SALT AND PEPPER NOISE REDUCTION USING MDBUTM FILTER WITH FUZZY BASED REFINEMENT

T. Vimala*

Abstract:

A Modified Decision Based Unsymmetrical Trimmed Median Filter (MDBUTMF) followed by Fuzzy Noise Reduction Method (FNRM) is proposed for the restoration of color images that are highly corrupted by salt and pepper noise. The proposed filter (MDBUTMF) replaces the noisy pixel by trimmed median value when some of the elements with values 0’s and 255’s are present in the selected window. If all the pixel values in the selected window are 0’s and 255’s means then the noisy pixel is replaced by mean value of all the elements present in that selected window. The throughput of MDBUTMF is a partial noise removed image. It is further processed by FNRM. The FNRM has two sub-filters. The first sub-filter computes fuzzy distances between the color components of the central pixel and its neighborhood. The target of the second sub-filter is to correct the corrupted pixels. The throughput of FNRM is a fully noise removed image. Simulation results show the feasibility of the proposed method. The proposed method is tested against different color images and it gives excellent Peak Signal-to-Noise Ratio (PSNR) than the Median Filter (MF), Switching Median Filter (SMF), Boundary Discriminative Noise Reduction Algorithm (BDNRA), Decision Based Algorithm (DBA), and Decision Based Unsymmetric Trimmed Median Filter (DBUTMF).

***Keywords*:** Fuzzy filter, noise reduction, salt and pepper noise, trimmed median filter.

* II M.E. (Applied Electronics), Department of ECE, PSN college of Eng and Tech.

I. INTRODUCTION:

Digital images are often corrupted by impulse noise also known as salt and pepper noise due to channel transmission errors or introduced during the signal acquisition stage. The goal of impulse noise removal is however achieved by means of filters. The most commonly used filters are the Median Filter (MF), Switching Median Filter (SMF), Boundary Discriminative Noise Reduction Algorithm (BDNRA), Decision Based Algorithm (DBA), and Decision Based Unsymmetric Trimmed Median Filter (DBUTMF). Among these the Median Filter (MF) is used widely because of its effective noise suppression capability [1]. However MF has a drawback that is, it modify both noise and noise-free pixels. To overcome this disadvantage Adaptive Median Filter (AMF) [2] is proposed. AMF perform well at low noise densities. But at high noise densities the window size has to be increased which may leads to blurring the image.

In order to overcome the above drawback, Switching Median Filter (SMF) [3] is proposed. In SMF the noise-free pixels are left unchanged. But it is effective only up to 40% of noise density. To avoid this demerit Boundary Discriminative Noise Reduction Algorithm (BDNRA) is proposed. In order to separate the pixels into three groups the BDNRA [4] uses two boundaries. These boundaries are very useful to provide high noise detection accuracy. The BDNRA achieves zero miss-detection rates only up to 70% of noise density. To avoid this drawback Decision Based Algorithm (DBA) [5], [7] is proposed. The higher correlation between corrupted pixel and neighborhood pixel leads to better edge preservation. But the continuous replacement of corrupted pixel by neighborhood pixel leads to streaking effect.

To overcome this drawback Decision Based Unsymmetric Trimmed Median Filter (DBUTMF) is proposed [6]. In case if the selected window contains all the values as 0's and 255's means then trimmed median value cannot be obtained, so this is also not an effective one. The proposed Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) and Fuzzy Noise Reduction Method (FNRM) avoid the above drawback even at high noise densities. The proposed method provides better Peak Signal-to-Noise Ratio (PSNR) than the existing methods.

The rest of this paper is structured as follows. The detailed description of the proposed method is presented in Section II. Section III contains the illustration of Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF). A brief introduction of Fuzzy Noise

Reduction Method (FNRM) is given in Section IV. Simulation results and conclusions are finally presented in Section V and VI.

II. PROPOSED METHOD:

In the proposed method first the noisy image is read then based on some decision salt and pepper noise detection takes place. At the end of the detection stage the noisy and noise-free pixels get separated. The noise-free pixel is left unchanged and the noisy pixel is given to the Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF). The MDBUTMF produces an image as its throughput that is a partially noise removed one. And it is further processed by Fuzzy Noise Reduction Method (FNRM). Finally the FNRM provides a restored image that is fully free from noise. The proposed method provides the final output image with higher PSNR value. It is clearly shown in Fig.1.

