Image and Vision Computing 28 (2010) 1611–1623

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A new wavelet-based fuzzy single and multi-channel image denoising

Jamal Saeedi [a,*], Mohammad Hassan Moradi [b], Karim Faez [a]

[a] Department of Electrical Engineering, Amirkabir University of Technology (Tehran Polytechnic), 424, Hafez Ave., Tehran, Iran
[b] Department of Bioelectric, Biomedical Engineering, Amirkabir University of Technology (Tehran Polytechnic), 424, Hafez Ave., Tehran, Iran

ARTICLE INFO

Article history:
Received 16 September 2009
Received in revised form 17 February 2010
Accepted 30 April 2010

Keywords:
Image denoising
Dual-tree discrete wavelet transform
Fuzzy membership function
Multi-channel image

ABSTRACT

In this paper, we propose a new wavelet shrinkage algorithm based on fuzzy logic. In particular, intra-scale dependency within wavelet coefficients is modeled using a fuzzy feature. This feature space distinguishes between important coefficients, which belong to image discontinuity and noisy coefficients. We use this fuzzy feature for enhancing wavelet coefficients' information in the shrinkage step. Then a fuzzy membership function shrinks wavelet coefficients based on the fuzzy feature. In addition, we extend our noise reduction algorithm for multi-channel images. We use inter-relation between different channels as a fuzzy feature for improving the denoising performance compared to denoising each channel, separately. We examine our image denoising algorithm in the dual-tree discrete wavelet transform, which is the new shiftable and modified version of discrete wavelet transform. Extensive comparisons with the state-of-the-art image denoising algorithm indicate that our image denoising algorithm has a better performance in noise suppression and edge preservation.

1. Introduction

Denoising has become an essential step in image analysis. Indeed, due to sensor imperfections, transmission channels defects, as well as physical constraints, noise weakens the quality of almost every acquired image. Three main types of noise exist: impulse noise, additive noise, and multiplicative noise. Impulse noise is usually characterized by some portion of image pixels that are corrupted, leaving the remaining pixels unchanged. Examples of impulse noise are fixed-valued impulse noise and randomly valued impulse noise. We consider here additive noise when a value from a certain distribution is added to each image pixel, for example, a Gaussian distribution. Multiplicative noise is generally more difficult to remove from images than additive noise because the intensity of the noise varies with the signal intensity (e.g., speckle noise) [1].

Because of the importance and commonality of preprocessing in most image and video systems, there has been an enormous amount of research dedicated to the subject of noise removal, and many different mathematical tools have been proposed. Variable coefficient linear filters [2,3], adaptive nonlinear filters [4,5], DCT-based solutions [6], cluster filtering [7], genetic algorithms [8], fuzzy logic [9,10], etc., have all been proposed in the literature.

The main goal of an image denoising algorithm is then to reduce the noise level, while preserving the image features (such as edges, textures, etc.). The multi-resolution analysis performed by the wavelet transform has been shown to be a powerful tool to achieve these goals [11]. Indeed, in the wavelet domain, the noise is uniformly spread throughout the coefficients, while most of the image information is concentrated in the few largest ones. Classical wavelet-based denoising methods consist of three steps (see Fig. 1):

1) Compute the discrete wavelet transform (DWT) or other kinds of multi-resolution transforms.
2) Remove noise from the wavelet coefficients.
3) Reconstruct the enhanced image by using the inverse wavelet transformation.

Due to the linearity of the wavelet transform, additive noise in the image domain remains additive in the transform domain. If $\gamma_{s,d}(i,j)$ and $x_{s,d}(i,j)$ denote the noisy and the noise-free wavelet coefficients of scale s and orientation d respectively, then we can model the additive noise in the transform domain as:

$$y_{s,d}(i,j) = x_{s,d}(i,j) + n_{s,d}(i,j) \quad (1)$$

where $n_{s,d}(i,j)$ is the corresponding noise component. We will only consider additive Gaussian white noise following a normal law defined by a zero mean and a known[1] variance, that is $n \sim N(0,\sigma^2)$.

The denoising process is known as wavelet shrinkage or thresholding. Among the various thresholding strategies, soft thresholding is the most popular and has been theoretically justified by Donoho and Johnstone [12]. These researchers have shown that the shrinkage rule

* Corresponding author. Tel.: +98 935 3592459; fax: +98 21 66468186.
E-mail addresses: jamal.saeedi@yahoo.com (J. Saeedi), mhmoradi@aut.ac.ir (M.H. Moradi), kfaez@aut.ac.ir (K. Faez).

[1] In practice, the noise standard deviation can be accurately estimated using a robust median estimator [11].

0262-8856/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2010.04.004

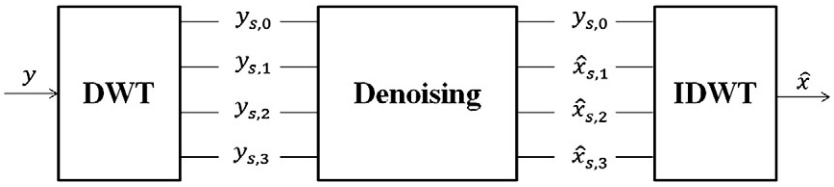

Fig. 1. Principle of wavelet denoising.

is near-optimal in the minimax sense and provided the expression of the optimal threshold value T, called the "universal threshold" as a function of the noise power σ^2 when the number of samples N is large: $T = \sqrt{2\sigma^2 \log N}$. The use of the universal threshold for denoising images in the wavelet domain is known as *VisuShrink* [13]. *VisuShrink* is very simple, but its disadvantage is to yield to overly smoothed images because the universal threshold T is too large. Just like *VisuShrink*, *SureShrink* also applies to the soft shrinkage rule, but it uses independently chosen thresholds for each sub-band through the minimization of the Stein's unbiased risk estimate (SURE) [14]. *SureShrink* performs better than *VisuShrink*, producing more detailed images.

After the influential work of Donoho and Johnstone, many alternative methods have come forth. One of the most popular was proposed by Chang et al., who derived their threshold in a Bayesian framework, assuming a generalized Gaussian distribution for the wavelet coefficients. This solution to the wavelet denoising problem is known as *BayesShrink* [15] and has a better MSE performance than *SureShrink*.

However, algorithms that exploit the different kinds of dependencies between the wavelet coefficients can result in better denoising performance, compared with the ones derived using an independence assumption. The wavelet coefficients are statistically dependent mainly due to two properties of the wavelet transform of natural images:

1) Large coefficients will propagate across the scales (inter-scale dependencies).
2) If a coefficient is large/small, some of the neighboring coefficients are also likely to be large/small (intra-scale dependencies).

Among the methods in which inter-scale dependency for image denoising have used: Sendur and Selesnick used a bivariate shrinkage function, which models the statistical dependence between a wavelet coefficient and its parent. It needs to estimate the marginal variance of the coefficient in a local neighborhood [16,17]; Luisier et al. directly parameterized the denoising 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 [18], and among the methods in which intra-scale dependency for image denoising have used: Chen et al. used local mean of neighbor coefficients in wavelet sub-bands as a feature to shrink wavelet coefficients and their method is called *NeighShrink* [19]; Pizurica and Philips used a probabilistic shrinkage function. Its core is estimating the probability that a given coefficient contains a significant noise-free component. Then the wavelet coefficient is multiplied with the probability [20]; Schulte et al. introduced a fuzzy version of probabilistic shrinkage method. Its core is shrinkage based on local mean of wavelet coefficients and some fuzzy rules [21].

Many other techniques have combined inter- and intra-scale dependencies. For example, denoising methods based on Gaussian Scale Mixture models, often employ the neighboring coefficients on the same and adjacent scales [22]. Local contextual HMT models have been developed, which capture both inter-scale and intra-scale information [23,24].

In image denoising, where a trade-off between noise suppression and the maintenance of actual image discontinuity must be made, solutions are required to detect important image details and accordingly adapt the degree of noise smoothing. In this paper, we model the intra-scale dependency in wavelet transform domain as a fuzzy feature. We always postulate that noise is uncorrelated in the spatial domain; it is also uncorrelated in the wavelet domain. With respect to this principle, we use a fuzzy feature for single channel image denoising to enhance image information in wavelet sub-bands, and then using a fuzzy membership function to shrink wavelet coefficients, accordingly. This feature space distinguishes between important coefficients, which belong to image discontinuity and noisy coefficients. In multi-channel images, different channels are correlated: an image discontinuity from one channel is probably to occur in at least some of the remaining channels. As for this reason and uncorrelated nature of noise, we use a new feature to extend our method for denoising multi-channel images.

DWT provides a good time frequency analysis of the signal, with a non-redundant signal representation and an optimal representation of singularities. Nevertheless, DWT suffers from five fundamental, intertwined shortcomings [25]: oscillations, aliasing, shift-variance, poor directionality, and absence of phase information. Shift invariance and directional selectivity are essential to the quality of wavelet-based image denoising results. Because of the down-sampling operation in the DWT filter bank (FB), it is shift-variance and will cause some visual artifacts (such as Gibbs phenomena) in thresholding-based denoising [26]. In addition, if the directional selectivity of a FB is defined as the ability to extract directional features into separate images, then the 2-D DWT has very poor directional selectivity because 2-D DWT has four sub-images, which are usually referred to as LL, LH, HL, and HH images.

