# DT-CWT Based Block Nonlinear De-noising

Sushil Kumar
Department of Mathematics, Rajdhani College, University of Delhi, New Delhi, India

**Article Info**

Article history:
Received 02 April 2016
Received in revised form
20 May 2016
Accepted 28 May 2016
Available online 15 June 2016

**Keywords**

DT-CWT, Wavelet Thresholding, BiShrink, BlockShrink, PSNR

**Abstract**

For image denoising, Block thresholding is considered to be a better strategy than the term-by-term thresholding. There are number of Wavelet based thresholding techniques for image denoising such as VisuShrink, BayesShrink, SureShrink, NeighShrink, BiShrink, ProbShrink, Sure-LET and BlockShrink. Selesnick has extended the BiShrink method to DT-CWT and he has shown that the DT-CWT achieves better results than DWT for image denoising. Dengwen and Xiaoliu have shown that BlockShrink enjoys a number of advantages over the other conventional image denoising methods. Their experimental results show that BlockShrink outperforms significantly classic SureShrink method and NeighShrink method. In this paper we extend BlockShrink approach to DT-CWT and compare it with the BiShrink approach proposed by Selesnick. We have analysed these methods of noise removal from degraded images with Gaussian noise and compared the results in term of PSNR.

## 1. Introduction

Image denoising is used to recover the image X from its noisy image, Y given by Y= X Θ ϵ, where ϵ is the contaminating noise and Θ is any composition of additive noise and/or multiplicative noise. Wavelet thresholding (or "shrinkage") is one of the techniques used for image deonoising. We first apply the wavelet transform, T, to the noisy image Y, and then apply a nonlinear estimation operator, D to the wavelet coefficients of high frequency sub-bands, either individually or in a group of coefficients. It has been shown that the reduction of absolute value in wavelet coefficients is successful in signal restoration [19]. Finally, compute the inverse transform, $T^{-1}$ to get an estimated image, X. In other words, X = $T^{-1}$ D (TY).

Two well-known shrinkage methods are hard thresholding and Soft thresholding [12]. Hard thresholding consists of setting to zero all wavelet coefficients whose magnitude is less than a threshold value whereas in Soft thresholding, the wavelet coefficients above the threshold are shrunk toward the origin. In practice, hard thresholding is preferred to soft thresholding, since for soft thresholding even large coefficients lying out of noise can shrunk and hence creates undesirable bias [27].

In fact, there are two basic approaches to modifying the coefficient, namely, probabilistic wavelet shrinkage and selective wavelet shrinkage. In the first method, the magnitude of the wavelet coefficient is reduced by the probability of its contribution to the overall quality of the image. The second method uses a binary method where the reduction of coefficient magnitude is either 0 or 1, i.e. coefficients are either selected or removed [1]. It is noted that the first method usually involves more computation not necessarily resulting in better performance. Balster et al. [1] have shown that the selective wavelet shrinkage method which either selects or rejects a wavelet coefficient is statistically better than the probabilistic method because the former can identify a narrow interval for the estimated parameter, which is used to adjust the wavelet coefficient, with a higher confidence level than the latter.

Some popular wavelet shrinkage methods are VisuShrink [13], BayesShrink [5], SureShrink[17], NeighShrink [6], BivariateShrink[25], ProbShrink [19], Sure-LET [3] and BlockShrink[9].

In the VisuShrink method, the wavelet coefficients are shrinked according to the soft-shrinkage rule using the universal Threshold, VisuShrink shows better denoising performance than the Unversal threshol, but it yield an overly smoothed images because the universal threshold T is high if the number of pixels in the image are high. Just like VisuShrink, SureShrink also applies the soft shrinkage rule, but it uses independently chosen thresholds for each subband through the minimization of the Stein's unbiased risk estimate (SURE) [26]. SureShrink performs better than VisuShrink, producing more detailed images.[10]. In Bayes shrink, thresholding is done at each sub band in the wavelet decomposition which improves outcome and also completely denoise the flat regions of the image. But it is less sensitive to the noise around edges [11].

In the *NeighShrink* approach, the wavelet coefficients are shrunk in overlapping blocks rather than individually or term by term as *VisuShrink* or *SureShrinK*. It is observed that *NeighShrink* outperforms *VisuShrink* and *SureShrink*.

