

## An adaptive image denoising method based on local parameters optimization

HARI OM and MANTOSH BISWAS\*

Department of Computer Science and Engineering, Indian School of Mines,  
Dhanbad, 826 004, India  
e-mail: hariom4india@gmail.com; mantoshb@gmail.com

MS received 14 January 2013; revised 5 April 2013; accepted 12 April 2013

**Abstract.** In image denoising algorithms, the noise is handled by either modifying term-by-term, i.e., individual pixels or block-by-block, i.e., group of pixels, using suitable shrinkage factor and threshold function. The shrinkage factor is generally a function of threshold and some other characteristics of the neighbouring pixels of the pixel to be thresholded (denoised). The threshold is determined in terms of the noise variance present in the image and its size. The VisuShrink, SureShrink, and NeighShrink methods are important denoising methods that provide good results. The first two, i.e., VisuShrink and SureShrink methods follow term-by-term approach, i.e., modify the individual pixel and the third one, i.e., NeighShrink and its variants: ModiNeighShrink, IIDMWD, and IAWDMBMC, follow block-by-block approach, i.e., modify the pixels in groups, in order to remove the noise. The VisuShrink, SureShrink, and NeighShrink methods however do not give very good visual quality because they remove too many coefficients due to their high threshold values. In this paper, we propose an image denoising method that uses the local parameters of the neighbouring coefficients of the pixel to be denoised in the noisy image. In our method, we propose two new shrinkage factors and the threshold at each decomposition level, which lead to better visual quality. We also establish the relationship between both the shrinkage factors. We compare the performance of our method with that of the VisuShrink and NeighShrink including various variants. Simulation results show that our proposed method has high peak signal-to-noise ratio and good visual quality of the image as compared to the traditional methods: Weiner filter, VisuShrink, SureShrink, NeighBlock, NeighShrink, ModiNeighShrink, LAWML, IIDMWT, and IAWDMBNC methods.

**Keywords.** Thresholding; maximum likelihood estimation (ML); peak signal-to-noise ratio (PSNR).

### 1. Introduction

Image denoising is one of the major research topics in image processing. An efficient image denoising method is that in which a compromise has to be found between the noise reduction

---

\*For correspondence

and the preservation of significant useful image details. It has to be adaptive to image discontinuities to achieve good performance in this respect. The denoising methods are an important subsystem of any signal processing system used for image enhancement because these methods remove undesirable signal components from the signal of interest. The denoising methods may be classified into two main categories: non-wavelet and wavelet based methods. The non-wavelet methods are also referred to as filtering methods (Jain 1989; Gonzalez & Woods 2008). In literature, some important filtering methods have been discussed for denoising the images including, natural images, 3-D images, and magnetic resonance images. In Coupe *et al* (2012), an adaptive multiresolution version of the block-wise non-local (NL)-means filter is discussed for denoising the 3-D magnetic resonance (MR) images by considering two types of noise: Gaussian noise and Rician noise. The filter discussed in Coupe *et al* (2012) implicitly adapts the amount of denoising according to the spatial and frequency information present in the image on the basis of an adaptive soft wavelet coefficient mixing. In general, the filters blur sharp edges, destroy lines and other fine details present in the image. It has also been observed that the filtering approach does not produce considerable denoising for the images corrupted with the Gaussian noise.

To overcome the problems of non-wavelet techniques, the wavelet-based techniques have been introduced that preserve the signal structure. The wavelets play a very important role in noise removal, especially the Gaussian noise (Edwards 1992). The wavelet-based denoising methods have several advantages. (a) They have good energy compaction. (b) They do not produce the blocking artifacts such as in the discrete cosine transform. (c) They allow to locally define their degree of smoothness because of the orthogonal property. (d) They nearly decorrelate the images with respect to intrascale statistics. (e) They partially reflect the anisotropy of the human visual system vortex decomposition due to orientations and multiresolution representation. Due to easier reconstruction and the variable window size, the wavelet based methods are capable of providing the time and frequency information simultaneously. The noise removal is highly dependent on the threshold that is used to modify the wavelet coefficients. Two types of thresholding are commonly used: term-by-term and block-by-block. The term-by-term thresholding removes the noise using individual pixels, and the block-by-block thresholding uses a group of pixels to remove the noise. Some of the important methods model the wavelet coefficients as some known probability distributions. In Bhuiyan *et al* (2008), the wavelet coefficients have been modelled as a symmetric normal inverse Gaussian distribution whose parameters are estimated using a Bayesian minimum mean-squared error (MMSE) estimator. In that work, it has been assumed that the coefficients obtained using the Bayesian MMSE estimator are approximately noise-free that are used to obtain the variances corresponding to the noise-free coefficients. The paper (Guo *et al* 2008) models the wavelet coefficients as Gaussian distributions and accordingly the shrinkage factors in each scale have been obtained. The shrinkage factors in consecutive scales are combined in order to detect the edges. The detected wavelet coefficients corresponding to the edge pixels and non-edge pixels are updated by different shrinkage factors to remove their noise. In Yin *et al* (2012), the denoising of the infrared images is discussed in which the image is first non-subsampled contourlet transformed and the transform coefficients are modelled as generalized Gaussian distribution. According to the characteristics of different energies in each direction of the subbands, a modified Bayesian threshold is obtained that is used to estimate the noise-free coefficients and the inverse non-subsampled contourlet transformation is used to get the denoised image. Liu *et al* (2012) also discussed the contourlet transform for denoising the natural images. In this paper, the natural image is disintegrated into subband and angular decompositions using the contourlet transform. The subband decomposition is done using the wavelet-based contourlet transform rather than the Laplacian pyramid (LP) in order to reduce

the computations and the directional decomposition is done using the directional filter banks (DFB). Then, the Donoho and Johnstone's threshold is used to modify the coefficients, which in turn provide the noise-free image on applying the inverse contourlet transform. This method is more time and space consuming because it requires the original image to be represented in two images: subband image and angular image. Some researchers have combined both filtering and wavelet approaches for denoising the images, which are named as hybrid filtering methods. Lal & Chandra (2012) discussed a hybrid filtering method, which is applied in two phases. In first phase, the noisy wavelet transformed image is processed by the nonlinear fourth order partial differential equation (PDE) and in second phase the bivariate shrinkage function is applied to further process the output of non-linear fourth order PDE. The fourth order PDE smoothens the small gradients and the bivariate shrinkage function removes large gradients. This method also requires more computation and space both.

Chen *et al* (2012) discussed an important denoising method by exploiting the statistical dependency between a complex wavelet coefficient, its parent, and its children of the dual-tree complex wavelet coefficients. The threshold is derived from this statistical dependency that is used to modify the wavelet coefficients. At first level, where no parent exists, the bivariate threshold is used to modify the wavelet coefficients. In almost all the denoising methods, the basic goal has been to estimate suitable thresholds, which are used to modify the noisy image coefficients. If there is no suitable threshold estimated, the ad-hoc choice of the threshold is used. Furthermore, the specific distributions of the signal (wavelet coefficients) and noise may not be well-matched at different scales/decomposition levels. The most common and well-known threshold is Universal threshold that is also known as VisuShrink (Donoho & Johnstone 1994). The VisuShrink threshold method however gives overly smoothed images and also does not deal with minimizing the mean squared error. It may be considered as a general purpose threshold method that exhibits near optimal minimax error properties. The mean squared error minimization is discussed in (Donoho & Johnstone 1995) by using the SureShrink threshold that depends upon the Stein's Unbiased Risk Estimator (SURE) and takes the combination of the Universal and SURE thresholds into account. Most of the wavelet shrinkage denoising approaches use the Donoho's shrinkage thresholding (Donoho 1995). Chen and Bui have extended the Cai's method, namely NeighBlock and NeighCoeff to 2-D and multi-wavelet domains (Cai & Silverman 2001; Chen & Bui 2003; Chen *et al* 2004). The NeighBlock, NeighCoeff, and NeighShrink thresholding methods are adaptive and use block thresholding in which the characteristics of the neighbouring coefficients are used to determine the shrinkage threshold. Mohideen *et al* (2008) have developed the ModiNeighShrink that improves the NeighShrink by using the modified shrinkage factor and the VisuShrink threshold (Mohideen *et al* 2008). The improved image denoising method based on wavelet thresholding (IIDMWT) (Om & Biswas 2012), a modified version of NeighShrink and ModiNeighShrink, uses the modified threshold of the VisuShrink and the ModiNeighShrink shrinkage factors and performs better than the NeighShrink and ModiNeighShrink both. Another method improved adaptive wavelet denoising method based on neighbouring coefficients (IAWDMNC) (Jiang *et al* 2010) uses both new threshold and shrinkage factor and outperforms the NeighShrink and ModiNeighShrink. The above discussed methods simultaneously either kill or keep all the coefficients in groups. Mihcak *et al* (1999a, b) discussed an efficient method that is also known as locally adaptive window-based denoising using maximum likelihood (LAWML) (Mihcak *et al* 1999a, b). In this method, the wavelet coefficients satisfy the Gaussian distribution with zero-mean. The estimated variance of the wavelet coefficients can be obtained by maximum likelihood estimation in the square neighbourhood and the denoised wavelet coefficients are obtained using the minimum mean squared error-like (MMSE-like) estimation. Some important recent works for selecting

the threshold for denoising a noisy image using wavelets are discussed in Chen *et al* (2004), Mohideen *et al* (2008), Om & Biswas (2012), Jiang *et al* (2010), Shengqian *et al* (2002a, b), Atto *et al* (2011).