Many solutions to the shift-variance and lack of directionality of the DWT have been suggested in the literature. A simple approach to shift-variance is to remove the decimation blocks in the FB, so that there is no aliasing in the output sub-band signals. In this case, the sub-bands signals are perfectly shift-invariant (undecimated discrete wavelet transform). The new properties resulting from the use of this highly redundant transformation have been obtained at the expense of the loss of orthogonality, a considerably more intensive memory usage, and a higher computational cost than that of the original DWT. The latter point becomes a major concern in multi-channel image denoising, particularly when the number of channels is large. However, a reduced form of translation invariance exists, namely, energy shift-invariance or "shiftability" [27], which means that the energy of the output signal is shift-invariant. In this paper, we examine our image denoising algorithm in the dual-tree DWT (DT-DWT), which provides both shiftable sub-bands and good directional selectivity and low redundancy [28,29] and [30].

This paper is organized as follows: In Section 2, the new fuzzy image denoising scheme for single channel image is explained. In addition, we formulate our image denoising algorithm for multi-channel image in Section 3. Section 4 gives various experimental results and performance comparisons. Finally, we conclude with a brief summary in Section 5.

2. Single channel image denoising

In image denoising, where a trade-off between noise suppression and the maintenance of actual image discontinuity must be made, solutions are required to detect important image details and accordingly adapt the degree of noise smoothing. We always postulate that noise is uncorrelated in the spatial domain; it is also uncorrelated in the wavelet domain. With respect to this principle, we use a fuzzy feature for single channel image denoising to enhance image information in wavelet sub-bands and then using a fuzzy membership function to shrink wavelet coefficients, accordingly. This feature space distinguishes between important coefficients, which belong to image discontinuity and noisy coefficient.

2.1. Fuzzy feature

We want to give large weights to neighboring coefficients with similar magnitude, and a small weight to neighboring coefficients with dissimilar magnitude. The larger coefficients, which are produced by noise, are always isolated or unconnected, but edge coefficients are clustered and persistent. It is well known that the more adjacent points are more similar in magnitude. So we use a fuzzy function $m(l,k)$ of magnitude similarity and a fuzzy function $s(l,k)$ of spatial similarity [31], which is defined as:

$$m(l,k) = \exp\left(-\left(\frac{y_{s,d}(i,j)-y_{s,d}(i+l,j+k)}{Thr}\right)^2\right) \tag{2}$$

$$s(l,k) = \exp\left(-\left(\frac{l^2+k^2}{N}\right)\right) \tag{3}$$

where $y_{s,d}(i,j)$ and $y_{s,d}(i+l,j+k)$ are central coefficient and neighbor coefficients in the wavelet sub-bands, respectively. $Thr = c\times\hat{\sigma}_n$, $3\le c\le 4$, $\hat{\sigma}_n$ is estimated noise variance, and N is the number of coefficients in the local window $k\in[-K\ldots K]$, and $l\in[-L\ldots L]$.

The following example shows spatial similarity for a 5×5 local window (i.e. $L=2$, and $K=2$). The central weight is set to zero that is very useful for removing isolated noise:

$$s(l,k) = \begin{pmatrix} 0.73 & 0.82 & 0.85 & 0.82 & 0.73 \\ 0.82 & 0.92 & 0.96 & 0.92 & 0.82 \\ 0.85 & 0.96 & 0 & 0.96 & 0.85 \\ 0.82 & 0.92 & 0.96 & 0.92 & 0.82 \\ 0.73 & 0.82 & 0.85 & 0.82 & 0.73 \end{pmatrix}$$

According the two fuzzy functions, we can get adaptive weight $w(l,k)$ for each neighboring coefficient:

$$w(l,k) = m(l,k)\times s(l,k) \tag{4}$$

Using the adaptive weights $w(l,k)$, we can obtain the fuzzy feature for each coefficient in the wavelet sub-bands as follows:

$$f(i,j) = \frac{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k)\times\left|y_{s,d}(i+l,j+k)\right|}{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k)} \tag{5}$$

In the literature, the local mean also is used for improving the wavelet coefficients' information in the shrinkage step [19,20] and [21]. Fig. 2 shows the difference between the fuzzy feature and local mean for distinguishing between important coefficients, which belong to edge structure and noisy coefficient in wavelet transform domain. As it can be seen in Fig. 2, the new fuzzy feature can effectively distinguish between edge structure and noise compare to the local mean.

2.2. Fuzzy shrinkage rule

The second step in the wavelet denoising procedure usually consists of shrinking the wavelet coefficients: the coefficients that contain primarily noise should be reduced to negligible values, while the ones containing a significant noise-free component should be reduced less. Here, we use a fuzzy rule based on the fuzzy feature for shrinking the wavelet coefficients.

Fuzzy logic was proposed by Zadeh [32], and has application in a large number of fields. The fuzzy sets and fuzzy rules form the knowledge base of a fuzzy rule-based reasoning system. Fuzzy rules are linguistic IF-THEN constructions that have the general form "IF A THEN B", where A and B are (collections of) propositions containing linguistic variables. A is called the premise or antecedent and B is the consequence of the rule [33]. After finding the fuzzy feature, we will form Linguistic IF-THEN rules for shrinking wavelet coefficients as follows:

IF the fuzzy feature $f(i,j)$ is *large* THEN shrinkage of wavelet coefficients $y_{s,d}(i,j)$ is *small*.

In fact, the fuzzy feature indicates how coefficients in the noisy sub-band should be shrunk. Fuzzy membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1.MF is often given the designation of μ. The input space is sometimes referred to as the universe of discourse, a fancy name for a simple concept. The only condition a membership function must really satisfy is that it must vary between 0 and 1. The function itself can be an arbitrary curve whose shape we can define as a function that suits us from the point of view of simplicity, convenience, speed and efficiency. MF is built from several basic functions such as: piece-wise linear functions, the Gaussian distribution function, the sigmoid curve, quadratic and cubic polynomial curves.

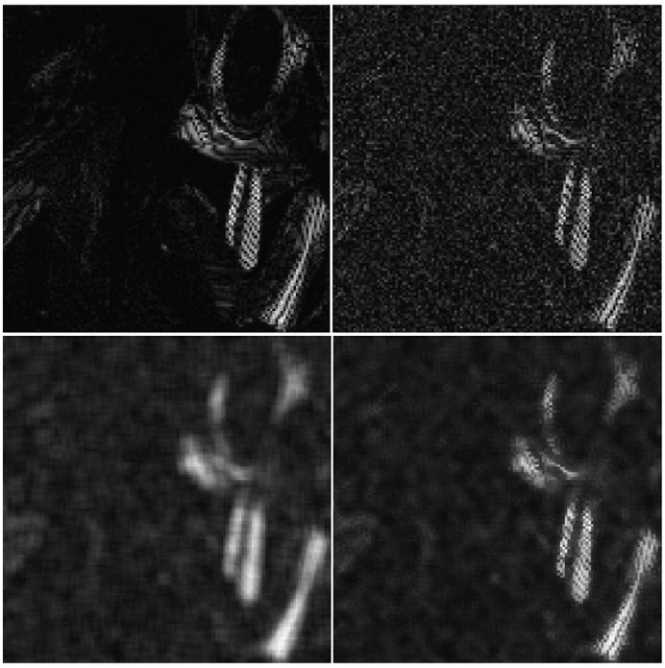

Fig. 2. From top left, clockwise: a sub-band of DT-DWT for "Barbara" image, noisy sub-band, the fuzzy feature, and local mean.

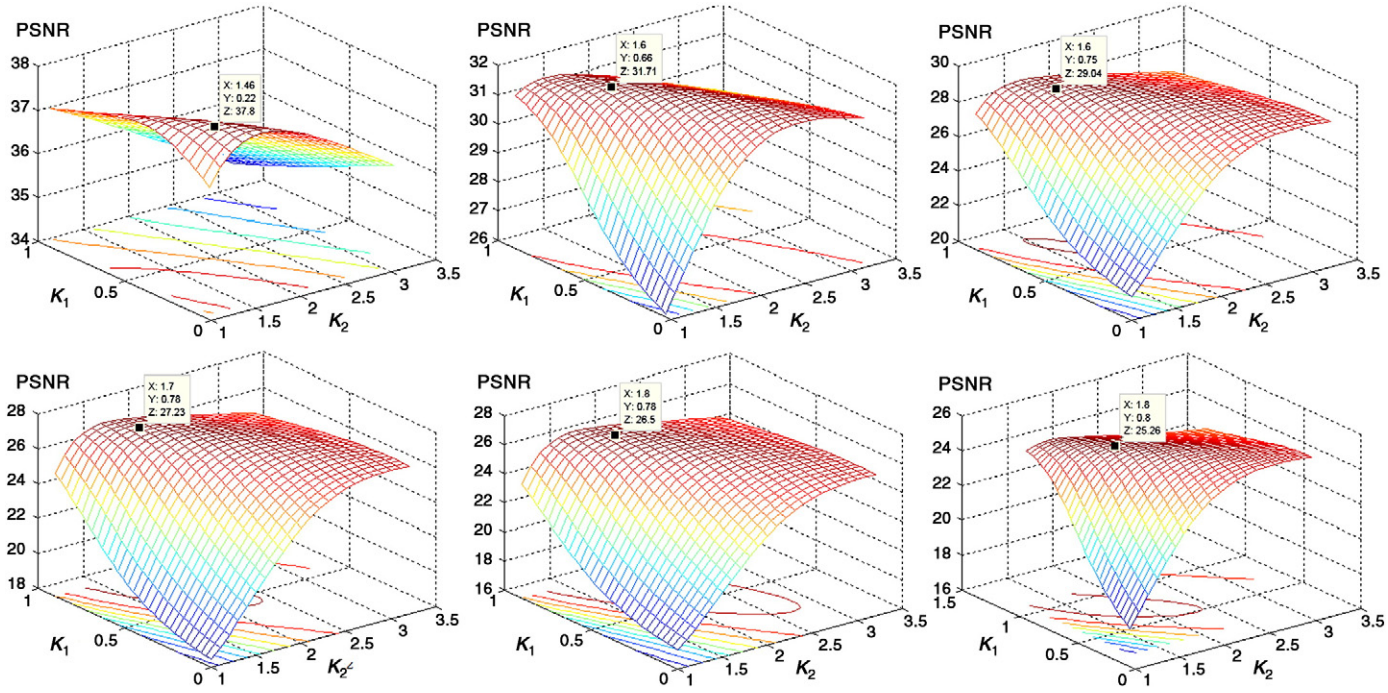

Fig. 3. 3D curves to obtain best values for the K_1 and K_2 at the different noise variances, from top left, clockwise: 5, 15, 25, 35, 40, and 50.