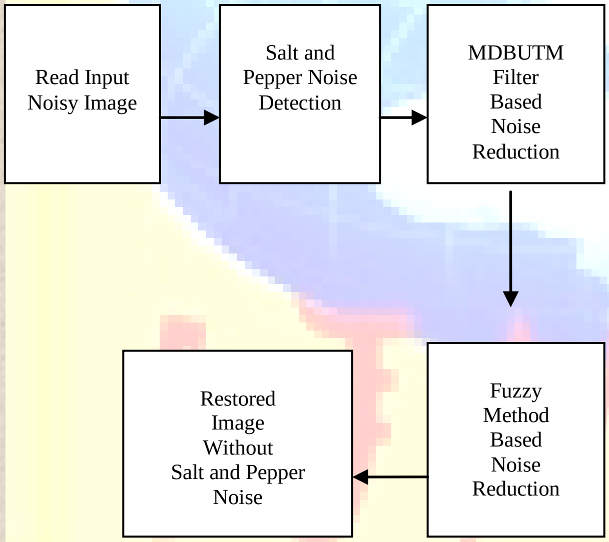

Read Input Noisy Image
Salt and Pepper Noise Detection
MDBUTM Filter Based Noise Reduction
Fuzzy Method Based Noise Reduction
Restored Image Without Salt and Pepper Noise

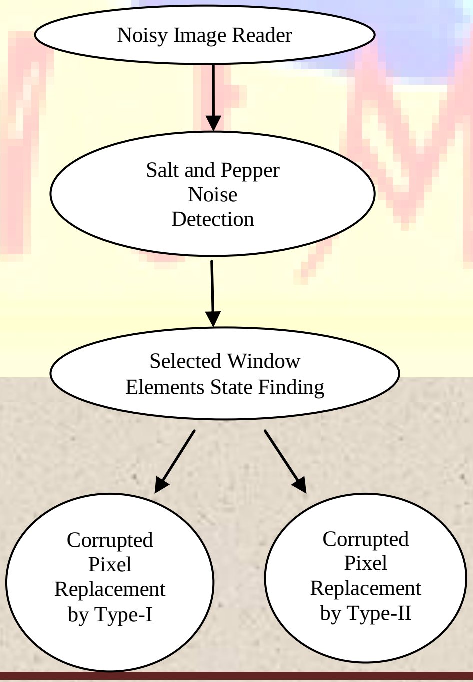
Fig. 1. Block Diagram of Proposed Method.

The data flow of the proposed method is shown in Fig.2. The flow goes in the way that, first the noisy image is given to a noisy image reader. Followed by this is the salt and pepper noise detection. After this, based on the state of the elements the corrupted pixel is either replaced by Type-I or replaced by Type-II. As a result of this a partially noise removed image is being produced. It is further processed by Fuzzy First Sub-Filter and Fuzzy Second Sub-Filter. Finally a restored image without Salt and Pepper noise is obtained as throughput.

Type-I): If the selected window contains all the elements as 0’s and 255’s means, then replace the processing pixel by the mean value of the elements present in that window.

Type-II): If the selected window contains not all elements as 0’s and 255’s. Then eliminate 0’s and 255’s and find the median value of the remaining elements. Replace the processing pixel with the median value.

The clear explanation of Type-I and Type-II with examples is given in Section III. The output images produced by the combination of Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) plus the Fuzzy Noise Reduction Method (FNRM) contain excellent Peak Signal-to-Noise Ratio (PSNR) than the existing methods. A higher Peak Signal-to-Noise Ratio (PSNR) would normally indicate that, the reconstruction is of very high quality. The Peak Signal-to-Noise Ratio (PSNR) is defined as the ratio of the maximum possible power of a signal to the power of corrupting noise signal. The expression for Peak Signal-to-Noise Ratio (PSNR) is given in section V. The clear explanation of Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) is given in Section III and the explanation for Fuzzy Noise Reduction method is given in Section IV.


