Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2010, Article ID 923748, 13 pages
doi:10.1155/2010/923748

Research Article

An Interactive Procedure to Preserve the Desired Edges during the Image Processing of Noise Reduction

Chih-Yu Hsu,[1] Hsuan-Yu Huang,[2] and Lin-Tsang Lee[2]

[1] *Department of Information and Communication Engineering, ChaoYang University of Technology, Taichung 41349, Taiwan*
[2] *Department of Applied Mathematics, National Chung-Hsing University, Taichung 40227, Taiwan*

Correspondence should be addressed to Chih-Yu Hsu, tccnchsu@gmail.com

Received 1 December 2009; Revised 5 February 2010; Accepted 30 March 2010

Academic Editor: Yingzi Du

The paper propose a new procedure including four stages in order to preserve the desired edges during the image processing of noise reduction. A denoised image can be obtained from a noisy image at the first stage of the procedure. At the second stage, an edge map can be obtained by the Canny edge detector to find the edges of the object contours. Manual modification of an edge map at the third stage is optional to capture all the desired edges of the object contours. At the final stage, a new method called Edge Preserved Inhomogeneous Diffusion Equation (EPIDE) is used to smooth the noisy images or the previously denoised image at the first stage for achieving the edge preservation. The Optical Character Recognition (OCR) results in the experiments show that the proposed procedure has the best recognition result because of the capability of edge preservation.

1. Introduction

Digital images are noisy due to environmental disturbances. To ensure image quality, image processing of noise reduction is a very important step before analysis or using images. Optical Character Recognition (OCR) system is an example that is very sensitive to noise. The quality of documents influences the results of recognition. Image noise decreases the accuracy of the recognition of documentations by OCR (optical character recognition) software because of blurred edges. Great damage will be caused in defense and security applications when OCR software is used for the scanning and recognition of documents such as passports and ID cards in busy airports where speed and accuracy are critical for processing thousands of documents daily. The most important image processing technique for noise reduction is the image denoising method. The purpose of image denoising method is to increase signal-to-noise ratio (SNR) in an image. However noise reduction always induces blurred edges by an image denoising process. Development for edge-preserved image denoising method is necessary for OCR software. The paper is to develop a denoising procedure with the edge preservation capability. The OCR system is a research field in pattern recognition [1, 2] and is used to convert papers, books, and documents into electronic files [3]. Researchers developed several methods in order to remove these image noise including Gaussian noise, salt and pepper noise [4]. There are some image filters, which are used for image denoising [5, 6] and the Gaussian filter is a well-known one [7]. In the period between 1984 and 1987, Koenderink and Hummel showed how Gaussian filters removed noise that was equal to dispersion effects of the isotropic diffusion equation, so Gaussian filters are called Diffusion Filters.

Isotropic diffusion equations can reduce noise but blur the contours of images. In order to improve on this drawback, Perona and Malik improved the diffusion coefficient of isotropic diffusion filters to produce anisotropic diffusion filters (ADFs) with a function of image gradients in the 1990 [8]. The coefficients of isotropic diffusion equations are constants, but the diffusion coefficients of anisotropic diffusion equation decrease as image gradients increase. Anisotropic diffusion equations are more effective in edge preservation. Because the gradients of noises are larger, then the coefficients of anisotropic diffusion filters are smaller. This solution cannot solve the problem very well. Researches [9, 10] are continuously focusing on improving the diffusion coefficients. However, these methods may remove image noise, but edges cannot be preserved.

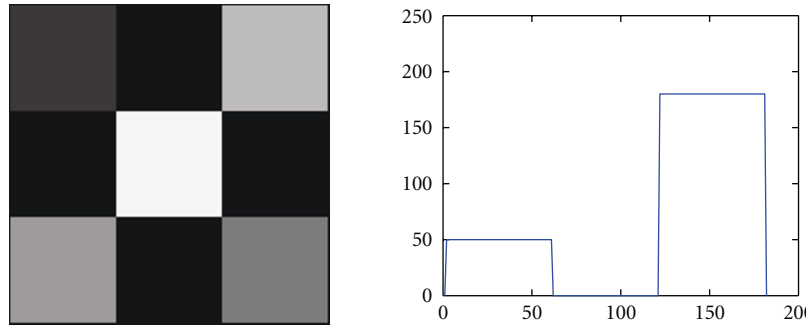


Figure 1: (a) Synthetic image and (b) the grayscale value of the 36th row of (a).

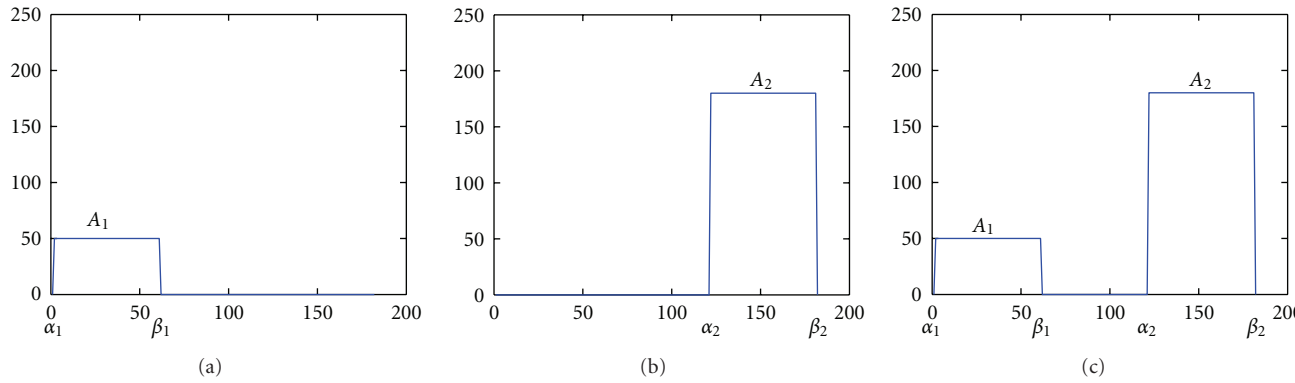


Figure 2: (a) The front of signal of Figure 1(b), (b) the back of signal of Figure 1(b), and (c) the signal is superimposed by (a) and (b).

In this paper, we propose a new procedure including four stages. At the first stage of the procedure, any kind of denoising algorithm can be applied on an original noisy image to get a well-denoised image. At the second stage, an edge map can be obtained to find the edges of the object contours by the Canny edge detector applied on the previously denoised image at the first stage. Since the contour edges are not found completely, then the users maybe need interactively modify the edge map to keep the edges of the desired object contours. At the third stage, manually modify the edges of edge map to match the desired edges. At the final stage, a new method Edge Preserved Inhomogeneous Diffusion Equation is used to smooth the original noisy image or the previously denoised image at the first stage and achieve preserving desired edge. The proposed procedure has the edge preservation capability that makes OCR results the best in this experiment.

2. Mathematic Formulation

Section 2.1 introduces the digital image as a matrix, and one row can be considered as a signal. Section 2.2 introduces how to find the solutions of a one-dimensional inhomogeneous diffusion equation by using Fourier series. Section 2.3 proposed a flow chart of the EPIDE denoising method.

