

IMAGE DENOISING USING IMPROVED FUZZY BASED APPROACH

Jenita Subash

Research Scholar, School of Electronics Engineering (SENSE), VIT University,
Vellore, Tamil Nadu, India

Jenita.subash2014@vit.ac.in

<http://vit.ac.in>

Dr.Kalaivani.S*

Associate Professor, School of Electronics Engineering (SENSE), VIT University,
Vellore, Tamil Nadu , India

*Corresponding author: kalaivani.s@vit.ac.in

<http://vit.ac.in>

Abstract

Digital image gets corrupted due to unlimited and number of unrecognized sources. The denoising of image in algorithmic approaches uses a sliding window that first detects the noisy or corrupted pixel and denoises by comparing with the nearby pixels of same window. The limitations of conventional filters for image denoising and their not-so-efficient accuracy are some major flaws that seek attention. In this paper, the improved fuzzy logic is implemented for image filtering. The medical images are sourced for experimentation and the design is tested on three noises i.e., Poisson, Gaussian and salt & pepper noises. Four parameters are selected to evaluate the performance of proposed system (PSNR, SSIM, IQI and IEF). Proposed architecture works well with salt & pepper and poisson noises. As noise density increases the performance of system is much better compared to performance of traditional median filter.

Keywords: Image Denoising; Fuzzy Decision; Median Filtering; PSNR; SSIM; IQI; IEF.

1. Introduction

In digital image capturing the various sources add spurious and extraneous information known as noise. This information in general terms is referred as noise that corrupts the original pixel information. Robert A. Schowengerdt [1] in his research performed a detailed analysis of noise and their tendency of image corruption. The salt and pepper noise is occurred due to inefficient channels or due to transmission errors. This noise has two intensity values high and low. High values have intensity 1 and low have 0. The white and black combination of noise in image has given it a name salt and pepper. Poisson noise is an impulsive noise that occurs at given interval based on probability of occurrence. This type of noise is either present or absent in noise and cannot be scaled on measuring devices. Another form of statistical noise called as Gaussian noise is the mixture of every single possible noise that can affect the image. The sources of this noise are undetermined and PDF of noise is equal to normal distribution or Gaussian distribution. The noise in any image is vulnerable for studies and detection of alien elements. In medical language the term 'texture' is used to locate the noise. The noise reduction methods attempt to maximize the signal to noise ratio and conserve edges and lines in any image. Before actually segmenting the pixels and classifying the corrupted information, the images are filtered as the preface action. The filtering of image has a rich literature and the techniques are classified in spatial domain filtering and transform domain filtering:

Spatial filtering is a conventional tool that approximates the pixels with vague information due to its high frequency or in form of mean square error (MSE) [5] [6]. Spatial filters are generally non-linear or linear in nature respectively based on approximation stated above. Non-linear filtering employs a low pass filter to process the images in response with assumption that noise inherited in image has high amplitudes in frequency spectrum [7]. The non-linear filtering uses Fourier transform to transform time domain image in frequency domain [8] [9]. If the range of peaks in frequency domain is highly subtle, median filtering is used. Median filters possess low sensitivity in comparison with mean filtering and in low noise content the edges could be preserved [10] [11] [12]. The median filters are further developed into weighted median, relaxed median and conditioned rank selection categories [13]. In linear mode, mean square error is calculated from given window size. Wiener filter is a linear filter that based on the information of stationary signal filters the additive noise to minimize mean square error [10] [14]. However, many researchers have indicated that Wiener is non-efficient for additive noise suppression

cases [15] [16]. Mean Filters were originally invented to minimize the salt & pepper noise [17]. The filter computes the mean of a sliding window of user selected size. The centre pixel of window is modified according to the average of nearby windows. This processing is good for the images that have low noise density. Least Mean Square adaptive filters could be customized for non-stationary images that have discrete values of intensity [18]. The filters have gained considerable interests among researchers for their low computational cost and higher performance than averaging (mean) filters and are employed mainly in cases of salt and pepper noise. The spatial filtering though having low computational cost persist severe consequences in output. The filtered images from both linear and nonlinear methods have blurred edges and other fine details [19] [20].