Here, we use the spline-based curve, which is a mapping on the vector $\boldsymbol{x}$, and is named because of its S- shape. The parameters T_1 and T_2 locate the extremes of the sloped portion of the curve as given by:

$$\mu(x) = \begin{cases} 0 & x \le T_1 \\ 2 \times \left(\frac{x - T_1}{T_2 - T_1}\right)^2 & T_1 \le x \le \frac{T_1 + T_2}{2} \\ 1 - 2 \times \left(T_2 - \frac{x}{T_2 - T_1}\right)^2 & \frac{T_1 + T_2}{2} \le x \le T_2 \\ 1 & x \ge T_2 \end{cases} \tag{6}$$

Finally, the estimated noise-free signal is obtained using the following formula:

$$\hat{x}_{s,d}(i,j) = \mu(f(i,j)) \times y_{s,d}(i,j) \tag{7}$$

where s and d are scale and orientation of wavelet sub-bands, and $f(i,j)$ is the fuzzy feature obtained using Eq. (5).

For building fuzzy membership function, two thresholds (T_1 and T_2), must be determined. We found out that T_1 and T_2 are related with the $\hat{\sigma}_n$, which is the estimated noise variance. In order to find these relations, we have done some experiments using test images. We found out that T_1 and T_2 have nonlinear relation with the $\hat{\sigma}_n$. For achieving the nonlinear relation, we have tested our noise reduction algorithm with the different noisy images. In each different noise variance, we obtained best values for T_1 and T_2. As it can be observed in Fig. 3, best values for K_1 and K_2 based on following equations are obtained for different noise variances:

$$T_1 = K_1 \times \hat{\sigma}_n \tag{8}$$

$$T_2 = K_2 \times \hat{\sigma}_n \tag{9}$$

where K_1 and K_2 are constant values, and $\hat{\sigma}_n$ is the estimated noise variance using *median estimator* [12].

We use polynomial curve fitting in MATLAB toolbox, and fit a fifth order polynomial on the points of K_1 and K_2 versus estimated noise

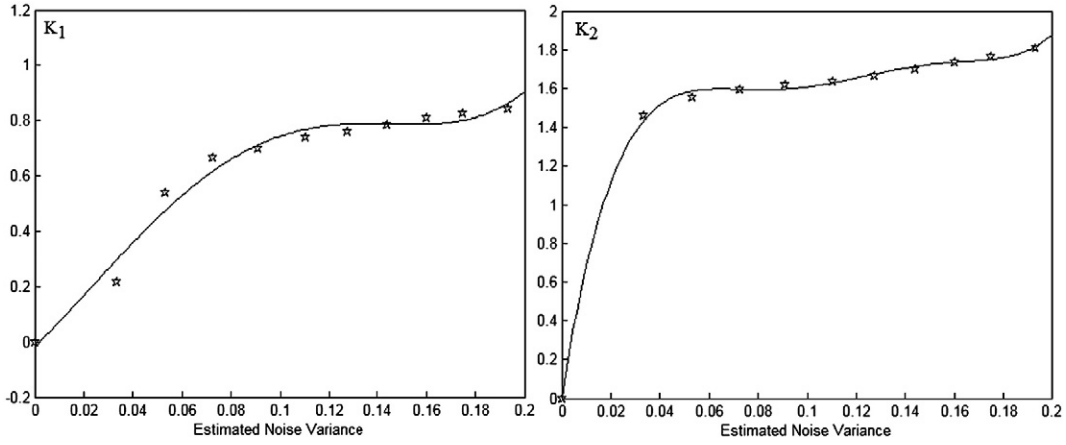

Fig. 4. K_1 and K_2 versus estimated noise variances, and the fitted polynomials on these points.

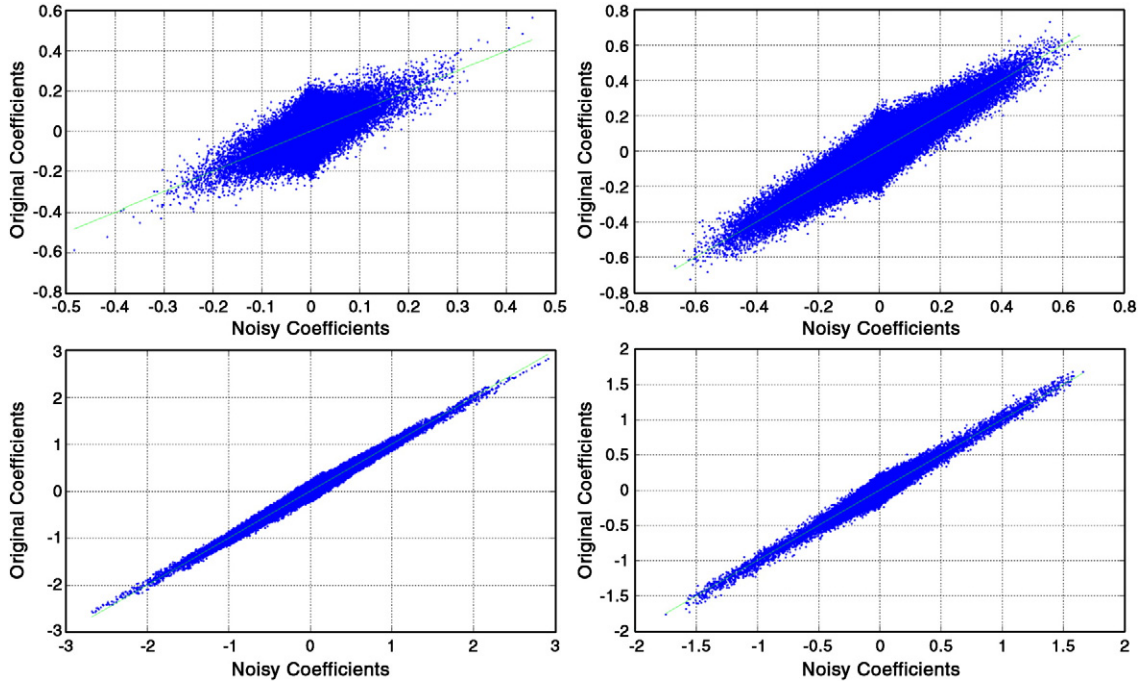

Fig. 5. Noisy coefficients versus original ones in the different level of decompositions. From top left, clockwise: first level, second level, third level and fourth level of decomposition.

variances (see Fig. 4). We use polynomial coefficients to obtain best values for T_1 and T_2 at different noise variances:

$$T_1 = \left(\sum_{i=1}^{5} p_{1_i} \times (\hat{\sigma}_n)^i\right) \times \hat{\sigma}_n \tag{10}$$

$$T_2 = \left(\sum_{i=1}^{5} p_{2_i} \times (\hat{\sigma}_n)^i\right) \times \hat{\sigma}_n \tag{11}$$

where p_1 and p_2 are the polynomials coefficients:

$$p_1 = -8.87_{e+04}, 4.64_{e+04}, -8.5_{e+03}, 595.8, -4.58, -0.0037,$$

$$p_2 = 1.11_{e+05}, -6.67_{e+04}, 1.52_{e+04}, -1.63_{e+03}, 82.8, 0.0027$$

When a noisy image passes low frequency filter banks of wavelet transform at each level, it will be smoother, and therefore, noise and high frequency regions in image (i.e. edges) are reduced. Fig. 5 shows noisy coefficients versus original ones in the different level of wavelet decomposition for a sample image. As it can be seen in Fig. 5, in coarser level, linearity of this curve is increased, which means in coarser level of wavelet decomposition, because of low-pass filtering of the noisy image, it will be more similar to the noise-free image. For this reason, we add a descending term to the T_2. In other words, wavelet coefficients in lower levels are less shrunk. Here, we use exponential descending function for this purpose; however, it is possible to use other descending functions:

$$T_2 = \left(\sum_{i=1}^{5} p_{2_i} \times (\hat{\sigma}_n)^i\right) \times \hat{\sigma}_n \times \exp\left(-\left(\frac{l-1}{T_3}\right)\right) \tag{12}$$

where l is the level of decomposition, and $8<T_3<10$.