2.1. Digital Images and Signals. We defined a $m \times n$ grayscale digital image as a function $u(x, y)$. The value of the function $u(x, y)$ is an image intensity that is between 0 and 255. For a gray image, the function u has grayscale values of image pixels. The coordinates (x, y) are locations of the pixels in an image. The grayscale values of the ith row of $u(x, y)$ are denoted by $u(i, 1 : n)$ which can be considered as a one-dimensional signal with length n. For example, the red line as shown in Figure 1(a) is the 36th row of the image. As shown in Figure 1(b), the grayscale values profile is composed of two Box functions.

2.1.1. One-Dimensional Signals. One-dimensional signals can be considered as piecewise constant functions, Heaviside function is suitable to discrete piecewise constant functions [11]. Heaviside function $H(x)$ is defined as:

$$H(x) = \begin{cases} 0, & x < 0, \\ 1, & x > 0. \end{cases} \tag{1}$$

The Heaviside function $H(x)$ is discontinuous at $x = 0$, and the value is usually defined by 1/2 at $x = 0$. If the Heaviside function $H(x)$ is shifted a, then Heaviside function is $H(x - a)$. Box Function ϕ can be represented by Heaviside function $H(x)$ as $\phi(x) = (H(x) - H(x-1))$. The Box function can be represented as follows:

$$\phi(x) = \begin{cases} 1, & 0 < x < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

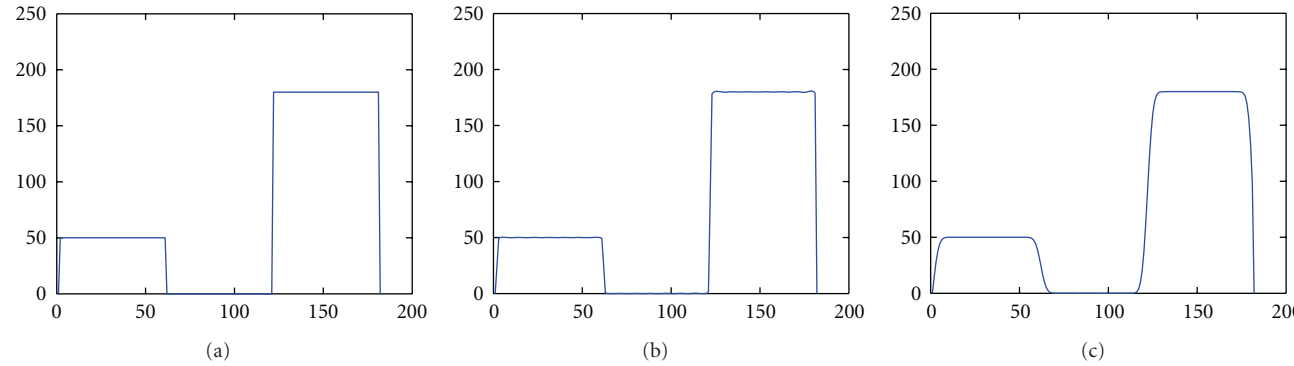


Figure 3: (a) The grayscale value of the 36th row of Figure 1(a), (b) the Fourier series with 300 terms, and (c) diffused result by diffusion equation at $t = 3$.

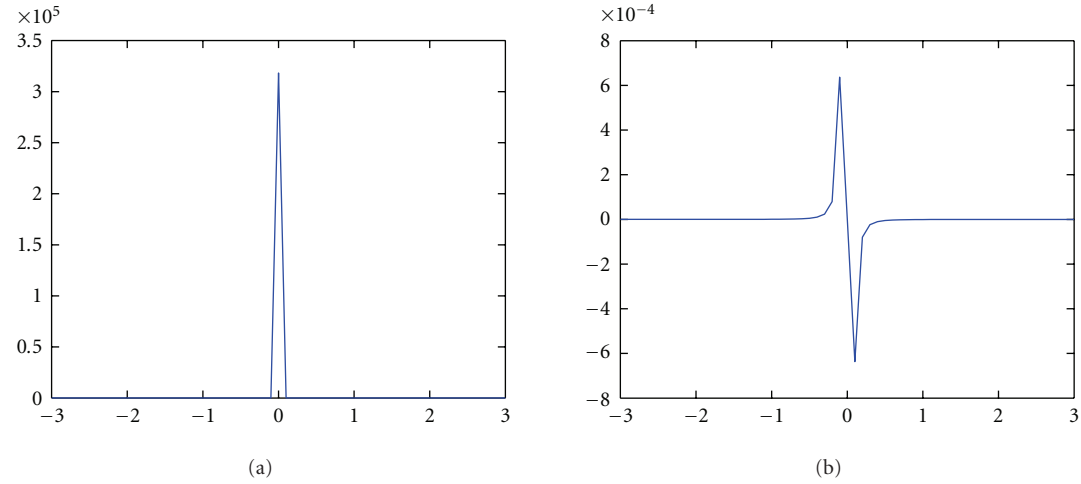


Figure 4: (a) $\delta(x) = (\varepsilon/\pi)/(x^2 + \varepsilon^2)$, $\varepsilon = 10^{-6}$ and (b) $\delta'(x) = -2\varepsilon x/\pi(x^2 + \varepsilon^2)^2$, $\varepsilon = 10^{-6}$.

As shown in Figure 1(b), the 36th row of $u(x, y)$ is $u(36, 1 : n)$ and the profile of the row is represented by two Box functions as in the following equation:

$$u(36, 1 : n) = \sum_{k=1}^{2} A_k[H(x - \alpha_k) - H(x - \beta_k)]. \quad (3)$$

One signal can be superimposed by two signals. Figure 2 shows how to use two box functions to superimpose the signal in Figure 1(b).

If there are M Box functions in the ith row of $u(x, y)$, the profile of $u(i, 1 : n)$ can be represented as follows:

$$u(i, 1 : n) = \sum_{k=1}^{M} A_k[H(x - \alpha_k) - H(x - \beta_k)]. \quad (4)$$

The letter α_k is the left-location value of the kth Box Function and β_k is value of the right location of the kth Box Function. The letter M denotes the total number of Box functions and the symbol A_k are coefficient constants.

2.1.2. Fourier Series of Box Function. According to (4), the function $u(i, 1 : n)$ as one signal can be represented by summation of the finite Box functions. If $u(i, 1 : n)$ is an integrable function on $[0, \pi]$, then $u(i, 1 : n)$ can approximate the continuous Fourier series [12] as follows:

$$u(i, 1 : n) = \frac{1}{2}a_0 + \sum_{k=1}^{\infty}(a_k \cos kx + b_k \sin kx), \quad (5)$$

where the coefficients are represented by $a_k = (1/\pi)\int_{-\pi}^{\pi} f(x)\cos kx dx$ and $b_k = (1/\pi)\int_{-\pi}^{\pi} f(x)\sin kx dx$ equations.

For example as shown in Figure 3, the grayscale value of the 36th row of Figure 1(a) is shown in Figure 3(a). Figure 3(b) shows a profile to approximate signal of Figure 3(a) by using Fourier series with 300 terms. Figure 3(c) shows defused result at t = 3 where the variable t will be explained in Section 2.2.1.