Transform Domain Filtering can be used either as adaptive or non-adaptive. Adaptive approaches such as ICA tends to create a matrix of independent components with respect to some maximum function [21]. The ICA was originally formulated for signal filtering that seeks separate sources of signal and noise. The algorithm considers the difference in noise and original signal source based on pre-defined function and same is criteria for separation. However in images, the image and noise is sourced from same end thus making the filtration unfeasible. Non-adaptive approaches such as wavelet transform segments the image into daughter wavelets and each wavelet is studied for noise prediction [22]. This approach is beneficial in its approach but the number of components in this method is inversely proportional to SNR of image. That means for low SNR, the number of components are greater that resembles shift invariance.

2. Detection of Noisy pixel

In this paper, image denoising is proposed via fuzzy based decision algorithm for image filtering. To show the novelty of propose method, the tests are performed on three images and three noise sources. We compared our architecture with Median filtering. The organization of paper is as follows: the following section describes the implementation of fuzzy method for image denoising. The mathematical analysis of algorithm is presented in constraints of parameters used for evaluation. The following results section analyzes the performance of proposed method and its efficiency. The paper is concluded with conclusion.

2.1. Variance Method

There are many formats of an image. The RGB format of an image consists of 3 pages with each page having information in two dimensions. To calculate variance the image is required to be transformed in a single page 2D format. The rgb2gray conversion in MATLAB converts the 3 page RGB image to 1 page 2D gray format. A window of customized size is selected to calculate the mean of image. The information of every pixel is summed and averaged by the total number of pixels. For example: if the window size is 3×3, a matrix could be formulated against it:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (1)$$

The mean of matrix is calculated as:

$$S_n = \frac{\sum_{i=1}^n a_i}{n} \quad (2)$$

Here, S_n represents mean of n components for given matrix and a_i is the pixel information of each pixel. For the matrix given in equation 1, the mean in extended form is written as:

$$S_n = \frac{a_{11}+a_{12}+a_{13}+a_{21}+a_{22}+a_{23}+a_{31}+a_{32}+a_{33}}{9} \quad (3)$$

The variance of matrix is defined as averaging the square of difference in pixel information and mean value. The difference of individual pixels with their mean in matrix form is:

$$\begin{bmatrix} a_{11} - S_n & a_{12} - S_n & a_{13} - S_n \\ a_{21} - S_n & a_{22} - S_n & a_{23} - S_n \\ a_{31} - S_n & a_{32} - S_n & a_{33} - S_n \end{bmatrix} \quad (4)$$

The variance of given matrix is:

$$V_n = \frac{[A-S_n]^2}{n} = \frac{\sum_{i=1}^k [a_{ik}-S_n]^2}{n} \quad (5)$$

For the pixels that are corrupted, the variance value will be different from the non-corrupted pixels. Thus by employing a threshold value corrupted pixels could be filtered out. This filtering is done by fuzzy clustering based on its membership and distance functions. The window size in an image could range according to requirements. For example, in a 256×256 image the window size used is generally 8×8 or 3×3. But in additional effort for correcting pixels at boundaries, a 3×3 window is unfit. Hence, some authors prescribe a 3×2 window use for edges.