In order to implement our fuzzy-shrink method, two parameters (the size of local window for fuzzy feature extraction and level of wavelet decomposition), must be defined. We found out that these two parameters are directly related to the estimated noise variance (using median estimator). Therefore, we use an adaptive method for selecting the parameters. Indeed, this method is a simple fuzzy IF-THEN rule, which assigns smaller local window and smaller level of decomposition when the estimated noise variance is small, and vice versa. Table 1 shows the choices for the size of local window and level of wavelet decomposition in the different noise variances, which are obtained using test images and try and error.

2.3. Post-processing

Processing artifact usually result from a modification of the spatial correlation between wavelet coefficients (often caused by zeroing of small neighboring coefficients) or using DWT, which is shift invariance and will cause some visual artifacts (such as Gibbs phenomena) in thresholding-based denoising. For this reason, we use a fuzzy filter on the results of our fuzzy-shrink algorithm to reduce artifacts to some extent. First, we use a window of size $(2L+1)\times(2K+1)$ centered at (i,j) to filter the current image pixel at position (i,j). Next, we calculate the similarity of neighboring pixels to the center pixel using following formula:

$$m(l,k) = \exp\left(-\left(\frac{\hat{X}(i,j)-\hat{X}(i+l,j+k)}{Thr}\right)^2\right) \tag{13}$$

where $\hat{X}(i,j)$ and $\hat{X}(i+l,j+k)$ are central pixel and neighbor pixels in the denoised image, respectively. N is the number of pixels in the local window $k\in[-K\ldots K]$, and $l\in[-L\ldots L]$, and $2.55<Thr<7.65$.

Table 1
The size of local window and level of wavelet decomposition at different noise levels.

Estimated noise variance	$0<\hat{\sigma}_n<10$	$0<\hat{\sigma}_n<20$	$30<\hat{\sigma}_n<40$	$40<\hat{\sigma}_n<50$
Window size	3×3	5×5	7×7	9×9
Level of decomposition	$N-5$	$N-4$	$N-3$	$N-2$

N is the possible number of decomposition level for an image (for a 512×512 image, $N=8$).

Table 2
Some well-known triangular norms.

T-norms	Operation
Minimum	$\min(x,y)$
Algebraic product	(x,y)
Weak	$\begin{cases} \min(x,y) & if\ \max(x,y)=1 \\ 0 & otherwise \end{cases}$
Bounded sum	$\max(0,x,y-1)$

According the fuzzy feature, we can get adaptive weight $w(l,k)$ for each neighboring coefficient:

$$w(l,k) = m(l,k) \times s(l,k) \tag{14}$$

where $s(l,k)$ is obtained using Eq. (3).

Finally, the output of post-processing step is determined as follows:

$$\tilde{X}(i,j,c) = \frac{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k) \times \hat{X}(i+l,j+k,c)}{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k)} \tag{15}$$

where $\hat{X}$ is the denoised image, which is obtained via our fuzzy-shrink algorithm.

3. Multi-channel image denoising

In this section, we consider image with C channels. Typically, C is equal to three-color channels for RGB images, but for biological (fluorescence) and multi-channel satellite images, C might be larger. We denote these multi-channel images by:

$$X = [x(i,j,1), x(i,j,2) \ldots x(i,j,C)] \tag{16}$$

These images are corrupted by an additive Gaussian white noise following a normal law defined by a zero mean and a known σ^2 variance, that is $n \sim N(0,\sigma^2)$:

$$N = [n(i,j,1), n(i,j,2) \ldots n(i,j,C)] \tag{17}$$

We denote the resulting noisy image by:

$$Y = X + N \tag{18}$$

Due to the linearity of the wavelet transform, additive noise in the image domain remains additive in the transform domain. Then we can model the additive noise in the transform domain as:

$$Y_{s,d} = X_{s,d} + N_{s,d} \tag{19}$$

where s and d are the scale and orientation of wavelet sub-bands.

The easiest method for denoising multi-channel image is simply to apply an existing denoising algorithm separately in each channel. However, this solution is beyond optimal, due to the presence of strong common information between the various channels.

In literatures, inter-channel information in both spatial and wavelet domain is used for image denoising. S. Schulte et al. used a fuzzy filter in spatial domain for multi-channel image denoising. The filter consists of two sub-filters. The first sub-filter computes the fuzzy distance between the color components of the central pixel and its neighborhood. These distances determine to what degree each component must be corrected. The second sub-filter corrects pixels where the color components differences are corrupted so much that they appear as outliers in comparison to their environment [1]. Luisier's et al. multi-channel *Sure-Let* work follows their *SURE-LET* approach [16] where the denoising algorithm is parameterized as a linear expansion of thresholds (LET) and optimized using Stein's unbiased risk estimate (SURE). The proposed wavelet thresholding function is point-wise and depends on the coefficients of same location in the other channels, as well as on their parents in the

Fig. 6. Test images used in the experiments. From top left, clockwise: Barbara, Peppers, Boat, Goldhill, Cameraman and Zelda.

coarser wavelet sub-band [34]. Pizurica et al. have also adapted their original *ProbShrink* by including the inter-channel dependencies in the definition of their local spatial activity indicator [20].

3.1. Fuzzy multi-channel feature

In multi-channel images, different channels are correlated: an image discontinuity from one channel is probably to occur in at least some of remaining channels. As for this reason and uncorrelated nature of noise, we use a new feature to extend our method for denoising of multi-channel images.

Improving wavelet coefficients' information in the shrinkage step is the main idea. We use two fuzzy features for this purpose. The first feature is similar to the fuzzy feature for single channel image denoising, which is a nonlinear averaging in the local neighborhood of central pixels as:

$$f_1(i,j,c) = \frac{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k,c) \times \left|y_{s,d}(i+l,j+k,c)\right|}{\sum_{l=-L}^{L}\sum_{k=-K}^{K} w(l,k,c)} \tag{20}$$

where $w(i,j,c)$ is obtained from Eq. (5), and C is the number of image channels. In addition, we use estimated noise variance in each channel $\hat{\sigma}_n^c$ for calculating the magnitude similarity $m(l,k,c)$ in Eq. (2). In other words, we use the estimated noise variance in each channel for calculating its corresponding weights, separately.

The second fuzzy feature that we have used for multi-channel image denoising is defined as:

$$f_2(i,j) = \frac{\sum_{c=1}^{C}\left(\sum_{l=-L}^{L}\sum_{k=-K}^{K}\left(\prod_{c=1}^{C} w(l,k,c)\right) \times \left|y_{s,d}(i+l,j+k,c)\right|\right)}{C \times \sum_{l=-L}^{L}\sum_{k=-K}^{K}\left(\prod_{c=1}^{C} w(l,k,c)\right)} \tag{21}$$

The second fuzzy feature equally is used to shrink wavelet coefficients in each channel. In fact, for denoising one channel, the second feature simultaneously takes the remaining channels' information into account in the denoising process. Then, we use the Takagi–Sugeno model based on the two fuzzy features for shrinking wavelet coefficients [35].

3.2. Fuzzy shrinkage rule

Having the fuzzy features, we will define Linguistic IF-THEN rules for shrinking wavelet coefficients in each channel as follows:

IF $f_1(i,j,c)$ (the first fuzzy feature) is *large* AND $f_2(i,j)$ (the second fuzzy feature) is *large* THEN shrinkage of wavelet coefficient in channel c i.e. $y_{s,d}(i,j,c)$ is *small*.

The idea behind this simple fuzzy rule is to assign small shrinkage weights to the wavelet coefficients, which have similar values to the neighbors and corresponding coefficients in the remaining channels. For color images, it is important to treat pixels as color components not as three separate colors. When only the separate channels are considered, more artifacts are introduced.

For building standard rule from linguistic rule, each linguistic value is represented by a membership function. The membership functions that are used to represent the two fuzzy sets of large $f_1(i,j,c)$ (the first fuzzy feature) and large $f_2(i,j)$ (the second fuzzy feature), are denoted as $\mu(f_1(i,j,c))$ and $\mu(f_2(i,j))$, respectively.

Here, we use the spline-based curve as fuzzy membership function, which is a mapping on the vector $\boldsymbol{x}$, and is named because of its S-shape. The parameters T_1 and T_2 locate the extremes of the sloped portion of the curve, as given by Eq. (7). For the pair of values $(\mu(f_1(i,j,c)),\mu(f_2(i,j)))$, the degree of satisfaction of the antecedent part of the rule determines the *firing strength* of the rule:

$$\tau(i,j,c) = f_1(i,j,c) \;\; AND \;\; f_2(i,j) \tag{22}$$

The AND operation is typically implemented as minimum, but any other triangular norms (t-norm) may be used. Originally, t-norms appeared in the context of probabilistic metric spaces [36]. Then they were used as a natural interpretation of the conjunction in the semantics of mathematical fuzzy logics and they are used to combine criteria in multi-criteria decision making [37]. Some well-known triangular norms are shown in Table 2. In such applications, the minimum or product t-norm is usually used because of a lack of motivation for other t-norms [36]. We have chosen algebraic product for the AND operation, because it is the standard semantics for strong conjunction in product fuzzy logic and continues as a function.

$$\tau(i,j,c) = f_1(i,j,c) \times f_2(i,j) \tag{23}$$

Finally, the estimated noise-free signal is obtained using the following formula:

$$\hat{x}_{s,d}(i,j,c) = \tau(i,j,c) \times y_{s,d}(i,j,c) \tag{24}$$

Similar to single channel image denoising, for building fuzzy membership function we used test images in different noise levels, to obtain best values for the thresholds. In addition, in the post-processing step we use the following formula to calculate the similarity of neighboring pixels to the center pixel instead of using Eq. (13):

$$m(l,k) = \prod_{c=1}^{C} \exp\left(-\left(\frac{\hat{X}(i,j,c) - \hat{X}(i+l,j+k,c)}{Thr}\right)^2\right) \tag{25}$$

where $\hat{X}(i,j,c)$ and $\hat{X}(i+l,j+k,c)$ are central pixel and neighbor pixels in the denoised image of channel c, respectively. N is the number of pixels in the local window $k \in [-K \ldots K]$, and $l \in [-L \ldots L]$, and $2.55 < Thr < 7.65$.