The choice of threshold is very crucial as it plays a very important role in noise removal because the denoising techniques most frequently produce the smoothed images. The NeighShrink (Cai & Silverman 2001; Chen & Bui 2003; Chen *et al* 2004) and its variants: ModiNeighShrink (Mohideen *et al* 2008), IIDMWD (Om & Biswas 2012), and IAWDMBMC (Jiang *et al* 2010), do not give very good visual quality of the denoise image due to removing of too many noisy coefficients completely because the threshold value is at higher side. In this paper, we present a new image denoising method that follows term-by-term and block-by-block thresholding. It takes image pixel statistics into account that include: mean, variance, minimum and maximum values of raw coefficients, sum of square of pixels within the neighbouring window, locally adaptive maximum likelihood estimation, division of sum of square of pixels within the window by window size, and finally the scale factor with noise level. In our proposed method, the threshold function mainly follows the noise variance using LAWML (Mihcak *et al* 1999a, b) methods, whereas the methods (Cai & Silverman 2001; Chen & Bui 2003; Chen *et al* 2004; Mohideen *et al* 2008; Om & Biswas 2012; Jiang *et al* 2010) follow the mean absolute deviation (MAD) (Donoho 1995) estimator. Thus, our method provides better results. The rest of the paper is organized as follows. Section 2 briefly discusses the wavelet transformation. Section 3 reviews the related works. Section 4 presents the proposed method. Section 5 discusses the simulation results, and finally the conclusion is given in section 6.

## 2. Wavelet principle

Wavelets are the functions that are used to analyse the data according to scale or resolution or both (Graps 1995). Basically, the term ‘wavelet’ refers to a set of orthonormal basis functions generated by dilation and translation of the scaling function  $\phi$  and mother wavelet  $\psi$  (Antoniadis & Bigot 2001). The finite scale multiresolution representation of a discrete function may be called as discrete wavelet transform (DWT) (Edwards 1992). The wavelet basis, unlike the Sine and Cosine in the Fourier transform, is quite localized in space and frequency. The orthonormal wavelet basis is defined as

$$\psi_{j,k}(x) = 2^{(j/2)}\psi(2^j x - k). \quad (1)$$

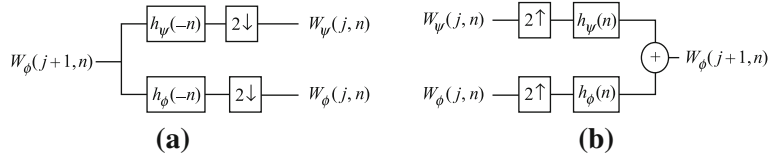
The scaling function is given by

$$\phi_{j,k}(x) = 2^{(j/2)}\phi(2^j x - k), \quad (2)$$

where,  $\psi$  is called the wavelet function,  $j$  and  $k$  are integers that signify scaling and dilation of the wavelet function. For numerical computation, the wavelet function is dilated by powers of two and is translated by  $k$ . In terms of the wavelet coefficients, the wavelet function is given by (Edwards 1992)

$$\psi(x) = \sum_k^{N-1} g_k \sqrt{2\phi(2x - k)}, \quad (3)$$

where  $g_0, g_1, g_2, \dots, g_{N-1}$  are the high pass wavelet coefficients.



**Figure 1.** Wavelet filter model: (a) analysis (decomposition) filter bank, (b) synthesis (reconstruction) filter bank.

Writing the scaling function in terms of the scaling coefficients gives

$$\phi(x) = \sum_k^{N-1} h_k \sqrt{2\phi(2x - k)}. \tag{4}$$

The coefficients  $h_0, h_1, h_2, \dots, h_{N-1}$  are low pass scaling coefficients. The wavelet and scaling coefficients are related by the quadrature mirror relationship as given by

$$g_n = (-1)^n h_{1-n+N}. \tag{5}$$

Here,  $N$  signifies the number of vanishing moments that helps wavelets classifying into a family (Edwards 1992). The wavelet function produces different wavelet families like Daubechies, Haar, Coiflets, etc. (Edwards 1992). Within each family of wavelets, there are wavelet subclasses distinguished by the number of coefficients and the level of iterations. A graphical representation of the decomposition and reconstruction of wavelet transform is shown in figure 1.

### 3. Review of previous work

During transmission, the image signal  $x, \{x_{i,j}; 1 \leq i, j \leq M, M \times M \text{ is the original image size}\}$ , is corrupted by the independent and identically distributed (i.i.d) zero mean, white Gaussian noise, i.e.,  $n \sim G(0, \sigma^2)$ ,  $G$  is Gaussian distribution. At the receiver end, the noisy observation  $y$  of the original image  $x$  is obtained. Thus, we have

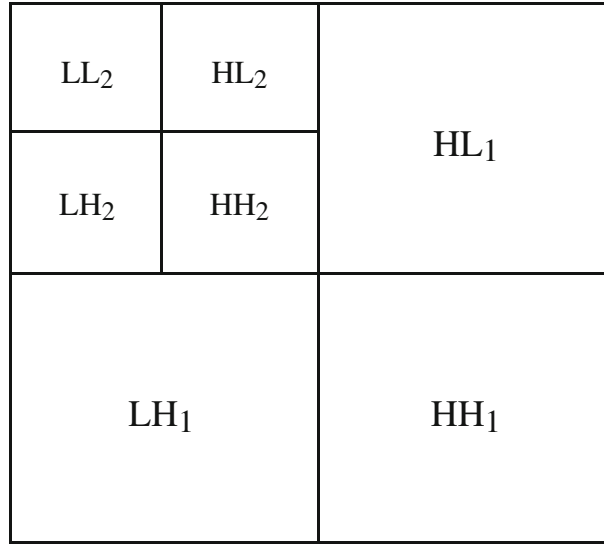
$$y_{i,j} = x_{i,j} + n_{i,j}, \quad i, j = 1, 2, \dots, M, \tag{6}$$

Here, the goal is to estimate the signal  $x$  from the noisy observation  $y$  with the condition that the mean squared error (MSE) is minimum. For this, a 2-D orthogonal discrete wavelet transform (DWT) (Edwards 1992; Antoniadis & Bigot 2001)  $W$  is applied onto  $y$  to get the wavelet coefficients  $Y$  that have four subbands: low-low (LL), low-high (LH), high-low (HL), and high-high (HH) (refer figure 2), i.e.

$$Y = Wy \tag{7}$$

The wavelet thresholding method processes each coefficient  $Y_{i,j}$  from the detail subbands: LH, HL, and HH, using suitable threshold function to obtain  $\bar{X}$ . The denoised estimate  $\hat{x}$  is obtained by applying the inverse wavelet transform (IDWT) (Edwards 1992; Antoniadis & Bigot 2001)  $W^{-1}$  to  $\bar{X}$ , i.e.,

$$\hat{x} = W^{-1}\bar{X}. \tag{8}$$



**Figure 2.** 2D-DWT with 2-Level decomposition.

The Wiener filtering method requires the information about the spectra of the noise and the original signal (Jain 1989; Gonzalez & Woods 2008). It works well for smooth signals and the Gaussian noise. The Wiener filter in Fourier domain is defined as follows

$$G(i, j) = \frac{H^*(i, j) P_s(i, j)}{|H(i, j)|^2 P_s(i, j) + P_n(i, j)}, \quad (9)$$

where  $H(i, j)$  denotes the degradation function and  $H^*(i, j)$  its complex conjugate;  $P_n(i, j)$  and  $P_s(i, j)$  denote the power spectral densities of the noisy and noise-free images, respectively.

The Wiener filtering technique fails to produce satisfactory results for a broad range of low contrast images and is also computationally expensive. To overcome the weaknesses of the Wiener filter, Donoho *et al* have discussed the wavelet based image denoising threshold that removes the noise significantly. Donoho and Johnston have discussed the soft thresholding method for image denoising (Donoho & Johnstone 1994, 1995; Donoho 1995). In soft thresholding, the wavelet coefficients that are greater than the threshold are shrunk towards zero and those that less than the negative of threshold are increased by the threshold value. The coefficients whose absolute value is less than the threshold value are set to zero. The soft-thresholding function in compact form is defined as follows:

$$\hat{x}_{i,j} = \begin{cases} x_{ij} - T & \text{if } x_{ij} \geq T \\ x_{ij} + T & \text{if } x_{ij} \leq -T \\ 0 & \text{if } |x_{ij}| < T \end{cases}, \quad (10)$$

where  $x$  and  $\hat{x}$  represent the noisy and noise-free image signals, respectively;  $T$  is threshold value.