2.2. One-Dimensional Inhomogeneous Diffusion Equation. We want to solve the problem of finding the intensity $u(x, t)$ of every row in an image. At both sides of the interval $0 \le x \le n$, the intensity values are set to be zero. By adding the inhomogeneous terms into the diffusion equation with the derivative of Delta functions,

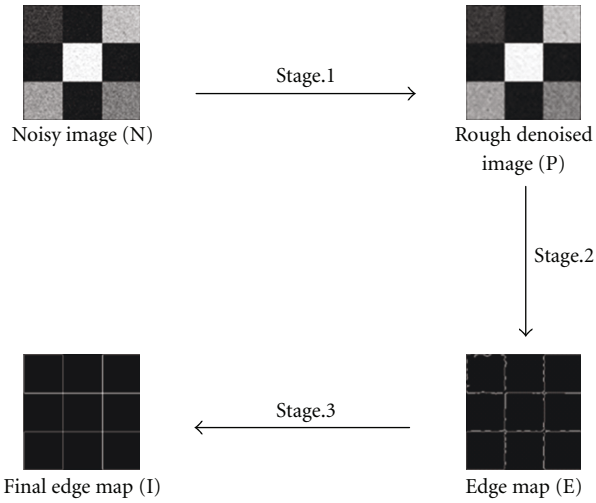


Figure 5: The flow chart of finding the edge map during the three stages.

the proposed denoising method is called the Edge Preserved Inhomogeneous Diffusion Equation (EPIDE) method.

2.2.1. Diffusion Equation Formulation. Consider the inhomogeneous differential equation [13] as follows:

$$\frac{\partial u(x,t)}{\partial t} = K\nabla^2 u(x,t) + F(x), \tag{6}$$

where the variables x are spatial coordinates and t is time, but the temperature $u(x,t)$ now is replaced by the intensity in an image that is function of the position x and time t, and K is a constant called the "thermal diffusivity" of the material. The function $F(x)$ is an inhomogeneous term that will be explained in (7).

The function $F(x)$ is used to have the effect of preserving edges and can be obtained from derivative of the right side of (4):

$$F(x) = \sum_{k=1}^{M} A_k[\delta'(x-\alpha_k) - \delta'(x-\beta_k)]. \tag{7}$$

The function δ' is a Dipole distribution and is derivative of the Delta (or impulse) function δ.

The relation of the Delta function and step function is as follows:

$$\delta(x) = \frac{dH(x)}{dx}, \tag{8}$$

where step function can be approximated by $H(x) = (1/2)(1+(2/\pi)\arctan(x/\varepsilon))$, and $\delta = H'(x) = (\varepsilon/\pi)/(x^2+\varepsilon^2)$, $\varepsilon = 10^{-6}$.

The $\delta(x)$ and $\delta'(x)$ functions are shown in Figures 4(a) and 4(b).

The Delta function is a generalized function, the properties of the Delta function are as follows [14]:

$$\int_a^b \delta(x-\xi)dx = \begin{cases} 1, & a \le \xi \le b, \\ 0, & a, b < \xi \text{ or } \xi < a, b, \end{cases} \tag{9}$$
$$\delta(x-\xi) = 0, \quad x \ne \xi,$$

where a, b, and ζ are constants.

Let (7) be into (6):

$$\frac{\partial u(x,t)}{\partial t} = K\nabla^2 u(x,t) + \sum_{k=1}^{M} A_k[\delta'(x-\alpha_k) - \delta'(x-\beta_k)]. \tag{10}$$

Equation (10) is an inhomogeneous diffusion equation used to preserve the edges. In (4), (7), and (10), the locations of edges α_k and β_k can be decided by the location of edge pixels in the signal that is one row in an image. Since in the edge locations it is not easy to obtain a noisy image, some images preprocessing techniques and Canny Edge detection method [15] are used to find the edge map of the object contours. The locations α_k and β_k are decided by the edge map. Modifying the edge map, the user can decide to keep the contours for their requirements.

2.2.2. Fourier Series Solutions. According to (6), the function $F(x)$ can be represented by Fourier series [12] as follows:

$$F(x) = \sum_{n=1}^{\infty} \beta_n \cdot \sin\frac{n\pi x}{L}. \tag{11}$$

The solution $u(x,t)$ can be solved by the Fourier series and the initial condition is represented as follows:

$$u(x,0) = \sum_{k=1}^{M} A_k[H(x-\alpha_k) - H(x-\beta_k)], \tag{12}$$
$$u(0,t) = u(L,t) = 0.$$

where L is the length of the signal. The solution $u(x,t)$ and the function $F(x)$ can be expanded by Fourier sine series:

$$u(x,t) = \sum_{n=1}^{\infty} b_n(t) \cdot \sin\frac{n\pi x}{L}, \tag{13}$$

where the coefficients $\beta_n = (2/\pi)\int_0^\pi u(x,0)\sin nx dx$ are determined by the function $F(x,t)$ and the coefficients b_n can be decided by substituting (13) into (11):

$$\sum_{n=1}^{\infty} \frac{\partial b_n(t)}{\partial t} \cdot \sin\frac{n\pi x}{L} = \sum_{n=1}^{\infty}\left(-\frac{n^2\pi^2}{L^2} b_n(t) + \beta_n\right) \cdot \sin\frac{n\pi x}{L}. \tag{14}$$

Comparing coefficients of $\sin(n\pi x/L)$ on both sides yields

$$\frac{\partial b_n(t)}{\partial t} + \frac{n^2\pi^2}{L^2} b_n(t) = \beta_n. \tag{15}$$

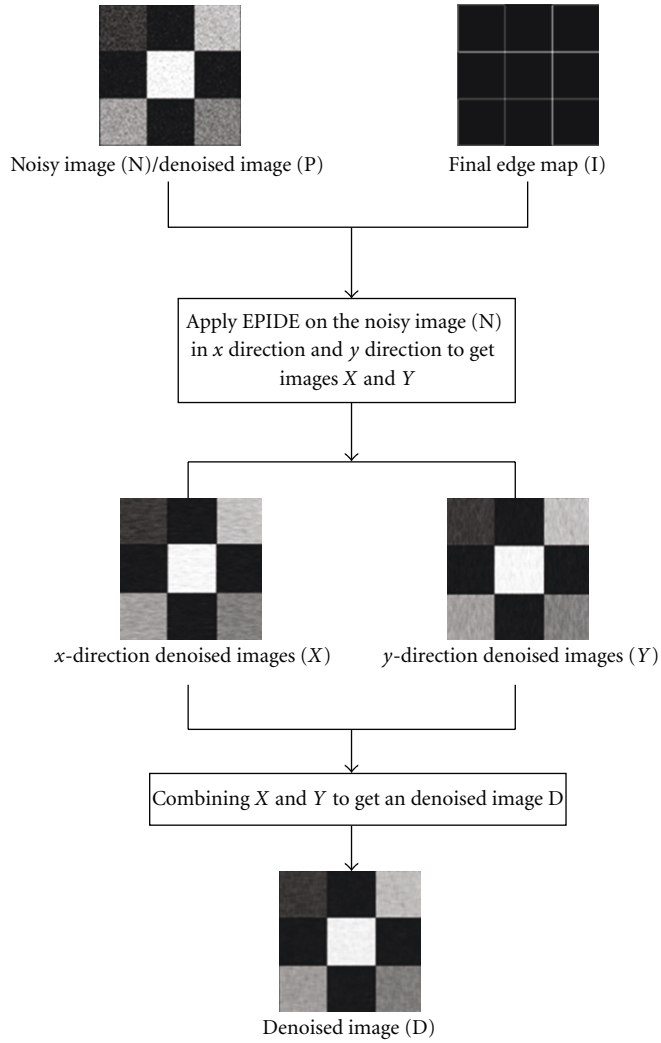


Figure 6: The flow chart to get denoised image at the final stage.