2.2. Summing the Difference

Consider two windows of size 3×3 represented in form of matrices A and B

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}; B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \quad (6)$$

The difference among both the matrices will be:

$$D = \begin{bmatrix} a_{11} - b_{11} & a_{12} - b_{11} & a_{13} - b_{11} \\ a_{21} - b_{11} & a_{22} - b_{11} & a_{23} - b_{11} \\ a_{31} - b_{11} & a_{32} - b_{11} & a_{33} - b_{11} \end{bmatrix} \quad (7)$$

If the pixels in any of the windows are corrupted, the respective components of matrix D will be high. If the difference is minimal, the pixels are considered to be uncorrupted. In case they have noise, the difference with adjacent pixel will be high. The corrupt information of pixel could exist either in matrix A or in matrix B. If the pixel is corrupted in matrix A, the magnitude of difference will be high and in else case the difference will be negative. The additional benefit of this method is real time predicting of noisy pixel with sliding window. The difference is calculated by storing the values of components of one window and subtracting it with next matrix formed by window during propagation. The generalized equation of matrix D can be written for unknown component μ_{ijkl} in terms of difference in respective component of matrix A $\alpha_{i,j}$ and matrix B $\alpha_{i+ki,j+l}$. $a_{11} - b_{11} = \mu_{ijkl} = |\alpha_{i,j} - \alpha_{i+ki,j+l}|$. The maximum value δ for k^{th} pixel is calculated from maximum function $\delta_{i,j} = \text{Max}\{\mu_{ijkl}\}$

2.3. Median Filter

The median is defined as the centered value of any progression. The corrupted information of pixels is deleted and new information has to be introduced. As the window slides in the image, the median intensity value of the window fills the required information of vacant pixels. Since, determining the actual information of pixel is next to impossible, the approximate value of neighboring pixels in small window provides suitable information. To determine the median value, the values of all the pixels are arranged in increasing order and centered value is predicted as show in Fig.1.

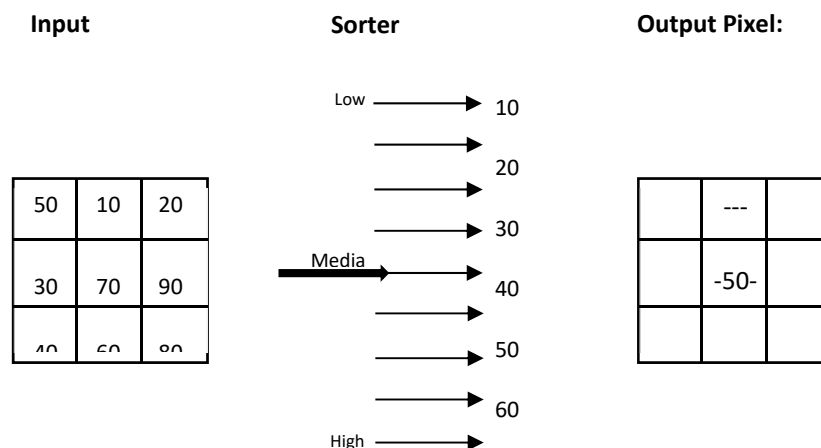


Fig. 1 Median Filter Operation

In case if the window size is of even coordinates, there exist two centers. In this scenario, the mean of both centered values is considered as the median value of window.

3. Fuzzy Approach

In contrast to binary decisions, fuzzy rules tend to find the truth values in the range 0 to 1 where, 1 refers to the completely truth and 0 signifies completely false output. Along with linguistic variables the degrees can be manipulated by using specific functions. Fuzzy has its applications in wide range of domains i.e. from communication to artificial intelligence. A Fuzzy system is setup of a Knowledge Base (KB) and Inference Mechanism (IM). KB is the assembly of fuzzy rules that is backbone of system. These rules are inherited by IM to classify the given objects.

3.1. Membership Function

Fuzzy classification is based on membership function. The values that are assigned as 1 lies within the set and values assigned as zero lies outside the set. In image processing every pixel of the image is a membership function. The functions hold a membership degree based on their distance from KB. For suppose, if the pixel has intensity value that is above 1 or below 0, (just for an assumption), the pixel does not lie in the set of clean pixels. The pixels that falls in this range are good pixels. The range of KB can also be redefined based on the interest of developer. The max and min intensity can be set to 0.75 and 0.25 for clean pixels also.

