

Removal of Impulse Noise using First Order Neighborhood Mean Filter

Priyanka Shrivastava
M.Tech Scholar
Computer Science Department
L.N.C.T, Bhopal

Uday Pratap Singh
Assistant Professor
Computer Science Department
L.N.C.T, Bhopal

Vineet Richhariya
Head of Department
Computer Science Department
L.N.C.T, Bhopal

ABSTRACT

Noise removal is one of the most important parts to get original image from highly corrupted image. The image can be corrupted with noise during transmission from noisy channel, sensors or due to some environmental conditions. This makes the image visually unpleasant. Impulsive noise may occur during transmission which highly corrupts the image. In this paper an algorithm is designed to remove the impulsive noise (salt and pepper) from corrupted gray scale and color images. In past years researchers proposed many algorithms to remove the impulse noise but they fail to give better results at high noise density i.e. 80%-90%. The proposed algorithm works on two stages first stage is to detect the noisy pixel and the second stage is to replace the noisy pixel. This algorithm considers first order neighborhood pixels for detecting the noisy pixel and mean filter is used for de-noising. Color images are also de-noised by extracting the R, G and B pixels from noisy image and then they are de-noised separately and then merged together again to form the color image. All the other algorithms are compared with the proposed algorithms and found that the proposed algorithm have good noise removal capabilities at high densities. The presented algorithm shows better results than Standard Median Filter (SMF), Adaptive Median Filter (AMF), Progressive Switched Median Filter (PSMF), Decision Based Algorithm (DBA), Modified Decision Based Algorithm (MDBA), Modified Decision Based Unsymmetrical Trimmed Median Filter (MDBUTMF), and Modified Non-Linear Filter (MNF). Different grayscale and color images are tested by using the algorithm and it gave better Peak Signal Noise Ratio (PSNR) and Image Enhancement Factor (IEF) at low, medium and high noise densities.

Keywords

Salt and Pepper (SNP), Mean Filter (MF), Peak Signal Noise Ratio (PSNR), Mean Square Error (MSE), Image Enhancement Factor (IEF)

1. INTRODUCTION

Image enhancement is one of the important stages for processing the image in digital image processing field. Image enhancement is the process of making images more useful and it also improves the quality of image. The reason why image enhancement is performed because it highlights interesting details in the image, removes noise from the image and makes the image visually appealing. There are two broad categories of image enhancement techniques spatial domain technique and frequency domain technique. Spatial domain techniques works directly on the manipulation of image pixels whereas frequency domain is based on modifying the Fourier or wavelet transform of image. If manipulation is done directly

on image pixels and if the image is noisy this means any unwanted information is added to the image then de-noising is performed in two parts detection of noise and removal of that particular noise. Noise generally comes from sensors, environmental conditions (rain, snow, lightening etc.) and transmission through noisy channel. De-noising is performed because the image could be visually unpleasant, bad compression or bad analysis. There are different noise types independent of spatial location and spatially dependent. The noises which are independent of spatial location are impulsive noise and AWGN (Additive white Gaussian noise) and the spatially dependent noise considers periodic noise. Impulse noise is again of two types fixed value impulse noise that is also known as salt and pepper and random value impulse noise [1]. Fixed value impulse noise has given this name because the intensity value of image is changed to 0 or 255 when the image is corrupted by noise. Salt and pepper name is used for fixed value noise as 0 refers to pepper because it is a black dot and 255 refers to salt because it is a white dot. In the presence of this noise the image gets corrupted. Therefore, this type of noise is to be removed as it is critical for the extraction of accurate and reliable information from the images [2]-[3]. Filters are better option to remove noise from the image as they are easy to implement on hardware.