The VisuShrink, an important denoising method, uses a threshold, denoted by  $T_{\text{visu}}$ , that is proportional to the standard deviation of noise (Donoho & Johnstone 1994). It follows

the soft thresholding rule, which is also referred to as the Universal threshold. The  $T_{\text{Visu}}$  is defined by

$$T_{\text{Visu}} = \sigma \sqrt{2 \log M}, \tag{11}$$

where  $\sigma^2$  is noise variance defined based on the median absolute deviation (MAD) and is given by (Donoho 1995)

$$\sigma^2 = \left[ \frac{\text{median} |HH_1|}{0.6745} \right]^2.$$

The modified threshold of the VisuShrink threshold, denoted by  $T_{\text{NEW}}$ , is given by (Om & Biswas 2012)

$$T_{\text{NEW}} = \sigma \sqrt{2 \log \hat{M}}, \tag{12}$$

where  $\hat{M} = \frac{M}{2^l}$ , an image dimension at  $l^{\text{th}}$  decomposition level.

The SureShrink method that also follows the soft thresholding rule suppresses the noise by thresholding the empirical wavelet coefficients (Donoho & Johnstone 1995). Its threshold, denoted by  $t^*$ , is defined as follows:

$$t^* = \min \left( t, \sigma \sqrt{2 \log M} \right), \tag{13}$$

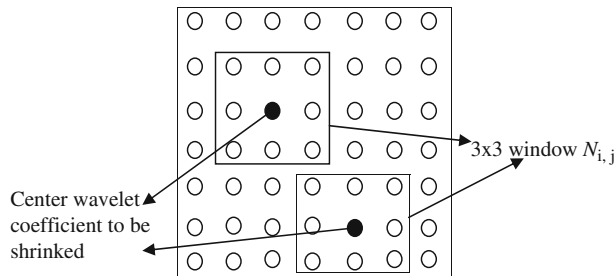
where  $t$  denotes the threshold value at each decomposition level of the wavelet transform.

Mahqak *et al* discussed an efficient method, called locally adaptive window-based denoising method, that uses the maximum likelihood (LAWML) (Mihcak *et al* 1999a, b). In this method, the signal variance  $\sigma_{i,j}^2$  is estimated by using the LAWML that gives reasonably good results. The signal variance  $\sigma_{i,j}^2$  is calculated as follows:

$$\sigma_{i,j}^2 = \max \left( 0, \frac{1}{b} \sum_{Y_{i,j} \in N(i,j)} Y_{i,j}^2 - \sigma^2 \right), \tag{14}$$

where  $N(i, j)$  represents the coefficients in a local window with length  $L$  centered at  $(i, j)$  coefficient and  $b$  is the number of coefficients in  $N(i, j)$ . Here  $L$  is a positive odd number denoting window size, e.g., the window size can be  $3 \times 3$ ,  $5 \times 5$  and so on (refer figure 3). Figure 3 shows  $3 \times 3$  neighbourhood window, whose blackened wavelet coefficients are to be thresholded.

One of the most popular methods of NeighShrink incorporates the neighbouring coefficients in the thresholding process by considering a local (square) window with length  $L$  centered at the



**Figure 3.** Wavelet coefficients in neighbourhood  $3 \times 3$  window.

pixel to be thresholded (refer figure 3) (Cai & Silverman 2001; Chen & Bui 2003; Chen *et al* 2004). This method is described below in brief.

Let  $Sq_{i,j}$  denote the summation square of the wavelet coefficients  $Y_{i,j}$  in the neighbourhood window  $N(i, j)$  i.e.,

$$Sq_{i,j} = \sum_{Y_{i,j} \in N(i,j)} Y_{i,j}^2. \quad (15)$$

Shrink the wavelet coefficients  $Y_{i,j}$  according to James–Stein (JS) rule as follows

$$\bar{X}_{i,j} = Y_{i,j} \left( \left( 1 - \frac{T_{\text{Visu}}^2}{Sq_{i,j}^2} \right)_+ \right). \quad (16)$$

Here, + sign at end of the formula signifies that the positive value remains unchanged and the negative value is made as zero.

On applying the above shrinkage factor (16) to the wavelet coefficients, it has been observed that some noise is still in the reconstructed image. This problem has been overcome using the following shrinkage factor (Mohideen *et al* 2008):

$$\bar{X}_{i,j} = Y_{i,j} \left( \left( 1 - \frac{3 T_{\text{Visu}}^2}{4 Sq_{i,j}^2} \right)_+ \right). \quad (17)$$

The IAWDMBNC method takes several useful values of the neighbouring coefficients, which are needed for choosing an appropriate threshold, as defined below (Jiang *et al* 2010):

$$\begin{aligned} Sq_{\max} &= \max (Sq_{i,j}^2) \\ Sq_{\min} &= \min (Sq_{i,j}^2) \end{aligned},$$

here  $Sq_{\max}$  and  $Sq_{\min}$  represent maximum and minimum summation square of the wavelet coefficients  $Sq_{i,j}$  defined in (15) at the same level, respectively. In this method, an adaptive threshold is defined using the following relation:

$$T_{i,j} = T_{\text{Visu}} \left( \frac{Sq_{\max} - Y_{i,j}}{Sq_{\max} - Sq_{\min}} \right). \quad (18)$$

The wavelet coefficients are shrunk using the following expression

$$\bar{X}_{i,j} = Y_{i,j} \left( 1 - \left( \frac{T_{i,j}^2}{Sq_{i,j}^2} \right)_+^l \right), \quad (19)$$

where  $1 \leq l \leq J$ ;  $J$  is number of decomposition levels.

In this method, different wavelet coefficient subbands are shrunked independently, but the threshold and the neighbouring window size is kept same in all subbands.

#### 4. Proposed method

Our proposed method consists of estimation of two shrinkage factors, their relationships, threshold estimation, and finally the denoising process.



#### 4.1 Shrinkage factor estimation

In wavelet coefficients, the local maximum must contain at least one important information about the signal such as singularity, transient phenomena (i.e., peaks). Such information is one of the main features of the signal and should be kept in the denoising process. Inspired by this idea, a new adaptive threshold method is developed, which combines the three criteria for distinguishing the useful coefficients from the noise: *summation square of the coefficients in the neighbouring window*, *LAWML of the coefficients*, and *the scale factor with noise levels from the noisy image*. These three criteria are combined in an adaptive manner in our method, which optimizes the local parameters. Our threshold function shrinks the wavelet coefficients using one of the following two shrinkage factors, denoted by  $\alpha_{i,j}$  and  $\beta_{i,j}$ .

##### 4.1a Shrinkage factor $\alpha_{i,j}$ :

It is defined as

$$\alpha_{i,j} = \left( 1 - \left( \frac{T_{New}}{Sq_{i,j}} \right)_+^l \right), \quad (20)$$

here  $l = 1, 2, \dots, J$ ; where  $J$  is pre-specified number of decomposition levels. The function of  $T_{New}$  is given later in (30).

##### 4.1b Shrinkage factor $\beta_{i,j}$ :

The shrinkage factor  $\alpha_{i,j}$  can be improved further, denoting it by  $\beta_{i,j}$ , as follows:

$$\beta_{i,j} = \left( 1 - \left( \frac{n}{n+1} \right) \left( \frac{T_{New}}{Sq_{i,j}} \right)_+ \right). \quad (21)$$

Here,  $n$  is any positive integer, i.e.,  $0 < n < \infty$ . The choice of  $n$  is independent of the image, noise, subband, or scale. The values of noisy wavelet coefficients become low as  $n$  increases since this formula can keep more information of the noisy image.

The main motive of our proposed method is to keep more and more image details such as edge, background, etc. as well as to remove the noise from the noisy image coefficients. Here we have discussed two shrinkage factors. Now, we discuss why the shrinkage factors used in Om & Biswas (2012), Jiang *et al* (2010) do not provide better quality of the reconstructed image. In (Om and Biswas 2012), the shrinkage factor is  $\left( 1 - \frac{3}{4} \frac{T_{NEW}^2}{Sq_{i,j}^2} \right)_+$ , where  $T_{NEW} = \sigma \sqrt{2 \log \hat{M}}$  and  $\hat{M} = \frac{M}{2^l}$ . The value of  $T_{NEW}$  decreases as the decomposition level increases and hence its square will also be small. This value is furthermore multiplied by  $\left( \frac{3}{4} \right)$  which is less than 1 and hence the resultant value of  $\left( \frac{3}{4} \frac{T_{NEW}^2}{Sq_{i,j}^2} \right)$  will move towards 0. Therefore, the value of  $\left( 1 - \frac{3}{4} \frac{T_{NEW}^2}{Sq_{i,j}^2} \right)$  will be close to 1. This means that the noise will not be removed significantly and it will remove the more noisy coefficients.

