

## Research Article

# A Residual-Based Kernel Regression Method for Image Denoising

Jiefei Wang,<sup>1</sup> Yupeng Chen,<sup>1</sup> Tao Li,<sup>1</sup> Jian Lu,<sup>1</sup> and Lixin Shen<sup>2</sup>

<sup>1</sup>College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, China

<sup>2</sup>Department of Mathematics, Syracuse University, Syracuse, NY 13244, USA

Correspondence should be addressed to Jian Lu; [jianlu1979@163.com](mailto:jianlu1979@163.com)

Received 7 October 2015; Revised 4 January 2016; Accepted 18 January 2016

Academic Editor: Kacem Chehdi

Copyright © 2016 Jiefei Wang 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.

We propose a residual-based method for denoising images corrupted by Gaussian noise. In the method, by combining bilateral filter and structure adaptive kernel filter together with the use of the image residuals, the noise is suppressed efficiently while the fine features, such as edges, of the images are well preserved. Our experimental results show that, in comparison with several traditional filters and state-of-the-art denoising methods, the proposed method can improve the quality of the restored images significantly.

## 1. Introduction

In image processing, noise reduction with preserving fine features is a long-standing research problem. Many applications such as image segmentation and medical image analysis require effective noise suppression to obtain reliable results. In recent years, many efficient mathematical methods such as kernel regression based methods and locally adaptive based methods for the purpose of image denoising have been developed in the literature (see [1–10] and the references cited therein). For example, Tomasi and Manduchi proposed the well-known bilateral filtering for image filtering in [1], which exploits the local structural patterns to regularize the image filtering procedure and assume that images are locally smooth except at edges. The bilateral filter can preserve certain details of images, while more information inherent in the image still remains to be discovered and utilized. Tschumperlé [2] proposed a common framework for image filtering which is based on the iterative local diffusion in the image guided by the local structure tensor. The method makes use of the local gradients of the image to control the shape of the kernel and hence to keep texture details of the image efficiently. However, for a very noisy image, the gradients of the image are likely to be corrupted by the noise and thus might raise huge errors in terms of gradient estimator in smooth areas,

which might generate negatively “ripple” phenomena. To overcome the drawbacks mentioned above, Takeda et al. [3] further proposed an iterative steering kernel regression filter, of which an iterative filtering procedure is used to exploit the output (less noisy) image of each iteration to improve the local orientation estimates used in the kernel in the next iteration. The implementation of the method is very time-consuming, even exceeding that of nonlocal methods [11, 12]; and it shows instability in estimating the local gradients of the image under high-level noisy case. López-Rubio and Florentín-Núñez [4] proposed to adopt a general kernel regression framework to 3D MR image denoising. The method is rooted on a zeroth-order 3D kernel regression, which calculates a weighted average of the pixels over a regression window; and the weights are obtained from the similarities among small sized feature vectors associated to each pixel. One common feature for these kernel regression methods is the exploration of the local structural regularity properties in natural images/videos. Based on this observation, the kernel regression based methods (or combined with other methods) are also successfully applied to image and video deblurring, upscaling, interpolation, fusion [3], superresolution [6, 13–16], registration [17], JPEG image deblocking [18], and so forth. On the other hand, the residual-based image denoising methods have also been studied. For

instance, in [19], the corrupted image was first denoised with the total variation minimization algorithm; then the residual image was denoised with the adaptive Wiener filter; and finally, based on a statistical test, the filtered residuals were added back to the denoised image. Similarly, in [20], the authors first adopted nonlocal means filter to denoise the degraded image and added the final filtered residual image to the denoised image to recover the lost image details. In [21], the image denoising was implemented in the wavelet transform domain. In the method, based on the kurtosis statistic of the residual image, a criterion was proposed for determining wavelet coefficient thresholds.

In this paper, different from the residual-based methods in [19–21], we propose a new residual-based denoising method, which combines the bilateral filter and the local structure adaptive kernel filter by making use of their respective advantages. The proposed method can prevent the fine details loss caused by the bilateral kernel regression method and avoid image ripple phenomena produced from the local structure adaptive kernel method. Numerical experiments demonstrate that, for images contaminated with Gaussian noise, the performance of our combined approach is generally superior to those of some traditional and state-of-the-art restoration methods in terms of the image quality measured by PSNR and SSIM [22].

## 2. Classic Kernel Regression Methods

In this section, we briefly review bilateral kernel and structural adaptive kernel methods which will be used in the development of our new method later.

Classical regression methods for image processing rely on a specific model of the signal of interest and calculate the parameters of the model in the presence of noise. Compared to the parametric methods, kernel regression methods [23, 24] rely on the data itself to dominate the structure of the model. Generally, the traditional kernel regression methods are based on solving the following optimization problem:

$$\tilde{y}(\mathbf{x}) = \arg \min_{\beta_0} \sum_{j \in \mathcal{N}(\mathbf{x})} (y_j - \beta_0)^2 K(\mathbf{x}_j - \mathbf{x}), \quad (1)$$

where  $\mathcal{N}(\mathbf{x})$  is a set that consists of the index numbers of the neighbors of  $\mathbf{x}$ ;  $y_j$  is the pixel observation at  $\mathbf{x}_j$ ;  $\tilde{y}(\mathbf{x})$  denotes the estimate of the pixel at location  $\mathbf{x}$  of the image.  $K(\mathbf{x}_j - \mathbf{x})$  is a generic kernel (e.g., Gaussian, exponential, or other forms) which is used as the penalty for distance between  $\mathbf{x}_j$  and  $\mathbf{x}$ , so that the influence of  $\mathbf{x}$  on  $\mathbf{x}_j$  decreases as the distance between these two points increases. Considering  $\beta_0$  as the first coefficient of some regression function on the coordinates  $\mathbf{x}$ , (1) is essentially a zero-order estimation. In order to approximate the image local structure better, higher order estimation was used in [3, 25]. Specifically, assuming that the image is locally smooth to some order, we can rely on the local expansion of the function using the Taylor series