Color image processing is also done for color images it can be divided into two areas: pseudo-color image processing and full-color image processing. Pseudo-color image processing is used to enhance the grayscale images with color whereas full-color image processing is used to enhance the color images. There are different color models that are used in color image processing RGB, CMY and HSI. These models are hardware-oriented models. Color images are also corrupted by noise and are de-noised to get visually pleasant image. Color image is a digital image that includes information about each and every color pixel. It is provided three color channels for each pixel these are interpreted as coordinates in some color model. RGB color model is commonly used in computer displays. In this model red, green and blue light is added together in various ways to produce array of colors. De-noising of a color image done by converting them to gray image, de-noised the image and get back to color image. Information might get lost while reconverting gray-level to color image. The information of color images are maintained by extracting the R, G and B pixel from noisy image, de-noised them separately and merged them together to form the color image.

2. LITERATURE SURVEY

For removing the noise from the corrupted image filtering is performed where different types of filters are used for noise removal. These filters suppress the noise from the image and make the image noise free. Several filters have been proposed

by different researchers for removing the noise from the images that are corrupted by impulse noise and they are the best option to remove noise as they are easy to implement on hardware. Different filters that were used include median filter, mean filter, switching median filter, alpha trimmed mean filter, un-symmetric trimmed median filter etc.

Many researchers have suggested various filtering techniques for removing salt and pepper noise. Among these Standard Median Filter (SMF) is easy to implement and is also reliable. However, its major drawback is that this filter is effective only at low densities. When density level is increased over 50% then the edge details of original image is not preserved [4]. To overcome this drawback several methods have been proposed to remove salt and pepper noise at high densities. Filtering with 3x3 mask is used for keeping the computation time of implementation minimum. Use of small filtering window for removing noise is insufficient. So, Adaptive Median Filter (AMF) has been proposed where the filtering window size is expanded pixel by pixel to get noise free pixel. This filter performs well at low densities. But at high densities the expansion of window size leads to blurring of image [5]. After that researchers have introduced Switching Median Filter [6], [7]. This filter uses predefined threshold value for recovering the corrupted image. Major drawback of this filter is that defining robust decision is difficult and details and edges are not recovered at high densities noise level.

To overcome the above filters drawback Decision Based Algorithm (DBA) has been proposed [8]. In this algorithm image is de-noised using 3x3 window. Here the pixel is processed only if its value is either 0 or 255 otherwise it is left unchanged. At high density noise level this leads to median value of 0 or 255 which is again noisy. In such case neighborhood pixel is used for replacement. But the repeated replacement of neighboring pixel produces streaking effect [9]. In order to avoid this drawback, Decision Based Un-symmetric Trimmed Median Filter (DBUTMF) [10] is proposed. This filter belongs to the DBA family. In this filter instead of removing from neighborhood pixel un-symmetric trimmed median value is taken. At high densities if the selected window contains all 0 or 255 or both then trimmed median value cannot be used. So this results bad at high densities that is at 80% to 90%. To avoid this we go for Modified Decision Based Un-symmetric Trimmed Median Filter (MDBUTMF) [11]. All the above algorithms do not perform well at high densities. To overcome Modified Non-Linear Filter (MNF) [12] is proposed. It gives better result at high densities. It yields better results than all previous algorithms at high densities with better Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF) values. But the output result is not much satisfied and the proposed algorithm provides much better result than all the previously designed algorithms.

The outline of the paper is structured as follows. Section 3 describes proposed algorithm and the steps that are involved in the algorithm. The detailed description of proposed algorithm is involved in Section 4. Simulation results with colored lena image are presented in Section 5. Finally Section 6 concludes the paper.

3. PROPOSED ALGORITHM

The proposed noise removal using First Order Neighborhood Mean Filter (FONMF) algorithm processes each and every pixel of an image by detecting the noisy pixel in the image.

