

# Image Denoising using Patch based Processing with Fuzzy Gaussian Membership Function

Krishan Kundu

Department of Computer Science & Engineering

Govt. College of Engineering & Textile Technology, Serampore, Hooghly, 712201, India

## ABSTRACT

In this paper image denoising scheme based on fuzzy Gaussian membership function. For a given corrupted image at first converted to the fuzzy values using the fuzzification method. Then extract all patches with overlaps, after extracting all patches each patch is to be permuted and apply the fuzzy Gaussian membership function. Extraction means all patches with overlaps, refer to these as coordinates in high-dimensional space, and arrange them such that they are chained in the shortest possible path. The obtained ordering, applying the fuzzy defuzzification method to convert fuzzy values to the crisp values to what should be a normal signal. This enables us to get high-quality recovery of the clean image by applying relatively simple one-dimensional smoothing operations to the reordered set of pixels. The performance of this approach is experimentally verified on a diversity of images and noise levels. The results presented here demonstrate that proposed technique is on similarity or more than the existing state of the art, in terms of both peak signal-to-noise ratio and subjective visual quality.

## Keywords

Patch-based processing, fuzzification, defuzzification, gaussian membership function, traveling salesman, pixel permutation, denoising.

## 1. INTRODUCTION

Image denoising is an important image processing task, both as a process itself, and as a component in other processes. Very many ways to denoise an image or a set of data exists. The main properties of a good image denoising model are that it will remove noise while preserving edges. Now days, image processing field with its local patches become highly popular and was shown highly effective for representing work. In this work the idea is that extract all possible patches of the given corrupted image to be processed which are very small as compared to original image. A typical patch size would be 8×8 pixels. The processing is carried out by working on these patches and exploiting interrelations between them. The modified patches are then put back into the image to form the resulting clean image.

In various ways, the relations between patches can be taken into account weighted averaging of pixels with similar neighboring patches, as the NL-Means algorithm does [1], clustering the patches into disjoint sets and treating each set differently, as performed in [2]–[6], looking for a representative dictionary for the patches and using it to sparsely represent them, collecting

group of similar patches and applying a sparsifying transform on them. A general idea, too many of these methods is the expectation that every patch taken from the image may find similar ones extracted to another place in the image. A joint action of these patches supports the restoration process by introducing a non-local force, thus enabling better recovery. The honor of state-of-the-art is held by block-matching with 3D filtering (BM3D) [7], aging over eight years. Levin and Nadler found that for natural images, BM3D is close to the theoretical limit of denoising, but artificial and highly correlated images still have potential for improvement. However, only a handful of methods numerically improve over BM3D, with modest increase in visual quality. State-of-the-art image denoising methods still produce visible artifacts, especially on sharp edges and in smooth regions of the original image. Such features are common for natural images, like clear sky and human skin, not to speak of synthetic images, where edges and gradients are abundant. Another nuisance is that current methods are complex and thus prohibit thorough analysis. In paper [8]–[10] shows that image denoising Based on Fuzzy and Intra-scale dependency in wavelet transform Domain.

Currently, Idan Ram, Michael Elad has been proposed image processing using smooth ordering of its patches [11]. I extend their work using the fuzzy based approach for image denoisy. In this paper, propose an image denoising scheme based on fuzzy Gaussian membership function. For a given corrupted image is converted to the fuzzy values using the fuzzification method. Then, extract all patches with overlaps, after extracting all patches each patch is to be permuted and apply the fuzzy Gaussian membership function. Now extract all patches with overlaps, refer to these as coordinates in high-dimensional space, and arrange them such that they are chained in the “shortest possible path,” basically solving the traveling salesman problem. The obtained ordering, applying the fuzzy defuzzification method to convert fuzzy values to the crisp values, to what should be a regular signal. This enables us to get good recovery of the clean image by applying comparatively simple one-dimensional smoothing operations (such as filtering or interpolation) to the reordered set of pixels. The performance of this approach is experimentally tested on a variety of images and different noise levels. The results presented here demonstrate that the proposed method is better than the current state of the art, both visually and quantitatively.