$$y(\mathbf{x}_i) = \beta_0 + \beta_1^\top (\mathbf{x}_i - \mathbf{x}) + \beta_2^\top \mathbf{tril} \{ (\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^\top \} + \dots, \quad (2)$$

where  $\mathbf{tril}$  denotes the operator that extracts the lower triangular part of a matrix and stacks it to a column vector. Then, the optimal regression coefficients  $\{\beta_k\}_{k=0}^N$  can be calculated from the following optimization problem:

$$\{\beta_k\}_{k=0}^N = \arg \min_{\{\beta_k\}} \sum_{j \in \mathcal{N}(\mathbf{x})} \left[ y_j - \beta_0 - \beta_1^\top (\mathbf{x}_j - \mathbf{x}) - \beta_2^\top \mathbf{tril} \{ (\mathbf{x}_j - \mathbf{x})(\mathbf{x}_j - \mathbf{x})^\top \} - \dots \right] K(\mathbf{x}_j - \mathbf{x}). \quad (3)$$

Although all regression coefficients  $\{\beta_k\}_{k=0}^N$  can be estimated effectively, the first element  $\beta_0$  is the desired pixel value estimation at  $\mathbf{x}$ . For the details of computing  $\beta_0$ , one can refer to [3, 23, 24].

As mentioned above, the traditional kernel methods simply take the distance of the pixels into consideration. That is, large weight is assigned to nearby pixels, while smaller weights are assigned to farther pixels. This may lead to obscurity of the image, especially on the edges. To relieve the problem, Tomasi and Manduchi [1] proposed the bilateral filter as solving the following optimization problem:

$$\tilde{y}(\mathbf{x}) = \arg \min_{\beta_0} \sum_{j \in \mathcal{N}(\mathbf{x})} (y_j - \beta_0)^2 K(\mathbf{x}_j - \mathbf{x}, y_j - y), \quad (4)$$

where  $K(\mathbf{x}_j - \mathbf{x}, y_j - y) = K_1(\mathbf{x}_j - \mathbf{x}) \cdot K_2(y_j - y)$ . Here, the kernel functions  $K_1$  and  $K_2$  are used for penalizing the spatial distance between the pixel of interest  $\mathbf{x}$  and its neighbors  $\{\mathbf{x}_j\}$  and the radiometric “distance” between the corresponding pixels  $y$  and  $\{y_j\}$ . The main idea of bilateral kernel is that, when calculating weights, it considers both the pixel distance and space distance between the points at  $\mathbf{x}$  and  $\mathbf{x}_j$ . Then, due to considerable variation of the pixel values around the image edges, the values of the associated weights  $K_2$  are relatively small, which contributes to preserving the edges of the image. In addition, the generalizations of the bilateral filter, including the choices of various kernels and higher order estimation, were reported in [10, 26, 27].

Although the bilateral filter can solve the indistinction problem of image margins to some extent, only distance information of the pixels is used, which might make a loss of details in the textures of image margins. Tschumperlé [2] further proposed to utilize correlation information of pixels to estimate the local directions of the image. And he constructed a local structure adaptive kernel as follows:

$$K(\mathbf{x}) = \frac{1}{2\pi\delta^2 \sqrt{\det(T)}} \cdot \exp\left(-\frac{1}{2\delta^2} \mathbf{x}^\top T^{-1} \mathbf{x}\right), \quad (5)$$

where  $T$  is the diffusion tensor controlling the spatial structure of the kernel and  $\delta$  is the parameter that controls the bandwidth of the structural kernel. In order to obtain  $T$ ,



(a) Lena



(b) Pirate



(c) Boat



(d) House

FIGURE 1: Four test images.

we construct a structure tensor, which can be defined as follows:

$$S(\mathbf{x}) = \begin{bmatrix} \sum_{j \in \mathcal{N}(\mathbf{x})} w(\mathbf{x}_j - \mathbf{x}) dx(\mathbf{x}_j)^2 & \sum_{j \in \mathcal{N}(\mathbf{x})} w(\mathbf{x}_j - \mathbf{x}) dx(\mathbf{x}_j) dy(\mathbf{x}_j) \\ \sum_{j \in \mathcal{N}(\mathbf{x})} w(\mathbf{x}_j - \mathbf{x}) dx(\mathbf{x}_j) dy(\mathbf{x}_j) & \sum_{j \in \mathcal{N}(\mathbf{x})} w(\mathbf{x}_j - \mathbf{x}) dy(\mathbf{x}_j)^2 \end{bmatrix}, \quad (6)$$

where  $w(\cdot)$  is a Gaussian kernel and  $dx(\mathbf{x})$  and  $dy(\mathbf{x})$  are the  $x$ -gradient and  $y$ -gradient at coordinates  $\mathbf{x}$ , respectively. Here we use Prewitt operator [23, 24] to estimate the gradient. After performing eigenvalue decomposition on  $S(\mathbf{x})$ , we get

$$S(\mathbf{x}) = [\mathbf{u} \ \mathbf{v}] \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \begin{bmatrix} \mathbf{u}^\top \\ \mathbf{v}^\top \end{bmatrix}, \quad (7)$$

where  $\alpha_1, \alpha_2$  are the eigenvalues of  $S(\mathbf{x})$  with  $\alpha_1 \leq \alpha_2 \neq 0$ , reflecting the strength of the gradient along each eigenvector

direction and  $\mathbf{u}$  and  $\mathbf{v}$  are the eigenvectors of  $S(\mathbf{x})$ . According to methods for finding diffusion tensor described in [2, 24], we set

$$T = f \mathbf{u} \mathbf{u}^\top + \frac{1}{f} \mathbf{v} \mathbf{v}^\top, \quad (8)$$

where

$$f = \frac{\alpha_2 + \gamma}{\alpha_1 + \gamma}. \quad (9)$$



FIGURE 2: Comparison of the three denoising methods: (a) noise image with  $\sigma = 20$ , (b) bilateral filter, (c) structure adaptive filter, and (d) the proposed method.