This algorithm is based on windowing technique so a least size window 3 x 3 is taken to reduce the complexity. Here the pixel of interest is the centre pixel known as processing pixel $P(i, j)$. Processing pixel is checked whether it is noisy or noise free by verifying that the pixel lies between maximum (255) and minimum (0) grey level values. If the pixel is in between the range of grey level then the pixel is noise free otherwise the pixel is corrupted pixel and it is processed to be replaced with the noise free pixel value. Uncorrupted pixels that lie in the range are left unchanged. In this work RGB color model is chosen to represent the color image. Noisy color images are formed by adding salt and pepper noise independently to each of these color components Algorithm is applied to each R, G and B planes and finally merged to get the color image. The steps for the algorithm are as follows:-

Step 1: First we take an initial color image and apply on it fixed valued impulse noise (Salt and Pepper noise). This color image is read as Y.

Step 2: In the second step split the image into RGB component. By taking Red, Green and Blue pixels for checking noise.

Step 3: Now in third step the pixel is read and processed by using the following steps:-

Step 3.1 Firstly check whether the pixels are between 0 to 255 ranges or not, here two cases are generated. If the processing pixel lies in between 0 and 255 ($0 < l(i, j) < 255$) then Case 1 is followed otherwise Case 2 is followed. Here $l(i, j)$ is the image processing pixels.

Case 1- If Pixels are between $0 < l(i, j) < 255$ then, they are noise free and move to restore the image.

Case 2- If the pixels does not lie in the range then they are moved to step 3.2.

Step 3.2: In the second step we will work on noisy pixel of step3.1 now select a window $W(i, j)$ of size 3 x 3. Assume that the processing noisy pixels are $X(i, j)$, that is processed in the next step.

Step 3.3: If the preferred window contains not all elements as 0's and 255's. Then remove all the 0's and 255's from the window, and send to restore the image. Now find the mean of the remaining pixels. Replace $X(i, j)$ with the mean value. This noise removed image restores in de-noised image at the last step.

Step 4: Repeat steps 3.1 to 3.3 for RGB components. For green the noisy pixel are represented by $m(i, j)$ and for blue the noisy pixel are represented by $n(i, j)$. Also the restored image is represented by $Y(i, j)$ and $Z(i, j)$ respectively.

Step 5: Whole process is performed until all pixels red, green and blue in the whole image are processed. And finally the RGB components are merged to get the final de-noised image.

Hence a better de-noised image is obtained with improved PSNR, IEF and also shows a better image with very low blurring and improved visual and human perception. Flowchart for algorithm is shown in figure 1.