Equation (15) can easily be solved to obtain b_n:

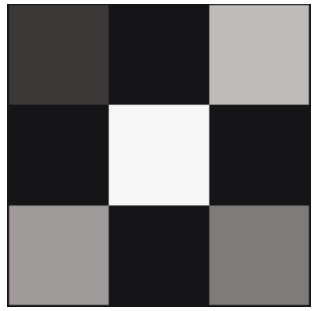
$$b_n(t) = \left[\exp\left(-\frac{n^2\pi^2 t}{L^2}\right)\right] \int_0^t \beta_n \cdot \exp\left(\frac{n^2\pi^2 s}{L^2}\right) ds. \quad (16)$$

Then solution $u(x,t)$ is obtained by substituting this formula for b_n into (13).

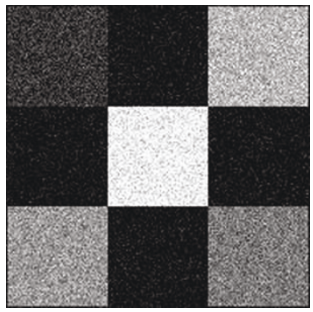
2.3. Proposed Procedure. The goal of the proposed procedure with four stages is to preserve the desired edges during the image processing of noise reduction, so EPIDE method plays an important role. However, the edges of object contours in an image should be extracted previously for EPIDE method. Canny edge detector can automatically find some edges in images. Since the contour edges are not all found, then the users want to interactively modify the edges capture all desired object contours.

In the first stage of the procedure, any kind of denoising algorithm can be applied on an original noisy image to obtain a denoised image. In the second stage, an edge map can be obtained to find the edges of the object contours by the Canny edge detector applied on the previously denoised image at the first stage. Since the contour edges are not found completely, then the users may be need to interactively modify the edge map to keep the edges of the desired object contours. At the third stage, users can manually modify the edges of edge map to match the desired edges. At the final stage, Edge Preserved Inhomogeneous Diffusion Equation (EPIDE) method is used to smooth the original noisy image or the previously denoised image at the first stage and achieve preserving desired edge. Two flow charts of the proposed procedure are shown in Figures 5 and 6.

Figure 5 shows the flow chart of the three stages. At the first stage, any kind of denoising algorithm can be applied on a noisy image (N) to get a previously denoised image (P). At the second stage, an edge map (E) can be obtained to find the edges of the object contours by the Canny edge detector applied on the previously denoised image at the first stage. At the third stage, modified edge map (I) captures all the desired edges manually. The flow chart to get a denoised image at the final stage with EPIDE method is shown in Figure 6.

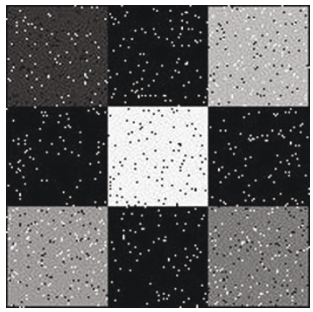


(a)

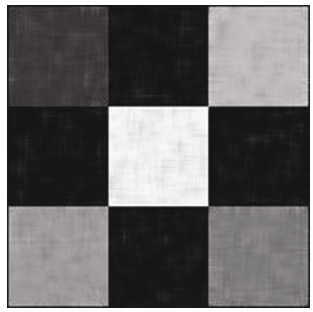


(b)

(c)

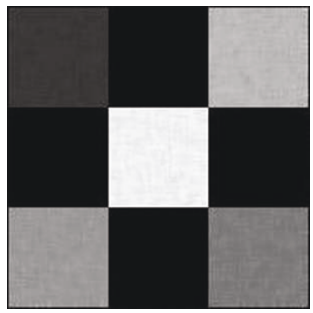


(d)



(e)

(f)



(g)

Figure 7: Comparing the performance of the various noises on "Nine Square Regions". (a) Original Image. (b) Gaussian noise image, $\sigma = 0.01$. (d) Salt and Pepper noise image, noise density is 0.05. (f) Poisson noise image, (c), (e), and (g) The proposed procedure denoised by (b), (d), (f).

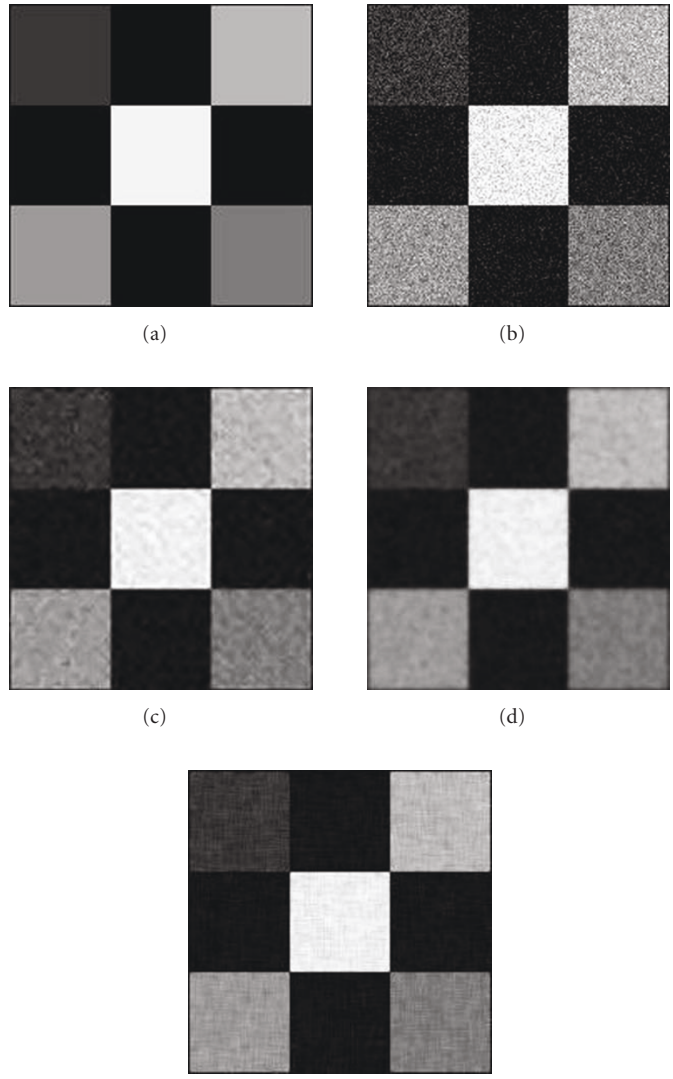


Figure 8: (a) "Nine Square Regions", (b) noisy image with Gaussian noise with $\sigma = 0.01$, (c) wavelet, PSNR = 25.493, (d) ADF, PSNR = 23.835, and (e) the proposed procedure, PSNR = 28.495.

At the final stage, the EPIDE method is used to smooth the noisy image (N) or the previously denoised image (P) with the modified edge map (I). Both in x-direction and y-direction, two images X and Y are generated by the EPIDE method. Finally a denoised image (D) can be obtained by an average combination of the image X and Y.

