International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-2 Issue-3, February 2013

Removal of High Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Adaptive Median Filter

Vivek Chandra, Sagar Deokar, Siddhant Badhe, Rajesh Yawle

Abstract— An algorithm based on adaptive and unsymmetric trimmed median filter is proposed in this paper. This algorithm is proposed for restoration of gray-scale as well as color images which are highly corrupted by salt and pepper noise. The proposed algorithm replaces the noisy pixel by a value which is either a mean or a median of all other non-noisy pixels in the selected window. The proposed algorithm also adaptively controls the window size depending on the relative amount of noisy pixels compared to non-noisy pixels in the selected window. This proposed algorithm substantially outperforms all existing median-based filters, in terms of suppressing salt and pepper noise while preserving image details. The proposed algorithm is tested against different gray-scale and color images giving better Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF) at different noise densities.

Index Terms— adaptive, median filter, restoration, salt and pepper noise.

I. INTRODUCTION

Salt & Pepper Noise in the images is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel [1]. For images corrupted by salt-and-pepper noise, noisy pixels can take only the maximum or the minimum values. There are many works on the restoration of images corrupted by salt & pepper noise. The median filter was once the most popular nonlinear filter for removing salt & pepper noise because of its good denoising power and computational efficiency [2]. However, when the noise level is over 50%, some details and edges of the original image are smeared by the filter.

Different remedies of the median filter have been proposed, e.g., the adaptive median filter [3], the switching median filter [4], Decision Based Algorithm (DBA) [5], Modified Decision Based Unsymmetric Trimmed Median Filter (MDBUTMF) [6]. These filters first identify possible noisy pixels and then replace them by using the median filter, while leaving all other pixels unchanged. These filters are good at detecting and removing noise even at a high noise level. Their main drawback is that the noisy pixels are replaced by some median value in their vicinity without taking into account local features such as the possible presence of edges. Hence, details and edges are not recovered satisfactorily, especially when the noise level is high.

In this paper, we propose a powerful algorithm which combines the method proposed in [6] with the adaptive median filter [3]. The proposed Modified Decision Based Unsymmetric Trimmed Adaptive Median Filter (MDBUTAMF) algorithm processes the corrupted image pixel by pixel. If the processing pixel value is 0 or 255, it is processed or else it is left unchanged. Since no changes are made to the non-noisy pixels and window size is changed dynamically depending on the noise density, the performance of our combined approach is much better than that of either one of the methods. Salt and pepper noise with noise density as high as 90% can be cleaned quite efficiently.

The outline of the paper is as follows. A brief introduction of unsymmetric trimmed adaptive median filter is given in Section II. Section III describes about the proposed algorithm. The detailed description of the proposed algorithm with an example is presented in Section IV. Simulation results along with comparison table and comparison graph are presented in Section V. Finally conclusions are drawn in Section VI.

II. UNSYMMETRIC TRIMMED ADAPTIVE MEDIAN FILTER

The idea behind a trimmed adaptive filter is to reject the noisy pixel from dynamically selected window (3x3 or 5x5). Alpha Trimmed Mean Filtering (ATMF) is a symmetrical filter where the trimming is symmetric at either end. In this procedure, even the uncorrupted pixels are also trimmed. This leads to loss of image details and blurring of the image. In Unsymmetric Trimmed Median Filter (UTMF), window size is fixed (3x3). At high noise densities, when 3/4th or more pixels in the selected 3x3 window are noisy, at most 2 pixels will be available for reference which are not enough to estimate the output pixel value. In order to overcome this drawback, an Unsymmetric Trimmed Adaptive Median Filter (UTAMF) is proposed.

Manuscript published on 28 February 2013.
* Correspondence Author (s)
Vivek Chandra, Department of Electronics and Telecommunication, Sinhgad Academy of Engineering, University of Pune, Pune, India.
Sagar Deokar, Department of Electronics and Telecommunication, Sinhgad Academy of Engineering, University of Pune, Pune, India.
Siddhant Badhe, Department of Electronics and Telecommunication, Sinhgad Academy of Engineering, University of Pune, Pune, India.
Rajesh Yawle, Department of Electronics and Telecommunication, Sinhgad Academy of Engineering, University of Pune, Pune, India.

In this UTAMF, window size is selected dynamically depending on the total number of noisy pixels in the selected window. If 3/4th or more pixels in the selected 3x3 window are noisy, then window of size 5x5 is selected. The selected 3x3 or 5x5 window elements are arranged in either increasing or decreasing order. Then the pixel values 0's and 255's in the window (i.e., the pixel values responsible for the salt and pepper noise) are removed from the window. Then the median value of the remaining pixels is taken. This median value is used to replace the noisy pixel. This filter is called trimmed adaptive median filter because the pixel values 0's and 255's are removed from the dynamically selected window.

III. PROPOSED ALGORITHM

The proposed Modified Decision Based Unsymmetrical Trimmed Adaptive Median Filter (MDBUTAMF) algorithm processes the corrupted images by first detecting the salt and pepper noise. The processing pixel is checked whether it is noisy or noise free. If the processing pixel lies between maximum and minimum gray level values, then it is noise-free pixel and it is left unchanged. If the processing pixel takes the maximum or minimum gray level, then it is noisy pixel which is processed by MDBUTAMF.