3.2. Fuzzy Distance

The benefit of fuzzy is that it provides suitable range of true values that are necessary in image processing. Every single pixel holds a different value and thus a range of intensity values are uncorrupt pixels. If the algorithm is only based on 0 and 1 for selection of pixels, the results may lead an error. Fuzzy distance is the difference in the attributes of an individual to the given set of rules. This distance is useful in classification of a pixel into the corrupted or not corrupted domains. For suppose the limits of an uncorrupted pixel is defined in the terms of intensity values having maximum (W_{max}) and minimum (W_{min}) limits. The pixels if falls in this range is uncorrupt or in else case is corrupt pixels. The distance of membership functions with respect to their degrees is calculated in following manner:

If membership-function ψ is written in mathematical form as:

$$\psi_s(a) = \begin{cases} \frac{(P-x)}{P} & \text{if } x \leq P \\ 0 & \text{if } x > P \end{cases} \quad (8)$$

The membership degree of each pixel in a given matrix can be computed from the maximum function and membership function

$$\psi_s(ijkl) = \left(\frac{\delta_{i,j} - \alpha_{i+kl,j+l}}{\delta_{i,j}} \right) \quad (9)$$

Here, $\delta_{i,j}$ is substituted for P and $\alpha_{i+kl,j+l}$ is used against x. From these membership values in windows, the pixels with values very close to 0 and 1 are ruled out. Here, 0 and 1 are hypothetical and can be replaced with another suitable values. The FBDA decision rule is given as following:

$$W_H \leq C_{central} \leq W_L \quad (10)$$

$$S < T_1 \text{ and } V < T_2 \quad (11)$$

Here, W_H and W_L are the maximum and minimum intensity values of a window respectively. The centre pixel if lies in this range, will be considered as uncorrupted pixel or otherwise will be chosen by FBDA for filtering. In summing the difference method the sum should be less than threshold value. Normally, T_1 and T_2 are selected to be 50 and 20 respectively.

4. Experimental setup

The experiments of proposed technique are performed in three noise scenarios declared above. The intensity of noise are chosen at seven levels starting from 10% to 70%. The subjects are images from open source and manual noise is added for evaluation. Three images (Fig. 2) are used for this on various noise densities. The whole system is implemented in MATLAB and the comparison is made among median filtering and Fuzzy based Decision algorithm. The comparison in both the techniques is quantified on 4 parameters given in following Eq. (4).

$$PSNR = 10 * \log_{10} \left(\frac{255^2}{MSE} \right) \quad (12)$$

$$MSE = \sum_{m,n} \frac{[O(m,n) - R(m,n)]^2}{M \times N} \quad (13)$$

$$SSIM = L(O, R) * C(O, R) * S(O, R) \quad (14)$$

$$L(O, R) = (2\mu_0\mu_R + C_1) / (\mu_0^2 + \mu_R^2 + C_1) \quad (15)$$



Fig.2 Three images used in experimentation

$$C(O, R) = (2\sigma_0\sigma_R + C_2) / (\sigma_0^2 + \sigma_R^2 + C_2) \quad (16)$$

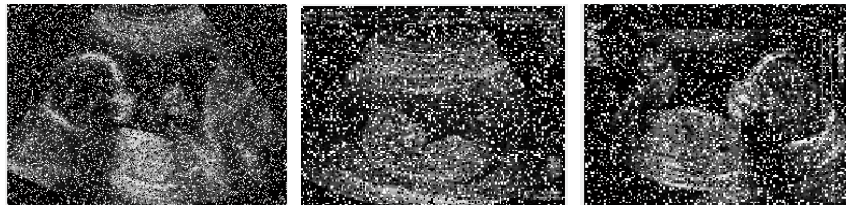
$$\text{Where, } S(O, R) = (\sigma_O \sigma_R + C_3) / (\sigma_O^2 + \sigma_R^2 + C_3)$$

$$C_1 = (K_1 * G)^2, C_2 = (K_2 * G)^2, C_3 = C_2/2$$

$$G = 255; K_1, K_2 \ll 1, (K_1 = 0.001, K_2 = 0.002)$$

$$IEF = (\sum_{m,n} [P(m,n) - O(m,n)]^2) / (\sum_{m,n} [R(m,n) - O(m,n)]^2) \quad (17)$$