Here,  $\gamma \geq 0$  is a sensible parameter so that the shape of the local structure adaptive kernel will respond to the local gradient of the image.

### 3. A Residual-Based Joint Denoising Method

Bilateral filter and structure adaptive denoising methods described above can restore the noisy images efficiently. For example, the original noise-free image “Lena” shown in Figure 1(a) is degraded by an additive white Gaussian noise with noise variances  $\sigma^2 = 20^2$ , as shown in Figure 2(a). We test the two methods on the Noisy “Lena” image, respectively. As we can see, the bilateral filter with PSNR = 27.56 dB, SSIM = 0.86 and the structure adaptive denoising method with PSNR = 30.20 dB, SSIM = 0.84 exhibit acceptable image quality as shown in Figures 2(b) and 2(c), respectively. On the other hand, as shown in the figure, the bilateral filter can smooth the Noisy “Lena” efficiently but erase certain details of the image in the meanwhile; and the local self-adaptive

method can deliver the fine features, such edges, of the image, whereas it might generate image ripples caused by noise in the smooth area. Based on this observation, we here propose a new method, that is, a residual feedback-based joint image denoising method using bilateral filter and structure adaptive methods.

Firstly, we define an image residual model as follows:

$$\hat{\epsilon}_i = g(x_i) - y(x_i), \quad i = 1, \dots, M, \quad (10)$$

where  $g(x_i)$  is the pixel value at point  $x_i$  of the noisy image with  $M$  pixels;  $y(x_i)$  is the pixel value at point  $x_i$  of the denoised image; and  $\hat{\epsilon}_i$  is the estimate of the location disturbance of  $x_i$ , namely, image residual. We assume that image residual  $\hat{\epsilon}_i$  still obeys Gaussian distribution with mean of 0 and a low-level variance; that is,

$$E(\hat{\epsilon}_i) = 0, \quad i = 1, \dots, M. \quad (11)$$

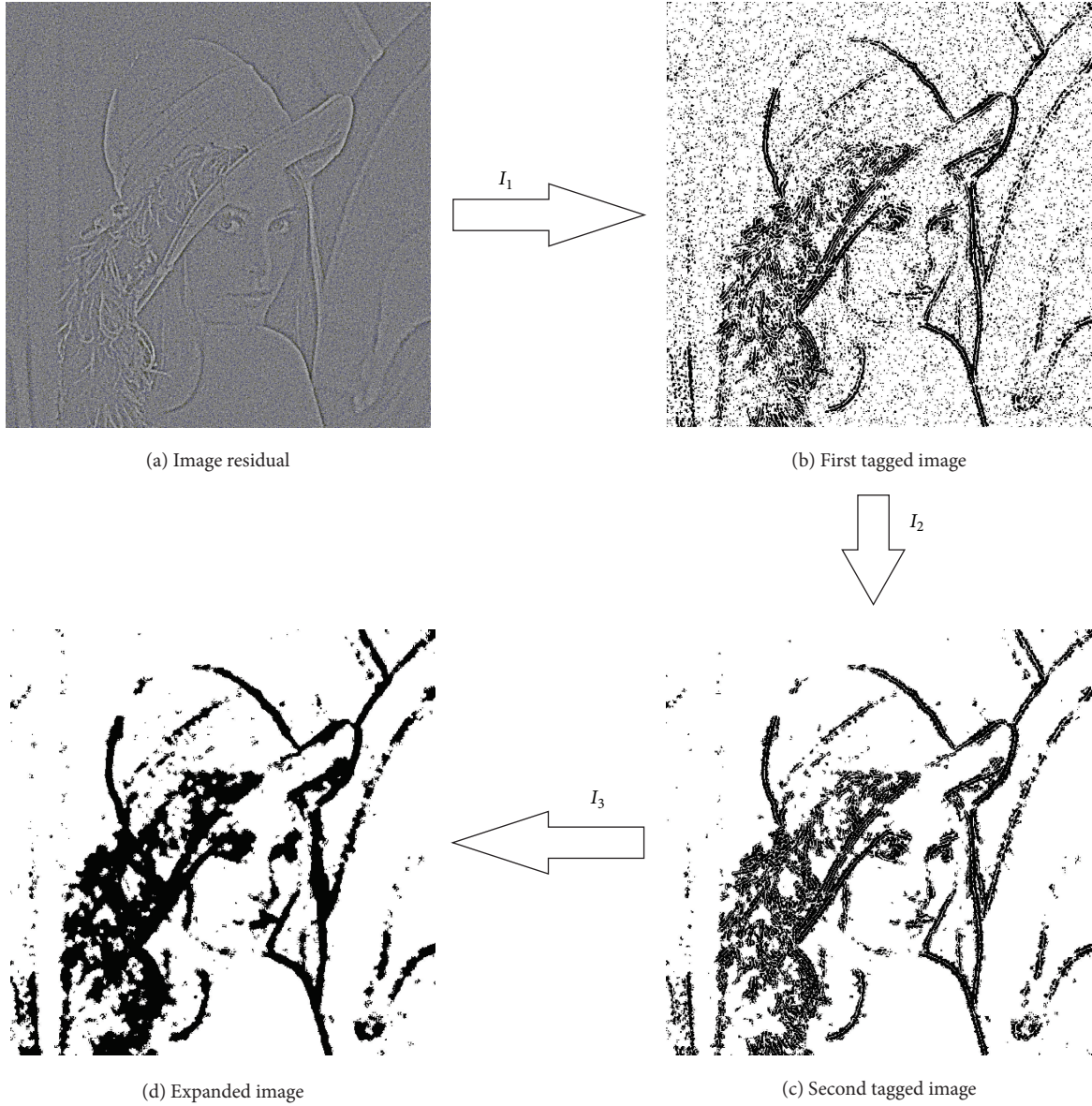


FIGURE 3: Tagging procedure of bilateral filter.

