method	**39.1414**	**34.8926**	**41.2256**	**35.7943**	**31.4041**	**30.0693**
	Image 10 (200 ∗ 150)		Image 11 (187 ∗ 171)		Image 12 (187 ∗ 171)	
Adaptive wavelet	29.4247	27.5268	26.7443	25.4201	15.9075	16.9781
Bilateral filtering	23.6728	23.0581	22.2984	21.7973	22.7751	22.6673
Nonlocal means	30.3106	27.9794	28.7439	26.6227	17.4105	17.2673
BM3D	30.8146	28.3780	31.8744	28.4922	31.0366	30.1990
Proposed method	**40.2191**	**35.9371**	**38.7096**	**35.0295**	**24.4256**	**25.7756**



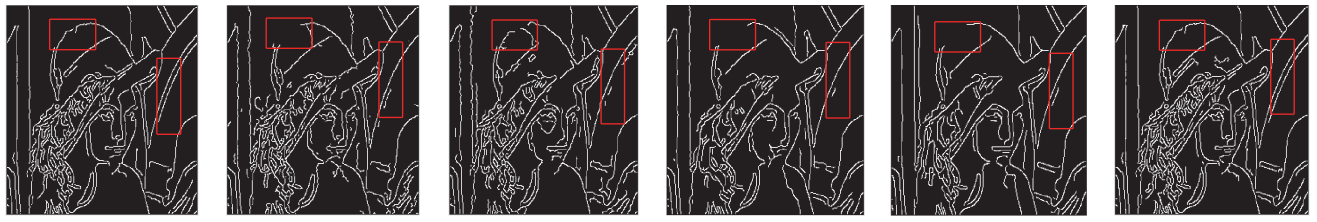
(a) $\sigma^2 = 0.01$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 11: Image 10: comparison of various experimental results under noise variance $\sigma^2 = 0.01$.



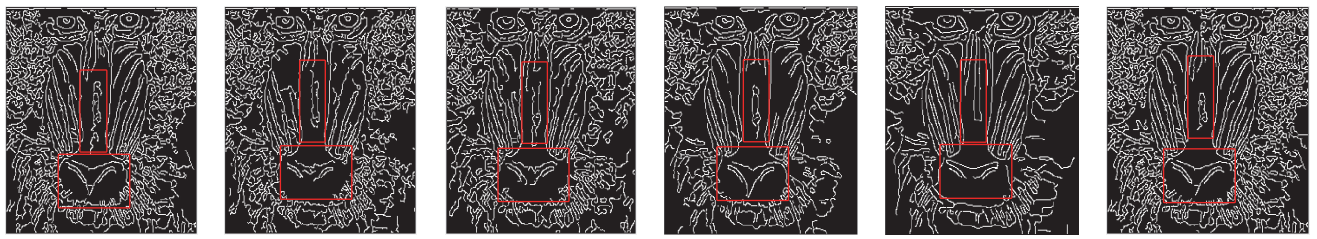
(a) $\sigma^2 = 0.03$ (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 12: Image 10: comparison of various experimental results under noise variance $\sigma^2 = 0.03$.



(a) Image 1 edge (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 13: Image 1: edge comparison of various experimental results under noise variance $\sigma^2 = 0.03$.



(a) Image 5 edge (b) Bilateral filtering (c) Adaptive wavelet (d) Nonlocal means (e) BM3D (f) Proposed method

Figure 14: Image 5: edge comparison of various experimental results under noise variance $\sigma^2 = 0.01$.