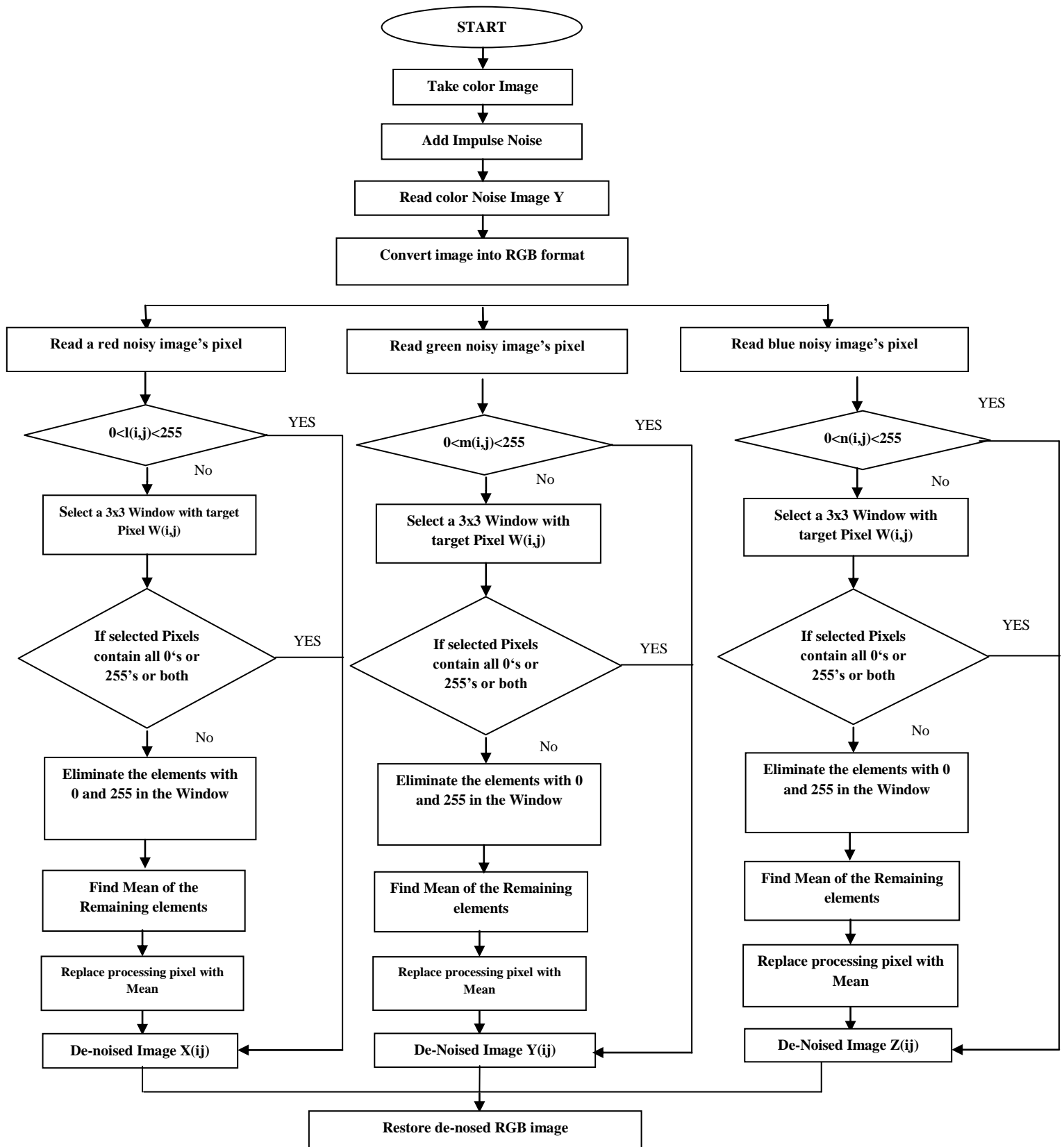


Fig 1: Flowchart of Proposed algorithm FONMF

4. ILLUSTRATION OF FONMF

Each and every pixel of the image is processed and checked for the presence of fixed valued impulse noise (salt and pepper noise). Three different cases are seen when 3x3 window is chosen for pixel processing and these three cases are illustrated in this section:-

90	85	254	0	255
210	255	255	0	74
83	0	(255)	0	255
255	255	0	255	82
145	123	98	96	0

Example I

98	42	255	62	123
232	255	123	0	75
83	78	(0)	58	25
0	145	255	93	255
155	74	87	255	177

Example II

165	68	255	0	146
212	82	99	27	75
83	46	(73)	69	128
255	80	78	92	255
155	132	87	47	143

Example III

Case i): In this case after the 3x3 window is selected the processing pixel is checked and if the processing pixel contains 0 or 255 pixel value then that pixel is noisy pixel i.e. salt and pepper noise. Now if the processing pixel is noisy the neighborhood pixels are checked if all the neighboring pixels value are also 0 or 255 then mean of the surrounding pixels are calculated and replaced with the value of processing pixel. If median of the surrounding pixel is taken then it would again give a noisy pixel value i.e. 0 or 255. So, mean is preferred as compared to median. The matrix form for the case is shown above in example I. The above matrix contains a dashed line rectangle which is the selected 3x3 window where '255' is the processing pixel and all the other values surrounding the processing pixel are 0 or 255.