We next define a criterion for testing image residual as follows:

$$G(x_i) = \sum_{j \in N_{h_1}(x_i)} \hat{\epsilon}_j, \quad i = 1, \dots, M, \quad (12)$$

where  $N_{h_1}(x_i)$  denotes a neighbor of  $x_i$  with the radius being  $h_1$ . As the expectation of image disturbance is 0, the criterion  $G(\cdot)$  should not be too large or too small in a flat area. Thus we can tag abnormal points by

$$I_1(x_i) = \begin{cases} 1, & G(x_i) < \theta_1 \text{ or } G(x_i) > \theta_2 \\ 0, & \text{otherwise,} \end{cases} \quad (13)$$

where  $\theta$  is a preprescribed value, used as an indicator of the toughness of tagging. As an example, the image residual

$\epsilon$  between Figures 2(a) and 2(b) is shown in Figure 3(a) and  $I_1$  shown in Figure 3(b) is obtained by using (13). From Figure 3(b), we can see that the details are tagged to some extent, but the noise still remains. So, for a tagged residual figure  $I_1(x_i)$ , we need to “filter” it a second time. Define

$$I_2(x_i) = \begin{cases} 1, & \sum_{t \in N_{h_2}(x_i)} I_1(t) > \beta_1, \quad I_1(x_i) = 1, \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

Here, similarly to  $N_{h_1}(x_i)$ ,  $N_{h_2}(x_i)$  is a neighbor of  $x_i$  and  $h_2$  is its radius; and  $\beta_1$  is a preselected threshold. The main idea of designing  $I_2$  in (14) is based on the fact that the useful information, such as edges, in the residual tagged image  $I_1(x)$



FIGURE 4: Comparison of the eleven denoising methods: (a) Noisy “Lena” image with  $\sigma = 30$ ; (b) bilateral filter; (c) structure adaptive kernel; (d) the proposed method; (e) Wiener filter; (f) Kuwahara filter; (g) NCD method; (h) BiShrink; (i) NeighShrinkSURE; (j) OWT-SURELET; (k) FP<sup>2</sup>O-ITV; (l) BM3D.

tends to be clustered while the noise-tags distribute randomly in the image  $I_1(x)$ . Thus we can determine whether a given pixel, say  $x_i$ , is informative or noisy by simply calculating  $I_2(x_i)$ . As shown in Figure 3(c), the noise scattered in the tagged image has been eliminated visibly after the second filtering process. In addition, it is worth noting that there exist some interrupted points in image  $I_2(x)$ , which might result from the fact that some useful and informative areas are covered by the noise. So, we further expand the tagged information by linking those unconnected abutting areas with following function:

$$I_3(x_i) = \begin{cases} 1, & \sum_{t \in N_{h_3}(x_i)} I_2(t) > \beta_2 \text{ or } I_2(x_i) = 1, \\ 0, & \text{otherwise.} \end{cases} \quad (15)$$

After the operation using  $I_3$ , we get a binary image shown as in Figure 3(d), which just tags useful information, such as edges, and meaningless noise has been eliminated clearly.

According to the analysis described above,  $I_3$  can clearly indicate the indistinct margins caused by bilateral denoising. On the other hand, as discussed in Section 2, the structure adaptive kernel based methods can protect fine features, such as edges, of an image as well as possible. So, we can avoid the appearance of indistinct margins and ripples in the restored image by combining the favorite properties of these two methods. Detailedly, let  $y_1$  and  $y_2$  be the images denoised by using structure adaptive kernel method and bilateral filter, respectively. Then the resulting image  $y_3$  can be formulated as follows:

$$y_3(x) = I_3(x) y_1(x) + (1 - I_3(x)) y_2(x). \quad (16)$$

It is worth noting that in order to get  $y_3$ , instead of filtering the whole noisy image to obtain  $y_1$  by structure adaptive kernel method, we only need to get some parts of  $y_1$  by filtering those pixels satisfying the first restriction in (15), which can reduce the computation complexity greatly. In addition, in order



FIGURE 5: Comparison of the eleven denoising methods: (a) Noisy “Pirate” image with  $\sigma = 30$ ; (b) bilateral filter; (c) structure adaptive kernel; (d) the proposed method; (e) Wiener filter; (f) Kuwahara filter; (g) NCD method; (h) BiShrink; (i) NeighShrinkSURE; (j) OWT-SURELET; (k) FP<sup>2</sup>O-ITV; (l) BM3D.

to erase those image edges caused by splicing, a smoothing formula is defined as

$$I'_3(x_i) = \begin{cases} 0.3, & \sum_{t \in N_2(x_i)} I_3(t) \neq \#N_2(x_i), \quad I_3(x_i) = 1, \\ I_3(x_i), & \text{otherwise,} \end{cases} \quad (17)$$

where  $\#A$  denotes the number of elements of set  $A$ . Thus we can replace  $I'_3$  with  $I_3$  in (16) to gain a smoothed splicing image.

In summary, the main steps of our proposed method are as follows:

- (1) Restore the noisy image  $g$  by bilateral filter and get the restored image  $y_2$ .
- (2) Calculate the residual  $\hat{\epsilon}$  by (10).
- (3) Compute  $I'_3$  by (13), (14), (15), and (17).

- (4) Apply structure adaptive method to denoise those pixels  $x_i$  of the noisy image  $g$  satisfying the first restriction of (15) and get  $y_1$ .

- (5) Get the restored image  $y_3$  by

$$y_3(x) = I'_3(x) y_1(x) + (1 - I'_3(x)) y_2(x). \quad (18)$$

Figure 2(d) illustrates the restored “Lena” image by using our proposed method. This image has PSNR = 31.65 dB and SSIM = 0.92. As we can see, the proposed method performs remarkably better than the bilateral filter and structure adaptive kernel method. Furthermore, the execution time of our proposed method with 55.16 seconds is much faster than that of the structure adaptive kernel method with 78.12 seconds. This can be explained by the fact that our proposed method only performs structure filter on tagged points indicated by  $I_3$ , instead of the whole image, leading to decrease of the computation complexity.