Noisy Image Reader
Salt and Pepper Noise Detection
Selected Window Elements State Finding
Corrupted Pixel Replacement by Type-I
Corrupted Pixel Replacement by Type-II

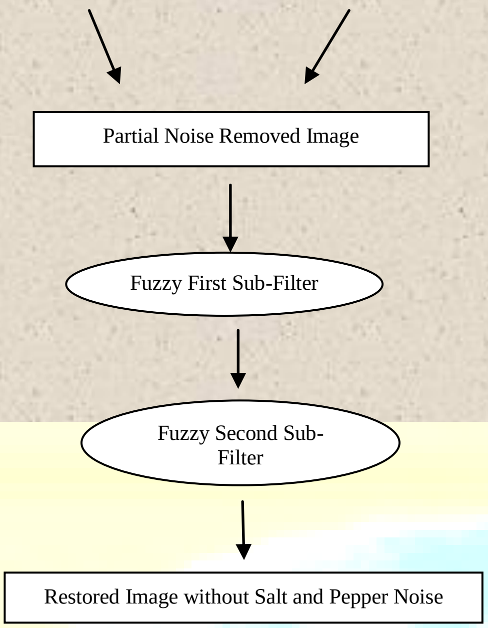

Partial Noise Removed Image
Fuzzy First Sub-Filter
Fuzzy Second Sub-Filter
Restored Image without Salt and Pepper Noise

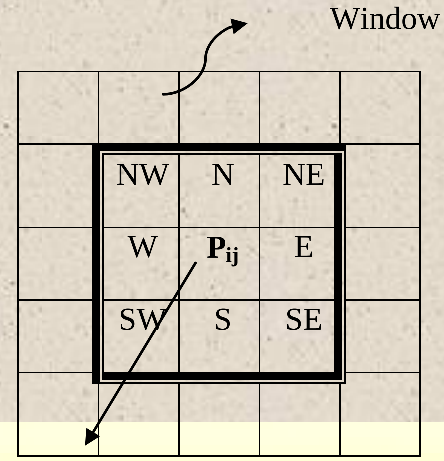
Fig. 2. Data Flow of Proposed Method.

III. ILLUSTRATION OF MDBUTM FILTER:

The Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) has several stages of operations. The stages are given below with examples.

Stage 1: The MDBUTM Filter selects a 2D-window of size 3×3. The center pixel in the selected window is the processing pixel and it is denoted as $P_{ij.}$ It is given in Fig.3. The neighboring pixels of the processing pixel P_{ij} are present in the directions NW, N, NE, W, E, SW, S, and SE. The positions of these directions are (i-1,j-1), (i-1,j), (i-1,j+1), (i,j-1), (i,j+1), (i+1,j-1), (i+1,j) and (i+1,j+1) respectively. The directions are clearly mentioned in the following Fig.3. The X-axis is considered for ‘i’ and Y-axis is considered for ‘j’. Followed by the Stage 1 are the Stage 2 and Stage 3. The Stage 3 consists of two types Type I and Type II.

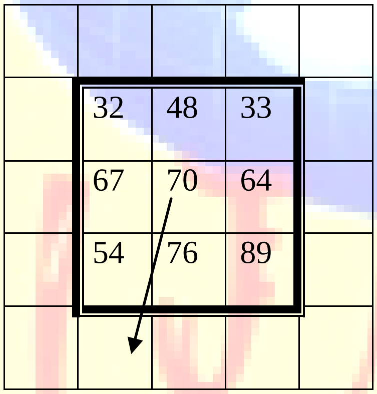
Window of size 3x3



Processing pixel

Fig. 3. Processing pixel in a selected window.

Stage 2: If the processing pixel (P_{ij}) lies between the values 0's and 255's ($0<P_{ij}<255$) then the processing pixel is considered as a noise free pixel and it is left unchanged. For example if the processing pixel value is 70 that is (0<70<255) then it is consider as a noise-free pixel and it is left unchanged. It is shown in Fig.4.



Processing pixel (P_{ij})

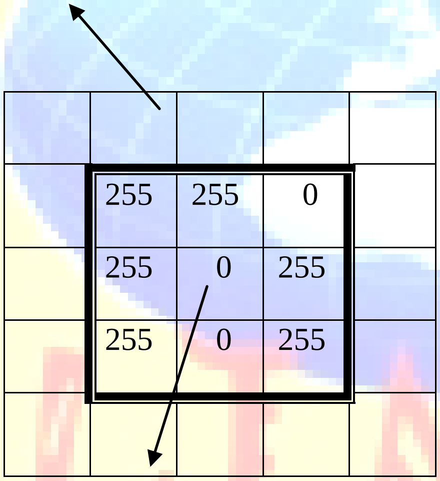
Fig.4. window contains noise-free pixel as a processing pixel.