4. Experimental results

In this section, we compare our fuzzy denoising algorithm with some of the best state-of-the-art techniques. First, we demonstrate the results of single channel image denoising algorithm, and then, the results of multi-channel image denoising will be illustrated. Finally, the computation time of different algorithms will be discussed.

4.1. Single channel image denoising results

In this section, we compare our single channel image denoising algorithm with some of the best state-of-the-art techniques: Sendur's et al. bivariate MAP estimator with local variance estimation [18], Pizurica's *ProbShrink* [20], Luisier's et al. *Sure-Let* [16], Chen's et al. *NeighShrink* [19], Fan's *HMT* [24] and Portilla's *BLS-GSM* [22]. Besides these methods, there are other promising algorithms such as Dabov's *BM3D* [38], Takeda's *kernel regression* [39], and Elad's *K-SVD* [40]. Here, we have only used previous wavelet-based methods for a fair comparison.

It should be mentioned that for comparing some methods, which are proposed in the discrete wavelet transform domain, we use a critically sampled orthonormal wavelet basis with eight vanishing moments (sym8) over four decomposition stages in our fuzzy-shrink method. In addition, for a fair comparison between methods, which have used a redundant wavelet transform, we use dual-tree discreet wavelet transform over four decomposition stages.

Table 3
Comparison of some of the most efficient denoising methods.

σ_n	5	10	15	20	25	30	50	5	10	15	20	25	30	50
Method	Barbara 512×512							Peppers 512×512						
Input PSNR	34.14	28.11	24.61	22.13	20.30	18.78	14.75	34.21	28.21	24.74	22.30	20.41	18.92	14.86
HMT (DWT)	31.62	28.18	26.20	25.05	24.27	23.80	22.54	34.34	32.15	30.52	29.23	28.26	27.44	25.01
NeighShrink (DWT)	32.69	29.04	26.87	25.24	24.02	23.33	21.91	33.39	31.49	29.97	28.66	27.62	26.73	24.42
ProbShrink (DWT)	36.00	31.57	29.65	28.06	27.01	26.15	23.63	35.38	33.26	29.88	29.22	28.45	27.93	25.92
Sure-let (DWT)	36.71	32.18	29.66	27.98	26.76	25.83	23.70	35.74	33.66	32.21	31.00	29.88	28.88	25.89
BLS-GSM Non-Redundant	36.63	32.29	29.88	28.24	27.05	26.14	23.82	36.69	33.76	32.05	30.78	29.77	28.94	25.96
Fuzzy-Shrink (DWT)	36.72	32.58	30.21	28.68	27.48	26.57	23.93	36.80	33.72	32.01	30.75	29.69	28.79	25.91
	36.90	32.87	30.45	28.90	27.63	26.68	23.91	37.08	34.28	32.50	31.20	30.05	29.08	26.10
Bi-Shrink (DT-DWT)	36.75	33.17	30.85	29.13	27.74	26.47	22.88	37.07	33.52	31.00	29.19	27.51	26.13	22.41
BLS-GSM Best Redundant	37.62	33.66	31.31	29.66	28.40	27.38	24.70	37.17	34.46	32.89	31.71	30.75	29.86	27.71
Fuzzy-Shrink (DT-DWT)	37.81	33.96	31.72	30.22	29.04	28.04	25.38	37.52	34.43	32.71	31.55	30.51	29.62	26.74
	37.75	33.99	31.81	30.31	29.11	28.11	25.31	37.56	34.68	33.09	31.79	30.71	29.77	26.86
Method	Boat 512×512							Zelda 512×512						
Input PSNR	34.15	28.14	24.66	22.17	20.28	18.74	14.58	34.17	28.16	24.67	22.24	20.33	18.83	14.73
HMT (DWT)	34.02	30.88	28.91	27.64	26.70	25.95	23.98	38.20	35.17	33.33	32.07	31.11	30.31	27.78
NeighShrink (DWT)	31.94	29.22	27.50	26.33	25.46	24.76	22.87	36.18	33.57	31.92	30.75	29.95	29.03	26.74
ProbShrink (DWT)	34.51	32.03	30.19	28.88	27.76	26.97	24.70	37.56	35.19	33.60	32.36	31.48	30.66	27.93
Sure-let (DWT)	35.02	32.55	30.72	29.46	28.43	27.63	25.39	38.55	35.89	34.24	32.98	31.97	31.03	28.17
BLS-GSM Non-Redundant	36.13	32.53	30.54	29.14	28.09	27.27	25.13	38.43	35.67	33.94	32.74	31.83	30.93	28.24
Fuzzy-Shrink (DWT)	36.26	32.54	30.52	29.15	28.11	27.26	24.96	38.51	35.69	33.97	32.79	31.82	30.95	28.20
	36.30	33.01	30.93	29.47	28.38	27.48	25.07	38.83	36.12	34.40	33.12	32.10	31.15	28.32
Bi-Shrink (DT-DWT)	35.92	32.97	30.71	29.03	27.55	26.27	22.84	38.13	34.80	32.32	30.27	28.76	27.31	23.25
BLS-GSM Best Redundant	36.86	33.38	31.40	30.01	28.94	28.09	25.90	39.21	36.68	35.10	33.93	33.01	32.26	30.11
Fuzzy-Shrink (DT-DWT)	37.02	33.49	31.53	30.13	28.96	28.21	25.56	39.23	36.41	34.96	33.62	32.71	31.86	28.93
	36.89	33.67	31.75	30.24	29.18	28.46	25.45	39.03	36.64	35.18	33.85	32.88	31.98	29.12
Method	Cameraman 256×256							Goldhill 512×512						
Input PSNR	34.12	28.27	24.88	22.44	20.55	19.02	14.91	34.14	28.13	24.64	22.15	20.27	18.73	14.68
HMT (DWT)	35.93	31.57	29.17	27.56	26.26	25.34	22.63	34.74	31.28	29.45	28.29	27.43	26.81	25.06
NeighShrink (DWT)	32.18	28.59	26.79	25.52	24.57	23.65	21.43	31.61	29.17	27.49	26.86	25.98	25.48	23.82
ProbShrink (DWT)	36.52	32.01	29.37	27.90	26.77	25.82	23.11	35.20	31.99	30.03	28.83	27.88	27.14	25.20
Sure-let (DWT)	35.70	32.21	30.12	28.52	27.39	26.51	23.47	35.80	32.54	30.71	29.51	28.60	27.89	25.82
BLS-GSM Non-Redundant	37.22	32.56	30.09	28.49	27.31	26.41	23.32	36.16	32.41	30.52	29.28	28.37	27.66	25.77
Fuzzy-Shrink (DWT)	36.93	32.35	29.88	28.24	27.08	26.01	23.21	36.18	32.29	30.34	29.14	28.20	27.44	25.53
	37.74	33.11	30.45	28.66	27.42	26.31	23.38	36.35	32.73	30.74	29.41	28.36	27.56	25.67
Bi-Shrink (DT-DWT)	36.61	32.33	29.96	28.03	26.65	25.27	21.83	35.28	32.20	30.14	28.45	27.06	26.07	22.66
BLS-GSM Best Redundant	37.40	33.03	30.67	29.16	28.05	27.17	24.81	36.77	33.04	31.15	29.90	28.98	28.27	26.48
Fuzzy-Shrink (DT-DWT)	37.65	33.21	30.75	29.24	27.96	26.91	23.83	36.82	33.14	31.25	29.96	29.03	28.35	26.15
	37.77	33.63	31.17	29.55	28.26	27.22	24.21	36.56	33.29	31.37	30.11	29.16	28.47	26.23

Output PSNRs have been averaged over five noise realizations. The best redundant results are obtained using the *BLS-GSM* 3×3 with an 8-orientations full steerable pyramid [22]. In addition, the results in the second row for our fuzzy method are obtained after post-processing.

The standard grayscale test images: "Barbara", "Peppers", "Boat", "Zelda", "Goldhill" (512×512) and "Cameraman" (256×256) were chosen as the experimental dataset (Fig. 6), which are captured by simulated additive Gaussian white noise at seven different power levels $\sigma_n = [5, 10, 15, 20, 25, 30, \text{and } 50]$. We objectively measured the experimental results by the peak signal-to-noise ratio (PSNR) in decibels (dB), which is defined:

$$PSNR = 10 \times \log_{10}\left(\frac{255^2}{MSE}\right) \tag{26}$$

where

$$MSE = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(\hat{X}(i,j) - X(i,j)\right)^2 \tag{27}$$

where $X(i,j)$ is the original image, $\hat{X}(i,j)$ is the estimated noise-free signal, and $M \times N$ is the number of pixels.