FIGURE 6: Comparison of the eleven denoising methods: (a) Noisy “Lena” image with  $\sigma = 40$ ; (b) bilateral filter; (c) structure adaptive kernel; (d) the proposed method; (e) Wiener filter; (f) Kuwahara filter; (g) NCD method; (h) BiShrink; (i) NeighShrinkSURE; (j) OWT-SURELET; (k) FP<sup>2</sup>O-ITV; (l) BM3D.

#### 4. Numerical Experiments

In this section, we present numerical results on simulated and read data to demonstrate the performance of the proposed method and compare it with several traditional methods and some existing state-of-the-art methods, which include bilateral filter, structure adaptive kernel method described in the previous Section 2, Wiener filter [28], Kuwahara filter [29], nonlinear complex diffusion (NCD) processes [30], BiShrink [31], NeighShrinkSURE [32], OWT-SURELET [33], FP<sup>2</sup>O-ITV [34], and BM3D [35]. We note that all algorithms are implemented by using MATLAB (version 7.10) and performed on Windows 7 with CPU E3V2 3.3 GHz and 8 GB memory. The denoising image quality is measured by *Peak Signal-to-Noise Ratio* (PSNR) defined by

$$\text{PSNR} = 10 \log_{10} \left( \frac{255^2}{(1/N) \sum_{i=1}^N (f_i - \hat{f}_i)^2} \right), \quad (19)$$

where  $f_i$  and  $\hat{f}_i$  are the pixel intensity of the original image and the retrieved image, respectively.  $N$  denotes the total

number of pixels. Moreover, we here also use the Structural SIMilarity (SSIM) Index introduced by [22] to measure the similarity between the original image  $f$  and the restored image  $\hat{f}$ .

Note that in this paper we use Gaussian type kernel functions in the three kernel based methods. For parameters in bilateral filter, corresponding to  $K_1$  and  $K_2$ , we choose  $\delta_1 = \delta_2 = 3$  and the window size  $h = 4$ ; for structure adaptive kernel method, we choose  $\delta = 3$ ,  $h = 5$ ; and for our proposed joint method, we choose  $h_1 = 3$  in (12),  $\theta_1$  and  $\theta_2$ , respectively, being 15% quantile and 85% quantile of the residuals in the set  $N_{h_1}(\cdot)$  in (13),  $\rho_1 = \rho_2 = 15$  in (14) and (15), and  $h_1 = h_2 = h_3 = 3$  in all the three equations. For the other test denoising methods, we experimentally select the parameters according to the suggestions discussed in the literature. For example, in the Wiener filter [28] and Kuwahara filter [29], we set the window size being  $6 \times 6$  and  $5 \times 5$ , respectively; in the BiShrink [31], we use the separable discrete wavelet transform with Daubechies' least asymmetric compactly supported wavelet with eight vanishing moments with four scales; also in the NeighShrinkSURE [32] and OWT-SURELET [33]



TABLE 1: A comparison of different denosing methods.