Stage 3: If P_{ij}=0 or P_{ij}=255 then the processing pixel is considered as a corrupted one and it is get processed by one of the following types. The selection among the types is based on the state of the elements present in the selected window. The types are given below with examples.

Type-I): If the processing pixel (P_{ij}) in a selected window takes the value as either 0 or 255 that is pepper or salt noise and the neighborhood pixels also take the values either as 0 or 255 or both. Then the mean value of the selected window is found and the processing pixel is replaced by the mean value. It is clearly given below by an example through Fig.5.

In Fig.5, the selected window contain the processing pixel (P_{ij}) value as 0 that is pepper noise and the neighborhood pixels also take the values as 0's and 255's. So if we take the median value of the selected window it will be either 0 or 255 which is again noisy. To avoid this problem the mean value of the selected window is found and the processing pixel P_{ij} is replaced by the mean value. Here the mean value of the selected window is 170. So the processing pixel is replaced by the mean value of 170. The mean value is found by adding all the pixel values present in the selected window and it is get divided by the total number of elements present in that selected window.

Selected window of size 3x3



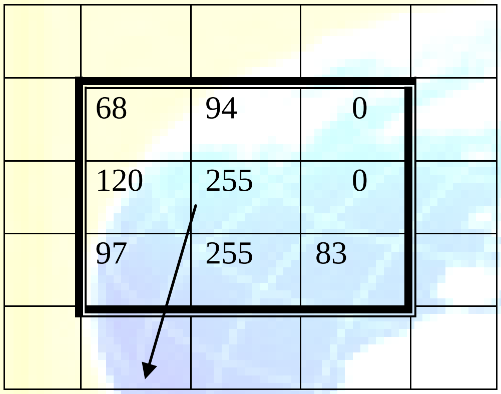
	255	255	0	
	255	0	255	
	255	0	255	

Pepper noise as processing pixel (P_{ij})

Fig. 5. Example of Type-I

Type-II): If the selected window contains some of the elements with values 0's and 255's, and the remaining elements with values between 0 and 255 means then we have to take the trimmed median value and replace the processing pixel by using that trimmed median value. The noisy pixel values (0's and 255's) are first removed from the selected window and then the median value is found by using the remaining pixel values present in that selected window. It is explained in Fig.6, through an example.

In Fig.6, the selected window contains the processing pixel with value 255 that is P_{ij}=255. And some of neighborhood contain noisy pixels with values 0's and 255's and some other contain noise-free pixels. So in order to find the trimmed median value we have to first form the 1-D array of the selected window elements as [68 94 0 120 255 0 97 255 83]. After eliminating the 0's and 255's from the selected window we have the remaining pixels from that selected window is [68 94 120 97 83]. From this we get a median element with value 120. Hence replace the processing pixel P_{ij} by 120.



	68	94	0	
	120	255	0	
	97	255	83	

Salt noise as processing pixel (P_{ij})

Fig.6. Example of Type-II.

Stage 4: Repeat stages 1 to 3 until all the pixels in the entire image are processed. This is the last stage of the process. Finally it produces a partial noise removed image.

IV. FUZZY NOISE REDUCTION METHOD:

The output of MDBUTMF is a partially noise removed one. So it is get further processed using Fuzzy Filter. The Fuzzy Filter consists of two sub-filters namely Fuzzy First Sub-Filter and Fuzzy Second Sub-Filter. The Fuzzy First Sub-Filter computes fuzzy distances between the color components of the processing pixel and its neighborhood. These distances determine in what degree each component should be corrected. The aim of the Fuzzy Second Sub-Filter is to correct the corrupted pixels. The detailed explanations of fuzzy filters are given below.

A. *FUZZY FIRST SUB-FILTER*

The main aim of the Fuzzy First Sub-Filter is to average the processing pixel by using its neighborhood pixel values. Meanwhile care should be taken in order to avoid the damage of the image features such as edges and color component distances. The color component distances are used to differentiate between the local variations caused by noise and the image structures such as edges. This is one of the main concerns of the Fuzzy First Sub-Filter.