3. Experimental Results

There are four test images "Nine Square Regions", "Number and Character", "Chinese Words", and "BarCode" corrupted by Gaussian noise with zero mean.

3.1. The Peak Signal-to-Noise Ratio (PSNR). The performance measure by using the peak signal-to-noise ratio is defined as follows:

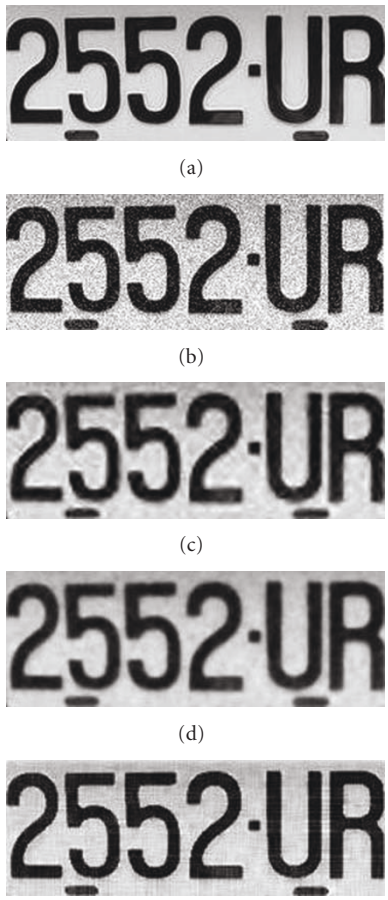
$$\text{PSNR} = 20 \cdot \log_{10} \frac{255}{\text{RMSE}}, \tag{17}$$

where RMSE is Root Mean Square Error, and it is defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} (f(i,j) - g(i,j))^2}. \tag{18}$$

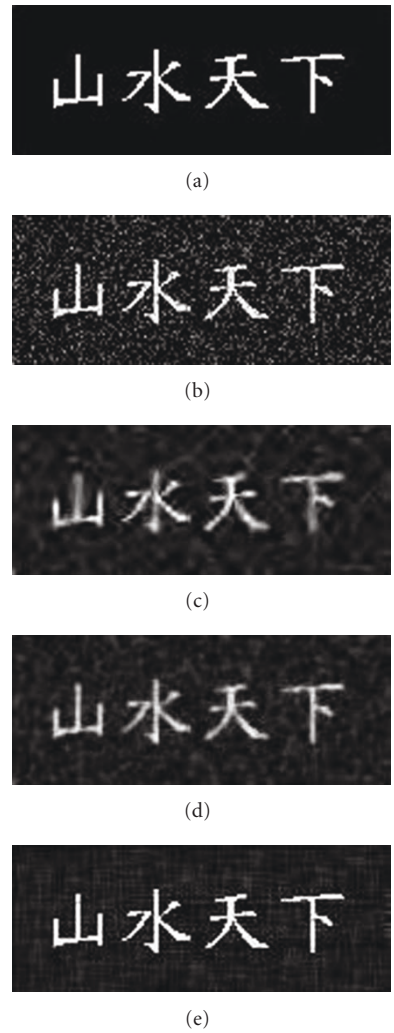
The functions $f(i,j)$ and $g(i,j)$ are original and denoised image, respectively. The numbers m and n are the size of an image.

3.2. Results and Discussions. Section 3.2.1 shows the denoised results of noise reduction test. Section 3.2.2 compares the denoised results of the proposed procedure with those of wavelet and ADF denoising methods. Section 3.2.3 describes the third stage in the proposed procedure. Section 3.2.4 shows the OCR application by the proposed procedure.



(a) (b) (c) (d) (e)

Figure 9: (a) "Number and Character", (b) noisy image with Gaussian noise with $\sigma = 0.01$, (c) wavelet, PSNR = 21.946, (d) ADF, PSNR = 22.285, and (e) the proposed procedure, PSNR = 22.495.



(a) (b) (c) (d) (e)

Figure 10: (a) Four Chinese words, (b) noisy image with Gaussian noise with $\sigma = 0.05$, (c) denoised image by wavelet, PSNR = 16.453, (d) denoised image by ADF, PSNR = 17.244, and (e) the proposed procedure, PSNR = 20.769.

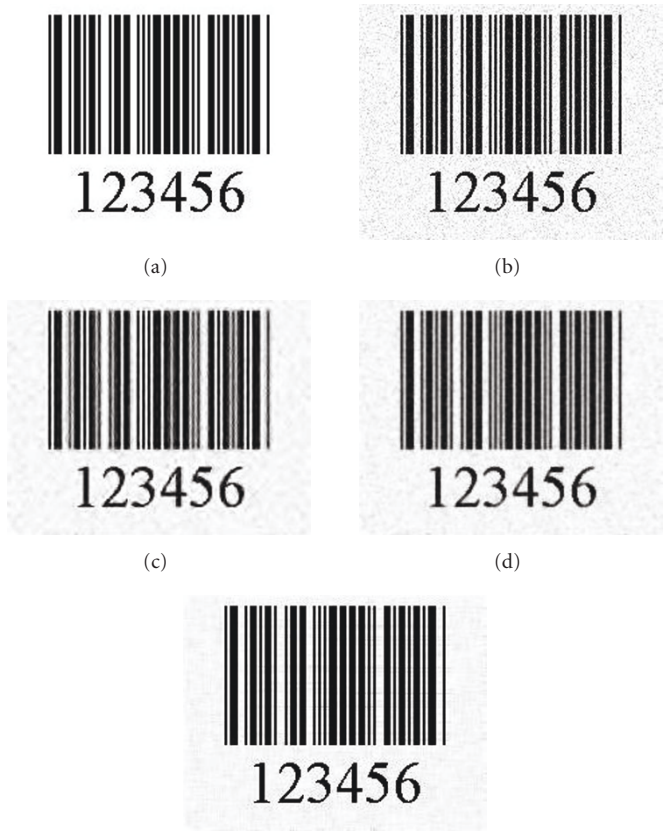
3.2.1. Noise Reduction Test. The experiments tested the synthetic image "Nine Square Regions" with Gaussian noise, Salt-and-Pepper noise, and Poisson noise, the result are shown in Figure 7.

The test image "Nine Square Regions" is a synthetic image shown in Figure 7(a). Figure 7(b) is the test image corrupted by adding Gaussian noise with variance 0.01. Figure 7(d) is the test image corrupted by adding salt and pepper noise with the density 0.05. Figure 7(f) is the test image corrupted by adding Poisson noise. Figures 7(c), 7(e), and 7(g) are images denoised by proposed procedure to preserve edges.

3.2.2. Comparison with Algorithms. In the experiments we have used the geometric images "Nine Square Regions", "Number and Character", "Chinese Words", and "BarCode" in order to demonstrate the edge preservation capability of the proposed procedure. Corresponding to three denoising methods, the values PSNR of all the denoised images are given in Table 1. These results of PSNR are 28.495, 22.495, 20.769, and 28.021 for four test images "Nine Square Regions", "Number and Character", "Chinese Words" and "BarCode" by the proposed procedure. The PSNR values of the proposed procedure are larger than those of the wavelet and ADF denoising method. The proposed procedure has better denoising capability.