The images in Fig.2 are fed to MATLAB simulation of proposed system. In this research, the images under consideration have pixel area of 256×256. The three noises are mixed to these images for evaluation. Fig. 3 is the images with 10% noise of all three images. The poisson noise being impulsive in nature cannot be categorized in classes of noise intensity. The noise in images can directly be predicted using visual inspection.



(a) 30% Salt and Pepper noise in 3 images



(b) Images with Poisson Noise



(c) Images with 20% Gaussian Noise

Fig. 3 Representation of Noise in Images. In Fig.3 (a) 30% Salt and Pepper noise is added to all three images under consideration. In Fig.3(b) Poisson noise is added to images. This noise being impulsive in nature does not possess intensity unit. In Fig.3 (c) Gaussian noise is added at 20% to system.

5. Result

Before processing these images with fuzzy system, the images are preprocessed using variance and summing the difference for neighbouring windows. The comparison among the median and fuzzy based denoising is given in Table 1 to Table 7.

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	37.8143	32.5438	0.99776	0.99239	0.99779	0.99248	238.1375	70.7677
20	32.3441	27.7041	0.99212	0.977	0.99221	0.97728	134.9371	46.3296
30	27.6536	22.8829	0.977	0.9328	0.97728	0.93358	69.2028	23.0702
40	22.4634	17.756	0.92644	0.80595	0.92728	0.80793	27.8384	9.4168
50	19.192	14.2148	0.85154	0.63421	0.85315	0.63727	16.2077	5.1566
60	15.9519	11.1848	0.71785	0.43359	0.72057	0.43712	9.3033	3.104
70	12.2626	8.5477	0.47735	0.23411	0.48124	0.2375	4.6833	1.991

Table 1. Comparison of Median filter and FBDA method for salt and pepper noise in Fig 2(a)

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	28.6629	24.6699	0.97859	0.9445	0.97889	0.9453	27.2392	10.8617
20	27.7213	23.3354	0.97338	0.92537	0.97375	0.92644	44.2166	16.1064
30	25.6972	20.0998	0.95775	0.85088	0.95834	0.8529	43.5226	11.9942
40	23.2732	16.7007	0.92715	0.71628	0.92815	0.71962	33.8049	7.4427
50	20.8578	15.5649	0.9357	0.79465	0.9388	0.79857	20.5485	5.7854
60	16.6594	12.4564	0.91764	0.74578	0.91769	0.74599	15.4572	2.5745
70	11.5979	9.48578	0.89245	0.69475	0.89350	0.69654	11.4798	2.0045

Table 2. Comparison of Median filter and FBDA method for salt and pepper noise in Fig. 2(b)

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	28.7508	25.2898	0.98048	0.95494	0.98073	0.95553	31.3145	14.1138
20	26.7158	23.2767	0.96886	0.92968	0.96927	0.9306	38.4328	17.4096
30	25.3616	21.1267	0.9577	0.89134	0.95824	0.8927	41.0637	15.4869
40	21.7631	16.708	0.90594	0.73961	0.9071	0.74241	24.9875	7.8021
50	18.2476	13.428	0.80356	0.54188	0.80579	0.54584	13.9185	4.5881
60	16.3587	12.5687	0.78011	0.40046	0.78154	0.41658	7.65887	3.6547
70	13.57898	11.2578	0.77854	0.35467	0.77978	0.35911	4.32454	1.6548

Table 3. Comparison of Median filter and FBDA method for salt and pepper noise in Fig. 2(c)