The denoising process has been performed over five different noise realizations for each standard deviation and the resulting PANRs are averaged over these five runs. The parameters of each method have been set according to the values given by their respective authors in

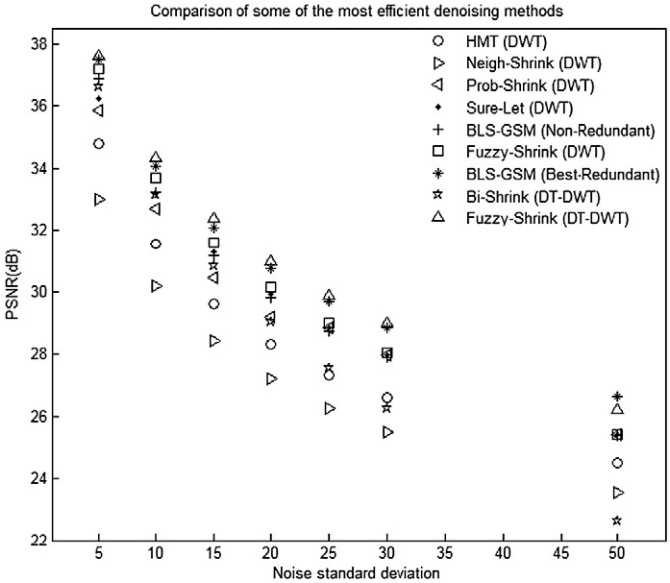

Fig. 7. Average PSNRs of different methods for the six test images used in the experiments.

the corresponding referred papers. Table 3 summarizes the results obtained. As it can be observed in Table 3, our method, which uses a simple fuzzy rule without taking into account inter-scale dependencies, has better results in almost all cases. In addition, Fig. 7 shows average PSNRs of different methods for the six images used in the experiments at the different noise variances.

When looking closer at the results, we observe the following:

- Our method outperforms the Fan's *HMT*, which integrates both inter- and the intra-scale dependencies more than +1.76 dB on average.
- Our method gives better results than Chen's et al. *NeighShrink*, which uses only intra-scale dependency (average gain of +2.99 dB).
- Our method gives better results than Pizurica's *ProbShrink*, which integrates the intra-scale dependency (average gain of +0.662 dB).
- Our method outperforms the Luisier's et al. *Sure-Let*, which integrates only the inter-scale dependency more than +0.321 dB.
- Our method outperforms the Portilla's *BLS-GSM* (Non-Redundant), which integrates both inter and the intra-scale dependencies more than +0.286 dB.
- Our method outperforms the Portilla's *BLS-GSM* (Best Redundant), which integrates both inter and the intra-scale dependencies more than +0.122 dB.
- Our method provides better results than Sendur's *Bi-Shrink*, which integrates both inter and the intra-scale dependencies (average gain of +2.04 dB).

Fig. 8. A part of noise-free 512×512 "Barbara" image, noisy version of it: PNSR=18.78, and denoised result using the *ProbShrink* (DWT): PSNR=26.15 (top-row, left to right). Denoised result using the *Sure-Let* (DWT): PSNR=25.83, denoised result using the *BLS-GSM* (Non-Redundant): PSNR=26.14, and denoised result using our Fuzzy-Shrink (DWT): PSNR=26.68 (middle-row, left to right). Denoised result using the Bi-Shrink (DT-DWT): PSNR=26.47, denoised result using the *BLS-GSM* (Best Redundant): PSNR=27.38, and denoised result using our Fuzzy-Shrink algorithm: PSNR=28.11 (bottom-row, left to right).

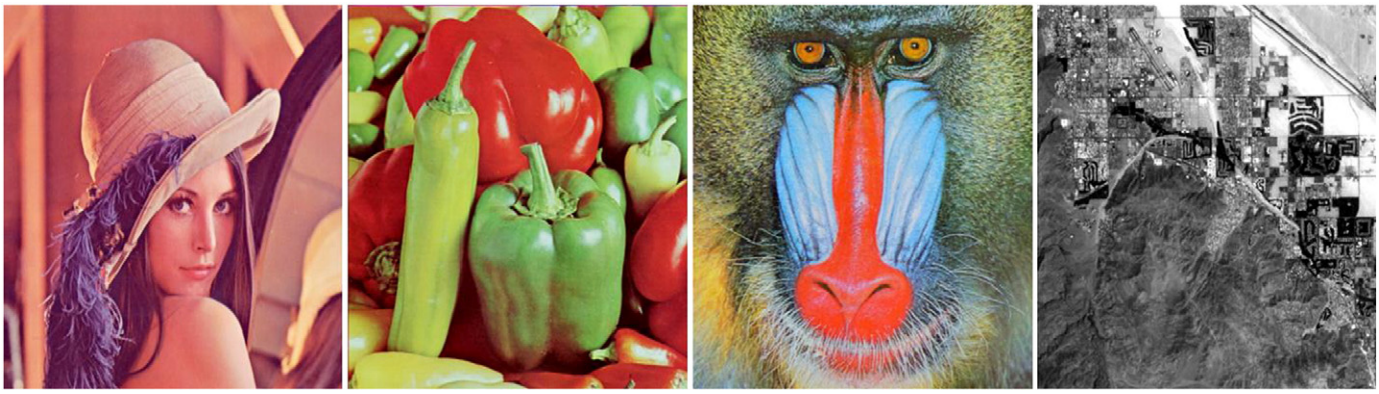

Fig. 9. Test images used in the experiments. From left to right: Lena, Peppers, Baboon, and first band of a Landsat image showing a part of southern California.

- The best result is related to the "Barbara" image, which has many texture areas and the fuzzy feature can effectively distinguish these regions from noise. Also in the results for images, which have many smooth regions such as "Peppers" and "Cameraman", the differences between the methods are smaller.

For subjective evaluation of methods, there are two important criteria: the visibility of processing artifacts and preserving image edges. Fig. 8 illustrate the results of noisy "Barbara" image from different methods. Additionally, we would like to stress that our new fuzzy method exhibits the fewest number of artifacts and preserves most of edges compared to other methods.

4.2. Multi-channel image denoising results

In the satellite systems, it may be desirable to perform denoising before the image compression step in order to improve the compression efficiency. Therefore, we use both color and multispectral satellite images for experimental dataset (Fig. 9), which was captured by simulated additive Gaussian white noise.

In this section, we compare our multi-channel fuzzy denoising method with three state-of-the-art wavelet-based techniques: Pizurica's multi-channel *ProbShrink* [20] and Luisier's et al. multi-channel *Sure-Let* [34]. In addition, we compare our multi-channel image denoising algorithm with the Portilla's *BLS-GSM* [22], which denoised each channel, separately.

It should be mentioned that for comparing *ProbShrink* [20] and *Sure-Let* [34] methods, which are proposed in the discrete wavelet transform domain, we use a critically sampled orthonormal wavelet basis with eight vanishing moments (sym8) over four decomposition stages in our fuzzy-shrink method. On the other hand, *BLS-GSM* method has used a redundant wavelet transform (an 8-orientations full steerable pyramid [22]). Therefore, for a fair comparison between *BLS-GSM* and our fuzzy-shrink methods, we use dual-tree discreet wavelet transform over four decomposition stages for wavelet analysis.

We measured the experimental results by the peak signal-to-noise ratio (PSNR) in decibels (dB), objectively, which is defined by Eq. (26), and MSE is defined as:

$$MSE = \frac{1}{C \times M \times N} \sum_{c=1}^{C} \left(\sum_{i=1}^{M} \sum_{j=1}^{N} \left(\hat{X}(i,j,c) - X(i,j,c) \right) \right)^2 \tag{28}$$

Table 4 summarizes the results obtained. As it can be observed in Table 4, our fuzzy multi-channel image denoising method is already competitive with the best techniques.

Table 4
Comparison of multi-channel image denoising algorithms (same noise level in each channel).

σ_n	5	10	15	20	25	30	50	5	10	15	20	25	30	50
Method	Lena 512×512							Peppers 512×512						
Input PSNR	34.15	28.16	24.69	22.23	20.37	18.86	14.82	34.25	28.25	24.78	22.33	20.47	18.98	15.01
ProbShrink-M (DWT)	35.61	33.18	31.40	30.25	29.31	28.51	26.08	34.86	31.49	30.35	29.39	28.12	27.82	25.29
Sure-let-M (DWT)	37.86	34.64	32.40	30.79	29.60	28.64	26.20	36.62	33.35	31.79	30.35	29.16	28.28	25.61
Fuzzy-Shrink-M (DWT)	36.81	33.83	32.08	30.88	30.01	29.31	26.77	36.13	32.95	31.26	30.14	29.17	28.49	25.86
	36.75	34.21	32.65	31.50	30.60	29.80	27.14	36.05	33.21	31.53	30.51	29.61	28.95	26.15
BLS-GSM Best Redundant	37.11	34.23	32.72	31.56	30.55	29.95	27.89	36.12	33.06	31.63	30.52	29.86	29.03	26.98
Fuzzy-Shrink-M (DT-DWT)	37.56	34.72	33.16	32.13	31.17	30.35	27.76	36.57	33.41	31.94	30.84	30.01	29.21	26.64
	37.42	34.87	33.29	32.23	31.28	30.46	27.92	36.41	33.62	32.23	30.99	30.24	29.45	26.78
Method	Baboon 512×512							Southern California 512×512						
Input PSNR	34.15	28.16	24.66	22.21	20.33	18.79	14.75	34.26	28.31	24.81	22.33	20.44	18.92	14.88
ProbShrink-M (DWT)	33.11	29.03	26.44	24.87	23.73	22.89	20.94	33.29	29.49	27.13	25.54	24.31	23.43	21.06
Sure-let-M (DWT)	34.46	29.78	26.82	25.00	24.03	23.27	21.43	33.36	29.94	27.74	26.15	24.95	24.05	21.66
Fuzzy-Shrink-M (DWT)	34.01	29.52	27.25	25.56	24.41	23.47	21.49	35.21	30.72	28.25	26.54	25.31	24.41	21.99
	33.86	29.42	27.44	26.02	24.86	23.96	21.79	35.35	31.06	28.41	26.86	25.64	24.76	22.42
BLS-GSM Best Redundant	35.01	30.13	27.56	26.03	24.75	24.01	21.62	35.31	30.79	28.31	26.65	25.32	24.68	22.13
Fuzzy-Shrink-M (DT-DWT)	34.33	29.67	27.66	26.22	25.25	24.57	22.28	35.23	31.39	28.94	27.47	26.15	25.23	22.68
	34.21	29.61	27.56	26.38	25.31	24.72	22.45	35.19	31.44	29.21	27.69	26.41	25.48	22.84

Output PSNRs have been averaged over five noise realizations. The best redundant results are obtained using the *BLS-GSM* 3×3 with an 8-orientations full steerable pyramid [22], and we have simply applied the *BLS-GSM* in each channel, independently. In addition, the results in the second row for our fuzzy method are obtained after post-processing.