In order to find the color component distances, the following couples are used. They are rg(i, j), rb(i, j) and gb(i,j). The couple red and green is given by rg(i,j)=(N(i,j,1), N(i,j,2)), the couple red and blue is denoted as rb(i,j)=(N(i,j,1),N(i,j,3)) and the couple green and blue is denoted by gb(i,j)=(N(i,j,2),N(i,j,3)). Where N (i, j, 1), N (i, j, 2) and N (i, j, 3) are the certain pixel positions of a noisy input image N for the red, green and blue components. The size of the window considered here is (2K + 1) × (2K + 1) and it is get centered at (i, j) that is the processing pixel. For each pixel in the selected window, certain weights are get assigned based on the color components. The weights that are get assigned for red, green and blue at position (i+k, j+l) are w(i+k, j+l, 1), w(i+k, j+l, 2) and w (i+k, j+l, 3) respectively with parameters $k,l \in \{-K \ldots K\}$. The distance between the couple rg(i,j) and rg(i+k, j+l) is calculated according to the Minkowski's distances. And it is given below in expression (1).

$$D\,(rg(i,j), rg(i+k, j+l)) = [(N(i+k, j+l, 1) - N(i,j,1))^{\delta} +$$

$$(N(i+k, j+l,2) - N(i,j,2))^{\delta}\,]^{1}\ /\ \square \qquad (1)$$

In order to find at what degree the distance between two couples be small, Fuzzy Set Small is used. These fuzzy sets are commonly represented by membership functions. It is denoted by 'μ'. By using such membership functions the membership degree is get derived. If the distance between two couples has a membership degree of one in the Fuzzy Set Small means then the distance is considered as small for sure. But if the distance between them has a membership degree of zero in the Fuzzy Set Small means then the distance is not consider as small. So here there is a kind of uncertainty is present, for the membership degrees occur between zero and one. The membership function small is denoted by 'μ_s'. If $x \leq p$ then the value of $\mu_s(x)$ is $(p - x/p)^{2}$ and for *x*>*p* the value of $\mu_s(x)$ is zero. The output of the Fuzzy First Sub-Filter for red component is given below in expression (2). It contains the weight function for red component that is w (*i*+*k*,*j*+*l*,1) at position (i+k,j+l). The weight function contains numerical number one that indicates that it belongs to red component.

$$F(i,j,1) = \frac{\sum_{k=-K}^{K} \sum_{l=-K}^{K} w(i+k,j+l,1) \times N(i+k,j+l,1)}{\sum_{k=-K}^{K} \sum_{l=-K}^{K} w(i+k,j+l,1)} \qquad (2)$$

The expression for $w(i+k,j+l,1)$, that is present in above expression (2) is given below in (3).

$$w(i+k,j+l,1) = \mu_{s1}(\gamma_{rg}(i,j,k,l)) \times \mu_{s2}(\gamma_{rb}(i,j,k,l)) \qquad (3)$$

Where $F(i,j,1)$ is the output image for the red component at position (*i*,*j*) and numerical number one indicates that it belongs to the red component. The numerical numbers two and three are used for green and blue components respectively. The above filtering method is similar for both green and blue components.

B. *FUZZY SECOND SUB-FILTER*

The window size consider here is (2L+1)×(2L+1), but it is not necessary that the value of 'L' must be equal to 'K' of the Fuzzy First Sub-Filter. To filter the current image pixel at position (i,j) the window is placed centered at (i,j). The local differences (LD) for red, green and blue components are denoted by LD_R, LD_G and LD_B. These are calculated as follows and given in expression (4).

$$LD_R(k,l) = F(i+k,j+l,1) - F(i,j,1)$$

$$LD_G(k,l) = F(i+k,j+l,2) - F(i,j,2)$$

$$LD_B(k,l) = F(i+k,j+l,3) - F(i,j,3) \qquad (4)$$

These differences are finally used to calculate the following correction term given in expression (5).

$\varepsilon(k,l) = ⅓ (LD_R(k,l) + LD_G(k,l) + LD_B(k,l))$ (5)

Where $\varepsilon(k,l)$ is the correction term for the neighboring pixel F(i+k,j+l,1), F(i+k,j+l,2) and F(i+k,j+l,3) of the processing pixel.