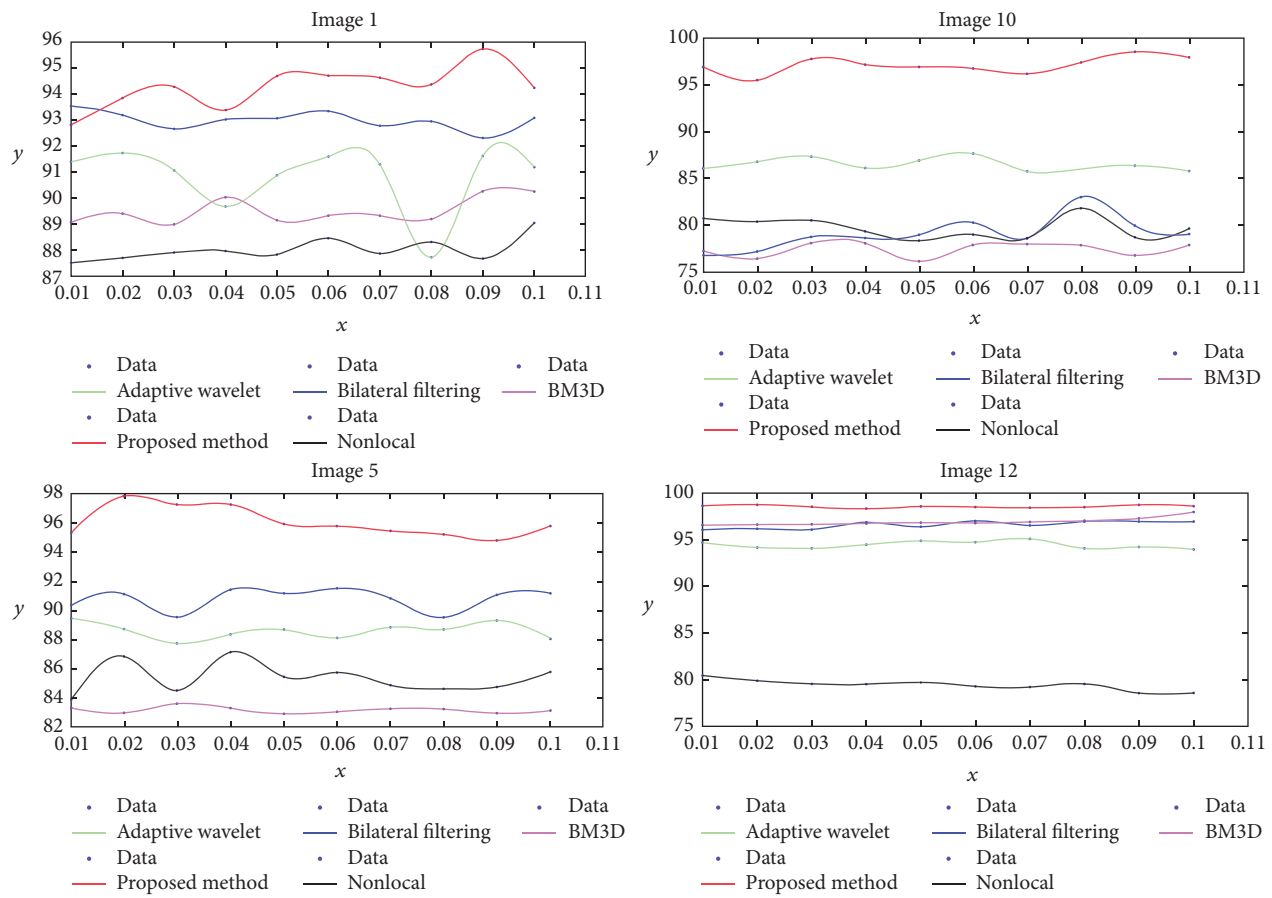


Figure 15: Pratt's figure of merit.

denoising techniques. This will be the focus of future research and exploration on color image denoising.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61370138, 61572077, and U1301251) and the Beijing Municipal Natural Science Foundation (no. 4162027); the Project of Oriented Characteristic Disciplines (no. KYDE40201701); the Project of Construction of Innovative Teams and Teacher Career Development for Universities and Colleges Under Beijing Municipality (no. IDHT20170511).

References

[1] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the International Conference on Computer Vision IEEE Computer Society*, 839 pages, 1998.

[2] Y. Zhang, X. Tian, and P. Ren, "An adaptive bilateral filter based framework for image denoising," *Neurocomputing*, vol. 140, pp. 299–316, 2014.

[3] J. S. Lim, *Two-dimensional signal and image processing*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1990.

[4] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1200–1224, 1995.

[5] D. L. Donoho, "De-noising by soft-thresholding," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[6] T. D. Bui and G. Chen, "Translation-invariant denoising using multiwavelets," *IEEE Transactions on Signal Processing*, vol. 46, no. 12, pp. 3414–3420, 1998.

[7] C. Srisailam, P. Sharma, and S. Suhane, "Color image denoising using wavelet soft thresholding," *International Journal of Emerging Technology and Advanced Engineering*, vol. 4, no. 7, pp. 474–478, 2014.

[8] K. K. Gupta and R. Gupta, "Feature adaptive wavelet shrinkage for image denoising," in *Proceedings of the 2007 International Conference on Signal Processing, Communications and Networking, ICSCN 2007*, pp. 81–85, ind, February 2007.

[9] E. Ordentlich, G. Seroussi, S. Verdu, M. Weinberger, and T. Weissman, "A discrete universal denoiser and its application to binary images," in *Proceedings of the International Conference on Image Processing (ICIP 2003)*, vol. 1, pp. 20–117, 2003.

[10] W. Dong and H. Ding, "Full frequency de-noising method based on wavelet decomposition and noise-type detection," *Neurocomputing*, vol. 214, pp. 902–909, 2016.

[11] I. Elyasi and S. Zermchi, "Elimination noise by adaptive wavelet threshold," *World Academy of Science Engineering & Technology*, vol. 3, no. 8, pp. 1541–1545, 2009.

[12] A. K. Bhandari, D. Kumar, A. Kumar, and G. K. Singh, "Optimal sub-band adaptive thresholding based edge preserved satellite image denoising using adaptive differential evolution algorithm," *Neurocomputing*, vol. 174, pp. 698–721, 2016.

[13] T. T. Cai and B. W. Silverman, "Incorporating information on neighbouring coefficients into wavelet estimation, Sankhyā," *The Indian Journal of Statistics, Series B (1960-2002)*, vol. 63, no. 2, pp. 127–148, 2001.

[14] G. Y. Chen, T. D. Bui, and A. Krzyzak, "Image denoising using neighbouring wavelet coefficients," *Integrated Computer-Aided Engineering*, vol. 12, no. 1, pp. 99–107, 2005.

[15] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.

[16] M. Nasri and H. Nezamabadi-pour, "Image denoising in the wavelet domain using a new adaptive thresholding function," *Neurocomputing*, vol. 72, no. 4–6, pp. 1012–1025, 2009.

[17] X. Liu, H. Wan, Z. Shang, and L. Shi, "Automatic extracellular spike denoising using wavelet neighbor coefficients and level dependency," *Neurocomputing*, vol. 149, pp. 1407–1414, 2015.

[18] C.-C. Kao, J.-H. Lai, and S.-Y. Chien, "VLSI architecture design of guided filter for 30 frames/s full-HD Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 513–524, 2014.

[19] A. Gadge and S. S. Agrawal, "Guided filter for color image," *IJIREEICE*, vol. 4, no. 6, pp. 250–252, 2016.

[20] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 60–65, June 2005.

[21] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[22] P. N. T. Wells, *Digital Image Processing*, W. K. Pratt John Wiley, Chichester, UK, 1991, (Medical Engineering & Physics, vol. 16 no. 1, p. 83, 1994).