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	18.1262	19.2432	0.84315	0.87821	0.84387	0.87862	0.2625	0.1628
20	13.5236	13.8188	0.71246	0.73998	0.71318	0.7404	1.1153	1.1937
30	10.3315	10.4495	0.60125	0.62548	0.60129	0.62586	1.059	1.0882
40	7.9524	7.9842	0.51128	0.53281	0.51196	0.53319	0.0348	1.0424
50	6.1494	6.13	0.4376	0.4578	0.43838	0.45796	1.0262	1.0216
60	4.7278	4.6719	0.3707	0.38863	0.37177	0.38939	1.0219	0.0088
70	3.5938	3.5026	0.30191	0.31549	0.3036	0.31689	1.1099	0.99866

Table 4. Comparison of Median filter and FBDA method for Gaussian noise in Fig. 2(a)

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	17.847	18.5007	0.82642	0.84502	0.82761	84611	1.1854	3779
20	3.3882	3.6392	0.70519	0.72133	0.70519	0.72232	1.0808	1.1451
30	10.2467	10.3425	0.59762	0.61333	0.59866	0.61421	1.0419	1.0652
40	7.8804	7.8891	0.50835	0.5248	0.5094	0.52562	1.0294	1.0315
50	6.1333	6.0704	0.43777	0.45673	0.43893	0.45756	1.0265	1.0118
60	4.8574	4.7172	0.39849	0.41242	0.3994	0.41309	1.0100	1.0101
70	3.8574	3.6742	0.39821	0.40803	0.39947	0.4095	0.0914	1.0901

Table 5. Comparison of Median filter and FBDA method for Gaussian noise in Fig.2(b)

Noise Ratio (%)	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
10	17.7887	18.6432	0.79391	0.82141	0.79498	0.82234	1.1704	1.4249
20	13.3682	13.6309	0.64334	0.66461	0.64438	0.66545	1.0837	1.1513
30	11.3254	11.4512	0.62587	0.65874	0.64175	0.65945	1.0945	1.0989
40	7.5681	8.1547	0.5287	0.56487	0.5299	0.56852	1.0548	1.0658
50	6.5987	6.4596	0.4632	0.46321	0.4654	0.47864	1.0454	1.0455
60	5.12457	4.9193	0.41578	0.44658	0.41658	0.44893	1.0235	1.0398
70	4.9865	4.1258	0.38564	0.39569	0.3947	0.39999	0.0756	0.0958

Table 6. Comparison of Median filter and FBDA method for Gaussian noise in Fig. 2(c)

Image	PSNR		IQI		SSIM		IEF	
	FBDA	Median	FBDA	Median	FBDA	Median	FBDA	Median
Fig. 2(a)	27.4502	24.9763	0.9716	0.9481	0.972	0.94884	0.4045	0.22884
Fig. 2(b)	28.0903	25.774	0.97723	0.95937	0.97753	0.9599	0.35214	0.20657
Fig. 2(c)	32.3918	32.6965	0.99221	0.99263	0.99231	0.99272	1.1591	1.2434

