ISSN (Online): 2349-7084
GLOBAL IMPACT FACTOR 0.238
DIIF 0.876

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

Removing Salt and Pepper Noise using Modified Decision- Based Approach with Boundary Discrimination

Aaditya Sharma, R. K .Pateriya
Computer Science &Engineering Department
Maulana Azad National Institute of Technology, Bhopal, India
Manit.aaditya@gmail.com, rkpateriya@gmail.com

Abstract: This article proposes a two-stage median filter for the removal of salt and pepper noise. In first stage, to identify the noisy pixels modified approach of BDND algorithm is used, and in second stage, before restoration of noisy pixels once again it is confirmed that whether current pixel is noisy or not using modified decision-based algorithm. Experimentation results show that our proposed algorithm performs better than several existing algorithms in terms of subjective quality of image as well as objective quality. Extensive experimentation shows that our proposed algorithm performs better than Standard Median Filter (MF), Adaptive Median Filter (AMF), Decision-Based Algorithm(DBA) , Modified Decision-Based Algorithm (MDBA), and Modified Decision-Based Unsymmetric Trimmed Median Filter (MDBUTMF). To compare the performance of our proposed algorithm, several matrices such as Peak Signal to Noise Ratio (PSNR), Mean Absolute Error (MAE), and Structural Similarity Index (SSIM) have been used.

Index: Noise models, Decision based algorithm, BDND, PSNR, SSIM, MAE

---*** --- ----------

1. INTRODUCTION

Digital images get corrupted by impulse noise either during acquisition or during transmission. Due to impulse noise, signal information gets corrupted and some irrelevant information gets added to original signal. Impulse noise can be categorized in two types, random valued impulse noise, and salt and pepper impulse noise. Salt and pepper impulse noise corrupt the pixel values in such a way that a pixel will hold either low intensity or high intensity value. . Several nonlinear filtering approaches have been proposed to remove salt and pepper noise [1, 2, 3, 4, 5, 6, 7]. Median filter (MF) is one such approach which works very well for less noisy images, but as the noise density gets increased, MF loses image details due to which images gets blurred [8, 9, 10, 12].When noise density cross 50% then standard median filter loses much details present in the image. Adaptive median filter (AMF) [11] performs well for less noisy image; however, as noise density increases then window size has to be increased due to which image gets blurred.

LetY_{ij} be the gray level of original pixel and X_{ij} be the gray level of noisy pixel and p is the noise ratio and Z_{ij} be the gray level $\in [n_{min}, n_{max}]$, where for salt and pepper noise n_{min} would be 0 and n_{max} would be 255. So noise model for salt and pepper noise can be defined as in equation 1.0 [11, 13].

$$X_{ij} = \begin{cases} Z_{ij} & with\ probability\ \ p \\ Y_{ij} & with\ probability\ 1-p \end{cases} \quad ...1.0$$

In switching median filter, decision is taken based on threshold; biggest drawback of this approach is that suggesting a suitable threshold is difficult [14, 15]. Switching median filter does not take into account the thin details present in image

Because the image quality degrades. To overcome these problems decision-based algorithm (DBA) was proposed [16]. This algorithm uses a window of 3*3 and checks whether pixel under consideration is having value 0 or 255 only then processing is done, otherwise it processes next pixel. When noise density is high then most of the pixels in the window will be either 0 or 255 then median value will be either 0 or 255, so noise would spread rather than being reduced.

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

At high noise density streaking effect is produced [17]. To remove the streaking effect at high noise density decision-based unsymmetrical trimmed median filter was proposed [21, 22]. When noise density is high then selected window contains all 255 or 0's or both, therefore, trimmed median value cannot be obtained. Therefore, at noise density > 80%, this filter does not produce better result. Later, to remove the above drawback at high noise density, Modified Decision-Based Unsymmetric Trimmed Median Filter (MDBUTMF) was proposed [18]. When the selected window contains all the pixels as either 0 or 255 then instead of taking mean if we count the number of pixels with low intensity and number of pixels with high intensity and later replace the noisy pixel by bigger of them then better result can be obtained. Our proposed algorithm uses this idea along with boundary selection approach of boundary discriminative noise detection (BDND) [15] algorithm, and it gives better peak signal to noise ratio (PSNR), structural similarity (SSIM) index, and mean absolute error (MAE) value than the many existing algorithms. The organization of this article is as follows: section 2 gives a brief introduction of noise removal algorithms, section 3 contains the proposed algorithm, section 4 contains experimental results and comparison analysis, and section 5 concludes the article.