Case ii): In this case if the selected 3x3 window contains processing pixel as 0 or 255 and the neighborhood pixels are not all 0 or 255 then the processing pixel is noisy and it has to be replaced so the pixel is replaced by taking one dimensional array of the matrix. This case matrix form is shown in example II where the dashed line rectangle shows the selected 3x3 window with '0' as the processing pixel and not all the neighborhood pixel are 0 or 255. Now to remove noise we will take one dimensional array of the surrounding pixels here in example II the array will be [255 123 0 78 0 58 145 255 93]. Then 0 and 255 are removed from the array so the array becomes [123 78 58 145 93]. Now mean of the values are calculated and processing pixel is replaced with the mean of the values in array. Now the processing pixel value is noise free and not 0 or 255.

Case iii): The last case shows that if the selected window contains a noise free pixel not 0 or 255 but the value between 0 to 255 as the processing pixel then it does not require any changes and it is left unchanged. As shown in example III the

dashed line shows the 3x3 selected window with processing pixel as '73' which is a noise free pixel. Since it a noise free pixel it does not require any processing and left unchanged.

All the three cases are checked for each and every pixel value after 3x3 window is formed and the processing of the pixel is performed as discussed in different cases. This processing gives the noise free image with no noisy pixel. Here noise means salt and pepper as we are considering noisy pixel as 0 or 255.

5. RESULT ANALYSIS AND COMPARISON

We have used Matlab R2012b as the simulation tool. Performance of the proposed algorithm is tested with different gray scale and color images. The image of size 256x256 is taken and up to 90 % of noise density is taken and algorithm is applied for results. Here in this paper 'Lena.jpg' is used as original image and tested for performance comparing to different algorithms. The image is corrupted by fixed value impulse noise i.e. salt and pepper noise. Performances are quantitatively measured with various noise densities for Peak-Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE) and Image Enhancement Factor (IEF) defined (1), (2) and (3) respectively:

$$PSNR = 10 \log_{10} \frac{(255)^2}{MSE} \quad (1)$$

$$MSE = \frac{\sum_{i=1}^m \sum_{j=1}^n \{Y(i, j) - \hat{Y}(i, j)\}^2}{m \times n} \quad (2)$$

$$IEF = \frac{\sum_{i=1}^m \sum_{j=1}^n \{\eta(i, j) - Y(i, j)\}^2}{\sum_{i=1}^m \sum_{j=1}^n (\hat{Y}(i, j) - Y(i, j))^2} \quad (3)$$

Here m x n is the size of the image. Y (i, j) represents the original image and $\hat{Y}(i, j)$ represents de-noised image and $\eta(i, j)$ represents noisy image. The noise density is varied from 10% to 90%. The results show improved performance.

Table I shows the comparison of PSNR values of different algorithms tested on Lena image. The PSNR calculated by proposed algorithm is much better as compared to other algorithms. Figure 2 shows the graphical representation for the same. The graph of different algorithm is designed and proposed algorithm give better result by yellow color line with the values that were calculated during simulation. Table II shows the comparison of IEF values for different algorithms on Lena image. The IEF calculated is better than other algorithms. Figure 3 shows graphical representation for IEF values in which proposed algorithm is shown by yellow line and values shows better result than other algorithms. Figure 4 and 5 shows the results for the applied proposed algorithm on color Lena image at 80% and 90% noise densities respectively on different algorithms MF, AMF, PSMF, DBA, MDBA, MDBUTMF, MNF and proposed algorithm FONMF.