In (Jiang *et al* 2010), the shrinkage factor is  $\left( 1 - \left( \frac{T_{i,j}}{Sq_{i,j}} \right)^l \right)$ , where  $T_{i,j} = T_{Visu} \left( \frac{Sq_{max} - Y_{i,j}}{Sq_{max} - Sq_{min}} \right)$ . In other words, the value of  $\left( \frac{Sq_{max} - Y_{i,j}}{Sq_{max} - Sq_{min}} \right)$  is multiplied by the fixed quantity  $T_{Visu} = \sigma \sqrt{2 \log M}$ , where  $M$  is original image size. Furthermore, the value of  $\left( \frac{T_{i,j}}{Sq_{i,j}} \right)^l$  will

**Table 1.** Various cases for different combinations of parameters defined over the noisy wavelet coefficients ( $Y_{i,j}$ ).

Cases	Cases
1. $\left(\frac{Sum_{\max} - Sum_{\text{mean}}}{Sum_{\max} + Sum_{\text{mean}}}\right)$	6. $\left(\frac{D_{\max} - D_{\min}}{D_{\max} + D_{\min}}\right)$
2. $\left(\frac{Sq_{\text{mean}} - Sq_{\min}}{Sq_{\text{mean}} + Sq_{\min}}\right)$	7. $\left(\frac{D_{\max} - D_{\text{mean}}}{D_{\max} + D_{\text{mean}}}\right)$
3. $\left(\frac{Sq_{\max} - Sq_{\min}}{Sq_{\max} + Sq_{\min}}\right)$	8. $\left(\frac{\hat{\sigma}_{\text{mean}}^2 - \hat{\sigma}_{\min}^2}{\hat{\sigma}_{\text{mean}}^2 + \hat{\sigma}_{\min}^2}\right)$
4. $\left(\frac{Sq_{\max} - Sq_{\text{mean}}}{Sq_{\max} + Sq_{\text{mean}}}\right)$	9. $\left(\frac{\hat{\sigma}_{\max}^2 - \hat{\sigma}_{\min}^2}{\hat{\sigma}_{\max}^2 + \hat{\sigma}_{\min}^2}\right)$
5. $\left(\frac{D_{\text{mean}} - D_{\min}}{D_{\text{mean}} + D_{\min}}\right)$	10. $\left(\frac{Coeff_{\max} - Coeff_{\text{mean}}}{Coeff_{\max} + Coeff_{\text{mean}}}\right)$

move towards to 1 as  $l$  increases. Hence, the value of  $\left(1 - \left(\frac{T_{i,j}^2}{Sq_{i,j}^2}\right)_l\right)$  will be slightly higher than 0. Therefore, it removes the noise significantly but also removes the more noisy coefficients. In other words, the methods (Om & Biswas 2012; Jiang *et al* 2010) remove too many noisy coefficients and give the overly smoothed image that is also shown in our results.

In our proposed method, we have two shrinkage factors  $\alpha_{i,j} = \left(1 - \left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)^l\right)$  and  $\beta_{i,j} = \left(1 - \left(\frac{n}{n+1}\right)\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)\right)$ . Here  $T_{\text{New}} = \chi(\text{Case}_c) f$ , where  $f = \sigma \sqrt{\log \frac{\hat{M}}{l}}$ ,  $\hat{M} = \frac{M}{2^l}$  and  $1 \leq c \leq 10$ . (ref. table 1).

Here the value of  $T_{\text{New}}$  decreases rapidly as the decomposition level increases since the value of  $f$  depends on the decomposition level, i.e.  $f = \sigma \sqrt{\log \frac{\hat{M}}{l}}$ , where  $\hat{M} = \frac{M}{2^l}$ . Thus, the value of  $\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)^l$  will also be very small that is very nearer to 0 for the high decomposition level  $l$ . Hence, the value of  $\alpha_{i,j}$  i.e.  $\left(1 - \left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)^l\right)$  will be closer to 1. Furthermore, the value of  $\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)$  has been multiplied by  $\left(\frac{n}{n+1}\right)$ . Since the value of  $\left(\frac{n}{n+1}\right)$  is less than 1, the result of  $\left(\frac{n}{n+1}\right)\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)$  will be further reduced and the value of  $\beta_{i,j} = \left(1 - \left(\frac{n}{n+1}\right)\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)\right)$  will also be closer to 1. Therefore, our proposed method using the shrinkage factors  $\alpha_{i,j}$  or  $\beta_{i,j}$  does not kill many noisy coefficients. The noisy coefficients are modified by using these factors rather than being killed. In other words, our proposed method removes the noise effectively and retains the image features as much as possible.

#### 4.1c Relationship between shrinkage factors $\alpha_{i,j}$ and $\beta_{i,j}$ :

We can write (20) as follows:

$$\left(\frac{T_{\text{New}}}{Sq_{i,j}}\right)^l = (1 - \alpha_{i,j}). \quad (22)$$

Taking natural logarithm on both sides of (22) gives

$$\log \left( \frac{T_{\text{New}}}{Sq_{i,j}} \right) = \frac{\log (1 - \alpha_{i,j})}{l}. \tag{23}$$

From (21), we have

$$\left( \frac{T_{\text{New}}}{Sq_{i,j}} \right) = (1 - \beta_{i,j}) \left( \frac{n + 1}{n} \right) \tag{24}$$

Taking natural logarithm on both sides of (24) gives

$$\log \left( \frac{T_{\text{New}}}{Sq_{i,j}} \right) = \log \left\{ (1 - \beta_{i,j}) \left( \frac{n + 1}{n} \right) \right\}. \tag{25}$$

From (23) and (25), we have

$$\log \left\{ (1 - \beta_{i,j}) \left( 1 + \frac{1}{n} \right) \right\} = \frac{\log (1 - \alpha_{i,j})}{l}.$$

We can write it as follows

$$l \left\{ \log (1 - \beta_{i,j}) + \log \left( 1 + \frac{1}{n} \right) \right\} = \log (1 - \alpha_{i,j}).$$

Writing log functions in series form gives

$$l \left\{ \left( -\beta_{i,j} - \frac{\beta_{i,j}^2}{2} - \frac{\beta_{i,j}^3}{3} - \dots \right) + \left( \frac{1}{n} - \frac{1}{2n^2} + \frac{1}{3n^3} - \dots \right) \right\} = -\alpha_{i,j} - \frac{\alpha_{i,j}^2}{2} - \frac{\alpha_{i,j}^3}{3} - \dots$$

Assuming  $\alpha_{i,j}, \beta_{i,j}$  as very small quantities, i.e.,  $0 < \alpha_{i,j}, \beta_{i,j} < 1$ , and taking  $n$  large value, we have the following relation between  $\alpha_{i,j}$  and  $\beta_{i,j}$ :

$$l \left\{ (-\beta_{i,j}) + \left( \frac{1}{n} \right) \right\} \approx -\alpha_{i,j}$$

Or

$$\beta_{i,j} \approx \frac{\alpha_{i,j}}{l} + \frac{1}{n}. \tag{26}$$

This is the relation between  $\alpha_{i,j}$  and  $\beta_{i,j}$ , which is the form of an affine transform. For very large value of  $n$ , at any decomposition level,  $\alpha_{i,j}$  and  $\beta_{i,j}$  are linearly related as we have from (26) the following relation:

$$\beta_{i,j} \approx \frac{\alpha_{i,j}}{l}.$$

Thus, both the shrinkage factors  $\alpha_{i,j}$  and  $\beta_{i,j}$  may be considered equivalent up to a linear multiplier factor.

#### 4.2 Threshold estimation

We define the following parameters over the noisy wavelet coefficients ( $Y_{i,j}$ ) whose different combinations can give a new threshold value.

$Sum_{i,j}$ : summation of the noisy wavelet coefficients  $Y_{i,j}$  in neighbourhood window  $N(i, j)$   
i.e.,

$$Sum_{i,j} = \sum_{Y_{i,j} \in N(i,j)} Y_{i,j} \quad (27)$$

$Sum_{\text{mean}}$ : mean of  $Sum_{i,j}$

$Sum_{\text{max}}$ : maximum of  $Sum_{i,j}$

$Sq_{\text{mean}}$ : mean of  $Sq_{i,j}$  ( $Sq_{i,j}$  is defined in (15)).