BiShrink [24] uses a bivariate shrinkage function taking into account the intrascale variability of wavelet coefficients by capturing the dependence between a wavelet coefficient and its parent. ProbShrink [20] estimate the probability that a

**Corresponding Author,**
**E-mail address:** skazad@rajdhani.du.ac.in
Phone No--+91-971-123-4705

given coefficient contains a significant noise-free component. SURE-LET [18] directly parameterizes the deonoising, process as a sum of elementary nonlinear processes with unknown weights. It need not hypothesize a statistical model for the noiseless image while it minimizes an estimate of the mean squared error between the noiseless image and the denoised one by the SURE. Consequently, it computes the unknown weights by solving a linear system of equations. Bi Shrink, Prob Shrink and SURE-LET methods have all been devised for both redundant and non redundant wavelet transforms

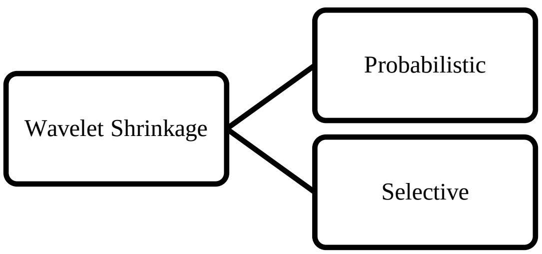


**Fig. 1** Block diagram of Wavelet Shrinkage methods

Block Shrink utilizes the pertinence of the neighbour wavelet coefficients by using the block thresholding scheme. It can decide the optimal block size and threshold for every wavelet subband by minimizing Stein's unbiased risk estimate (SURE). The block thresholding simultaneously keeps or kills all the coefficients in groups rather than individually, enjoys a number of advantages over the conventional term-by-term thresholding. The block thresholding increases the estimation precision by utilizing the information about the neighbor wavelet coefficients. It outperforms the classic SureShrink and NeighShrink [9], [11].

Dixit and Sharma [11] have observed that recently proposed wavelet methods like ProbShrink, BlockShrink and NeighShrink Sure produce better visual images.

Lal et al. [16] have compared performances of Separable DT-DWT, Real DT-DWT, Complex DT-DWT, RealDDT-DWT and Complex DDDT-DWT and observed that Complex DDDT-DWT outperforms the other wavelettransforms and is effective for the very highly corrupted images.

Chinnarao and Madhavilatha [7] have proposed a conatextual information based thresholding method in DT-CWT and observed that their method is highly suitable at high noise levels as compared to low noise levels.

In this paper, we extend the BlockShrink approach to DT-CWT and compare it with the BiShrink approach proposed by Selesnick. Since it is noted that many of the wavelet based denoising algorithms are suffering from shift variance and lack of directionality, Dual Tree Complex Wavelet Transform (DT-CWT) and complex Double Density Dual Tree Discrete Wavelet Transform (DDDT-DWT) have been proposed to decompose the image and shrinkage operation to eliminate the noise from the noisy image. We analyse these methods of noise removal from degraded images with Gaussian noise and compare the results in term of PSNR.

In the subsequent section 2, we summarize the features of Dual Tree Complex Wavelet Transform. In Section 3, we give the experimental results by implementing the proposed image denoising algorithms using Matlab 7.0 on different grey scale image formats of size 512 x 512.

**2. Dual Tree Complex Wavelet Transform**

Dual-Tree complex wavelet transform ([21],[14]) uses two real DWT trees to implement its real part and imaginary part, separately that result in decomposition with a much higher degree of directionality than that possessed by the traditional DWT. There are two versions of the 2D dual-tree wavelet transform: the real 2-D dual-tree DWT is 2-times expansive, while the complex 2-D dual-tree DWT is 4-times expansive. Both types have wavelets oriented in six distinct directions. The real 2-D dual-tree DWT of an image is implemented using two critically-sampled separable 2-D DWTs in parallel. Then for each pair of sub-bands we take the sum and difference. The six wavelets associated with the real 2D dual-tree DWT are strongly oriented in {+15º, +45º, +75º, -15º, -45⁰, -75º} direction (see Fig. 2). The complex 2-D dual-tree DWT also gives rise to wavelets in six distinct directions, however, in this case there are two wavelets in each direction ( as shown in Fig. 3). In each direction, one of the two wavelets can be interpreted as the real part of a complex-valued 2D wavelet, while the other wavelet can be interpreted as the imaginary part of a complex-valued 2D wavelet. Because the complex version has twice as many wavelets as the real version of the transform, the complex version is 4-times expansive.