2. LITERATURE REVIEW

In MDBUTMF [18] algorithm, they propose to remove salt and pepper noise from digital images. This proposed algorithm, first checks whether the pixel is noisy or not, and if the pixel value lies between minimum and maximum gray level, then the pixel is identified as correct pixel and therefore no processing will be done for such pixels. The main idea of algorithm is to check whether the pixel value is 0 or 255, based on which the pixel will be identified as either noisy or noiseless. Pixels which are found to be noiseless will be kept intact and remaining corrupted pixels then undergoes processing. If all the pixels in current window are either 0 or 255 then the mean of window will be calculated to restore the noisy pixel; however, if some of the pixels are having value greater than 0 and less than 255, then all the 0's and 255's are trimmed from the current window, and from the remaining pixels median is calculated to restore the noisy pixel.

BDND algorithm works in two iterations in which the invocation of the second iteration is conditional. Initially, a window of 21*21 is chosen to check whether the pixel under consideration is noisy or not. If pixel under processing is found to be corrupt then 2nd iteration will be called to assure that indeed pixel is corrupted, using window of size 3*3, a more confined locality. First, the pixels in window size 21*21 are sorted and stored in a vector V_o and median is calculated, and later a difference vector V_d is calculated from the sorted vector V_o. For the pixel intensities between 0 and median, in V_o, maximum intensity difference in V_d is calculated of the same range and marks its corresponding pixel in V_o as boundary b_1. Similarly, boundary b_2 is identified for pixel valued between median and 255. So in this manner three clusters are created: if the pixel under consideration belongs to middle cluster, then it is classified as uncorrupted pixel and no processing is needed for that and classification process stops here. Otherwise 2nd iteration will be invoked and a window of size 3*3 is chosen centered around the current pixel and algorithm is repeated. Now, for all the corrupted pixels standard median filter with adaptive window size is applied. Window size is kept 3*3 for 20% noise, 5*5 for 21–40% noise, and 7*7 for greater than 40% noise level.

3. PROPOSED ALGORITHM

Our proposed algorithm works in two phases: in the first phase, it identifies whether the pixel under consideration is noisy or not, and in 2nd phase noisy pixel is restored using the modified decision-based algorithm (MDBA). Initially, a window of size 17*17 is chosen and pixels in that window are classified into three clusters: pixels which are of low intensity placed in lower cluster, high intensity pixels are placed in upper cluster, and remaining pixels whose intensity is neither low nor high are classified in middle cluster. If current pixel exists in the middle cluster then it is identified as uncorrupted pixel and its value remains unchanged, otherwise a new window of size 3*3 is created to restore the noisy pixel. Detailed steps of algorithm are given below.

ISSN (Online): 2349-7084

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

ALGORITHM

FIRST PHASE

Step1. Select a window of size 17*17, which is centered at current pixel.

Step 2. Sort all the pixels in current window and obtain a sorted vector of pixels named S_o and calculate median of vector S_o named M.

Step 3. Calculate the intensity difference between all the adjacent pixels in sorted vector S_o and obtained a new difference vector S_d.

Step 4. Pixels having intensity between 0 and median (M) in sorted vector S_o compute the greatest intensity difference in difference vector S_d of the similar range and its corresponding pixel in sorted vector S_o is marked as boundary B_1.

*Step 5.*Pixels having intensity between median (M) and 255 in sorted vector S_o compute the maximum intensity difference in difference vector S_d of similar range and its corresponding pixel in vector $S_o$is marked as boundaryB_2.

Step 6. Pixels having intensity greater than boundary B_1 but less than boundary B_2 are classified as noise-free pixels and their pixel value remains unchanged, however pixels having intensity less than boundary B_1 and intensity greater than boundary B_2 are classified in lower intensity cluster and upper intensity cluster respectively.