Table 7. Comparison of Median filter and FBDA method for Poisson noise in Fig. 2

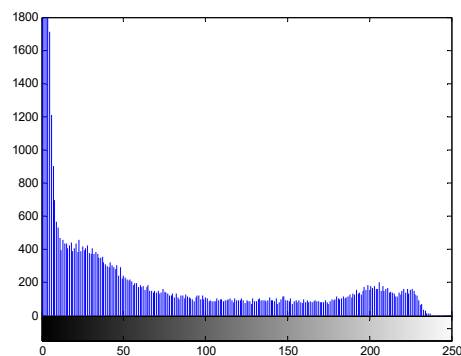


Fig.4 Histogram Graph of Original Image

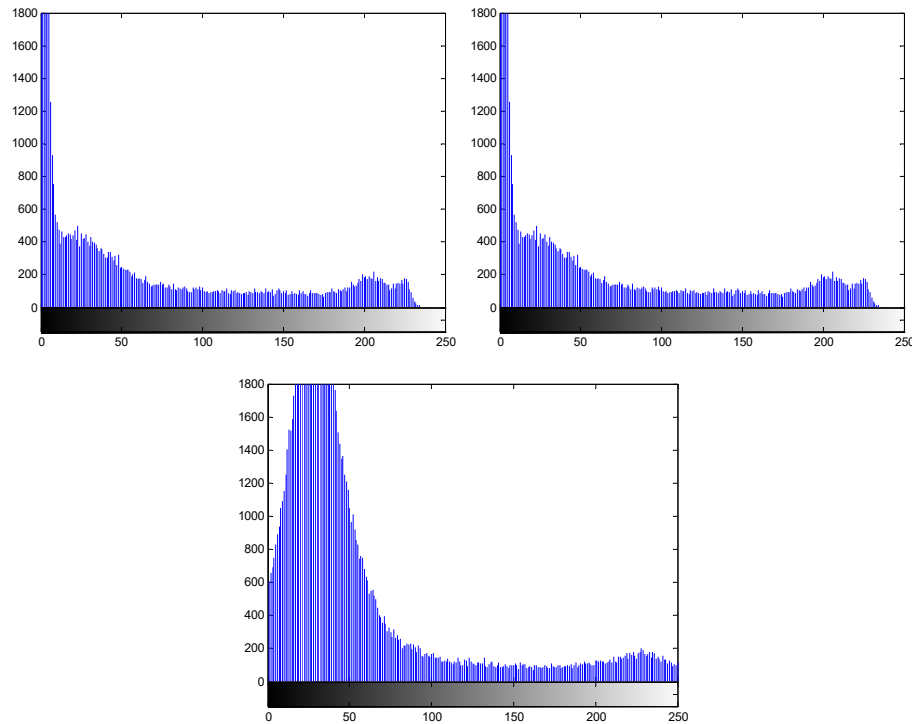


Fig.5 Histogram Graph of FBDA performance on (a) Poisson Noise (b) Salt & Pepper Noise (c) Gaussian Noise

The outputs of variance are fed to fuzzy system. The images are the MATLAB output of proposed algorithm. The FBDA algorithm has better results for salt and pepper and poisson noise for all three images (Table 1, Table 2 and Table 3). Fig. 5(a) and 5(b) depicts that histogram reconstruction is almost similar as of original image for salt & pepper and poisson noises. The median filter has better outputs for Gaussian noise but the difference is minimal only. The observation (Table 4, Table 5 and Table 6) shows with increase in noise intensity at above 50%, PSNR of Fuzzy based approach outperforms the median filter.

6. Conclusion

Image noise filtering using fuzzy is a better option compared to median filter for all types of noises. To witness our statement, three noises i.e. salt & pepper, poisson and Gaussian noises are considered here. Three test images of medical importance are acquired from an open source. The results depict that for impulsive noises such as poisson noise, the FBDA outperforms the Median filter. For salt and pepper noise also, the evaluation parameters voted positively about FBDA algorithm. The Gaussian noise was not efficiently filtered by FBDA algorithm in comparison with median filter, but the minor difference in results of Gaussian, PSNR of FBDA at high noise intensity and high performance in salt & pepper and poisson concludes that FBDA is a better selection for denoising than conventional filtering algorithm. For future work, fuzzy based methods can be improved for efficient working of Gaussian and Speckle noise also.