TABLE I. Comparison of PSNR Values of Different Algorithms for Lena Image at Different Noise Densities

Proposed filter	Noise Density %								
	10	20	30	40	50	60	70	80	90
MF	28.49	25.75	21.84	18.40	14.73	12.23	9.98	8.02	6.57
AMF	21.98	21.92	21.47	21.47	20.65	18.40	14.85	11.29	8.06
PSMF	30.64	28.20	25.55	22.69	19.44	12.85	10.52	8.48	6.78
DBA	36.75	33.26	30.56	28.26	26.28	24.53	22.77	20.14	17.12
MDBA	36.75	33.26	30.53	28.29	26.25	24.63	22.93	20.40	17.22
MDBUTMF	38.12	34.60	32.14	32.08	28.21	26.59	24.38	22.01	17.98
MNF	37.34	34.23	32.14	30.57	29.00	27.82	26.05	24.32	21.32
FONMF	38.79	35.59	33.26	31.66	30.06	28.56	26.88	24.85	21.56

TABLE II. Comparison of IEF Values of different Algorithms for Lena Image at Different Noise Densities

Proposed filter	Noise Density %								
	10	20	30	40	50	60	70	80	90
MF	20.5	21.3	13.1	7.8	4.2	2.8	1.9	1.4	1.1
AMF	4.2	8.9	12.9	16.0	16.4	11.8	6.0	3.0	1.6
PSMF	33.1	38.1	30.6	21.2	12.5	3.3	2.2	1.6	1.2
DBA	137.9	120.5	97.6	76.1	61.2	48.4	37.9	23.7	13.2
MDBA	137.9	120.5	96.9	76.8	60.7	49.5	39.2	25.2	13.5
MDBUTMF	189.1	164.0	140.4	116.0	95.5	77.8	54.8	36.5	16.2
MNF	158.0	150.8	140.4	129.9	114.6	103.3	80.6	62.3	34.9
FONMF	217.5	195.0	175.4	160.2	135.4	107.5	88.6	63.0	35.5