SECOND PHASE

Step 1. Pixel which is classified as corrupted in first phase, a new window of size 3*3 is created centered at current pixel. Suppose this pixel is represented by$pixel_{ij}$.

Step 2.$if\,(\,0 < pixel_{ij} < 255\,)$Then finally that is treated as noisy-free pixel, otherwise if $pixel_{ij} = 0\ or\ pixel_{ij} = 255$ then further processing is done.

Step 3. If all the pixels in current window contain 0 and 255 then count how many 0 as well as 255 are present. After counting compare them and if number of 0's are more than 255's than noisy pixel is restored by 0 otherwise it is restored by 255.

*Step 4.*If current window contains some other intensity values also along with 0's and 255's then all 0's and 255's are trimmed from the window and then from the remaining pixels weighted median is calculated to restore the noisy pixel.

Step 5. Execute the above steps to process the entire image. Execute the complete algorithm for N_{max} iterations. In general, N_{max} varies from 3 to 5 for highly corrupted images however at low noise density N_{max} varies from 1 to 2. .

Let us understand the above algorithm by taking an example, for convenience we take a window of size 7*7 centered at current pixel.

$$\begin{pmatrix} 255 & 255 & 50 & 255 & 50 & 255 & 40 \\ 60 & 100 & 255 & 255 & 0 & 0 & 0 \\ 0 & 0 & 30 & 90 & 255 & 120 & 255 \\ 255 & 0 & 255 & 210 & 255 & 0 & 0 \\ 0 & 0 & 56 & 78 & 100 & 255 & 255 \\ 80 & 255 & 37 & 49 & 255 & 0 & 255 \\ 35 & 255 & 0 & 0 & 0 & 255 & 47 \end{pmatrix}$$

Pixels in the window are sorted in ascending order and vector S_o is calculated and then median (M) 60 is picked.

S_o =[0,0,0,0,0,0,0,0,0,0,0,0,0,0,30,35,37,40,47,49,50,50,56, **60**,78,80,90,100,100,120,210,255,255,255,255,255,255,255, 255,255,255,255,255,255,255,255,255,255,255].

Difference vector of pixel intensities is calculated for all adjacent pixels in sorted vector $S_o, S_d =$ [0,0,0,0,0,0,0,0,0,0,0,0,0,30,5,2,3,7,2,2,1,0,6,4,18,2,10,10,0 ,20,90,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0].

Pixels with intensity between 0 and median (M) in sorted vector S_o the maximum intensity difference is 30 in difference vector S_d which is between 0 and 30 in

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

vector S_o so $B_1 = 0$. Similarly pixel intensity between median (M) and 255 the maximum intensity difference is 90 which is between 120 and 210 in sorted vector S_o so $B_2 = 120$.Now based on B_1 and B_2 three clusters are made and pixels are classified to them. Lower intensity cluster is [0,0,0,0,0,0,0,0,0,0,0,0,0,0], higher intensity cluster is

[210,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255] and middle intensity cluster is [30,35,37,40,47,49,50,50,56,60,78,80,90,100,100,120].

Now a window of size 3*3 is created centered at current pixel.

$$\begin{pmatrix} 30 & 90 & 255 \\ 255 & 210 & 255 \\ 56 & 78 & 100 \end{pmatrix}$$

In current window centered pixel intensity is neither 0 nor 255 so it is un-corrupted pixel and window slides over the image. Suppose during execution at some point of time window contains these elements.

$$\begin{pmatrix} 0 & 0 & 30 \\ 255 & 0 & 255 \\ 0 & 0 & 56 \end{pmatrix}$$

Now $pixel_{ij} = 0$ so this pixel is corrupted, window contains some intensity values including 0 and 255 so remove the intensity values 0 and 255 and from remaining pixels intensities calculate weighted median and replace noisy pixel by weighted median. While calculating weighted median weights are assigned to pixels in such a way that pixels which are close to centered pixel is given more weight and pixels which are far given lesser weight. To calculate the distance either standard Euclidean distance or generalized Minkowski distance can be used.