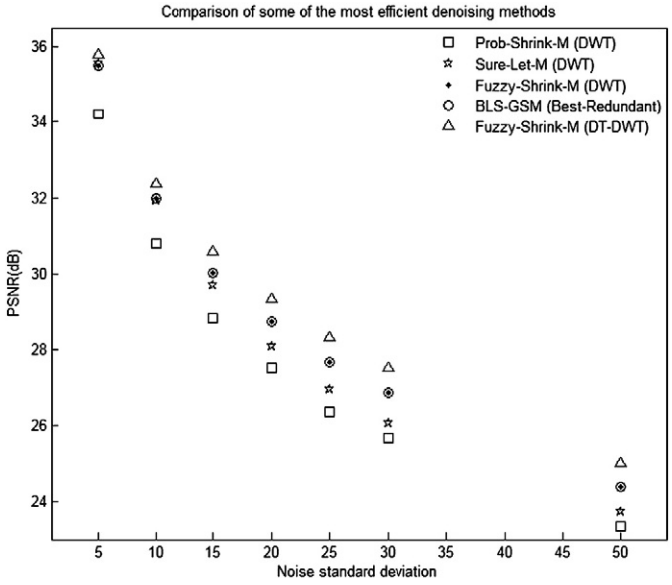

Fig. 10. Average PSNRs of different methods for the four test images used in the experiments.

When looking closer at the results, we observe the following:

- Our method (Fuzzy-Shrink using DWT) provides better results than Pizurica's multi-channel *ProbShrink*, which integrates the intra-scale dependencies (average gain of +1.199 dB).
- Our method (Fuzzy-Shrink using DWT) outperforms the Luisier's et al. multi-channel *Sure-Let*, which is used the inter-scale and inter-channel dependencies more than +0.449 dB on average.
- Our method (Fuzzy-Shrink using DT-DWT) also gives better results than Portilla's *BLS-GSM* (Best Redundant), which is used for each channel separately (average gain of +0.435 dB).
- The worst result is obtained for the "Baboon" image. Our explanation for this is that the "Baboon" image is very noisy (i.e. texture area in this image is similar to noise), and when we use the fuzzy features for taking into account the neighbor dependency, it will be smoother in the resulting image.

For better judgment between different methods, Fig. 10 shows average PSNRs of different methods for the test images used in the experiments. In addition, Figs. 11 and 12 illustrate the results of noisy color "Lena" and Multispectral "southern California" images from different methods. As it can be observed in Figs. 11 and 12, our multi-channel image denoising is successful in preserving image edges and fewer artifacts as visual criteria as compared to other methods.

Fig. 11. A part of noise-free "Lena" image, noisy version of it, PNSR = 18.86, denoised result using the multi-channel *ProbShrink* (DWT): PSNR = 28.51, and denoised result using the multi-channel *Sure-Let* (DWT): PSNR = 28.64 (second row). Denoised result using our multi-channel Fuzzy-Shrink (DWT): PSNR = 29.80, and denoised result using the single-channel *BLS-GSM* (Best Redundant): PSNR = 29.95 (third-row). Denoised result using our multi-channel Fuzzy-Shrink (DT-DWT): PSNR = 30.46, and the original "Lena" image.

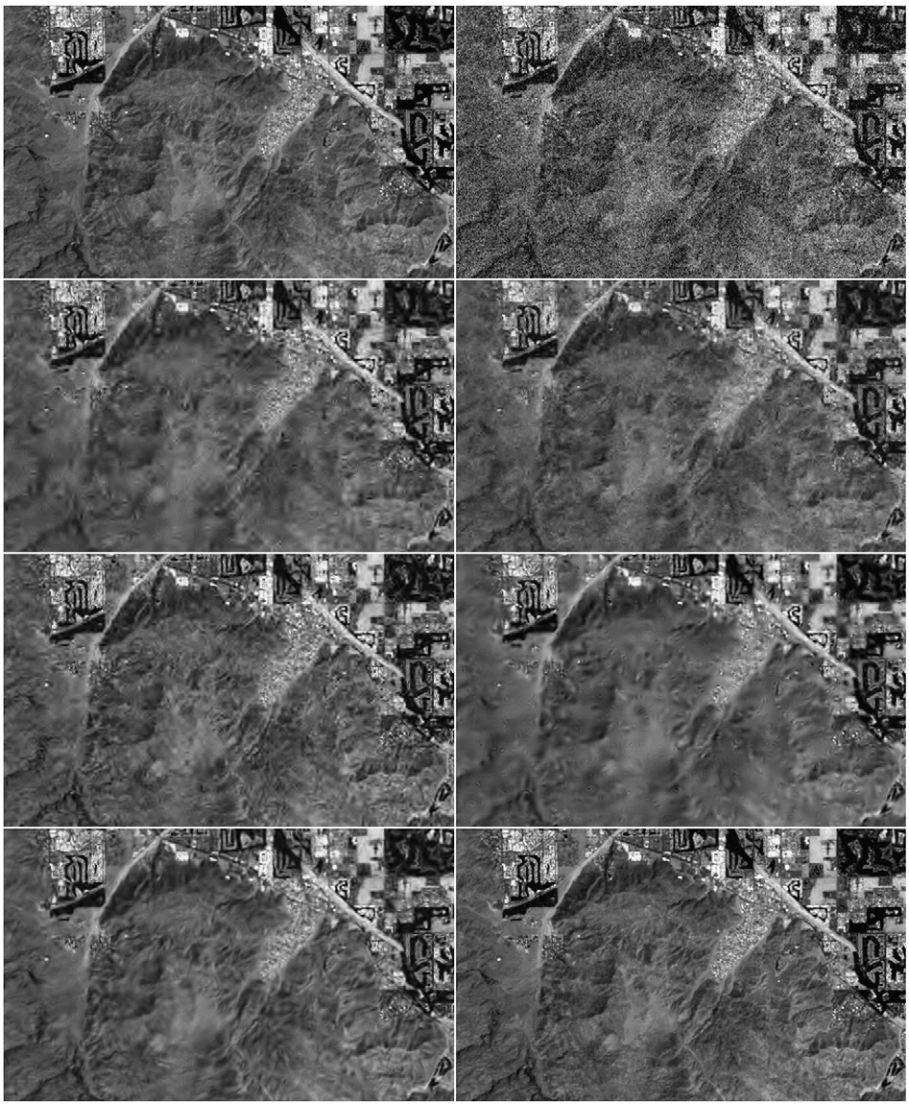

Fig. 12. A part of first band of noise-free 512×512 multispectral "southern California" image, noisy version of it: PNSR = 18.92 (first-row). Denoised result using the multi-channel *ProbShrink* (DWT): PSNR = 24.43, and denoised result using the multi-channel *Sure-Let* (DWT): PSNR = 24.05 (second row). Denoised result using our multi-channel Fuzzy-Shrink (DWT): PSNR = 24.76, and denoised result using the single-channel *BLS-GSM* (Best Redundant): PSNR = 24.68 (third-row). Denoised result using our multi-channel Fuzzy-Shrink (DT-DWT): PSNR = 25.48, and the original "Southern California" image (last-row).

4.3. Computation time

It is also interesting to evaluate the various denoising techniques from a practical point of view that is the computation time. Indeed, the results achieved by overcomplete representation are admittedly superior than the ones obtained by critically sampled wavelet transforms, but their weakness is the time they require (19.12 s for 512×512 images to obtain the redundant result of *BLS-GSM* method). With our simple fuzzy method, the total denoising process (including four iterations of an orthonormal wavelet transform) lasts approximately 0.43 s for 256×256 images (1.59 s for 512×512 images). In addition, the redundant results of our fuzz-shrink algorithm using DT-DWT are also promising (1.70 s for 256×256 images and 3.17 s for 512×512 images).

Table 5 summarizes the relative computation time of the various methods considered in this paper. All computations have been performed on a Pentium IV personal computer, using a 2.00-GHz processor, running Windows Vista. Note that our fuzzy-shrink algorithm is fully implemented in Matlab software. As it can be seen in Table 5, our fuzzy approach is placed between the methods with low computation time.

Table 5
Relative computation time of various denoising techniques.