The test image "Nine Square Regions" is a synthetic image shown in Figure 8(a). Figure 8(b) is the test image corrupted by adding Gaussian noise with variance 0.01. Figure 8(c) is the denoised image by wavelet denoising method. Figure 8(d) is the denoised image by ADF denoising method. Figure 8(e) is the denoised image by the proposed procedure. From the visual evaluation of images (c), (d), and (e), the proposed procedure has the best edge preservation.

The test image "Number and Character" as shown in Figure 9(a) is an image captured by camera from a car license plate. Figure 9(b) is the test image corrupted by adding Gaussian noise with a variance of 0.01. Figure 9(c) is the denoised image by wavelet denoising method. Figure 9(d) is the denoised image by ADF denoising method. Figure 9(e)



(a) (b) (c) (d) (e)

Figure 11: (a) BarCode image, (b) noisy image with Gaussian noise with $\sigma = 0.05$, (c) denoised image by wavelet, PSNR = 17.762, (d) denoised image by ADF, PSNR = 16.940, and (e) denoised image by the proposed procedure, PSNR = 28.021.

Table 1: RMSE and PSNR (dB) values of the denoised images by EPIDE, Wavelet, and ADF methods. There are four test images "Nine Square Regions", "Number and Character", "Chinese Words", and "BarCode".

Image	Nine Square Regions		Number and Character		Chinese Words		BarCode	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
Noise Image	21.809	21.358	23.885	20.568	40.206	16.045	18.197	22.931
EPIDE	9.5896	28.495	19.133	22.495	23.340	20.769	10.127	28.021
Wavelet	13.549	25.493	20.379	21.947	38.319	16.453	32.994	17.762
ADF	16.397	23.835	19.602	22.285	35.022	17.244	36.27	16.940

is the denoised image by the proposed procedure. From the visual evaluation of images (c), (d), and (e), the proposed procedure has the best edge preservation.

The test image "Chinese Words" is an image with four Chinese characters as shown in Figure 10(a). Figure 10(b) is the test image corrupted by adding Gaussian noise with a variance of 0.05. Figure 10(c) is the denoised image by wavelet denoising method. Figure 10(d) is the denoised image by ADF denoising method. Figure 10(e) is the denoised image by the proposed procedure. From the visual evaluation of images (c), (d), and (e), the proposed procedure has the best edge preservation.

The test image "BarCode" is an image without noise as shown in Figure 11(a). Figure 11(b) is the test image corrupted by adding Gaussian noise with variance 0.05. Figure 11(c) is the denoised image by wavelet denoising method. Figure 11(d) is the denoised image by ADF denoising method. Figure 11(e) is the denoised image by the proposed procedure. From the visual evaluationof images (c), (d), and (e), the proposed procedure has the best edge preservation.

3.2.3. Interactively Modified Edge Map. The edge-preserved performance of the EPIDE method depends on the edge detection. A noisy image is difficult to find all edges by Canny edge detector because of the noise interference. However, Canny edge detector has the same function as the Gaussian filter, but it is better to use any kind of denoising algorithm in the first stage of proposed algorithm. The following experiment is used to show how to modify the edge map by Canny

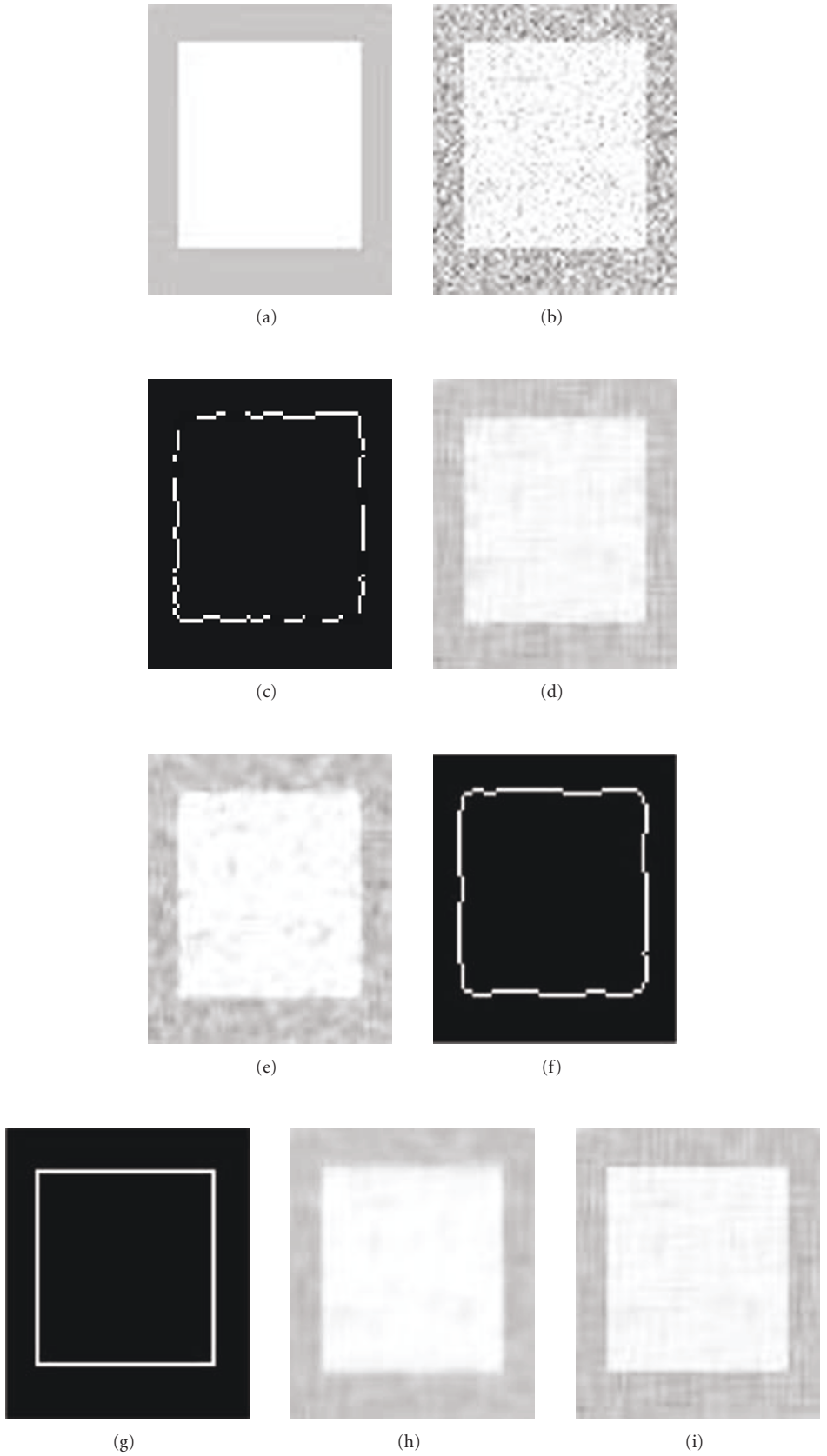


Figure 12: (a) Synthetic image, (b) noisy image with Gaussian noise with $\sigma = 0.05$, (c) edge map obtained from (b), (d) denoised image at final stage, (e) denoised image by neighborhood filters, (f) edge map obtained from (e), (g) modified edge map, (h) denoised image at final stage, and (i) denoised image at final stage.