$Sq_{\text{min}}$ : minimum of  $Sq_{i,j}$

$Sq_{\text{max}}$ : maximum of  $Sq_{i,j}$

$Coeff_{\text{mean}}$ : mean of noisy wavelet coefficients  $Y_{i,j}$

$Coeff_{\text{max}}$ : maximum of noisy wavelet coefficients  $Y_{i,j}$

$D_{i,j}$ : average of the square of noisy wavelet coefficients within the neighbouring window of size  $b$ , i.e.

$$D_{i,j} = \frac{1}{b} \sum_{Y_{i,j} \in N(i,j)} Y_{i,j}^2 \quad (28)$$

$D_{\text{mean}}$ : mean of  $D_{i,j}$

$D_{\text{max}}$ : maximum of  $D_{i,j}$

$D_{\text{min}}$ : minimum of  $D_{i,j}$

$\sigma_{\text{mean}}^2$ : mean of  $\sigma_{i,j}^2$  ( $\sigma_{i,j}^2$  is defined in (14)).

$\sigma_{\text{max}}^2$ : maximum of  $\sigma_{i,j}^2$

$\sigma_{\text{min}}^2$ : minimum of  $\sigma_{i,j}^2$ .

We get the noiseless coefficients  $\bar{X}_{i,j}$  after shrinking the noisy coefficients  $Y_{i,j}$  by using the shrinkage factor as follows

$$\bar{X}_{i,j} = \theta Y_{i,j}, \text{ where } \theta = \alpha_{i,j} \text{ or } \theta = \beta_{i,j}. \quad (29)$$

The new threshold,  $T_{\text{New}}$  is defined as

$$T_{\text{New}} = \chi (Case_c) f, \quad (30)$$

where,  $\chi (Case_c)$  is given below (ref. table 1) and the range of  $c$  varies from 1 to 10, i.e.  $1 \leq c \leq 10$ .

Here, the scale factor,  $f$ , is calculated as follows

$$f = \sigma \sqrt{\log \frac{\hat{M}}{l}}, \text{ where } \hat{M} = \frac{M}{2^l}. \quad (31)$$

#### 4.3 Denoising algorithm

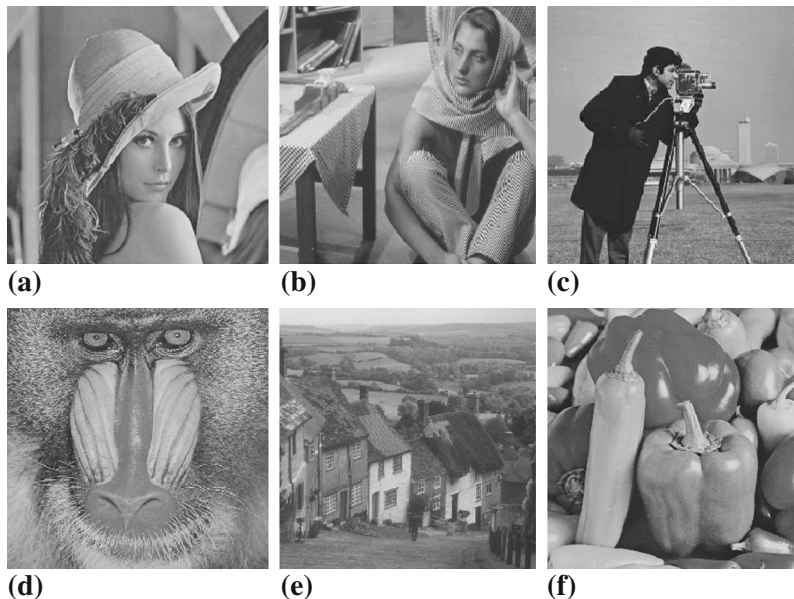
Our denoising algorithm has the following steps:

- (i) Perform decomposition on the noisy image using 2-D orthogonal wavelet transform  $W$  up to  $J^{\text{th}}$  level to generate several subbands (refer figures 1 and 2).

- (ii) For each level, compute all parameters required to evaluate threshold.
- (iii) For each subband: HH, HL, and LH (i.e., except the low pass residual LL)
  - Evaluate  $T_{\text{New}}$  using (30)
  - Apply shrinkage factor according to (29) to the noisy coefficients to get the modified coefficients.
- (iv) Perform inverse wavelet transform to the modified coefficients to obtain the denoised estimate image  $\hat{x}$ .

## 5. Results and discussions

We have computed the results using our proposed method for Lena, Barbara, Cameraman, Mandrill, Goldhill, and Pepper images (refer figure 4), each of size  $512 \times 512$ , with various additive Gaussian white noise levels using soft thresholding. We have used Symlet least asymmetric compactly supported wavelet with eight vanishing moments up to four scales of decomposition at noise levels: 20, 50, and 100. The window size has been taken as  $3 \times 3$ . The results are compared with that of the VisuShrink (Donoho & Johnstone 1994), SureShrink (Donoho & Johnstone 1995), non-wavelet method Weiner filter (Jain 1989; Gonzalez & Woods 2008), NeighShrink (Chen & Bui 2003; Chen *et al* 2004), NeighBlock (Cai & Silverman 2001), ModiNeighShrink (Mohideen *et al* 2008), IAWDMBNC (Jiang *et al* 2010), LAWML (Mihcak



**Figure 4.** Original test images of size  $512 \times 512$  pixels: (a) Lena (b) Barbara (c) Cameraman (d) Mandrill (e) Goldhill and (f) Pepper.

**Table 2.** Results (in PSNR) of denoising for Lena, Barbara, Cameraman, Mandrill, Goldhill, and Pepper images using VisuShrink, SureShrink and our proposed method with various cases using term-by-term thresholding.

Image name	Noise levels	Proposed method with various cases as taken in table 1												
		1	2	3	4	5	6	7	8	9	10			
Lena	20	26.40	26.43	28.38	28.35	28.22	28.38	28.35	28.22	28.38	28.35	28.38	28.35	28.36
	50	23.00	23.84	25.27	25.30	23.32	25.29	25.30	23.32	25.29	25.30	23.32	25.29	25.30
	100	20.76	22.38	22.39	22.67	18.34	22.43	22.67	18.34	22.43	22.67	18.34	22.43	22.68
Barbara	20	23.39	24.13	25.18	25.12	25.39	25.16	25.12	25.39	25.16	25.12	25.39	25.16	25.12
	50	20.75	21.32	22.50	22.46	21.70	22.50	22.46	21.70	22.50	22.46	21.70	22.50	22.46
	100	19.11	20.33	20.65	20.78	17.69	20.67	20.78	17.69	20.67	20.78	17.69	20.67	20.79
Cameraman	20	24.93	27.71	27.03	26.95	27.00	27.00	26.95	27.00	27.00	26.95	27.00	27.00	26.95
	50	21.62	22.20	23.81	23.79	22.86	23.82	23.79	22.86	23.82	23.79	22.86	23.82	23.79
	100	19.44	20.55	21.21	21.39	18.16	21.24	21.39	18.16	21.24	21.39	18.16	21.24	21.40
Mandrill	20	21.21	24.43	22.77	22.81	23.26	22.97	22.81	23.26	22.97	22.81	23.26	22.97	22.81
	50	19.34	19.62	20.53	20.55	20.30	20.62	20.55	20.30	20.62	20.55	20.30	20.62	20.53
	100	18.51	19.09	19.30	19.28	17.00	19.16	19.28	17.00	19.16	19.28	17.00	19.16	19.29
Goldhill	20	27.67	31.20	27.30	27.32	27.34	27.37	27.32	27.34	27.37	27.32	27.34	27.37	27.31
	50	23.99	25.12	24.61	24.61	23.08	24.59	24.61	23.08	24.59	24.61	23.08	24.59	24.61
	100	21.41	22.58	22.44	22.36	18.11	22.03	22.36	18.11	22.03	22.36	18.11	22.03	22.36
Pepper	20	26.19	27.95	28.36	28.38	28.39	28.42	28.38	28.39	28.42	28.38	28.39	28.42	28.36
	50	22.52	23.45	25.08	25.10	23.76	25.05	25.10	23.76	25.05	25.10	23.76	25.05	25.10
	100	20.15	21.85	22.53	22.48	18.49	22.24	22.48	18.49	22.24	22.48	18.49	22.24	22.49

**Table 3.** Results (in PSNR) of denoising for Lena, Barbara, Cameraman, Mandrill, Goldhill, and Pepper images using Weiner filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, IIDMWT methods and our proposed one with various cases using block thresholding and shrinkage factor  $\alpha_{i,j}$ .