## 2. PROPOSED METHOD

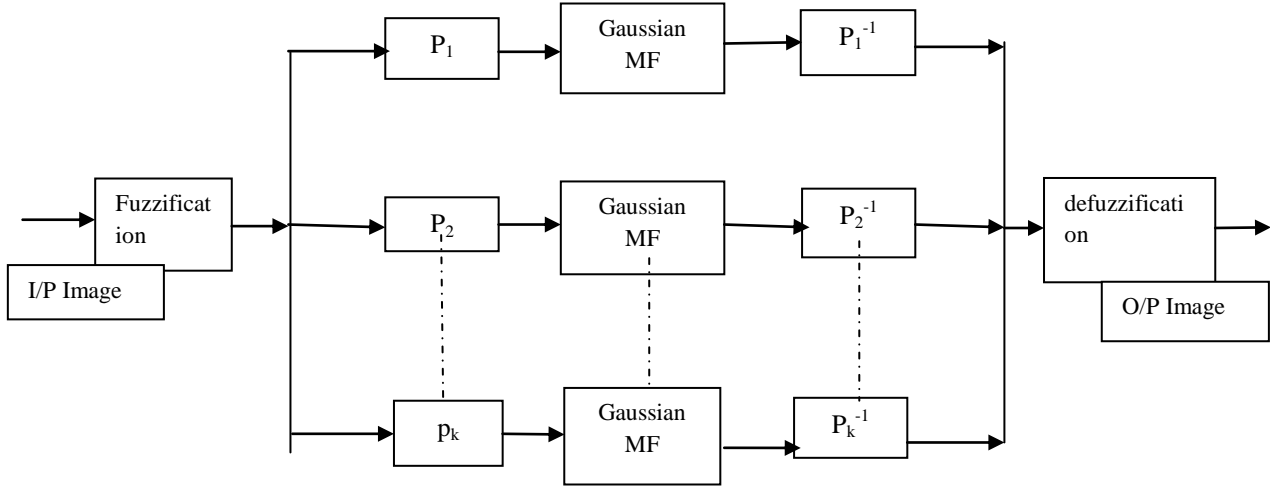


Fig.1: Proposed Method

In the Fig.1 shows that the proposed method for Gaussian noise reduction consists of three parts first part is fuzzification which basically convert the image pixel values into the fuzzy values, 2nd is the modification of the pixel values using the Gaussian membership function and third is the defuzzification which is generally convert the fuzzy values to the crisp pixel values. At first, extract all patches with overlaps, after extracting all patches each patch is to be permuted and apply the fuzzy Gaussian membership function. Extract all patches with overlaps, refer to these as coordinates in high-dimensional space, and arrange them such that they are chained in the “shortest possible path,” mainly solving the traveling salesman problem.

### 2.1 Proposed Scheme

Let  $Y$  be an image of size  $N = S \times T$ , and let  $Z$  be a noisy or corrupted version of  $Y$ . Also, let  $y$  and  $z$  be the column stacked versions of  $Y$  and  $Z$ , respectively. A column stacked version of a matrix is a column vector, which is produced by concatenating the columns of the matrix one by one. Then the corrupted image satisfies

$$z = My + v \quad (1)$$

Where the  $N \times N$  matrix  $M$  denotes a linear operator which corrupts the data and  $v$  denotes an additive white Gaussian noise independent of  $y$  with zero mean and variance  $\delta^2$ . The main objective of this method is to reconstruct the original image from the corrupted image i.e. reconstruct  $y$  from noisy image  $z$ , this method employ a permutation matrix  $P$  of size  $N \times N$ . It is assumed that when  $P$  is applied to the target signal, it produces a smooth signal.

This method start by applying the permutation matrix  $P$  to  $z$  and obtain  $z^P = Pz$ . After that it applies a simple one dimensional smoothing operator  $H$  such as interpolation or filtering on  $z$ . And finally, it applies inverse permutation matrix  $P^{-1}$  to the result, and obtains the reconstructed image.

$$\hat{y} = P^{-1}H\{Pz\} \quad (2)$$

The patch reordering proposed algorithm is given below:

Algorithm 1: The Patch Reordering Algorithm

Operation: Reorder the image patches  $x_j$

We use concept of “cycle spinning” method to better smooth the recovered image. For that, we randomly construct  $K$  different permutation matrices  $P_k$ , utilize each to denoising the image  $z$  and then average the results. This can be expressed by

$$\hat{y} = \frac{1}{K} \sum_{k=1}^K P_k^{-1} H\{P_k z\} \quad (3)$$

### 2.2 Construction the Permutation Matrix P

Design a Permutation matrix  $P$  which produces a smooth signal when it is applied to the target image  $y$ . When the image  $Y$  is known, the optimal solution would be to reorder it as a vector, and then apply a simple sort operation on the obtained vector. This is the interesting in the case where only have the corrupted image  $Z$  (noisy, containing missing pixels, etc.), and any permutation defined by simply reordering the corrupted pixels to a regular signal does not necessarily smooth  $y$ . Therefore, as the pixels in the corrupted image are not helpful and stay for a suboptimal ordering operation, using patches from the corrupted image.