Method	Unite of time	
	256×256 images	512×512 images
HMT (DWT)	1.89	4.37
NeighShrink (DWT)	3.85	5.11
ProbShrink (DWT)	4.13	5.42
Sure-let (DWT)	0.45	0.93
BLS-GSM (Non-Redundant)	0.72	2.39
Fuzzy-Shrink (DWT)	0.43	1.59
Bi-Shrink (DT-DWT)	0.51	1.35
BLS-GSM (Best Redundant)	4.45	19.12
Fuzzy-Shrink (DT-DWT)	1.70	3.17
ProbShrink-M (DWT)	11.4	17.2
Sure-let-M (DWT)	1.47	2.41
Fuzzy-Shrink-M (DWT)	2.1	4.63
Fuzzy-Shrink-M (DT-DWT)	5.12	10.11

The computation times have been averaged over ten runs.

5. Conclusion

In this paper, we propose a new wavelet shrinkage algorithm using intra-scale dependency as a fuzzy feature, and then, shrinkage wavelet coefficients with corresponding fuzzy membership function, for single channel image denoising. We also extended our image denoising algorithm for multi-channel images. In multi-channel images, we use inter-channel relation as a new feature and improve wavelet coefficients' information at the shrinkage step. We use the DT-DWT for wavelet analysis, because it is shift-invariant, and has more directional sub-bands compared to the DWT. In other words, proposing a new method for shrinking wavelet coefficients in the second step of the wavelet-based image denoising, for both single and multi-channel images, is the main novelty of this paper. The comparison of the denoising results obtained with our algorithm, and with the best state-of-the-art methods, demonstrate the performance of our fuzzy approach, which gave the best output PSNRs for most of the images. In addition, the visual quality of our denoised images exhibits the fewest number of artifacts and preserves most of edges compared to other methods. A future improvement of the proposing method is the ability of incorporate more information (e.g. inter-scale dependency of wavelet coefficients) by adding other fuzzy rules to improve the noise reduction performance. Future work should be done on this promising issue.

References

[1] S. Schulte, V.D. Witte, E.E. Kerre, A fuzzy noise reduction method for color images, Ieee Trans. Image Process. 16 (5) (MAY, 2007) 1425–1436.
[2] R. Dugad, N. Ahuja, Video denoising by combining Kalman and Wiener estimates, Proc IEEE Int. Conf. Image Process 4 (1999) 152–156.
[3] O. Ojo, T. Kwaaitaal-Spassova, An algorithm for integrated noise reduction and sharpness enhancement, IEEE Trans. Consum. Electron. 46 (5) (May 2000) 474–480.
[4] M. Meguro, A. Taguchi, N. Hamada, Data-dependent weighted median filtering with robust motion information for image sequence restoration, IEICE Trans. Fund. 2 (2001) 424–428.
[5] Zlokolica, W. Philips, D. Van De Ville, A new nonlinear filter for video processing, Proc. IEEE Benelux Signal Process. Symp., 2, 2002, pp. 221–224.
[6] S.D. Kim, S.K. Jang, M.J. Kim, J.B. Ra, Efficient block-based coding of noise images by combining pre-filtering and DCT, Proc. IEEE Int. Symp. Circuits Syst., 4, 1999, pp. 37–40.
[7] Y.F. Wong, E. Viscito, E. Linzer, Preprocessing of video signals for MPEG coding by clustering filter, Proc. IEEE Int. Conf. Image Process., 2, 1995, pp. 2129–2133.
[8] C. Vertan, C.I. Vertan, V. Buzuloiu, Reduced computation genetic algorithm for noise removal, Proc. IEEE Conf. Image Process. Applic., 1, Jul. 1997, pp. 313–316.
[9] W. Ling, P.K.S. Tam, Video denoising using fuzzy-connectedness principles, Proc. Int. Symp. Intell. Multimed., Video, Speech Process. (2001) 531–534.
[10] L. Shutao, W. Yaonan, Z. Changfan, M. Jianxu, Fuzzy filter based on neural network and its applications to image restoration, Proc. IEEE Int. Conf. Signal Process., 2, 2000, pp. 1133–1138.
[11] S. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (Jul 1989) 674–693.
[12] D.L. Donoho, I.M. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, J. Amer. Statist. Assoc. 90 (432) (Dec. 1995) 1200–1224.
[13] D.L. Donoho, I.M. Johnstone, Ideal spatial adaptation via wavelet shrinkage, Biometrika 81 (1994) 425–455.
[14] C. Stein, Estimation of the mean of a multivariate normal distribution, Ann. Stat. 9 (1981) 1135–1151.
[15] S.G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, IEEE Trans. Image Process. 9 (9) (Sep. 2000) 1135–1151.
[16] T. Blu, F. Luisier, The SURE-LET approach to image denoising, IEEE Trans. Image Process 16 (2007) 2778–2786.
[17] L. Ṣendur, W. Selesnick, Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency, *IEEE Trans.* Signal. Process 50 (11) (Nov 2002) 2744–2756.
[18] L. Ṣendur, I.W. Selesnick, Bivariate shrinkage with local variance estimation, *IEEE Signal Proc.* Lett. 9 (12) (Dec. 2002) 438–441.
[19] G.Y. Chen, T.D. Bui, A. Krzyzak, Image denoising with neighbour dependency and customized wavelet and threshold, Pattern Recog. 38 (2005) 115–124.
[20] A. Pizurica, W. Philips, Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising, IEEE Trans. Image Process. (2006) 654–665.
[21] S. Schulte, B. Huysmans, A. Pizurica, E.E. Kerrel, W. Philips, A New Fuzzy-based Wavelet Shrinkage Image Denoising Technique, Springer Verlag, 2006, pp. 12–23.
[22] J. Portilla, V. Strela, M. Wainwright, E. Simoncelli, Image denoising using Gaussian scale mixtures in the wavelet domain, IEEE Trans. Image Process. (2003) 1338–1351.
[23] G. Fan, X. Xia, Image denoising using local contextual hidden markov model in the wavelet domain, IEEE Signal Process. Lett. (2001) 125–128.
[24] G. Fan, X. Xia, Improved hidden Markov models in the wavelet domain, IEEE Trans. Signal Process. (2001) 115–120.
[25] I.W. Selesnick, R.G. Baraniuk, N.G. Kingsbury, The dual-tree complex wavelet transform, IEEE Signal Process. Mag. (2005) 124–152.
[26] L. Zhang, P. Bao, Hybrid inter and intra-wavelet scale image restoration, Pattern Recog. 36 (2003) 1737–1746.
[27] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, D.J. Heeger, Shiftable multiscale transform, IEEE Trans. Inf. Theory 38 (2) (Mar. 1992) 587–607.
[28] N.G. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, J. Appl. Computat. Harmonic Anal. 10 (3) (May 2001) 234–253.
[29] N.G. Kingsbury, A dual-tree complex wavelet transform with improved orthogonality and symmetry properties, Proc. IEEE Conf. Image Process. (2000) 375–378.
[30] I.W. Selesnick, The design of approximate Hilbert transform pairs of wavelet bases, IEEE Trans. Signal Process. 50 (5) (2002) 1144–1152.
[31] L. Jun, C. Guangmeng, H. Bo, Image Denoising Based on Fuzzy in Wavelet Domain, IMTC 2005 Instrumentation and Measurement Technology Conference, Ottawa, Canada, 17–19 MAY 2005, pp. 2019–2023.
[32] L. Zadeh, Fuzzy Sets, Inform. Contr. 8 (3) (1965) 338–353.
[33] W. Siler, J.J. Buckley, Fuzzy Expert Systems and Fuzzy Reasoning, By ISBN 0-471-38859-9, John Wiley & Sons, Inc, 2005.
[34] F. Luisier, T. Blu, SURE-LET multichannel image denoising: interscale orthonormal wavelet thresholding, Ieee Trans. Image Process. 17 (4) (April 2008) 482–492.
[35] L.A. Zadeh, Fuzzy logic and its application to approximate reasoning, Inf. Process. 74 (1973) 591–594.
[36] D. Driankov, H. Hellendoorn, M. Reinfrank, An Introduction to Fuzzy Control, Springer, Berlin/Heidelberg, 1993, ISBN 3-540-56362-8.
[37] P. Hájek, P. Hájek, Metamathematics of Fuzzy Logic, Kluwer, Dordrecht, 1998, ISBN 0-7923-5238-6.
[38] K. Dabov, A. Foi, V. Katkovnik, K.O. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, IEEE Trans. Image Process. 16 (8) (August 2007) 2080–2095.
[39] H. Takeda, S. Farsiu, P. Milanfar, Kernel regression for image processing and reconstruction, IEEE Trans. Image Process. 16 (2) (February 2007) 349–366.
[40] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, IEEE Trans. Image Process. 15 (12) (December 2006) 3736–3745.

Jamal Saeedi was born in Amol, Iran, in 1983. He received the B.Sc. degree in Biomedical Engineering from Sahand University of Tabriz, Iran in 2006 and M.Sc. degrees in Electronic Engineering from Amirkabir University of Tehran, Iran in 2009.He works in the field of signal and image processing, specializing particularly in multi-resolution analysis, image fusion, and fuzzy image processing.

Mohammad Moradi received the BS and MS degrees in electronic engineering from Tehran University, in 1988 and 1990, respectively, and the PhD degree from the University of Tarbiat Modarres, Tehran, in 1995.He has been with the faculty of biomedical engineering, Amirkabir University of Technology (AUT), since 1995, where he is currently an Associate Professor. His primary research and teaching interests involve the theory and application of medical instrumentation, biomedical signal and Image processing and fuzzy neural systems.