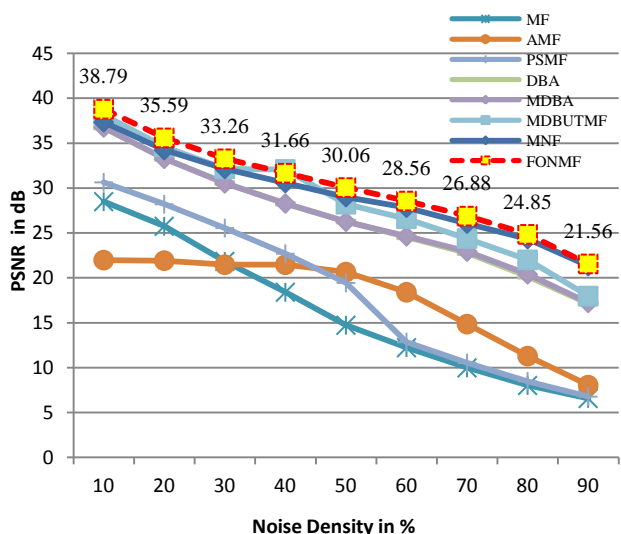


Fig. 2 Noise density versus PSNR (db) for Lena image

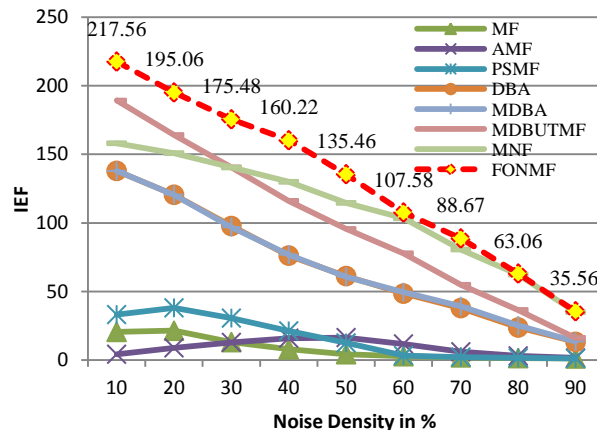


Fig. 3 Noise density versus IEF for Lena image

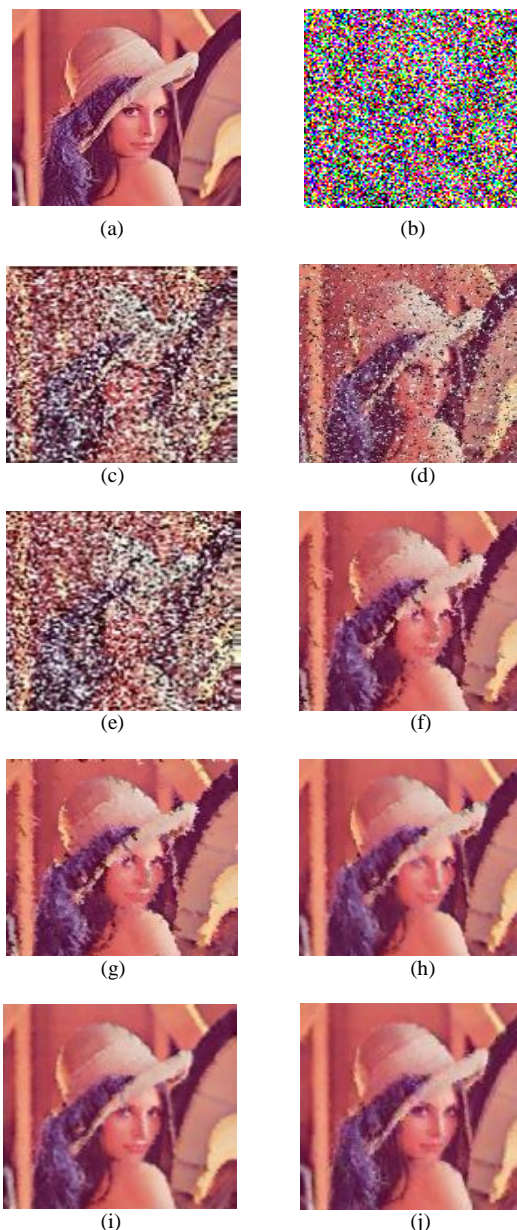


Fig. 4 Results for 80% noise corrupted lena color image (a) Original image (b) Noisy image (c) MF (d) AMF (e) PSMF (f) DBA (g) MDBA (h) MDBUTMF (i) MNF (j) FONMF