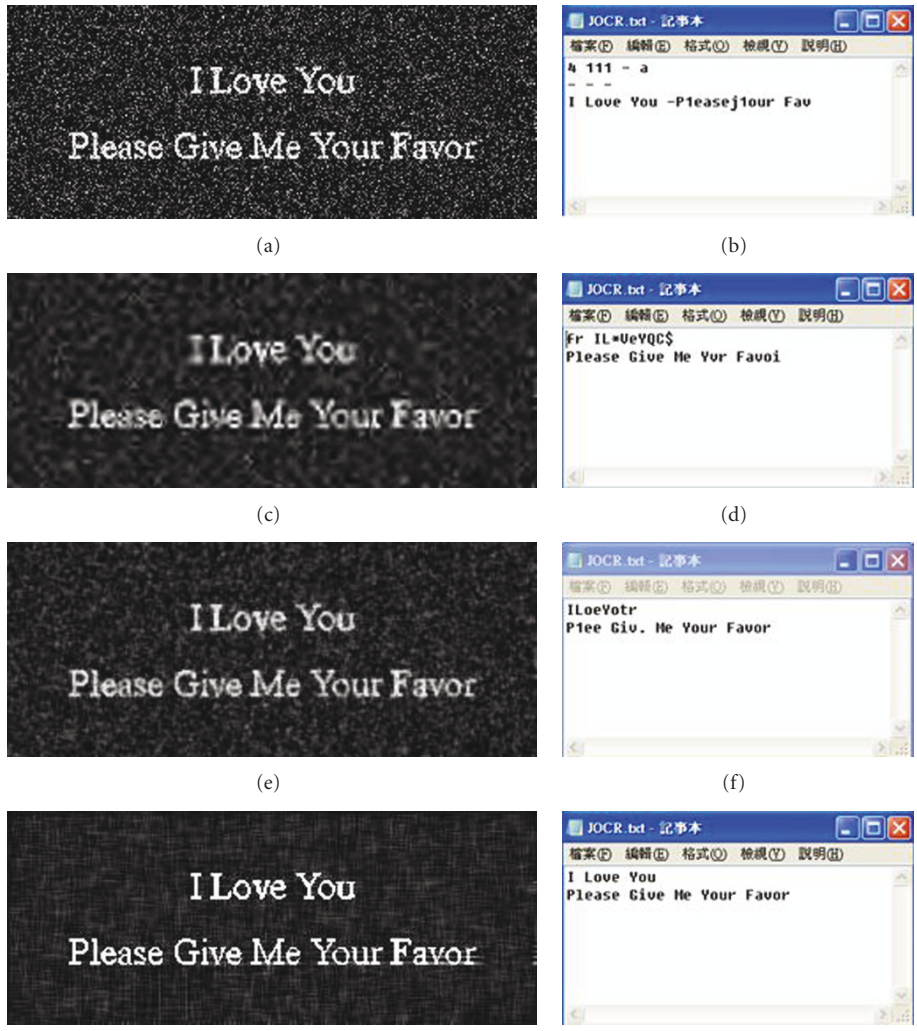


Figure 13: Left column is (a) noisy image and denoised image by (c) wavelet (e) ADF and (g) the proposed procedure, right column. (b), (d), (f), and (h) are OCR results.

edge detector and the results of denoised image by EPIDE method. Figure 12(a) is a synthetic image and Figure 12(b) is the noisy image with Gaussian noise with $\sigma = 0.05$. Figure 12(c) is an edge map obtained by Canny edge detector applied on Figure 12(b). Figure 12(d) is the denoised image by EPIDE method applied on Figure 12(b) with the edge map in Figure 12(c). Figure 12(e) is a denoised image by neighborhood filters applied on Figures 12(b) and 12(f) is an edge map by Canny edge detector applied on Figure 12(e). Figure 12(g) is a manually modified edge from the edge map of Figure 12(f). Figure 12(h) is the denoised image by EPIDE method with the edge map in Figure 12(g) applied on image in Figure 12(e), but Figure 12(i) is the denoised image by EPIDE method with the same edge map applied on image in Figure 12(b). By the visual evaluations, Figure 12(i) is better than Figure 12(h). From Table 2, the PSNR values of image of Figure 12(i) is higher than Figure 12(h). The first stage of proposed procedure can be an option for different images denoising cases.

Table 2: RMSE and PSNR (dB) values of denoised images B, D, H and I.

	B	D	H	I
RMSE	6.3711	3.7040	3.6321	3.1651
PSNR	32.049	36.757	36.928	38.123

* Character "B" represents Figure 12(b).
* Character "D" represents Figure 12(d).
* Character "H" represents Figure 12(h).
* Character "I" represents Figure 12(i).

3.2.4. The Proposed Procedure Applications. Now, the experiment is to demonstrate that the denoised images with a good edge-preserved can have better OCR result. The noise image is corrupted with Gaussian noise with variance of 0.08 as shown in Figure 12(a). In Figure 13, the left column the denoised image by Figures 13(c) wavelet 13(e) ADF 13(g) EPIDE method right column 13(b), 13(d), 13(f) and 13(h) are OCR results. Figure 13(a) is an image with Gaussian noise