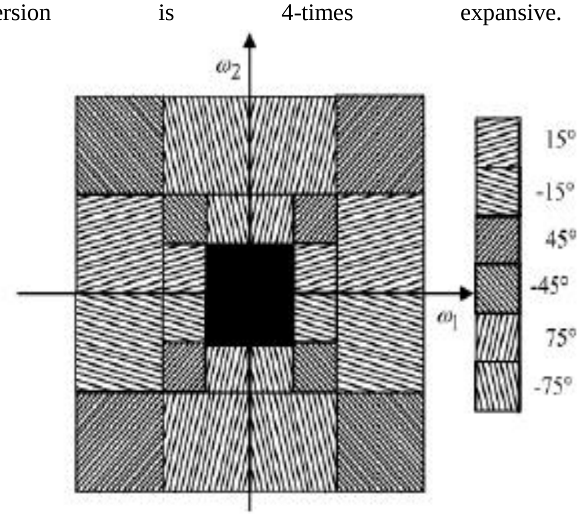


**Fig. 2** Frequency domain partition in DT-CWT resulting from two level decomposition [14, 21]

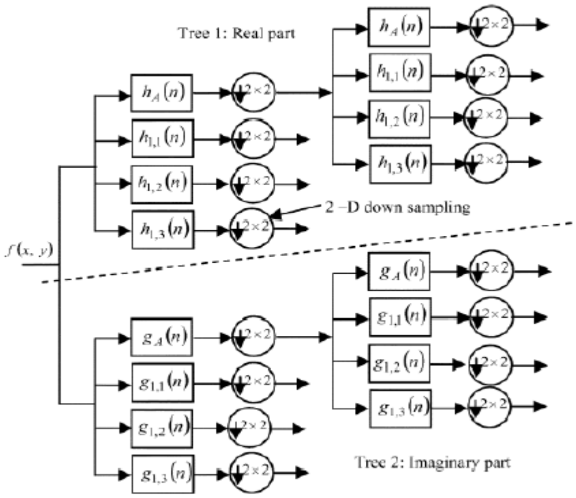


**Fig. 3** 2-D dual- tree complex wavelet transform[14, 21]

### 3. Review of Thresholding Functions

In this section we present the basic steps of two thresholding functions used in this paper. These are *Bivariate Shrinkage function* and *Block Shrinkage funciton.* The basic steps of wavelet based image denoising are:

1. Decompose corrupted image by noise using wavelet transform.
2. Compute threshold in wavelet domain and apply to noisy coefficients.
3. Apply inverse wavelet transform to reconstruct image.

Sender and Selesnick [23] provide the Matlab implementation of both wavelet-based denoising using the separable DWT and wavelet-based denoising using the dual-tree DWT with bivariate shrinkage function. This shrinkage function requires the prior knowledge of the noise variance and the signal variance for each wavelet coefficient. The main matlab files provided for the Bivaraiate Shrinkage usig DT-CWT are:

1. main_dtdwt.m: It loads the noisy image, calls the denoising routine and calculates the PSNR value of denoised image.
2. denoising_dtdwt.m: It call number of subfunctions for the calculations of the local adaptive image denoising.
   2a. symextend.m: This function is used to extend the noisy image using symmetric extension in order to reduce the boundary problem
   2b. cplxdual2D.m: It calculates the forward dual-tree DWT
   2c. expand.m: The parent matrix is expanded using this function in order to make the matrix size the same as the coefficient matrix.
   2d. bishrink.m: It estimates the magnitude of the complex coefficients. The coefficients are estimated using the magnitudes of the complex coefficient, its parent and the threshold value with this function.
   2e. icplxdual2D.m: It calculates the inverse wavelet transform

The package provided by Dengwen [8] contains the Matlab codes for denoisinig grey scale images using BlockShrink implemented with a decimated wavelet transform. The main matlab files provided are:

1. denoisefun.m: This function is the denoising main function using BlockShrink
2. SubbandThresholding.m: This function is used to threshold the noisy subband according to BlockShrink rule
3. parameters.m: This function select the optimal block size and corresponding threshold
4. Parameters.mexw32: This file is significantly faster than Parameters.m
5. Calc_MSE_PSNR: for computing MSE and PSNR
6. Test: This script is for denoising demonstration.