Image name	Noise levels	Proposed method with various cases as taken in table 1																	
		Weiner filter	Neigh Shrink	Neigh Block	ModiNeighShrink	LAWML	IAWDMBNC	IIDM WT	1	2	3	4	5	6	7	8	9	10	
Lena	20	28.99	28.58	30.17	28.89	28.77	29.64	29.13	30.72	30.76	30.72	30.87	30.76	30.72	30.87	30.72	30.72	30.72	30.72
	50	21.40	23.47	24.22	23.66	23.64	24.47	23.87	25.57	25.63	25.59	25.90	25.63	25.59	25.90	25.42	25.42	25.42	25.58
	100	15.49	22.06	22.06	22.11	22.09	22.51	22.26	22.71	22.69	22.68	22.84	22.69	22.68	22.84	22.66	22.66	22.66	22.69
Barbara	20	26.82	25.24	27.15	25.65	25.35	26.32	25.88	26.93	26.98	26.94	27.21	26.98	26.94	27.21	26.93	26.93	26.93	26.93
	50	20.66	21.07	21.52	21.19	21.13	21.60	21.34	22.26	22.30	22.27	22.42	22.30	22.27	22.42	22.26	22.26	22.26	22.27
	100	15.25	20.18	20.21	20.18	20.18	20.37	20.20	20.53	20.53	20.53	20.61	20.53	20.53	20.61	20.18	20.18	20.18	20.53
Cameraman	20	28.73	26.96	27.47	27.29	27.05	27.76	27.42	28.54	28.60	28.55	28.71	28.60	28.55	28.71	28.54	28.54	28.54	28.54
	50	21.23	21.90	22.64	22.18	21.88	23.01	22.37	24.10	24.15	24.11	24.29	24.15	24.11	24.29	24.10	24.10	24.10	24.10
	100	15.39	19.87	19.91	19.91	19.86	20.32	19.88	20.66	20.68	20.67	20.90	20.68	20.67	20.90	20.60	20.60	20.60	20.66
Mandrill	20	23.49	21.90	24.04	22.48	22.08	22.80	22.48	23.59	23.65	23.60	23.90	23.65	23.60	23.90	23.59	23.59	23.59	23.59
	50	19.13	19.37	19.56	19.46	19.38	19.60	19.46	19.88	19.92	19.89	20.05	19.92	19.89	20.05	19.88	19.88	19.88	19.89
	100	15.10	19.04	19.01	19.04	19.04	19.05	19.04	19.08	19.08	19.08	19.13	19.08	19.08	19.13	19.04	19.04	19.04	19.08
Goldhill	20	28.26	26.73	27.72	27.24	26.72	27.33	27.24	28.29	28.33	28.29	28.45	28.33	28.29	28.45	28.28	28.28	28.28	28.29
	50	21.27	23.43	23.81	23.67	23.45	23.78	23.67	24.56	24.59	24.56	24.74	24.59	24.56	24.74	24.45	24.45	24.45	24.56
	100	15.45	22.20	22.43	22.23	22.23	22.59	22.53	22.81	22.81	22.80	22.86	22.81	22.80	22.86	22.75	22.75	22.75	22.80
Pepper	20	29.01	28.45	29.11	29.09	28.59	29.37	29.09	30.29	30.33	30.29	30.37	30.33	30.29	30.37	30.29	30.29	30.29	30.29
	50	21.40	23.03	23.66	23.54	22.99	23.77	23.54	25.27	25.36	25.30	25.46	25.36	25.30	25.46	25.30	25.30	25.30	25.29
	100	15.48	21.48	21.48	21.48	21.48	22.51	21.48	22.00	22.01	21.97	22.32	22.01	21.97	22.32	21.48	21.48	21.48	21.98



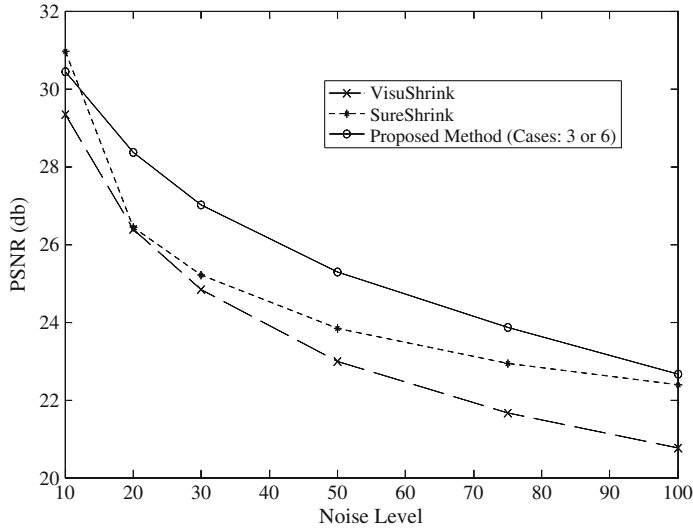


**Table 4b.** Results (in PSNR) of denoising for Lena, Barbara, Cameraman, Mandrill, Goldhill, and Pepper images using Weiner filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, IIDMWT methods and our proposed one with various cases using block thresholding and shrinkage factor  $\beta_{i,j}$  ( $n = 2$ ).

Image name	Noise levels	Weiner filter		Neigh Shrink		Neigh Block		ModiNeighShrink		LAWML		IAWDMBNC		IIDMWT		Proposed method with various cases as taken in table 1									
		WT	PSNRs (in db)	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10		
Lena	20	28.99	28.58	30.17	28.89	28.77	29.64	29.13	31.04	31.07	31.04	31.20	31.07	31.04	31.20	31.07	31.04	31.20	31.07	31.04	31.20	31.03	31.03	31.03	31.03
	50	21.40	23.47	24.22	23.66	23.64	24.47	23.87	25.78	25.84	25.79	26.10	25.84	25.80	26.10	25.84	25.80	26.10	25.84	25.80	26.10	25.53	25.53	25.53	25.79
	100	15.49	22.06	22.06	22.11	22.09	22.51	22.26	22.81	22.79	22.78	22.93	22.79	22.78	22.93	22.79	22.78	22.93	22.79	22.78	22.93	22.73	22.73	22.73	22.79
Barbara	20	26.82	25.24	27.15	25.65	25.35	26.32	25.88	27.72	27.77	27.73	27.99	27.77	27.73	27.99	27.77	27.73	27.99	27.77	27.73	27.99	27.73	27.73	27.73	27.72
	50	20.66	21.07	21.52	21.19	21.13	21.60	21.34	22.35	22.39	22.36	22.54	22.39	22.36	22.54	22.39	22.36	22.54	22.39	22.36	22.54	22.33	22.33	22.33	22.36
	100	15.25	20.18	20.21	20.18	20.18	20.37	20.20	20.59	20.59	20.58	20.68	20.59	20.58	20.68	20.59	20.58	20.68	20.59	20.58	20.68	20.18	20.18	20.18	20.58
Cameraman	20	28.73	26.96	27.47	27.29	27.05	27.76	27.42	29.22	29.29	29.23	29.43	29.29	29.23	29.43	29.29	29.23	29.43	29.29	29.23	29.43	29.23	29.23	29.23	29.22
	50	21.23	21.90	22.64	22.18	21.88	23.01	22.37	24.10	24.25	24.20	24.39	24.25	24.20	24.39	24.25	24.20	24.39	24.25	24.20	24.39	24.20	24.20	24.20	24.10
	100	15.39	19.87	19.91	19.91	19.86	20.32	19.88	20.75	20.77	20.75	21.02	20.77	20.75	21.02	20.77	20.75	21.02	20.77	20.75	21.02	20.61	20.61	20.61	20.75
Mandrill	20	23.49	21.90	24.04	22.48	22.08	22.80	22.48	24.30	24.37	24.30	24.67	24.37	24.30	24.67	24.37	24.30	24.67	24.37	24.30	24.67	24.30	24.30	24.30	24.30
	50	19.13	19.37	19.56	19.46	19.38	19.60	19.46	20.02	20.07	20.04	20.23	20.07	20.04	20.23	20.07	20.04	20.23	20.07	20.04	20.23	20.00	20.00	20.00	20.03
	100	15.10	19.04	19.01	19.04	19.04	19.05	19.04	19.11	19.12	19.11	19.16	19.12	19.11	19.16	19.12	19.11	19.16	19.12	19.11	19.16	19.04	19.04	19.04	19.11
Goldhill	20	28.26	26.73	27.72	27.24	26.72	27.33	27.24	28.64	28.69	28.64	28.84	28.69	28.64	28.84	28.69	28.64	28.84	28.69	28.64	28.84	28.61	28.61	28.61	28.64
	50	21.27	23.43	23.81	23.67	23.45	23.78	23.67	24.69	24.71	24.69	24.91	24.71	24.69	24.91	24.71	24.69	24.91	24.71	24.69	24.91	24.51	24.51	24.51	24.69
	100	15.45	22.20	22.43	22.23	22.23	22.59	22.53	22.82	22.82	22.81	22.88	22.82	22.81	22.88	22.82	22.81	22.88	22.82	22.81	22.88	22.75	22.75	22.75	22.82
Pepper	20	29.01	28.45	29.11	29.09	28.59	29.37	29.09	30.62	30.66	30.62	30.70	30.66	30.62	30.70	30.66	30.62	30.70	30.66	30.62	30.70	30.62	30.62	30.62	30.62
	50	21.40	23.03	23.66	23.54	22.99	23.77	23.54	25.39	25.47	25.41	25.61	25.47	25.41	25.61	25.47	25.41	25.61	25.47	25.41	25.61	25.41	25.41	25.41	25.41
	100	15.48	21.48	21.48	21.48	21.48	22.51	21.48	22.17	22.17	22.15	22.43	22.17	22.15	22.43	22.17	22.15	22.43	22.17	22.15	22.43	21.48	21.48	21.48	21.45

**Table 4c.** Results (in PSNR) of denoising for Lena, Barbara, Cameraman, Mandrill, Goldhill, and Pepper images using Weiner filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, IIDMWT methods and our proposed one with various cases using block thresholding and shrinkage factor  $\beta_{i,j}$  ( $n = 3$ ).