Image	Method	$\sigma = 20$		$\sigma = 25$		$\sigma = 30$		$\sigma = 35$		$\sigma = 40$	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Lena	BF	27.56	0.86	27.36	0.85	27.12	0.83	26.83	0.81	26.51	0.79
	SAF	30.20	0.84	28.73	0.78	27.43	0.73	26.30	0.68	25.30	0.63
	<i>Proposed</i>	31.65	0.92	30.64	0.89	29.67	0.87	28.78	0.84	27.95	0.81
	Wiener [28]	28.98	0.85	28.57	0.82	27.71	0.80	26.98	0.78	26.27	0.76
	Kuwahara [29]	27.16	0.82	26.14	0.77	25.12	0.72	24.26	0.68	23.44	0.63
	NCD [30]	29.94	0.86	28.99	0.82	27.92	0.76	26.94	0.71	25.90	0.67
	BiShrink [31]	31.17	0.90	30.16	0.88	29.41	0.87	28.64	0.84	28.18	0.83
	NeighShrinkSURE [32]	31.52	0.91	30.53	0.89	29.69	0.87	29.01	0.85	28.40	0.84
	OWT-SURELET [33]	31.37	0.91	30.35	0.89	29.56	0.87	28.88	0.85	28.31	0.84
	FP <sup>2</sup> O-ITV [34]	30.78	0.91	30.04	0.88	28.72	0.84	26.79	0.76	24.81	0.67
BM3D [35]	<b>33.02</b>	<b>0.93</b>	<b>32.05</b>	<b>0.92</b>	<b>31.25</b>	<b>0.91</b>	<b>30.53</b>	<b>0.89</b>	<b>29.85</b>	<b>0.87</b>	
Pirate	BF	26.04	0.81	25.91	0.80	25.74	0.78	25.54	0.77	25.29	0.75
	SAF	29.03	0.86	27.85	0.82	26.77	0.77	25.78	0.73	24.88	0.69
	<i>Proposed</i>	29.15	0.89	28.50	0.87	27.83	0.85	27.17	0.82	26.52	0.80
	Wiener [28]	28.41	0.85	27.21	0.83	26.53	0.81	25.93	0.78	25.31	0.76
	Kuwahara [29]	26.48	0.83	25.54	0.79	24.63	0.75	23.77	0.70	23.03	0.66
	NCD [30]	28.50	0.87	27.88	0.84	27.12	0.80	26.34	0.76	25.52	0.72
	BiShrink [31]	29.01	0.88	28.09	0.84	27.29	0.81	26.69	0.79	26.17	0.76
	NeighShrinkSURE [32]	29.30	0.88	28.30	0.85	27.51	0.83	26.90	0.80	26.36	0.78
	OWT-SURELET [33]	29.41	0.89	28.44	0.86	27.69	0.83	27.08	0.81	26.57	0.78
	FP <sup>2</sup> O-ITV [34]	29.05	0.87	28.61	0.86	27.65	0.83	26.09	0.77	24.34	0.71
BM3D [35]	<b>30.20</b>	<b>0.91</b>	<b>29.17</b>	<b>0.89</b>	<b>28.36</b>	<b>0.87</b>	<b>28.19</b>	<b>0.85</b>	<b>27.61</b>	<b>0.83</b>	
Boat	BF	24.95	0.79	24.83	0.78	24.69	0.76	24.51	0.75	24.30	0.73
	SAF	28.79	0.86	27.66	0.81	26.60	0.76	25.61	0.72	24.70	0.67
	<i>Proposed</i>	28.85	0.88	28.18	0.87	27.51	0.84	26.85	0.82	26.18	0.79
	Wiener [28]	28.21	0.85	26.76	0.82	26.07	0.80	25.43	0.77	24.90	0.74
	Kuwahara [29]	25.90	0.82	25.04	0.77	24.23	0.73	23.44	0.69	22.72	0.65
	NCD [30]	27.93	0.86	27.34	0.82	26.65	0.78	25.80	0.74	24.94	0.69
	BiShrink [31]	29.18	0.89	28.15	0.86	27.32	0.83	26.72	0.81	26.08	0.78
	NeighShrinkSURE [32]	29.41	0.89	28.36	0.87	27.52	0.84	26.84	0.81	26.26	0.79
	OWT-SURELET [33]	29.47	0.89	28.44	0.86	27.63	0.84	26.97	0.81	26.41	0.79
	FP <sup>2</sup> O-ITV [34]	28.82	0.87	28.32	0.86	27.25	0.83	25.86	0.77	24.30	0.70
BM3D [35]	<b>30.76</b>	<b>0.92</b>	<b>29.80</b>	<b>0.90</b>	<b>28.99</b>	<b>0.88</b>	<b>28.43</b>	<b>0.86</b>	<b>27.74</b>	<b>0.84</b>	
House	BF	30.77	0.88	30.32	0.86	29.81	0.84	29.26	0.82	28.68	0.80
	SAF	31.41	0.80	29.59	0.73	28.11	0.67	26.87	0.62	25.80	0.57
	<i>Proposed</i>	34.14	0.91	32.62	0.88	31.28	0.85	30.08	0.82	29.01	0.78
	Wiener [28]	31.57	0.89	30.33	0.86	29.34	0.82	28.33	0.78	27.49	0.75
	Kuwahara [29]	28.25	0.80	27.01	0.74	25.88	0.68	24.93	0.63	24.01	0.58
	NCD [30]	32.08	0.84	30.50	0.78	29.10	0.72	27.73	0.66	26.46	0.61
	BiShrink [31]	34.25	0.90	33.15	0.88	32.24	0.86	31.36	0.84	30.87	0.83
	NeighShrinkSURE [32]	34.90	0.91	33.75	0.89	32.81	0.87	32.00	0.86	31.28	0.84
	OWT-SURELET [33]	34.30	0.91	33.15	0.89	32.33	0.87	31.46	0.86	30.79	0.85
	FP <sup>2</sup> O-ITV [34]	33.88	0.91	32.62	0.89	30.34	0.82	27.76	0.71	25.37	0.61
BM3D [35]	<b>36.47</b>	<b>0.93</b>	<b>35.60</b>	<b>0.92</b>	<b>34.67</b>	<b>0.91</b>	<b>34.03</b>	<b>0.90</b>	<b>33.30</b>	<b>0.88</b>	

methods, the wavelet type used in the method is “sym8” with the number of wavelet decomposition levels equal to four, and we assume that the noise variances were known; in the algorithm FP<sup>2</sup>O-ITV [34], we fix  $\mu = 1/0.05$  and  $\lambda = 0.25$  for convenience.

We first choose the  $512 \times 512$  standard test images, “Lena,” “Pirate,” “Boat,” and “House” shown as in Figure 1, in our experiments. They were contaminated by the Gaussian random noise with standard deviation varying from  $\sigma = 20$  to  $\sigma = 40$  and the step size being 5. As we can



FIGURE 7: Comparison of the eleven denoising methods: (a) Noisy “Pirate” image with  $\sigma = 40$ ; (b) bilateral filter; (c) structure adaptive kernel; (d) the proposed method; (e) Wiener filter; (f) Kuwahara filter; (g) NCD method; (h) BiShrink; (i) NeighShrinkSURE; (j) OWT-SURELET; (k) FP<sup>2</sup>O-ITV; (l) BM3D.

see from Table 1, when the quality of the restored images is measured by PSNR and SSIM, the proposed method significantly performs better than the traditional methods, bilateral filter, structure adaptive kernel method, Wiener filter, and Kuwahara filter; and it generally outperforms the NCD, BiShrink, and FP<sup>2</sup>O-ITV, especially for high-level noisy cases (e.g.,  $\sigma \geq 30$ ). When compared with NeighShrinkSURE and OWT-SURELET methods, it is also promising. In addition, when only focusing on the SSIM-values of the table, our proposed method possesses particularly favorable results in comparison with all others except BM3D, which can be explained by the fact that the proposed method inherits the advantages of the bilateral filter and the local structure adaptive kernel filter, coinciding with our aforementioned analysis in Section 3. Note that the performance of BM3D nonlocal method is superior to all the other methods.

Figures 4–7 illustrate the comparative results of the corrupted “Lena” and “Pirate” images with noise-levels,  $\sigma = 30$  and  $\sigma = 40$ , respectively. As shown in the figures,

the proposed method can preserve the fine features of the images favorably in visual effects.