Let  $y_i$  and  $z_i$  denote the  $i^{\text{th}}$  samples in the vectors  $y$  and  $z$ , respectively. We denote by  $x_i$  the column stacked version of the  $\sqrt{n} \times \sqrt{n}$  patch around the location of  $x_i$  in  $Z$ . Assume that under a distance measure  $w(x_i, x_j)$ , proximity between the two patches  $x_i$  and  $x_j$  suggests proximity between the uncorrupted versions of their center pixels  $y_i$  and  $y_j$ . Thus, try to reorder the points  $x_j$  so that they form a smooth path, hoping that the corresponding reordered 1D signal  $y^p$  will also become smooth. The “smoothness” of the reordered signal  $y^p$  can be measured using its total-variation measure

$$\|y^p\|_{TV} = \sum_{j=2}^N \|y^p(j) - y^p(j-1)\| \quad (4)$$

Let  $\{x^p\}_{j=1}^N$  denote the points  $\{x_i\}_{i=1}^N$  in their new order. Then by similarity measure the “smoothness” of the path through the points the points  $x_j^p$  by the measure

$$X_{TV}^p = \sum_{j=2}^N w(x_j^p, x_{j-1}^p) \quad (5)$$

Parameter: Given the image patches  $\{x_j\}_{j=1}^N$  the distance function  $w$  and  $\epsilon$  which is used in the probability.

Step1. Start.

Step2. Choose a random index  $j$  and set  $\Omega(1) = \{j\}$

Step3. For  $k=1 \dots N-1$

Step4. Set  $A_k$  to be the set of indices of the  $B \times B$  patches around  $x_{\Omega(k)}$

Step5. If  $|\frac{A_k}{\Omega}| = 1$

Step6. Set  $\Omega(k+1)$  to be  $\frac{A_k}{\Omega}$

Else

Step7. If  $|\frac{A_k}{\Omega}| \geq 2$

Step8. Find  $x_{j1}$  - the nearest neighbor to  $x_{\Omega(k)}$  such that  $j1$  belongs to  $A_k$  and  $j1$  not belongs to  $\Omega$

Find  $x_{j2}$  -the second nearest neighbor to  $x_{\Omega(k)}$  such that  $j2$  belongs to  $A_k$  and  $j2$  not belongs to  $\Omega$

Step9. Else if  $|\frac{A_k}{\Omega}| = 0$

Step10. Find  $x_{j1}$  - the nearest neighbor to  $x_{\Omega(k)}$  such that  $j1$  not belongs to  $\Omega$

Find  $x_{j2}$  -the second nearest neighbor to  $x_{\Omega(k)}$  such that  $j2$  not belongs to  $\Omega$

Step11. Set  $\Omega(k+1)$  to be:

$\{j1\}$  with probability  $p1 = a \exp [-w(x_{j0}, x_{j1}) / \epsilon]$

$\{j2\}$  with probability  $p2 = a \exp [-w(x_{j0}, x_{j2}) / \epsilon]$

Step12. End.

Output: the set  $\Omega$  holds the proposed ordering

Minimizing  $XpTV$  comes down to finding the shortest path that passes through the set of points  $x_i$ , visiting each point only once. This can be regarded as an instance of the traveling salesman problem [13], which can become very computationally expensive for large sets of points. I choose a simple and crude approximate solution, which is to start from a random point and then continue from each point  $x_{j0}$  to its nearest neighbor  $x_{j1}$  with a probability  $p1 = a \exp w(x_{j0}, x_{j1})$ , or to its second nearest neighbor  $x_{j2}$  with a probability  $p2 = a \exp w(x_{j0}, x_{j2})$ , where  $a$  is determined such that  $p1 + p2 = 1$ , is a design parameter, and  $x_{j1}$  and  $x_{j2}$  are taken from the set of unvisited points.