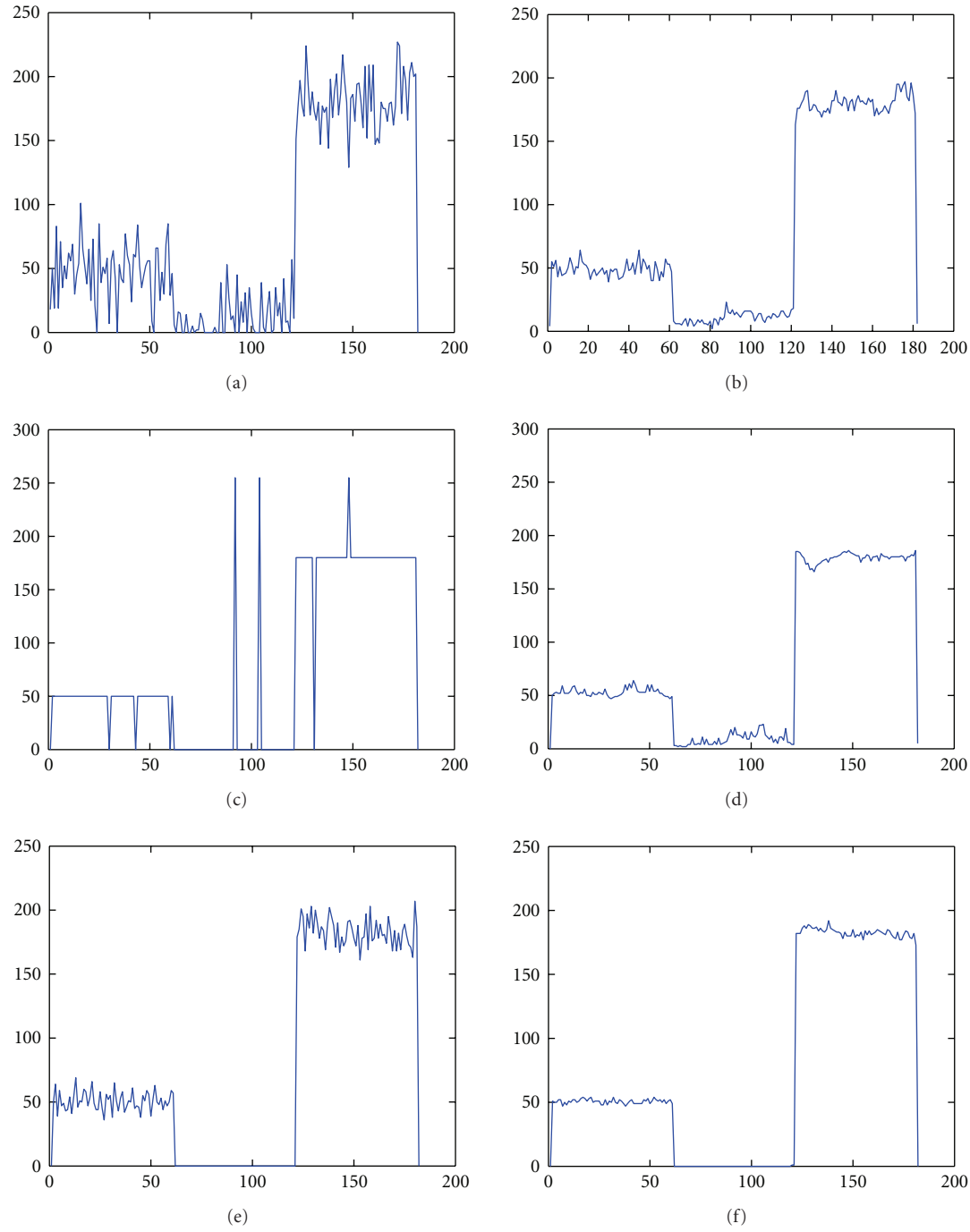


Figure 14: The 20th row of $u(x, y)$ of the image "Nine Square Regions" with the various noise. (a) Gaussian noise, (c) Salt and Pepper noise, (e) Poisson noise. The denoised results are, respectively, (b), (d), and (f)

and it's variance is 0.08. The denoised images are shown in Figures 13(c), 13(e) and 13(g) and they are denoised by wavelet, ADF and EPIDE methods. Figures 13(b), 13(d), 13(f) and 13(h) are OCR results of images in the left column. To evaluate the denoising performance of wavelet, ADF and EPIDE methods, it is suitable to use the character recognition software JOCR [16] to obtain the words in noised and denoised images. The experimental results show that the image denoised by EPIDE can have the best recognition results in Figure 13(h). All the characters in two sentences "I Love You" and "Please Give Me Your Favor" are correctly recognized in the image denoised by EPIDE method. There are some errors for character recognition are in Figures 13(b), 13(d) and 13(f). These results are obtained by JOCR software on the noisy and denoised images by wavelet and ADF methods.

The results of the above experiments demonstrate the effective of our proposed denoising procedure. In next

section, theoretical explanations are described to show why the proposed denoising procedure works well for any kind of noise.

3.2.5. The Denoising Capability of the Diffusion Equation. There are many types of image noise, such as Gaussian noise, Salt-and-pepper noise, Shot noise and Uniform noise. Noises are randomly distributed in image intensity value. At different pixels, the intensity values are independent of one another. For example, Gaussian noise, Shot noise and Uniform noise separately follow a Gaussian, Poisson, Fat-tail and Uniform distribution. The 20th row of $u(x, y)$ of the image "Nine Square Regions" in Figure 7 with Gaussian noise, Salt and Pepper noise and Poisson noise has three profiles as in Figures 14(a), 14(c), and 14(e). The three profiles of the denoised image are shown in Figures 14(b), 14(d), and 14(f). Three profiles as in Figures 14(a), 14(c), and 14(e) are the initial conditions of the diffusion equation (6). Three profiles as in Figures 14(b), 14(d), and 14(f) are the steady-state solutions of the steady-state diffusion equation (19)

$$K\nabla^2 u(x,t) + F(x) = 0. \tag{19}$$

The image denoising results in Figures 14(b), 14(d), and 14(f) are consistent with the theoretical explanation of (19).

4. Conclusion

The contribution of the paper is to propose a procedure to smooth the noisy or denoised image with any kind of denoising algorithm for desired edge preservation. To achieve preservation of designed edges, the inhomogeneous terms of the diffusion equation are formulated by the derivative of the Delta function. Fourier series is used to obtain the exact solution of the diffusion equation. The exact solution is a function of time and its value is the intensity of each pixel in an image. The Delta functions in the diffusion equation are used to locate the positions of edge pixels for each object in the image. To locate contour pixels for each object, it is necessary to use some image preprocessing methods and an edge detection method to find the edges of the object contours. Since the contour edges are not all found, then the user can interactively modify the edge map to keep the desired object contours. The proposed denoising method with edge preservation capability has the best OCR result in the experiment compared to the results from the wavelet denoising method and anisotropic diffusion filters.

Acknowledgment

The authors thank National Science Council (NSC) for partial financial support (NSC 97-2115-M-324-001) and (NSC 98-2115-M-324-001).

References

[1] W. E. Weideman, M. T. Manry, and H. C. Yau, "A comparison of nearest neighbor classifier and a neural network for numeric handprint character recognition," in *Proceedings of the IEEE International Conference on Neural Networks*, Washington, DC, USA, 1989.

[2] C. C. Tappert, "Recognition System for Run-on Handwritten Characters," US patent no. 4731857, International Business Machines Corporation, Armonk, NY, USA, March 1988.

[3] S.-H. Hahn, J.-H. Lee, and J.-H. Kim, "A study on utilizing OCR technology in building text database," in *Proceedings of the 10th International Workshop on Database and Expert Systems Applications*, pp. 582–586, 1999.

[4] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 2002.

[5] M. Welk and J. Weickert, "Semidiscrete and discrete well-posedness of shock filtering," in *Mathematical Morphology*, Springer, Berlin, Germany, 2005.

[6] S. Guillon, P. Baylou, M. Najim, and N. Keskes, "Adaptive nonlinear filters for 2D and 3D image enhancement," *Signal Processing*, vol. 67, no. 3, pp. 237–254, 1998.

[7] Y. B. Yuan, T. V. Vorburger, J. F. Song II, and T. B. Renegar, "A simplified realization for the Gaussian filter in surface metrology," in *Proceedings of the 10th International Colloquium on Surfaces*, M. Dietzsch and H. Trumpold, Eds., p. 133, Shaker, Chemnitz, Germany, January-February 2000.

[8] 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.

[9] S. K. Weeratunga and C. Kamath, "A comparison of PDE-based non-linear anisotropic diffusion techniques for image denoising," in *Image Processing: Algorithms and Systems II*, Proceedings of SPIE, Santa Clara, Calif, USA, January 2003.

[10] G. Gerig, O. Kubler, R. Kikinis, and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 221–232, 1992.

[11] R. P. Kanwal, *Generalized Functions: Theory and Applications*, Birkhäuser, Boston, Mass, USA, 3rd edition, 2004.

[12] G. B. Folland, *Fourier Analysis and Its Applications*, Brooks/Cole, Pacific Grove, Calif, USA, 1992.

[13] M. Sen, *Analytical Heat Transfer*, Department of Aerospace and Mechanical, Engineering University of Notre Dame, Notre Dame, Ind, USA, 2008.

[14] R. Bracewell, *The Fourier Transform and Its Applications*, McGraw-Hill, New York, NY, USA, 2nd edition, 1986.

[15] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[16] http://home.megapass.co.kr/~woosjung/Product_JOCR.html.