Finally, we apply the three denoising methods on the color image shown in Figure 8, which is contaminated by real film grain and scanning process noise. To obtain better color evaluations, we transform the RGB image to the YCrCb representation. Here, Y represents the luminance component and Cr and Cb denote the chrominance components. Then we apply the bilateral filter, the structure adaptive filter, and our proposed denoising method on each channel separately. The results are shown in Figures 8(b)–8(d), respectively. As we can see, the image in Figure 8(d) appears clearer than that in Figure 8(b). To further demonstrate this, Figures 8(e) and 8(f) show the absolute values of the residuals on the Y channel. It can be seen that the proposed method produces less features than bilateral filter and more noise-like residuals than structure adaptive method, which, in the absence of original noise-free image, is a reasonable indicator of superior performance.

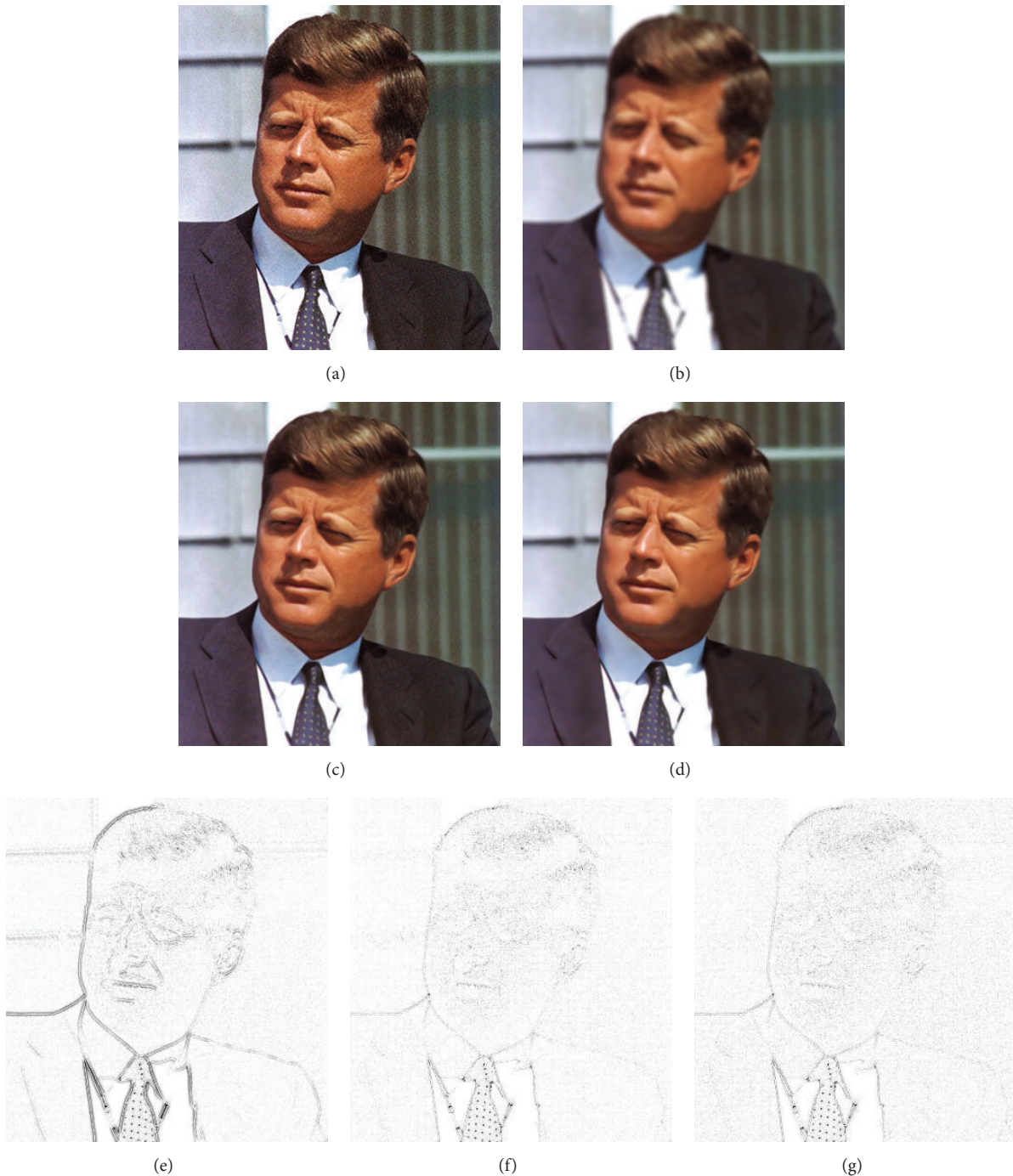


FIGURE 8: Performance of different denoising methods is compared on a color image with real noise; (e)–(g) show the residual between given noisy image and the respective estimates. (a) Real noisy image; (b) bilateral filter; (c) structure adaptive filter; (d) the proposed method; (e) bilateral filter; (f) structure adaptive filter; (g) the proposed method.

## 5. Conclusion

In this paper, we propose a new method for joint denoising with the help of residual feedbacks. This method combines favorable properties of bilateral filter and local self-adaptive kernel; and it can efficiently preserve the

fine structure of the image. Numerical experiments show that our new combined method has acceptable denoising performance even for handling images with heavy Gaussian noise. It remarkably exceeds traditional bilateral filter, local self-adaptive kernel, Wiener filter, and Kuwahara filter methods and generally outperforms some

state-of-the-art ones in terms of the quality of the restored images.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

This work was supported by the US National Science Foundation #DMS-1522332, the National Natural Science Foundation of China #61373087, #11201312, #61472257, and #61401283, the Foundation for Distinguished Young Teachers in Guangdong, China, #Yq2013144, the Guangdong Natural Science Foundation of China #2015A030313557 and #2015A030313550, the Specialized Research Fund for the Doctoral Program of Higher Education of China #20134408110001, the HD Video R & D Platform for Intelligent Analysis and Processing in Guangdong Engineering Technology Research Centre of Colleges and Universities #GCZX-A1409, and the Shenzhen Key Laboratory of Media Security.