4. EXPERIMENTAL RESULTS

Our proposed algorithm is tested using various images of size 512*512 such as Lena, Baboon, Cameraman, Peppers, and so on. For the performance evaluation, several matrices such as PSNR, SSIM index, and MAE have been used at low as well as at high noise density and proposed algorithm outperform the several other existing algorithms. PSNR, MAE, and SSIM index calculation is given in equations 1.1, 1.3, and 1.4, respectively.

$$PSNR = 10 * \log_{10}\left(\frac{255^2}{MSE}\right) \quad ...1.1$$

Where, MSE refers to mean square error which is calculated as given in equation 1.1.

$$MSE = \frac{\sum_i \sum_j (Y_{ij} - H_{ij})^2}{M*N} \quad ...1.2$$

Where, Y_{ij} original image and H_{ij} is the de-noised image, M*N is the size of image.

$$MAE = \frac{\sum_i \sum_j (Y_{ij} - H_{ij})}{M*N} \quad ...1.3$$

$$SSIM(Y, H) = \frac{(2*\mu_Y*\mu_H + C_1)*(2*\sigma_{YH} + C_2)}{(\mu_Y{}^2 + \mu_H{}^2 + C_1)*(\sigma_Y{}^2 + \sigma_H{}^2 + C_2)} \quad ...1.4$$

Where, Y and H are the original and restored image respectively of same size N*N, and $i$is counter variable.

$$\mu_Y = \sum_{i=1}^{N} Y_i \quad ...1.5$$

$$\mu_H = \sum_{i=1}^{N} H_i \quad ...1.6$$

$$\sigma_Y{}^2 = \frac{\sum_{i=1}^{N}(Y_i - \mu_Y)^2}{N-1} \quad ... \quad 1.7$$

$$\sigma_H{}^2 = \frac{\sum_{i=1}^{N}(H_i - \mu_H)^2}{N-1} \quad .... \quad 1.8$$

$$\sigma_{YH} = \frac{\sum_{i=1}^{N}(Y_i - \mu_Y)(H_i - \mu_H)}{N-1} \quad .... \quad 1.9$$

C_1 *and* C_2 Are constants.

SSIM index has been used to evaluate the image quality in order to predict the human preference [12, 20].Human visual system (HVS is highly adapted to perceive the structure of objects from the scene, so SSIM index provides the measure of good image quality Maximum image similarity between two images can be 1 if and only if both

Table 1 PSNR comparison for Lena Image

Noise (%)	MF	AMF	DBA	MDBA	MDBUTMF	PA
10	30.99	28.43	36.40	36.94	37.91	52.37
20	28.28	27.40	32.90	32.69	34.78	49.13
30	23.51	26.10	30.15	30.41	32.29	47.04
40	18.81	24.40	28.49	28.49	30.32	45.41
50	15.12	23.40	26.41	26.52	28.18	43.71
60	12.19	20.69	24.83	24.41	26.43	41.51
70	9.79	14.89	22.64	22.47	24.30	38.56
80	7.88	11.76	20.32	20.44	21.70	34.89
90	6.36	9.52	17.14	17.56	18.40	31.01

Table 2 MAE comparison for Lena Image

Noise (%)	MF	AMF	DBA	MDBA	MDBUTMF	PA
10	4.22	4.99	2.18	1.96	1.72	0.53
20	4.74	5.53	3.05	2.87	2.68	1.08
30	6.20	5.85	3.72	3.36	3.27	1.69
40	10.01	6.1	4.40	4.12	4.06	2.37
50	17.30	6.49	5.19	4.91	4.83	3.34
60	29.43	6.71	6.20	5.98	5.85	4.58
70	47.42	7.37	7.78	7.46	7.29	5.67
80	70.74	8.59	11.01	10.82	10.77	6.61
90	91.21	11.5	27.89	27.56	27.41	8.15

Table 3 SSIM index comparison for Lena Image

Noise (%)	MF	DBA	MDBUTMF	PA
10	0.965171	0.988422	0.995201	0.995659
20	0.920106	0.96227	0.989430	0.990859
30	0.770225	0.884744	0.982537	0.984003
40	0.513049	0.721124	0.974825	0.970964
50	0.303169	0.514991	0.964842	0.939473
60	0.173192	0.337029	0.952363	0.926669
70	0.094475	0.209600	0.933685	0.912371
80	0.048081	0.122030	0.898941	0.890812
90	0.02108	0.059019	0.809985	0.843219