The output of the Fuzzy Second Sub-Filter for the red component is given below in expression (6). It is denoted as Out (i,j,1), that is it represent the output image of Fuzzy Second Sub-Filter for red component at position (i,j). The numerical number one indicates that it belongs to red component. The numerical number two and three are used to represent green and blue.

$$Out(i,j,1) = \frac{\sum_{k=-L}^{+L}\sum_{l=-L}^{+L}(F(i+k,j+l,1)+\varepsilon(k,l))}{(2L+1)^2} \qquad (6)$$

The above procedure is similar for the case of green and blue components. The correction term $\varepsilon(k,l)$ is similar for both green and blue components. Where, $(2L+1)^2$ is the size of the window that is selected for the Fuzzy Second Sub-Filter and F (i+k,j+l,1) is the output of Fuzzy First Sub-Filter for red component.

V. SIMULATION RESULTS

The performance of the proposed algorithm is tested against different color images by varying the noise intensity from 10% to 90%. The noise removal performance of the proposed algorithm is quantitatively measured by using the Peak Signal-to-Noise Ratio (PSNR) as defined in (7).

$$PSNR\ in\ dB = 10log_{10}(\,255^2/MSE\,) \qquad (7)$$

$$MSE = \frac{\sum_i\sum_j(Y(i,j)-\hat{Y}(i,j))^2}{M\times N} \qquad (8)$$

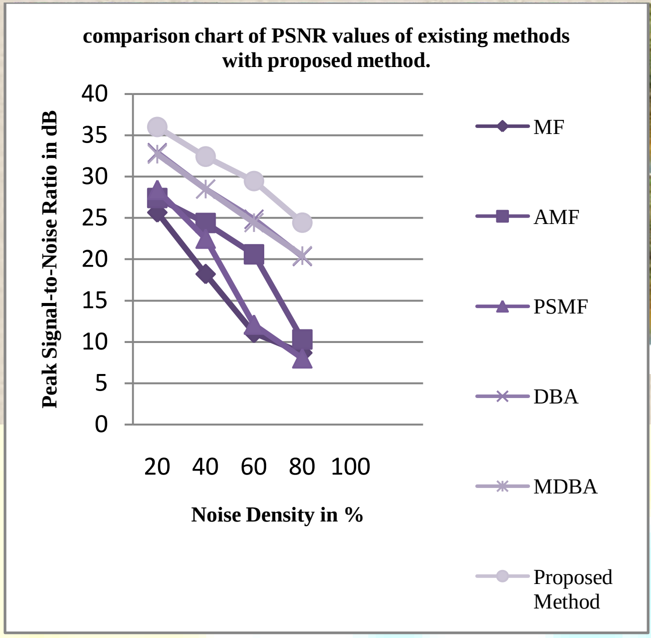
The Mean Square Error (MSE) that is needed for the calculation of PSNR is defined in expression (8). Where M×N is the size of the image, Y denotes the original image and $\hat{Y}$ represents the noisy image.

TABLE I

Comparison of PSNR values for different noise densities of Existing Methods with Proposed Method

Noise	Peak Signal-to-Noise Ratio (PSNR) in dB					
Density	MF	AMF	PSMF	DBA	MDBA	Proposed in % Method
10	26.34	28.43	30.22	36.40	36.94	39.91
20	25.66	27.40	28.39	32.90	32.69	35.98
30	21.86	26.11	25.52	30.15	30.41	34.99
40	18.21	24.40	22.49	28.49	28.49	32.40
50	15.04	23.36	19.13	26.41	26.52	30.93
60	11.08	20.60	12.10	24.83	24.41	29.43
70	9.93	15.25	9.84	22.64	22.47	25.97
80	8.68	10.31	8.02	20.32	20.44	24.41
90	6.65	7.93	6.57	17.14	17.56	22.98

The PSNR values of the proposed method for different noise densities are get compared with the existing methods namely Median Filter (MF), Adaptive Median Filter (AMF), Progressive Switching Median Filter (PSMF), Decision Based Algorithm (DBA) and Modified Decision Based Algorithm (MDBA) and it is given in Table I. The graphical representation of the comparison of PSNR values of existing methods with the proposed method which is given in Table I is shown in Fig.7. From the graph it is clear that the proposed Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) and Fuzzy Noise Reduction Method (FNRM) give excellent Peak Signal-to-Noise Ratio (PSNR) than the existing algorithms.