### References

- [1] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the IEEE 6th International Conference on Computer Vision*, pp. 839–846, IEEE, Bombay, India, January 1998.
- [2] D. Tschumperlé, *PDE's based regularization of multivalued images and applications [Ph.D. thesis]*, Université de Nice Sophia-Antipolis, Nice, France, 2002.
- [3] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.
- [4] E. López-Rubio and M. N. Florentín-Núñez, "Kernel regression based feature extraction for 3D MR image denoising," *Medical Image Analysis*, vol. 15, no. 4, pp. 498–513, 2011.
- [5] H. Takeda, S. Farsiu, and P. Milanfar, "Higher order bilateral filters and their properties," in *Computational Imaging V*, vol. 6498 of *Proceedings of SPIE*, San Jose, Calif, USA, January 2007.
- [6] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Image and video restorations via nonlocal kernel regression," *IEEE Transactions on Cybernetics*, vol. 43, no. 3, pp. 1035–1046, 2013.
- [7] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Non-local kernel regression for image and video restoration," in *Computer Vision—ECCV 2010*, vol. 6313 of *Lecture Notes in Computer Science*, pp. 566–579, Springer, Berlin, Germany, 2010.
- [8] H. Takeda, S. Farsiu, and P. Milanfar, "Deblurring using regularized locally adaptive kernel regression," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 550–563, 2008.
- [9] J. Lu, Z. Ye, and Y. Zou, "Huber fractal image coding based on a fitting plane," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 134–145, 2013.
- [10] H. Takeda, S. Farsiu, and P. Milanfar, "Regularized kernel regression for image deblurring," in *Proceedings of the 40th Asilomar Conference on Signals, Systems, and Computers (ACSSC '06)*, pp. 1914–1918, Pacific Grove, Calif, USA, November 2006.
- [11] 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, San Diego, Calif, USA, June 2005.
- [12] 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.
- [13] H. Takeda, P. Milanfar, M. Protter, and M. Elad, "Super-resolution without explicit subpixel motion estimation," *IEEE Transactions on Image Processing*, vol. 18, no. 9, pp. 1958–1975, 2009.
- [14] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Multi-scale non-local kernel regression for super resolution," in *Proceedings of the 18th IEEE International Conference on Image Processing (ICIP '11)*, pp. 1353–1356, IEEE, Brussels, Belgium, September 2011.
- [15] C. Huang, Y. Liang, X. Ding, and C. Fang, "Generalized joint kernel regression and adaptive dictionary learning for single-image super-resolution," *Signal Processing*, vol. 103, pp. 142–154, 2014.
- [16] H. Yu, F.-S. Chen, Z.-J. Zhang, and C.-S. Wang, "Single infrared image super-resolution combining non-local means with kernel regression," *Infrared Physics & Technology*, vol. 61, pp. 50–59, 2013.
- [17] R. Gallea, E. Ardizzone, R. Pirrone, and O. Gambino, "Three-dimensional fuzzy kernel regression framework for registration of medical volume data," *Pattern Recognition*, vol. 46, no. 11, pp. 3000–3016, 2013.
- [18] M. N. Florentín-Núñez, E. López-Rubio, and F. J. López-Rubio, "Adaptive kernel regression and probabilistic self-organizing maps for JPEG image deblocking," *Neurocomputing*, vol. 121, pp. 32–39, 2013.
- [19] D. Brunet, E. R. Vrscay, and Z. Wang, "The use of residuals in image denoising," in *Image Analysis and Recognition*, M. Kamel and A. Campilho, Eds., vol. 5627 of *Lecture Notes in Computer Science*, pp. 1–12, Springer, 2009.
- [20] M. Zhou, X. Yan, H. Xie, H. Zheng, and G. Yang, "An effective method for signal extraction from residual image, with application to denoising algorithms," in *Advances in Neural Networks—ISNN 2013*, vol. 7951 of *Lecture Notes in Computer Science*, pp. 657–663, Springer, Berlin, Germany, 2013.
- [21] T. A. Hearn and L. Reichel, "Image denoising via residual kurtosis minimization," *Numerical Mathematics: Theory, Methods and Applications*, vol. 8, no. 3, pp. 406–424, 2015.
- [22] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [23] R. L. Euband, *Nonparametric Regression and Spline Smoothing*, Statistics, Textbooks and Monographs, Marcel Dekker, New York, NY, USA, 2nd edition, 1999.
- [24] W. Härdle, *Applied Nonparametric Regression*, Cambridge University Press, Cambridge, UK, 1990.
- [25] T. Q. Pham, L. J. Van Vliet, and K. Schutte, "Robust fusion of irregularly sampled data using adaptive normalized convolution," *Eurasip Journal on Applied Signal Processing*, vol. 2006, Article ID 083268, 2006.
- [26] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1141–1151, 2002.

- [27] D. Barash, "A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 844–847, 2002.
- [28] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, 3rd edition, 2007.
- [29] M. Kuwahara, K. Hachimura, S. Eiho, and M. Kinoshita, "Processing of RI-angiocardigraphic images," in *Digital Processing of Biomedical Images*, K. Preston Jr. and M. Onoe, Eds., pp. 187–202, Plenum, New York, NY, USA, 1976.
- [30] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Image enhancement and denoising by complex diffusion processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1020–1036, 2004.
- [31] L. Şendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 438–441, 2002.
- [32] D. Zhou and W. Cheng, "Image denoising with an optimal threshold and neighbouring window," *Pattern Recognition Letters*, vol. 29, no. 11, pp. 1694–1697, 2008.
- [33] F. Luisier, T. Blu, and M. Unser, "A new SURE approach to image denoising: interscale orthonormal wavelet thresholding," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 593–606, 2007.
- [34] C. A. Micchelli, L. Shen, and Y. Xu, "Proximity algorithms for image models: denoising," *Inverse Problems*, vol. 27, no. 4, Article ID 045009, 2011.
- [35] 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.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