References

- [1] Schowengerdt, R. A. (1983). Techniques for image processing and classification in remote sensing. Academic Press
- [2] Thangavel, K., Manavalan, R., & Aroquiaraj, I. L. (2009). Removal of speckle noise from ultrasound medical image based on special filters: comparative study. ICGST-GVIP Journal, 9(3), 25-32
- [3] Sarode, M. V., & Deshmukh, P. R. (2011). Reduction of speckle noise and image enhancement of images using filtering technique. International Journal of Advancements in Technology, 2(1), 30-38
- [4] Muthukumar, S., & Raju, G. (2011, November). A non-linear image denoising method for salt-&-pepper noise removal using Fuzzy-Based approach. In Image Information Processing (ICIIP), 2011 International Conference on (pp. 1-5) IEEE
- [5] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. Image Processing, IEEE Transactions on, 13(4), 600-612
- [6] Horner, J. L. (1969). Optical spatial filtering with the least mean-square-error filter. JOSA, 59(5), 553-556
- [7] Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. Signal Processing, IEEE Transactions on, 50(2), 174-188
- [8] Starck, J. L., Candès, E. J., & Donoho, D. L. (2002). The curvelet transform for image denoising. Image Processing, IEEE Transactions on, 11(6), 670-684
- [9] Portilla, J., Strela, V., Wainwright, M. J., & Simoncelli, E. P. (2003). Image denoising using scale mixtures of Gaussians in the wavelet domain. Image Processing, IEEE Transactions on, 12(11), 1338-1351
- [10] Chen, T., Ma, K. K., & Chen, L. H. (1999). Tri-state median filter for image denoising. Image Processing, IEEE Transactions on, 8(12), 1834-1838

- [11] Ng, P. E., & Ma, K. K. (2006). A switching median filter with boundary discriminative noise detection for extremely corrupted images. *Image Processing, IEEE Transactions on*, 15(6), 1506-1516
- [12] Chan, R. H., Ho, C. W., & Nikolova, M. (2005). Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *Image Processing, IEEE Transactions on*, 14(10), 1479-1485
- [13] Hamza, A. B., Luque-Escamilla, P. L., Martínez-Aroza, J., & Román-Roldán, R. (1999). Removing noise and preserving details with relaxed median filters. *Journal of Mathematical Imaging and Vision*, 11(2), 161-177
- [14] Buades, A., Coll, B., & Morel, J. M. (2005, June). A non-local algorithm for image denoising. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (Vol. 2, pp. 60-65). IEEE.
- [15] Frost, V. S., Stiles, J. A., Shanmugan, K. S., & Holtzman, J. (1982). A model for radar images and its application to adaptive digital filtering of multiplicative noise. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2), 157-166.
- [16] Buades, A., Coll, B., & Morel, J. M. (2005). A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2), 490-530.
- [17] Esakkirajan, S., Veerakumar, T., Subramanyam, A. N., & PremChand, C. H. (2011). Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter. *Signal Processing Letters, IEEE*, 18(5), 287-290.
- [18] Kotropoulos, C., & Pitas, I. (1996). Adaptive LMS L-filters for noise suppression in images. *Image Processing, IEEE Transactions on*, 5(12), 1596-1609.
- [19] Aurich, V., & Weule, J. (1995). Non-linear gaussian filters performing edge preserving diffusion. In *Mustererkennung 1995 Springer Berlin Heidelberg*, (pp. 538-545).
- [20] Nitzberg, M., & Shiota, T. (1992). Nonlinear image filtering with edge and corner enhancement. *IEEE transactions on pattern analysis and machine intelligence*, 14(8), 826-833.
- [21] van Hateren, J. H., & Ruderman, D. L. (1998). Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265(1412), 2315-2320.
- [22] Kazubek, M. (2003). Wavelet domain image denoising by thresholding and Wiener filtering. *Signal Processing Letters, IEEE*, 10(11), 324-326.

Authors Profile:



S Kalaivani Narayanan, currently working at the School of Electronics Engineering (SENSE), VIT University. She completed her Ph.D in Anna University during 2011. She has published more than 32 papers in Scopus indexed journals. Her specialized subjects are Image and Signal Processing, Deep learning and Computer Vision.



Jenita Subash is an Assistant Professor in the Department of ECE, Cambridge Institute of Technology, Bangalore. She always focus on research in the area of digital image processing and Stastical Signal Processing. Her qualifications include M.E in Communication Systems (Anna University), B.E(ECE) – Anna University, and she is perusing Ph.D in Vellore Institute of Technology, Vellore, under the guidance of Dr. Kalaivani. S.