### 4. Proposed Algorithms

We present the image denoisinig algorithms using Bivariate Shrink thresholding and BlockShrink thresholding implemented with a dual tree complex wavelet transform. The matlab codes for Bivariate Shrinkage thresholding implemented on DT-DWT are provided by Selesnick [23], and for Blockshrink thresholding implemented on DWT are provided by Dengwen [8] as discussed in above section.

The basic steps of these algorithms are given below.

**Algorithm 1 (*BiShrink* [23])**

………………………………………………….

1. Read the image, s and resize it to 512 x 512.
2. Obtain the noisy image x, given as x = s + g, where g is the additive white Gaussian noise
3. Set the window size (=7).
4. Extend the noisy image using symmetric extension in order to improve the boundary problem with the function symextend.m.
5. Perform the 2D Dual tree DWT to level J = 6. using cplxdual2D.m.
6. Estimate the noise variance. The noise variance will be calculated using the robust median estimator.
7. Process each subband seperately in a loop. First the real and imaginary parts of the coefficents and the corresponding parent matrices are prepared for each subband, and the matrices corresponding to the real and imaginary parts of the parent matrix are expanded using a function expand.m in order to make the matrix size the same as the coefficient matrix.
8. Estimate the signal variance and the threshold value : The signal variance for each coefficent is estimated using the window size , and the threshold value for each coefficient will be calculated and stored in a matrix with the same size as the coefficent matrix.
9. Estimate the magnitude of the complex coefficients. The coefficients will be estimated using the magnitudes of the complex coefficient,

its parent and the threshold value with a Matlab function bishrink.m.

10. Calculate the inverse wavelet transform using icplxdual2D.m.
11. Extract the image. The neccessary part of the final image is extracted in order to reverse the symetrical extension.

…………………………………………

**Algorithm 2 (*BlockShrink* [8])**

………………………………………………

1. Read the image s, and resize it to 512 x 512.
2. Corrupt the image by additive White Gaussian Noise. It is given as x = s + g where x is noisy image corrupted by additive white Gaussian noise g of standard deviation . Both s and x are of same sizes.
3. Perform the 2D Dual tree DWT to x to level J = 5 using cplxdual2D.m.
4. Estimate the noise variance, sigma, using robust median estimator, if it is provided by the user
5. Normalise the noise level of the noisy coefficients.
6. Extract all six detail subbands in each scale.
7. Apply different threshold values with soft Thresholding for each detail subband coefficients with the function 'SubbandThresholding'.
8. Reconstruct the denoised image by taking the inverse DT-DWT.

……………………………………………………..

## 5. Experimental Results

In this section, we give the results obtained on implementing the proposed method of image denoising, i.e., Bivariate Shrink and Block Shrink based on DT-CWT thresholding methods on Matlab 7.0. The algorithm is implemented on different grey scale image formats of size 512 x 512, and some of the results are summarized in Table 1 and Table 2, respectively. The Table 3 summarizes the results of proposed algorithms for the brain image: mri_jpe wtth standard deviation, sigma of additive Gaussian noise taken from 10 to 100, respectively. The results of the tables 1 to 3 are shown through fig. 4 to fig. 6, respectively.

**Table 1.** PSNR values of denoised images resulting from the proposed method based on DT-CWT

| Image | Bishrink | BlockShrink |
|---|---|---|
| barbara.png | 33.3797 | 33.6762 |
| lena.png | 35.8798 | 35.1921 |
| c2.bmp | 36.3445 | 36.0390 |
| mri.jpe | 38.9919 | 39.3364 |
| lena.jpg | 38.4773 | 38.3097 |
| new7.tif | 37.9214 | 37.8906 |
| new11.tif | 35.8946 | 35.8222 |
| new12,tif | 38.1366 | 38.2194 |
| shagun.jpg | 36.4878 | 36.1837 |
| zoneplate.png | 28.0138 | 32.7194 |
| peppers.jpg | 36.9105 | 37.0584 |
| tooth1.jpg | 35.1693 | 35.5138 |
| cameraman.tif | 36.8094 | 36.7104 |
| New8.tif | 34.5610 | 34.5968 |