I restrict the nearest neighbor search performed for each patch to an adjacent square neighborhood which contains  $B \times B$

## 2.4 PSNR Measure

To evaluate the performance of the membership function uses the Peak Signal to Noise Ratio (PSNR) measure. The PSNR is used to calculate the difference between the original noise free image and the denoised image. Notation "Signal" in the PSNR measure refers to noise free image. The PSNR measure is calculated using (7). Note that the high value of the PSNR shows the high quality of the reconstructed image.

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \quad (6)$$

patches. When only one unvisited patch remains in that neighborhood, simply continue to this patch, and in the case that no unvisited patches remain, I search for the first and second nearest neighbors among all the unvisited patches in the image. This restriction decreases the overall computational complexity, and our experiments show that with a proper choice of  $B$  it also leads to improved results. The permutation applied by the matrix  $P$  is defined as the order in the found path. The patch reordering scheme is summarized in Algorithm 1.

## 2.3 Gaussian membership function

A membership function for a fuzzy set  $A$  on the universe of discourse  $X$  is defined as  $\mu_A: X \rightarrow [0, 1]$ , where each element of  $X$  is mapped to a value between 0 and 1. This value, called membership value or degree of membership, quantifies the grade of membership of the element in  $X$  to the fuzzy set  $A$ . Membership functions allow us to graphically represent a fuzzy set. The  $x$  axis represents the universe of discourse, whereas the  $y$  axis represents the degrees of membership in the  $[0, 1]$  interval. The fuzzy Gaussian function transforms the original values into a normal distribution. The midpoint of the normal distribution defines the ideal definition for the set, assigned a 1, with the remaining input values decreasing in membership as they move from away from the midpoint in both the positive and negative directions. The input values decrease in membership from the midpoint until they reach a point where the values move too far from the ideal definition and definitely not in the set and are therefore assigned zeros. The fuzzy Gaussian function is given below:

$$\text{Gaussian}(x; c, \sigma) = e^{-\frac{(x-c)^2}{2\sigma^2}}$$

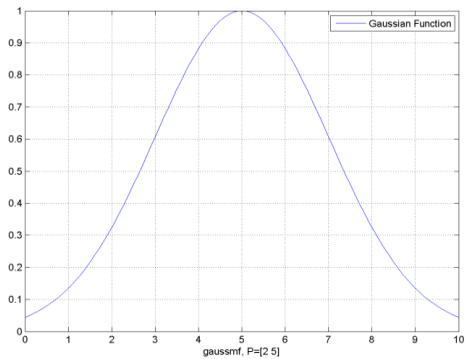


Fig.2: Gaussian Membership Function

A Gaussian MF is specified by two parameters: A Gaussian MF is determined complete by  $c$  and  $s$ ;  $c$  represents the MFs centre and  $s$  determines the MFs width. Fig.2. Plots a Gaussian MF defined by  $\text{Gaussian}(x; 2, 5)$ . I am applying the Gaussian membership function on the permutation matrix  $P_k$  where  $k = 1, 2 \dots k$ . After applying the Gaussian membership function I also get new values of each of the patches.

$$\text{PSNR} = 10 \times \log_{10} \left( \frac{MAX^2}{MSE} \right) \quad (7)$$

Equation (6) calculates the differences between the original noiseless image and the filtered image. In (6), notations  $I$  and  $K$  refer to the input noise free image and the filtered image, respectively, and size of the image (original/filtered) is  $m \times n$ . In (7),  $MAX^2$  shows the maximum pixel value in the image. For instance, in the gray level image,  $MAX^2$  is equal to 255.

### 3. EXPERIMENT

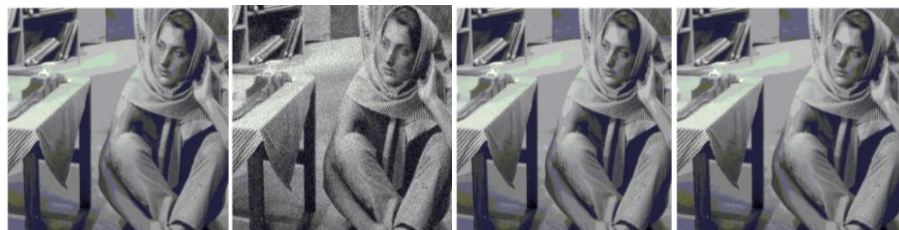
#### 3.1 Experimental Results

Table-I Denoising results (psnr in db) of noisy versions of 4 images, obtained with bm3d algorithms and paper [11] and the proposed scheme. For each image and noise level, the best result is highlighted