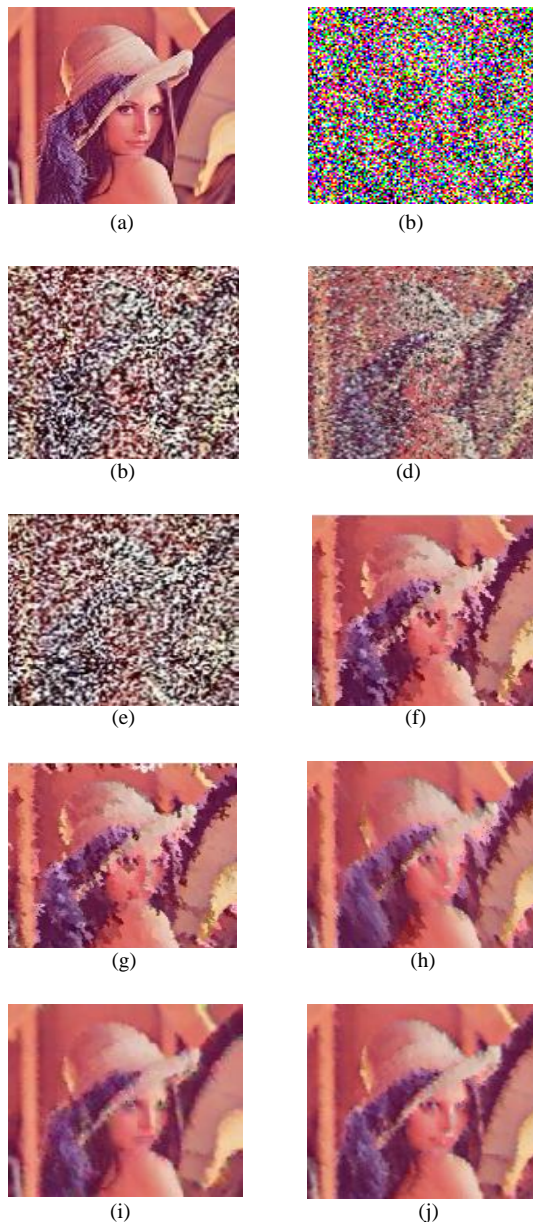


Fig. 4 Results for 90% noise corrupted lena color image (a) Original image (b) Noisy image (c) MF (d) AMF (e) PSMF (f) DBA (g) MDBA (h) MDBUTMF (i) MNF (j) FONMF

6. CONCLUSION

A new algorithm (FONMF) has been proposed for noise removal at higher densities noise 80% to 90%. This algorithm gives better results than MF, AMF, PSMF, DBA, MDBA and other existing algorithms in terms of PSNR and IEF. The figure 2 and 3 shows the graphical result for PSNR and IEF comparing other algorithms. The image performance has been tested at low, medium and high densities noise on both grayscale and color images. The figure 4 and 5 shows lena

color image result after applying the algorithm. At high density level this algorithm provides better results in comparison with other existing algorithms. Due to limited window size it requires less computing time. The proposed algorithm is effective for fixed valued impulse noise i.e. salt and pepper noise at low densities to high densities.

7. REFERENCES

- [1] T.A. Nodas and N.C. Gallagher, Jr., "The output distribution of median type filters," IEEE Trans. Communication, 32(5): 532-541, 1984.
- [2] H.Hwang and R.A.Haddad, "Adaptive median filters: new algorithms & results," IEEE Transactions on image processing, Vol. no:4, pp.499-502, 1995.
- [3] J.Astola and P.Kuosmaneen, Fundamentals of Nonlinear Digital Filtering, 1997.
- [4] S.Zhang and M.A. Karim, "A new impulse detector for switching median filters," IEEE signal Process.Lett., vol. 9, no. 11, pp.360-363, Nov. 2002.
- [5] Fanzhi Kong, Wenbin Ma, "A Fast adaptive Mean Filtering Algorithm," 2nd International Conference on Industrial and Information Systems, IEEE 2010.
- [6] P.E. Ng and K.K.Ma, "A switching median filter with boundary discriminative noise detection for extremely corrupted images," IEEE Trans. Image Process., vol. 15, no. 6, pp. 1506-1516, Jun 2006.
- [7] K.S. Srinivasan and D.Ebenezer, "A new fast and efficient decision based algorithm for removal of high-density salt and pepper noise in image," EURASIP J. Adv. Signal Process, 2010.
- [8] K.S. Srinivasan and D.Ebenezer, "A new fast and efficient decision based algorithm for removal of high-density impulse noises," IEEE Signal Processing Letters, vol. no:14 pp.189-192, 2007.
- [9] V. Jayaraj and D. Ebenezer, "A new switching-based median filtering scheme and algorithm for removal of high-density salt and pepper noise in image," EURASIP J. Adv. Signal Process., 2010.
- [10] K.S. Srinivasan, V.Jayaraj and D.Ebenezer, "A new and efficient algorithm for removal of high-density salt and pepper noise in images and videos," Second Int. Conf. Computer Modeling and Simulation, 2010, pp. 409-413.
- [11] S.Esakirajan, T.Veerakumar, Adabala N.Subramanyam, and C.H.PremChand, "Removal of high Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter," IEEE Signal Processing Letters, vol. 18, no.5, May 2011.
- [12] T.M.Benazir and B.M.Imran, "Removal of High and Low Density Impulse Noise From Digital Images Using Non Linear Filter," International Conference on Signal Processing, Image Processing and Pattern Recognition, [ICSPR) 2013.