Image name	Noise levels	Weiner filter	Neigh Shrink	Neigh Block	ModiNeighShrink	LAWML	IAWDMBNC	IIDM		Proposed method with various cases as taken in table 1									
								WT	PSNRs (in db)	1	2	3	4	5	6	7	8	9	10
Lena	20	28.99	28.58	30.17	28.89	28.77	29.64	29.13	30.87	30.91	30.87	31.05	30.91	30.87	31.05	30.86	30.86	30.86	30.87
	50	21.40	23.47	24.22	23.66	23.64	24.47	23.87	25.58	25.64	25.60	25.90	25.64	25.60	25.90	25.38	25.38	25.38	25.59
	100	15.49	22.06	22.06	22.11	22.09	22.51	22.26	22.72	22.71	22.70	22.83	22.71	22.70	22.83	22.66	22.66	22.66	22.71
Barbara	20	26.82	25.24	27.15	25.65	25.35	26.32	25.88	27.44	27.49	27.45	27.73	27.49	27.45	27.73	27.96	27.96	27.96	27.44
	50	20.66	21.07	21.52	21.19	21.13	21.60	21.34	22.25	22.28	22.26	22.42	22.28	22.26	22.42	22.41	22.41	22.41	22.26
	100	15.25	20.18	20.21	20.18	20.18	20.37	20.20	20.54	20.54	20.53	20.62	20.54	20.53	20.62	20.18	20.18	20.18	20.53
Cameraman	20	28.73	26.96	27.47	27.29	27.05	27.76	27.42	28.95	29.02	28.96	29.17	29.02	28.96	29.17	28.96	28.96	28.96	28.95
	50	21.23	21.90	22.64	22.18	21.88	23.01	22.37	24.02	24.06	24.02	24.21	24.06	24.02	24.21	24.02	24.02	24.02	24.02
	100	15.39	19.87	19.91	19.91	19.86	20.32	19.88	20.62	20.64	20.63	20.87	20.64	20.63	20.87	20.54	20.54	20.54	20.62
Mandrill	20	23.49	21.90	24.04	22.48	22.08	22.80	22.48	24.01	24.08	24.02	24.39	24.08	24.02	24.39	24.01	24.01	24.01	24.01
	50	19.13	19.37	19.56	19.46	19.38	19.60	19.46	19.93	19.97	19.94	20.12	19.97	19.94	20.12	19.92	19.92	19.92	19.94
	100	15.10	19.04	19.01	19.04	19.04	19.05	19.04	19.09	19.10	19.09	19.14	19.10	19.09	19.14	19.04	19.04	19.04	19.09
Goldhill	20	28.26	26.73	27.72	27.24	26.72	27.33	27.24	28.47	28.52	28.47	28.67	28.52	28.47	28.67	28.45	28.45	28.45	28.47
	50	21.27	23.43	23.81	23.67	23.45	23.78	23.67	24.56	24.58	24.55	24.76	24.58	24.55	24.76	24.42	24.42	24.42	24.55
	100	15.45	22.20	22.43	22.23	22.23	22.59	22.53	22.78	22.77	22.77	22.83	22.77	22.77	22.83	22.72	22.72	22.72	22.77
Pepper	20	29.01	28.45	29.11	29.09	28.59	29.37	29.09	30.47	30.52	30.48	30.57	30.52	30.48	30.57	30.48	30.48	30.48	30.48
	50	21.40	23.03	23.66	23.54	22.99	23.77	23.54	25.19	25.27	25.22	25.41	25.27	25.22	25.41	25.21	25.21	25.21	25.21
	100	15.48	21.48	21.48	21.48	21.48	22.51	21.48	22.04	22.05	22.03	22.30	22.05	22.03	22.30	21.48	21.48	21.48	22.03



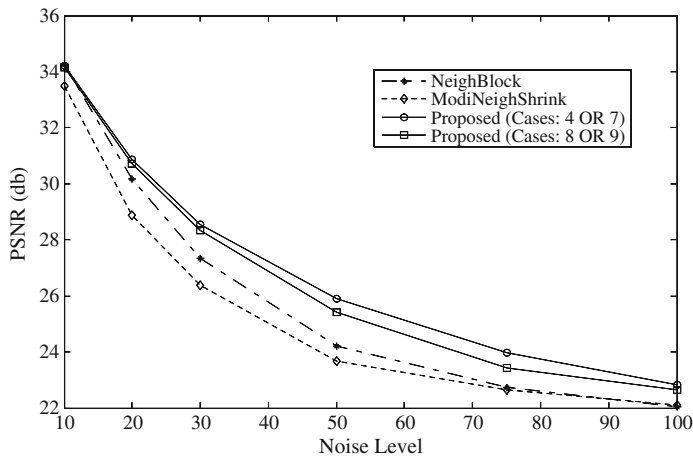
**Figure 5.** PSNR gain vs. Noise level for the proposed method with case 3 or 6 as taken from table 1 with VisuShrink, and SureShrink methods for Lena image using term-by-term thresholding.

et al 1999a) and IIDMWT (Om & Biswas 2012). The performance measure is evaluated in terms of PSNR, which is calculated as

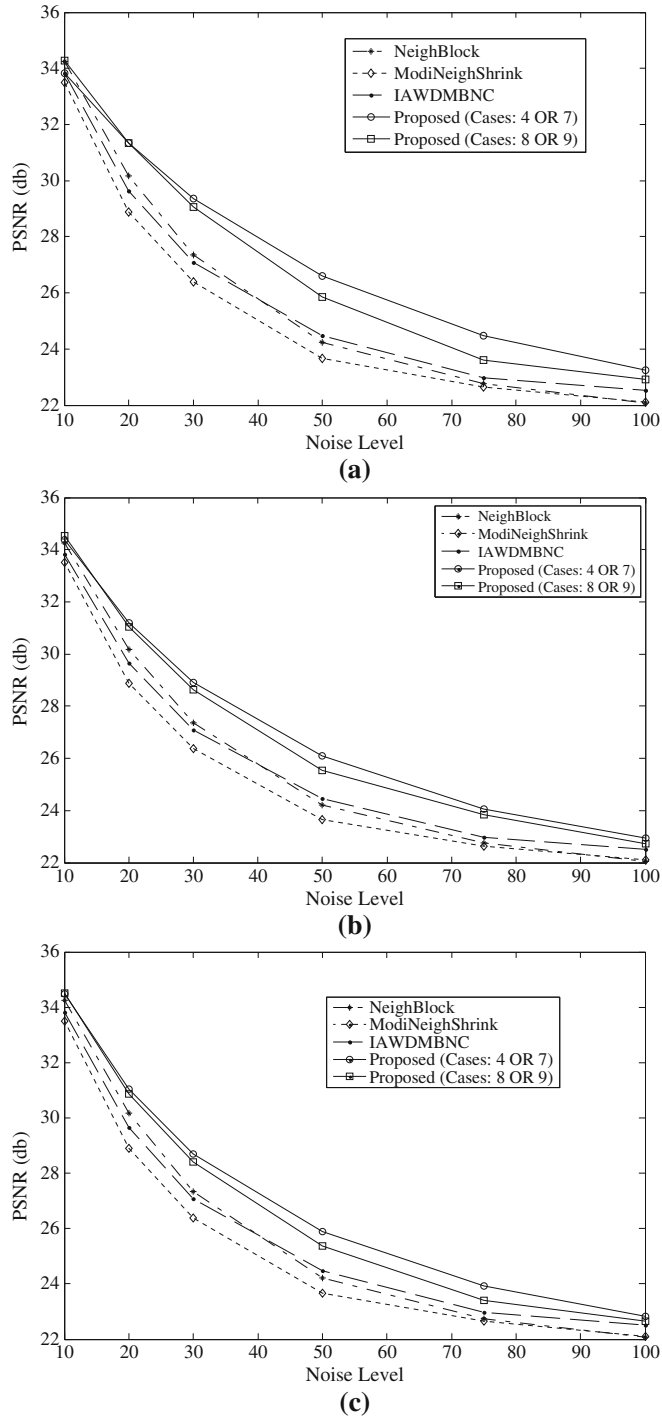
$$\text{PSNR (in db)} = 10 \log_{10} \left( (255)^2 / \text{MSE} \right),$$

where MSE is the mean squared error between the original and reconstructed images.