Image	Method	$\sigma$ /PSNR							
		5/34.14	10/28.12	15/24.60	20/22.10	25/20.17	50/14.14	75/10.62	100/8.12
Lena (512 × 512)									
	BM3D	38.710	36.905	35.566	34.592	33.855	31.487	30.541	29.913
	Paper[11]	38.222	36.265	34.641	33.234	32.342	31.564	30.634	29.922
	Proposed	<b>38.712</b>	<b>36.906</b>	<b>35.567</b>	<b>34.595</b>	<b>33.856</b>	<b>31.580</b>	<b>30.640</b>	<b>29.931</b>
Barbara (512 × 512)									
	BM3D	38.296	36.591	34.716	33.443	32.543	29.614	28.207	27.187
	Paper[11]	37.637	35.291	33.543	33.401	32.540	29.712	28.317	27.294
	Proposed	<b>38.297</b>	<b>36.596</b>	<b>34.790</b>	<b>33.511</b>	<b>32.613</b>	<b>29.782</b>	<b>28.323</b>	<b>27.304</b>
Boat (512 × 512)									
	BM3D	37.235	34.746	33.214	32.123	31.296	28.502	27.517	26.905
	Paper[11]	37.105	34.657	33.135	32.109	31.290	28.512	27.614	26.971
	Proposed	<b>37.240</b>	<b>34.752</b>	<b>33.227</b>	<b>32.128</b>	<b>31.299</b>	<b>28.520</b>	<b>27.622</b>	<b>26.978</b>
House (256 × 256)									
	BM3D	39.813	37.725	36.620	35.242	34.578	32.327	31.290	30.564
	Paper[11]	38.541	36.513	36.008	35.132	34.467	32.329	31.342	30.612
	Proposed	<b>39.815</b>	<b>37.728</b>	<b>36.627</b>	<b>35.246</b>	<b>34.610</b>	<b>32.345</b>	<b>31.349</b>	<b>30.620</b>



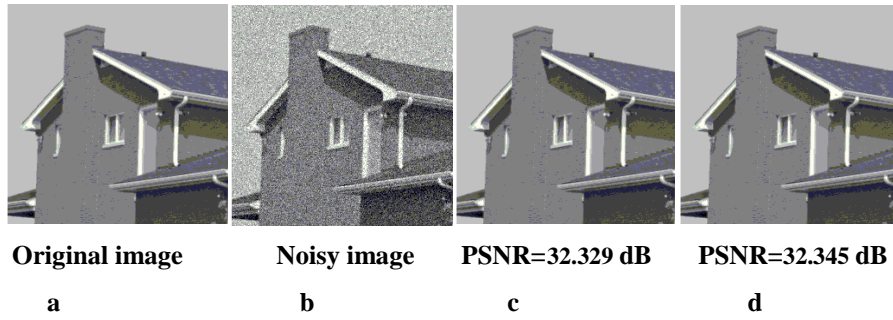
Original image      Noisy image      PSNR=31.564 dB      PSNR=31.580 dB



Original image      Noisy image      PSNR=29.712 dB      PSNR=29.782 dB



Original image      Noisy image      PSNR=28.512 dB      PSNR=28.520 dB



**Fig.3. Comparison of denoising results on different images with noisy image corrupted by WGN of  $\sigma=50$  (a) Original image, (b) noisy image, (c) Paper[11] PSNR value, (d) Proposed Method PSNR value**

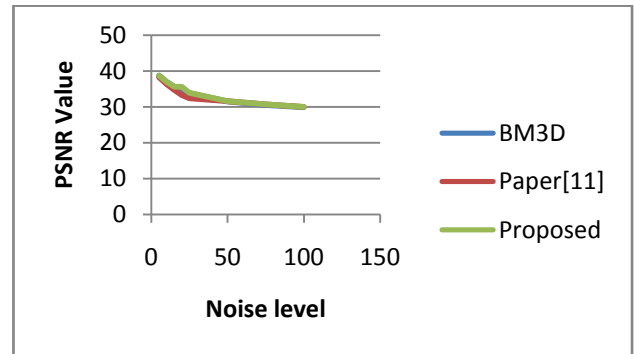
For this work, the four well known images have been tested such as "Lena", "Barbara", "Boat" and "House". The size of first three gray scale images is  $512 \times 512$  and remaining is  $256 \times 256$ . All images are tested for different values of  $\sigma$ , i.e. 5, 10, 15, 20, 25, 50 and 100. The framework for implementation and testing the algorithm is Matlab7. Table-I shows the PSNR values for the different method which were tested on the different additive noises level and also shows that the highlighted part of the table are the better PSNR values. Therefore, the proposed method is better than the state of the art method like BM3D, NLM and paper [11].