Table 4 SSIM index comparison for Peppers Image

Noise (%)	MF	DBA	MDBUTMF	PA
10	0.949893	0.962796	0.992541	0.993752
20	0.908894	0.905573	0.98466	0.98677
30	0.774421	0.819402	0.975928	0.978264
40	0.524926	0.671423	0.966759	0.964836
50	0.314949	0.491558	0.956919	0.946993
60	0.187795	0.329784	0.944586	0.94044
70	0.106933	0.211472	0.925431	0.928655
80	0.059008	0.127649	0.890185	0.910186
90	0.027221	0.063124	0.788459	0.867488

Table 5 PSNR comparison at 70% noise density for various test images

Image	MF	AMF	DBA	MDBA	MDBUTMF	PA
Lena	9.79	14.89	22.64	22.47	24.30	38.56
Cameraman	9.74	13.93	20.84	19.97	22.52	38.85
Baboon	**10.2**	**14.86**	**22.35**	**20.54**	**23.80**	**37.01**

Images are same. Tables 1–4 are shown below which contains the PSNR, MAE, and SSIM index value for test images using various algorithms SSIM index compares the local patterns of pixel intensity that have been normalized to average intensity [12]. PSNR, as mentioned in equation 1.1, at different noise density for Lena image have been shown in Table 1, Mean absolute error (MAE) as in equation 1.3 at different noise densities for Lena Image have been mentioned in Table 2, SSIMmentioned in equation 1.4 at varying noise level levels for Lena Image is shown in Table 3, and SSIM index for

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

Fig 1.a) Cameraman image corrupted with 30% noise density, (b) Output of PA, (c) Output of DBA, (d) Output of MF,(e) Output of MDBUTMF, (f) Original Image

ISSN (Online): 2349-7084

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

Peppers image at different noise densities have been mentioned in Table 4. From the obtained results, we can conclude that at lower noise density several algorithms perform well for different matrices, but as noise density increases above 50%, then many algorithms do not produce good result.. Our proposed algorithm produced better results at low and high noise density.. It also works very well for color images. Several colored images with different noise densities were tested on differing parameters. Later, we have shown the color version of restored images at varying noise densities using different algorithms From the Table 2, MAE generated is more on high noise density by several algorithms; even MDBUTMF produced high MAE than AMF at noise density 80-90%. MDBA, MDBUTMF produce relatively equal MAE butMDBUTMF produce less MAE than MDBA; however, our proposed algorithm produced least MAE at low as well as high noise density. . The efficiency of our proposed algorithm lies in the fact that number of iterations required to execute the algorithm for some noise density level. By performing experimentation, we found that when noise density is less than 30% then one round of processing is enough, but as the density increases from 30 to 50% then at the most three iterations are required. Further increasing the noise density up to 70% just 3 iterations are enough but at 80% noise density, four iterations are required whereas when the noise density goes above90%, five iterations of processing is required. . In each iteration, the number of corrupted pixels decreased, so more