**Table2.** Comparison of BiShrink and BlockShrink image denoising approaches using wavelet thresholding and DT-DWT thresholding methods (I1: barbara.png; I2:lena.png; I3:parliament.bmp; I4:lena.jpg)

| | BiSh_ DWT | BiSh_DT DWT | BlSh_ DWT | BlSh_ DTDWT |
|---|---|---|---|---|
| I1 | 35.5457 | 33.3797 | 35.727 | 33.6762 |
| I2 | 37.7429 | 35.8798 | 37.8393 | 35.1921 |
| I3 | 34.8633 | 36.3445 | 34.6432 | 36.0398 |
| I4 | 32.1841 | 38.4773 | 32.3865 | 38.3097 |

**Table 3**: Results of Proposed algorithms for the brain image: mri_jpe

| Sigma | Bi_ Shrink | Block Shrink |
|---|---|---|
| 10 | 38.996 | 39.3364 |
| 40 | 31.2812 | 31.5986 |
| 80 | 27.5422 | 27.9882 |
| 100 | 26.4751 | 26.8314 |

Volume 4, Issue2 (2016) 333-338 ISSN 2347 - 3258
International Journal of Advance Research and Innovation

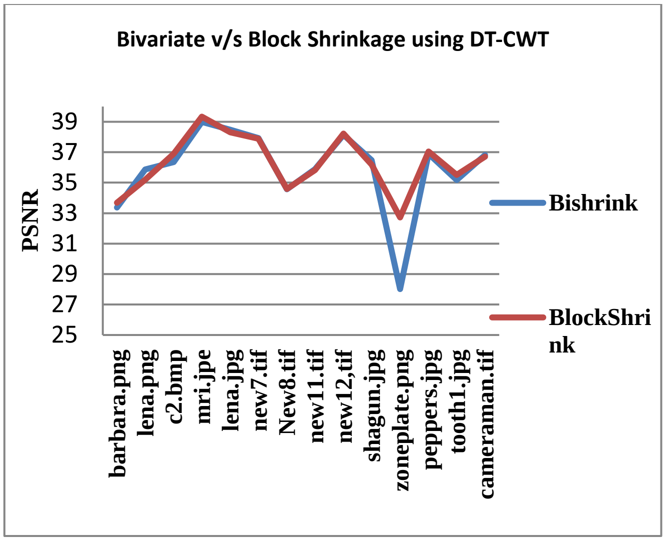


**Fig.4** The results of PSNR of proposed method based on Bishrink and BlockShrink thresholding methods

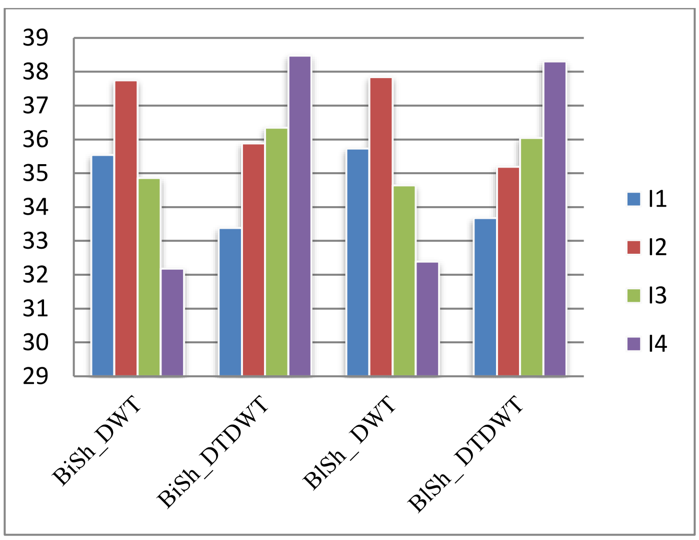


**Fig.5** PSNR values obtained of proposed algorithms compared with corresponding Wavelet based thresholding techniques for four images

From Fig. 4, we observe that overall Block shrinkage method using DT-DWT performs better than Bivariate Shrinkage method using DT-CWT for image denoising in terms of PSNR. In Fig. 6, we observe that even when noise density is increasing, Block shrink based proposed algorithm performs better than Bi-Shrink based algorithm in terms of PSNR. Finally, we compared the proposed algorithms based on Bi-Shrink function and Block Shrink function using DT-CWT with the corresponding Wavelet based thresholding, and summarized the results for some of the images in Fig. 5. We observe that the proposed algorithms perform better for the .bmp and .jpg formats but fail to show better results in terms of PSNR for .png image formats. Further, from the Fig. 4, one can observe that DT-CWT based denoising using Block Shrink function gives better result in terms of PSNR for the image "Zoneplate.png" that contains much more contours than the other images, as compared to DT_CWT based denoising using Bi-Shrink function. Some of the noised and denoised images obtained from the implementation of proposed algorithms are shown in Fig. 7 below.