The results are shown in tables 2, 3 and 4. Table 2 contains the results of our proposed method, VisuShrink, and SureShrink by using term-by-term thresholding. From this table, we



**Figure 6.** PSNR gain vs. Noise level of proposed method with cases 4 or 7 and 8 or 9 as taken from table 1 with NeighShrink, NeighBlock, and ModiNeighShrink methods for Lena image using block thresholding with shrinkage factor  $\alpha_{i,j}$ .



**Figure 7.** PSNR gain vs. Noise level of proposed method with cases 4 or 7 and 8 or 9 as taken from table 1 with NeighBlock, ModiNeighShrink, and IAWDMBNC methods for Lena image using block thresholding with shrinkage factor  $\beta_{i,j}$  for: (a)  $n = 1$  (b)  $n = 2$  and (c)  $n = 3$ .

observe that the results of our proposed method are either better than or comparable in almost all the cases with that of the VisuShrink and SureShrink methods. The results for Lena image are shown graphically also in figure 5. As evident from figure 5, our proposed method performs better than the VisuShrink and SureShrink methods for almost all values of noise levels. Numerical results of the Wiener filter and block thresholding methods: NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, IIDMWT and our proposed method for various cases and different noise levels: 20, 50, 100, and using shrinkage factor  $\alpha_{i,j}$  are given in table 3 and that for the shrinkage factor  $\beta_{i,j}$  are given in tables 4a–4c, for  $n = 1, 2, 3$ , respectively. Using shrinkage factor  $\alpha_{i,j}$ , we observe that the results of our proposed method are either better than or comparable in almost all the cases with that of the Wiener filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, and IIDMWT methods. Figure 6 shows the results graphically for Lena image by using NeighBlock, ModiNeighShrink, and our proposed method (cases 4 or 7 and 8 or 9). The remaining methods have intermediary performances; hence they are not shown graphically. Using shrinkage factor  $\beta_{i,j}$  and  $n = 1$ , our method performs better than the Wiener filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, and IIDMWT methods for almost all cases (refer table 4a). Figure 7a shows the results graphically for Lena image using NeighBlock, ModiNeighShrink, IAWDMBNC, and our proposed method (cases 4 or 7 and 8 or 9). The remaining methods have intermediary performances; hence they are not shown graphically. For shrinkage factor  $\beta_{i,j}$  and  $n = 2$  and 3, our method performs consistently better than the Wiener filter, NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, and IIDMWT methods for all noise levels and all cases under consideration (refer tables 4b and 4c). Figure 7b and c show the results graphically for Lena image using NeighBlock, ModiNeighShrink, IAWDMBNC, and our proposed method (cases 4 or 7 and 8 or 9).

We also observe that our results are same for using individual term-by-term thresholding for cases: 1 and 2, 3 and 6, 4 and 7, 5 and 8; and for using the block thresholding for cases: 2 and 5, 3 and 6, 4 and 7, 8 and 9, because the values of the corresponding shrinkage factors are same. On comparing the results for the shrinkage factors  $\alpha_{i,j}$  and  $\beta_{i,j}$ , we observe that the results for latter are significantly better than that of the former (refer tables 3 and 4a–4c). It has also been observed that, for low noise level, taking higher value of  $n$  in shrinkage factor  $\beta_{i,j}$  gives good quality of the image. We can say that the denoised image contains more shrunked coefficients of the corrupted image, if the value of the new threshold ( $T_{New}$ ) is less than the square of the summation of wavelet coefficients in this neighbouring window ( $Sq$ ), i.e.,  $T_{New} < Sq$ . Thus, the resultant image has good visual quality. On the other hand, the denoised image is dominated by the noisy coefficients, if  $T_{New} \geq Sq$ . In this case, the original image cannot be recovered from the noisy one. The cases: 4 and 7 (in our proposed method) perform well among all the discussed cases since they satisfy the condition  $T_{New} < Sq$ .

## 6. Conclusion

In this paper, we have discussed a new image denoising method — an adaptive image denoising method based on local parameters optimization that addresses the issues of image recovery from its noisy one. Our method provides smoothness and good quality of the image by optimizing local parameters in the neighbourhood window. This method has either better or comparable performance with respect to the VisuShrink and SureShrink that follow term-by-term approach and the non-wavelet method, i.e., Wiener filter; and NeighShrink, NeighBlock, ModiNeighShrink, LAWML, IAWDMBNC, and IIDMWT that follow block thresholding approach.

## References

- Antoniadis A and Bigot J 2001 Wavelet estimators in nonparametric regression: A comparative simulation study. *J. Stat. Softw.* 6: 1–83
- Atto A M, Pastor D and Mercier G 2011 Wavelet shrinkage: unification of basic thresholding functions and thresholds. *SIViP* 5: 11–28
- Bhuiyan M I H, Ahmad M O and Swamy M N S 2008 Wavelet-based image denoising with the normal inverse Gaussian prior and linear MMSE estimator. *IET Image Process.* 2(4): 203–217
- Cai T T and Silverman B W 2001 Incorporating information on neighbouring coefficients into wavelet estimation. *Sankhya: Ind. J. Stat. B, Pt. 2*(63): 127–148
- Chen G Y and Bui T D 2003 Multiwavelets denoising using neighbouring coefficients. *IEEE Signal Processing Lett.* 10(7): 211–214
- Chen G Y, Bui T D and Krzyzak A 2004 Image denoising using neighbouring wavelet coefficients. *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004*. Proceedings. (ICASSP '04), 17–21 May 2004, Montreal, Quebec, Canada, 917–920
- Chen G, Zhu W P and Xie W 2012 Wavelet-based image denoising using three scales of dependency. *IET Image Process.* 6(6):756–760
- Coupe P, Manjon J V, Robles M and Collins D L 2012 Adaptive multiresolution non-local means filter for three-dimensional magnetic resonance image denoising. *IET Image Process.* 6(5): 558–568
- Donoho D L and Johnstone I M 1994 Ideal spatial adaptation via wavelet shrinkage. *Biometrika* 81(3): 425–455
- Donoho D L and Johnstone I M 1995 Adapting to unknown smoothness via wavelet shrinkage. *J. Am. Stat. Assoc.* 90(432): 1200–1224
- Donoho D L 1995 De-Noising by soft thresholding. *IEEE Trans. Inf. Theory* 41(3): 613–627
- Edwards T 1992 *Discrete wavelet transforms: theory and implementation*. Discrete Wavelet Transforms, Stanford University
- Gonzalez R C and Woods R E 2008 *Digital image processing*. Pearson Education
- Graps A 1995 An introduction to wavelets. *IEEE Comput. Sci. Eng.* 2(2): 50–61
- Guo Z B, Zhao Z G, Jiao Y and Lu H X 2008 Wavelet shrinkage image Denoising based on scales consistency edge detection. *Guangdianzi Jiguang/Journal of Optoelectronics Laser* 19(10): 1415–1418
- Jiang J, Guo J, Fan W and Chen Q 2010 An improved adaptive wavelet denoising method based on neighbouring coefficients (IAWDMNC). *Proc of World Congress on Intelligent Control and Automation, China*, 2894–2898
- Jain A K 1989 *Fundamental of digital image processing*. NJ: Prentice Hall
- Lal S and Chandra M 2012 Image denoising of wavelet based compressed images corrupted by additive white Gaussian noise. *Res. J. Appl. Sci. Eng. Technol.* 4(17): 3108–3118
- Liu G, Liu J, Wang Q and He W 2012 The translation invariant wavelet-based contourlet transform for image denoising. *J. Multimed.* 7(3): 254–261
- Mihcak M K, Kozintsev I and Ramchandran K 1999a Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising. *IEEE International Conference on Acoustics, Speech, and Signal Processing* 6: 3253–3256
- Mihcak M K, Kozintsev I, Ramchandran K and Moulin P 1999b Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Process. Lett.* 6(12): 300–303
- Mohideen S K, Perumal S A and Sathik M M 2008 Image de-noising using discrete wavelet transform (ModiNeighShrink). *Int. J. Comput. Sci. Netw. Sec.* 8(1): 213–216
- Om H and Biswas M 2012 An improved image denoising method based on wavelet thresholding (IIDMWT). *J. Signal Inf. Process. (USA)* 3(1): 109–116
- Shengqian W, Yuanhua Z and Daowen Z 2002a Adaptive shrinkage de-noising using neighbourhood characteristic. *Electron. Lett.* 38(11): 502–503
- Shengqian W, Yuanhua Z and Daowen Z 2002b Local characteristics based wavelet shrinkage denoising algorithm. *Electron. Lett.* 38(9): 411–412
- Yin M, Liu W and Wang Z C 2012 Infrared image denoising based on statistical model in nonsubsampling Contourlet transform domain. *Guangdian Gongcheng/Opto-Electronic Engineering* 39(8): 46–54