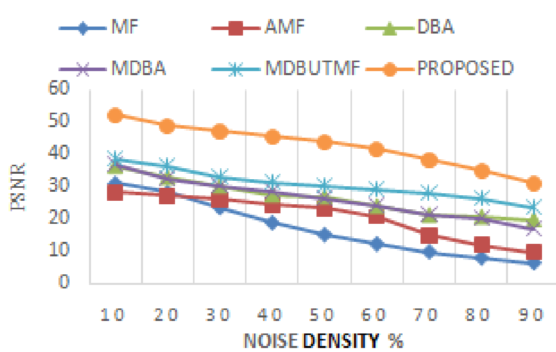

***Fig. 2.**PSNR comparison for Lena Image at varying noise density*

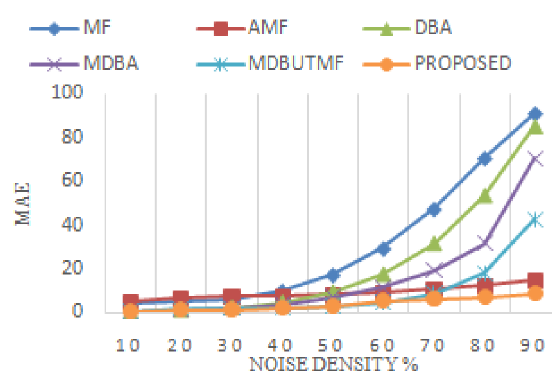

***Fig. 3**. MAE comparison for Lena Image at varying noise density*

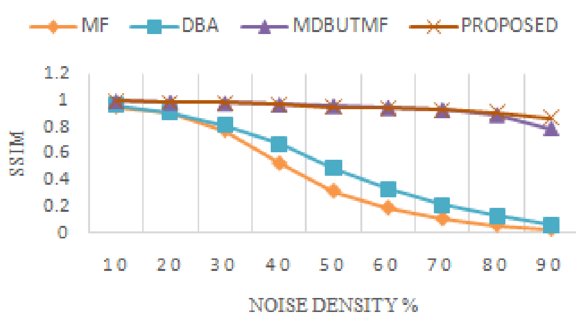

***Fig. 4**. SSIM index comparison for Lena Image at varying noise density*

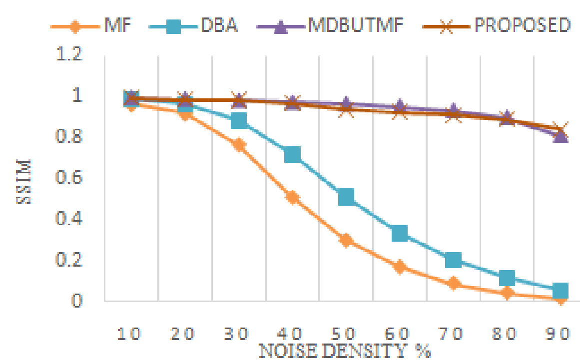

***Fig. 5**. SSIM index comparison for Peppers Image at varying noise density*

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

And more noisy pixels can be restored. Fig. 1 shows the PSNR value changes over varying noise density for various algorithms. Fig. 2 shows that as the noise density cross 50% then several algorithms generated less PSNR; however, our proposed algorithm produced highest PSNR.. Several test images that are corrupted at 70% noise density have been compared with our proposed algorithm in terms of PSNR given in Table 5. . It shows that our proposed algorithm produced better PSNR, and it does not depend on the nature of image In Figs.4 and 5, we have not used some algorithms which have been used in PSNR and MAE calculation because the produced result of those intermediary algorithms were not so much important, so we decided to select just a few of them so that Improvements of algorithms can be seen easily. Fig.1. is showing cameraman image which is corrupted at 30% noise density and then processed by various algorithms. We have shown some color restored images also for the sake of completeness. Peppers image corrupted with 90% noise density and corresponding restored images using various algorithms have been given in Fig.7. Fig. 6 contains Baboon image corrupted with 60% noise density and restored images various algorithms as MF, DBA, MDBUTMF, and proposed algorithm (PA).

5. CONCLUSIONS

This article proposes an efficient algorithm for removal of salt and pepper noise. Our proposed algorithm has been tested on various gray scale and color images and compared with several algorithms such as MF, AMF, DBA, MDBA, and MDBUTMF using PSNR, MAE, and SSIM. Tables 1–5 show performance of our proposed algorithm along with several other algorithms. Figures 5–7 that are shown at the end of this article shows output of various algorithms considered for noise removal including our proposed algorithm. This algorithm performs better on gray scale and color images at low and high noise density. Restored image obtained by proposed algorithm is having high sharpness. SSIM index is calculated for several images, and it is found that at high noise density proposed algorithm gives highest SSIM index among the other tested algorithms. Our proposed algorithm also provides better image subjective quality compared with other tested algorithms for salt and pepper noise. Also, it produces least MAE compared with other algorithms. Therefore, based on our