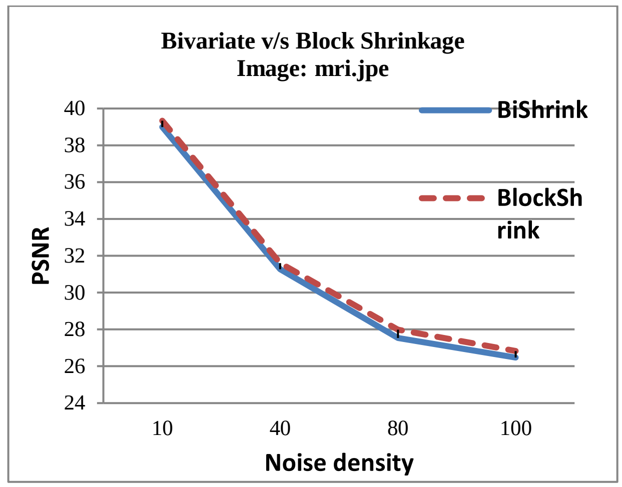


**Fig.6** The results of proposed method based on Bishrink and Block Shrink thresholding methods for the barin image: mri_jpe

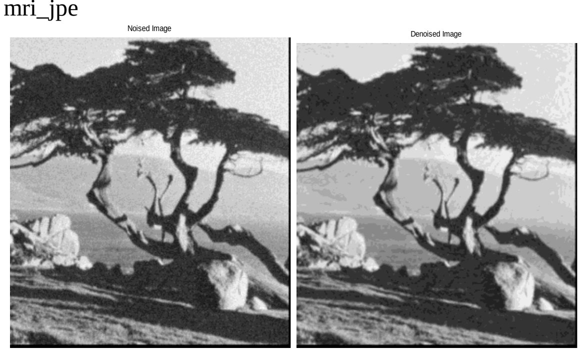

**Fig.7 (a)** The noised and denoised images new11.tif

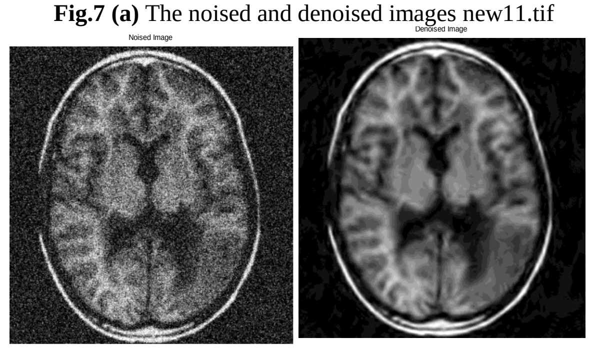

**Fig.7(b)** The noised and denoised images mri.jpe

## 6. Conclusions

Image denoising is an active and challenging topic of research. The major challenge lies in the fact that one does not know what the original signal is for a corrupted image. We have presented the DT-CWT based denoising using BiShrink function and BlockShrink functiion and compared the results for different grey scale image formats in terms of PSNR. We observe that Block Shrink outperform BiShrink as well as earlier wavelet domain methods.

## Acknowledgement

We wish to thank I. W. Selesnick and Zhou Dengwen for providing their Matlab software for the image denoising for the purpose of research to us.

## References

[1] Balster Eric J., Zheng Y. F., and Ewing R. L., Feature-Based Wavelet Shrinkage Algorithm for Image Denoising, IEEE Transactions on Image Processing, 14,(12), 2005

[2] Bhonsle D., Dewangan S., “Comparative Study of Dual-Tree Complex Wavelet Transform and Double Density Complex Wavelet Transform for Image Denoising Using Wavelet-Domain”, International Journal of Scientific and Research Publications, 2, 7, 2012

[3] Blu T., Luisier F., “The SURE-LET approach to image denoising”, IEEE Trans. Image Process., 16, 2007,2778–2786

[4] Cai T.T. Zhou H. H. A Data-Driven Block Thresholding Approach To Wavelet Estimation. Ann.Statist., http://www.stat.yale.edu/~hz68/.