### 3.2 Discussion

In this section proposed method compare to other popular denoising methods. In this approach Membership functions allow us to graphically represent a fuzzy set. The fuzzy Gaussian function transforms the original values into a normal distribution. The midpoint of the normal distribution defines the ideal definition for the set, assigned a 1, with the remaining input values decreasing in membership as they move from away from the midpoint in both the positive and negative directions. The input values decrease in membership from the midpoint until they reach a point where the values move too far from the ideal definition and definitely not in the set and are therefore assigned zeros. Another parameter that influences the performance is the patch size, which was empirically chosen to be  $8 \times 8$ .

The results obtained by this method are visually compared to some popular denoising methods in Fig. 3, where shows results obtained for different images strongly corrupted by AWGN with  $\sigma=25$ . Note how this method is capable of removing the noise while retaining much of the finer details, even for such high noise cases. Visually, this method can be seen to be superior to BM3D [7], paper [11].

I also provide quantitative evaluation of this method using peak signal-to-noise ratio (PSNR) as the performance metric. The PSNRs obtained by the methods over different noise levels are shown in Table-I. It can be seen there that this method consistently produces results that are also quantitatively comparable to the state-of-the-art.



**Fig.4 Noise Level vs. PSNR Value for the Lena image**

The Fig.4 shows that graphical representation of the BM3D, Paper [11] and proposed method. From this graphical representation shows that proposed method is better than the BM3D and paper [11] and also observe that whenever the noise level is increases in the different method but decreases the PSNR (peak signal to noise ratio) value. However, when the noise level is nearby 20 db or neighboring the 20 db the proposed method is better than the other two methods and also sees that whenever the noise level at close by 20 db noise is much reduced.

### 4. CONCLUSION

In this paper proposed a new image processing scheme which is based on smooth 1D ordering of the pixels in the given image. A fuzzy filter is described in this paper. The proposed filter is seen to be quite effective in reducing the Gaussian noise; in addition, the proposed filter preserves the image boundaries and fine details satisfactorily. The efficacy of the proposed filter is illustrated by applying the filter on various test images contaminated by different levels of noise. This filter outperforms the existing filters in terms of qualitative and quantitative measures. In addition, the filtered images are found to be pleasant for visual perception, since the filter is robust against Gaussian noise while preserving the image features intact. Further, the proposed filter is suitable for real-time implementation, and applications because of its adaptive in nature.

## 5. REFERENCES

- [1] Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, 2006.
- [2] P. Chatterjee and P. Milanfar, "Clustering-based denoising with locally learned dictionaries," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1438–1451, Jul. 2009.
- [3] G. Yu, G. Sapiro, and S. Mallat, "Image modeling and enhancement via structured sparse model selection," in *Proc. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1641–1644.
- [4] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2481–2499, May 2012.
- [5] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 457–464.
- [6] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 479–486.
- [7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [8] D. Van De Ville, M. Nachtegaele, D. Van der Weken, W. Philips, I. Lemahieu, and E. E. Kerre, "A new fuzzy filter for Gaussian noise reduction," in *Proc. SPIE Vis. Commun. Image Process.*, 2001, pp. 1–9.
- [9] S. Schulte, V. D. Witte and E. E. Kerre, "A Fuzzy Noise Reduction Method for Color Images," *IEEE Transactions on Image Processing*, Vol. 16, No.5, pp.1425–1436, 2007.
- [10] Jamal Saeedi, Mohammad Hassan Moradi, Ali Abedi, "Image Denoising Based on Fuzzy and Intra-scale Dependency in Wavelet Transform Domain", *ICPR, 2010, 2010 20th International Conference on Pattern Recognition (ICPR 2010)*.
- [11] Idan Ram, Michael Elad, Fellow, IEEE, and Israel Cohen, "Image Processing Using Smooth Ordering of its Patches", *IEEE Trans. Image Processing*, vol.22, no.7, pp. 2764-2774, 2013.
- [12] Rafael c. Gonzalez, Richard E. Woods, *Digital Image Processing*, 2nd Ed., Pearson Education, 2002, pp.147-163.
- [13] T. H. Cormen, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2001.
- [14] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.