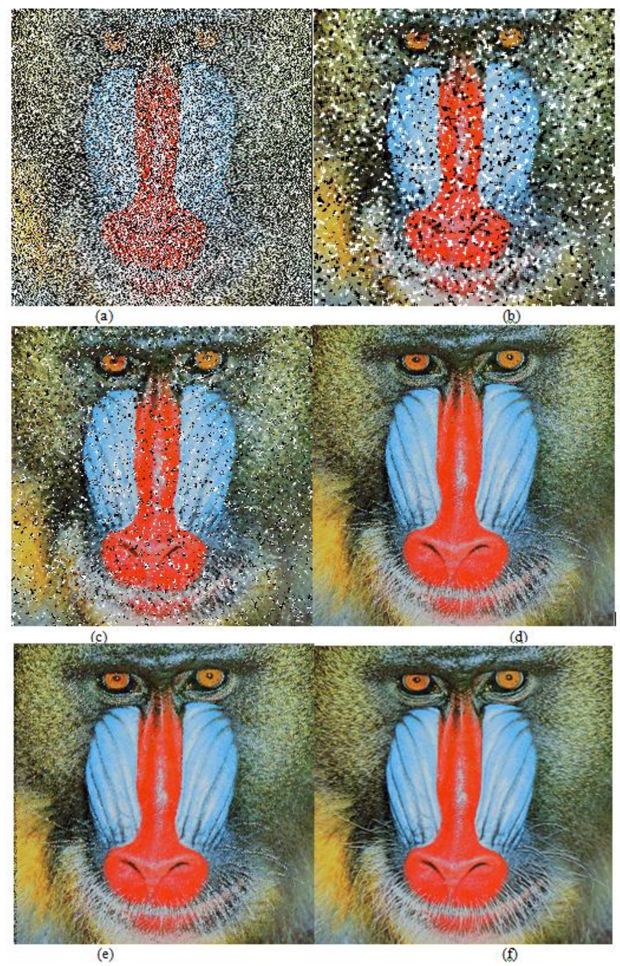

Fig. 6.(a) Baboon image corrupted with 60 % noise density,(b) restored image using MF,(c) restored image using DBA, (d) processed image using MDBUTMF, (e) restored image using PA, (f) original image

ISSN (Online): 2349-7084

INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

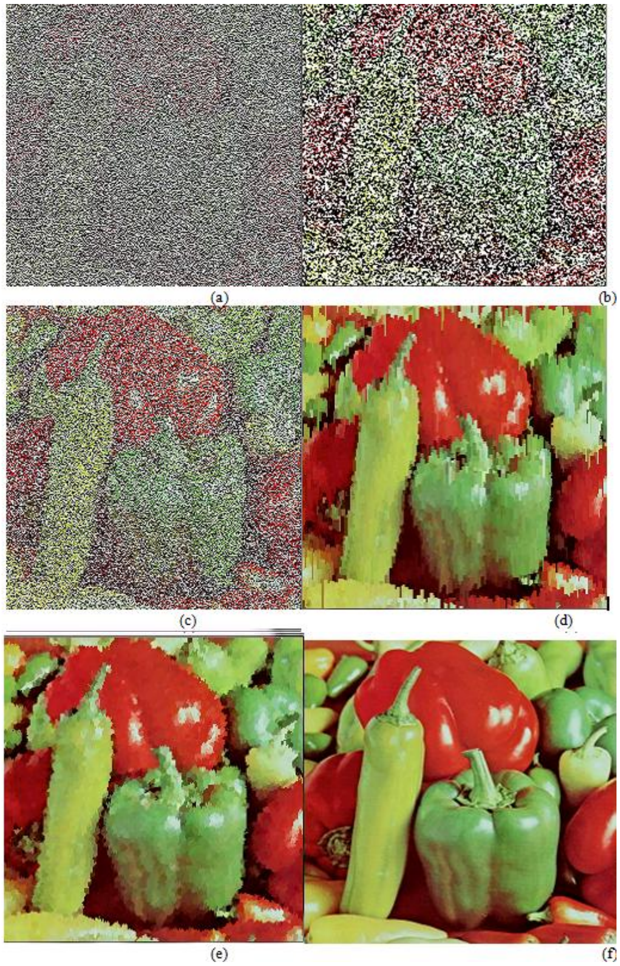

Fig. 7.(a) Peppers image corrupted with 90 % noise density, (b) Output of MF, (c) Output of DBA, (d) Output of MDBUTMF, (e) Output of PA, (f) Original Image

Results, we can say that proposed algorithm (PA) is an effective method for fixed valued impulse noise.