**[**5**]** Chang S. G., Yu B., and Vetterli M. Adaptive wavelet thresholding for image denoising and compression. IEEE Transaction Image Processing, 9, 2000, 1532-1546.

[6] Chen G.Y., Bui T.D., Krzyzak A. Image denoising using neighboring wavelet coefficients. Proceedings of IEEE international conference on Acoustics, speech and signal processing 04, 2004, 917-920

[7] Chinnarao B. Madhavilatha M. Improved Image denoising algorithm using Dual Tree Complex Wavelet Transform. International Journal of Computer Applications, 44, 20, 2012, 1-4

[8] Dengwen Zhou, 24430-BlockShrink-denoising, <http://www.mathworks.com/matlabcentral/ fileexchange/>

[9] Dengwen Zhou and Xiaoliu Shen, Image denoising using block thresholding,in Proc. 2008 congress on image and signal processing,Sanya, Hainan, China, 2008, 335–338

[10] Dengwen Zhou, Wengang Cheng. Image denoising with an optimal threshold and neighboring window. Elsevier pattern Recognition, 29, 11, 2008, 1694–169

[11] Dixit A., Sharma P., A Comparative Study of Wavelet Thresholding for Image Denoising, I.J. Image, Graphics and Signal Processing, 12, 2014, 39-46

[12] Donoho D.L., Johnstone I.M. Ideal spatial adaptation via wavelet shrinkage. Biometrika, 81, 1994, 425–455

[13] Donoho D.L., Johnstone I.M. Adapting to unknown smoothness via wavelet shrinkage. Journal of the American Statistical Association, 90(432), 1995, 1200-1224

[14] Kingsbury N.: Image processing with complex wavelets. Phil. Trans. Royal Society London A, 9 (29), 1999, 2543 – 2560

[15] Kumar S. Image Denoising using Wavelet-like Transform, ICASET, 2016, 21-22

[16] Lal S., Chandra M., Upadhyay G.K., Gupta D., Removal of Addditive Gaussian noise by complex double density dual tree discrete wavelet transform, MIT International Journal of Electronics and Communication Engineering, 1, 1, 2011, 8-16

[17] Luisier F., Blu T., Unser M., “A new SURE approach to image denosing: interscale orthonormal wavelet thresholding”, IEEE Transactions on Image Processing, 16 (3), 2007, 593-606

[18] Luisier F., Blu T., Unser M. SURE-LET Matlab Codes. 2007b.<http://bigwww.epfl.ch/demo/suredenoising/matlab.html>.

[19] Piˇzurica A. and Philips W. Estimating the probability of the presence of a signal of interest in multi resolution single and multiband image denoising. IEEE Trans. Image Process. 15, 3, 2006, 654–665

[20] Pizzeria A., Philips W. ProbShrink Matlab Codes, 2006 b. Located at the URL: <ttp:// telin.ugent.be/~sanja>.

[21] Selesnick I. W., The design of approximate Hilbert transform pairs of wavelet bases, IEEE Trans. Signal Processing, **5**(50), 2002, 1144 - 1152

[22] Sendur L. Selesnick I. W. Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency,” IEEE Trans. Signal Processing, 50,11, 2002, 2744–2756

[23] Sendur L. Selesnick I.W. DT-CWT and BiShrink Matlab codes, 2002b. <ttp://taco.poly.edu/WaveletSoftware/>.

[24] Selesnick I. W., Wavelet Software at Brooklyn Poly, <eeweb.poly.edu/iselesni/Wavelet Software/denoise2.html>

[25] Sendur L. and Selesnick I. W. Bivariate shrinkage with local variance estimation, IEEE Signal Process. Lett. 9, 12, 2002, 438–441

[26] Stein C.Estimation of the mean of a multivariate normal distribution, Ann. Statist. 9, 1981, 1135–1151

[27] Starck Jean-Luc, Murtagh Fionn, and Fadili Jaial M. Sparse Image and Signal Processing. Cambridge University Press, 2010