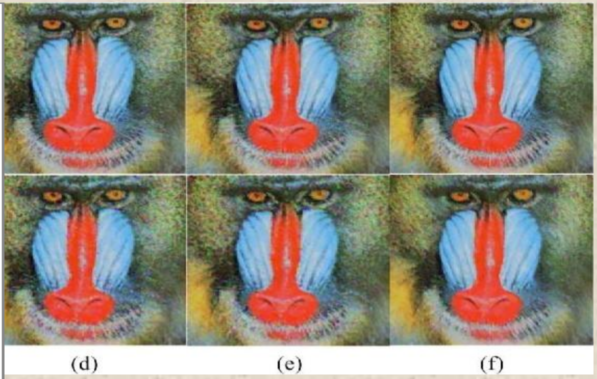


Fig.7. comparison graph of PSNR values of existing algorithms with proposed method.

The noise density is varied from 10% to 90% and the corresponding PSNR values of existing methods and proposed methods are clearly shown in Table I. From this it is clear that the performance of the proposed method is better than the performance of the existing methods.

The result of the proposed algorithm against existing algorithm is shown in Fig.8. The image used here is the image of a Baboon. The type of the image is color and the noise densities consider here is 70% and 80%. Subsequent columns represent the processed images of MF, AMF, PSMF, DBA, MDBA and the proposed algorithm. From the Fig.8, it is possible to observe that the quality of the restored image using proposed method is better than the quality of the restored image using existing algorithms. The quality is calculated using the Peak Signal-to-Noise Ratio (PSNR) that is already defined in expression (7). The proposed algorithm gives PSNR value of 39.91 dB for 10% of noise density. For 90% of noise density it produces the value of PSNR as 22.98 dB. These values are higher than the values produced by existing algorithms. From the Table-I, Fig.7 and Fig.8, it is very clear that the proposed method performance is excellent than the existing methods.

Fig. 8. Results of different algorithms for color Baboon image. (a) Output of MF. (b) Output of AMF. (c) Output of PSMF. (d) Output of DBA. (e) Output of MDBA. (f) Output of Proposed Method. Rows 1 and 2 show processed results of various algorithms for color image corrupted by 70% and 80% noise densities, respectively.

VI. **CONCLUSION:**

In this paper, Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) followed by Fuzzy Noise Reduction Method (FNRM) is proposed which gives better performance in comparison with Median Filter (MF), Adaptive Median Filter (AMF), Progressive Switching Median Filter (PSMF), Decision Based Algorithm (DBA), Modified Decision Based Algorithm (MDBF) and other existing noise removal algorithms in terms of PSNR. The performance of these algorithms has been tested at low, medium and high noise densities for color images. Even at high noise density levels the Proposed method gives better results in comparison with other existing algorithms. Both visual and quantitative results are demonstrated. The proposed Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) and Fuzzy Noise Reduction Methods (FNRM) together is effective for salt and pepper noise removal in color images even at high noise densities.

REFERENCES:

- J.Astola and P.Kuosmaneen," Fundamentals of Non-linear Digital Filtering ". Boca Raton, FL: CRC, 1997.
- H. Hwang and R. A. Hadded, "Adaptive median filter: New algorithms and results," *IEEE Trans. Image Process.*, vol. 4, no. 4, pp.499–502, Apr.1995.
- S.Zhang and M.A.Karim, "Anew impulse detector for switching Median filters", IEEE signal process. Lett, vol.9, no.11, pp. 360-363, Nov.2002.
- 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.

- K. S. Srinivasan and D. Ebenezer, “A new fast and efficient decision based algorithm for removal of high density impulse noise,” *IEEE Signal Process. Lett.*, vol. 14, no. 3, pp. 189–192, Mar. 2007.
- K. Aiswarya, V. Jayaraj, and D. Ebenezer, “A new and efficient algorithm for the removal of high density salt and pepper noise in images and videos,” in *Second Int. Conf. Computer Modeling and Simulation*, 2010, pp. 409–413.
- 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.