6. REFERENCES

[1] S. Indu and C. Ramesh," A noise fading technique for images highly corrupted with impulse noise," International Conference on Computing: Theory and Applications, ICCTA'07, Kolkata, India, pp.627-632.

[2] A. K. Tripathi, U. Ghanekar and S. Mukhopadhyay," Switching median filter: advanced boundary discriminative noise detection algorithm,"IET Image Process., vol.5, Issue 7, pp.598-610, 2011.

[3] Z. Wang and D. Zhang," Progressive switching median filter for removal of impulse noise from highly corrupted images," IEEE Trans. Circuits Syst.II, vol.46, no.1, pp.78-80, 1999.

[4] W. Luo, "An efficient algorithm for the removal of impulse noise from corrupted images," IEEE Trans. On Consumer Electronics, vol. 61, Issue 8, pp.551-553, Sept.2007.

[5] P.Y. Chen, C.-Y. Lien and C.-Y. Tsai," An effective impulse noise detector of switching median filter using min-max working window", Proceedings of IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing ,Splendor Kaohsiung Tiwan ,Nov. 2007,pp 1-4.

[6] A. S. Awad and H. Man," High Performance detection filter for impulse noise removal in images," Electronic Letters, vol.44, no.3, Jan.2008.

[7] F. Russo and G. Ramponi," A fuzzy filter for image corrupted by impulse noise," IEEE Signal Process. Lett. vol.3, no.6, pp.168-170, 1996.

[8]J. Astola and P. Kuosmaneen, Fundamentals of Nonlinear Digital Filtering. Boca Raton, FL: CRC, 1997.

[9]] T. Chen, H. R. Wu, "Adaptive impulse detection using centered –weighted median filters," IEEE Signal Processing, Letters,8 (2001)1-3.

[10]] H. –L. Eng., K. –K. Ma, "Noise adaptive soft switching median filter," IEEE Trans .Image Process.10 (2001) 242-251.

[11] 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.

[12] Zhou Wang, Alan C. Bovik," Image quality assessment: from error visibility to structural similarity," IEEE Trans. On Image Processing, vol. 13, no.4, April.2004.

[13] R. H. Chan, C.W.Ho, M. Nikolova, Salt –and-pepper noise removal by median type noise detectors and detail –preserving regularization, IEEE Trans. Image Process. 14(2005)1479-1485.

[14] 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.

[15] 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, June.2006.

[16] 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.

[17] V. Jayraj 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.

[18] S. Esakkirajan, T. Veerakumar, Adabala N. Subramanian, and C. H. PremChand, "Removal of high density salt and pepper noise through modified decision based unsymmetrical trimmed median filter," IEEE Signal Processing Letters,vol.18,no.5,May2011.

[19] Y. H. Lee, S. A. Kassam, "Generalized median filtering and related nonlinear filtering techniques," IEEE Trans. Acoust. Speech Signal Process. 33 (1985) 672-683.

ISSN (Online): 2349-7084
INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS
VOLUME 2, ISSUE 7, JULY 2015, PP 411-420

[20] W. S. Geisler, and M. S. Banks, “Visual performance,” in Hand-book of optics (M. Bass, ed.), McGraw-Hill, 1995.

[21] K .Aiswarya, V. Jayraj, 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.

[22] Aaditya Sharma, R. K. Pateriya, “A Survey on various Image Filtering Approaches to remove Impulse Noise " INTERNATIONAL JOURNAL OF COMPUTER ENGINEERING IN RESEARCH TRENDS VOLUME 2, ISSUE 5, MAY 2015, PP 322-328

Aaditya Sharma completed B.TECH (Hons.) in Computer Science & Engineering from Gautam Buddh Technical University Luck now, India in 2012. He is pursuing M.TECH in Advance Computing from Maulana Azad National Institute of Technology, Bhopal, India. His area of research includes image processing and parallel algorithms.

Dr. R. K. Pateriya, an Associate professor in the Department of Computer Science & Engineering, at Maulana Azad National Institute of Technology, Bhopal, India. He is a member of the IEEE. His current research interests includes image processing, cloud and Distributed computing, E-commerce, security etc. He has authored and published over 100 research papers in national and international journals.