While processing image, always noisy image is taken as reference for calculation of mean or median and processing pixel is replaced by mean or median in output image. This is explained as follows.

Let 'a' be the input noisy image and 'b' be the output image which initially is a copy of the input noisy image 'a'. Now the image 'a' acts as the reference image and it is processed pixel-by-pixel and the corresponding pixel in the image 'b' is replaced with the output pixel which is obtained as a result of processing done on image 'a'.

The steps of the MDBUTMF are explained as follows.

ALGORITHM

Step 1: Read Noisy Image.

Step 2: Select 2D window of size 3x3 with centre element as processing pixel. Assume that the pixel being processed is P_{ij}.

Step 3: If P_{ij} is an uncorrupted pixel (that is, $0 < P_{ij} < 255$), then its value is left unchanged.

Step 4: If $P_{ij} = 0$ or $P_{ij} = 255$, then P_{ij} is a corrupted pixel.

Step 5: If 3/4th or more pixels in selected window are noisy then increase window size to 5x5.

Step 6: If all the elements in the selected window are 0's and 255's, then replace P_{ij} with the mean of the elements in the window else go to step 6.

Step 7: Eliminate 0's and 255's from the selected window and find the median value of the remaining elements. Replace P_{ij} with the median value.

Step 8: Repeat steps 2 to 6 until all the pixels in the entire image are processed.

The flow chart of the proposed algorithm is shown in Fig. 1.

IV. ILLUSTRATION OF MDBUTAMF ALGORITHM

Each and every pixel of an image is checked for the presence of salt and pepper noise. Different cases are illustrated in this section. If the processing pixel is not noisy pixel, that is, its value lies between 0 and 255 is illustrated in Case i). If all pixels in the selected window are noisy, that is, either 0 or 255 is illustrated in Case ii). If 3/4th or more pixels

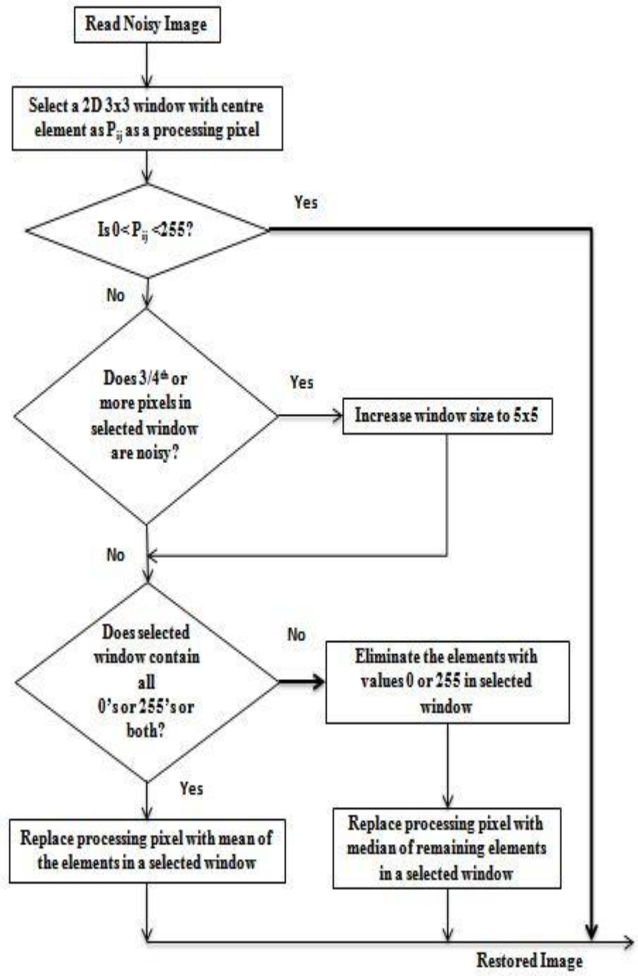

Fig. 1 Flow Chart of MDBUTAMF

in the selected window including processing pixel are noisy, that is, either 0 or 255 is illustrated in Case iii). If less than 3/4th pixels in the selected window including processing pixel are noisy, that is, either 0 or 255 is illustrated in Case iv).

Case i) If processing pixel in the selected 3x3 window is non-noisy pixel, it does not require further processing. For example, if processing pixel is 99, then it is non-noisy pixel:

73	50	175
211	**99**	114
91	39	56

Case ii) If all the pixel values in the selected window contains salt or pepper noise (i.e., 255 or 0 pixel value):

0	0	255	0	255
255	255	0	255	0

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

0	0	**0**	255	255
0	0	255	255	0
255	0	0	255	255

where "0" is processing pixel, i.e., P_{ij}.

Since all the elements surrounding are 0's and 255's, if one takes the median value it will be either 0 or 255 which is again noisy. To solve this problem, the mean of the selected window is found and the processing pixel P_{ij} is replaced by the mean value. Here the mean value is 142. Hence replace the processing pixel P_{ij} by 142.

Case iii) If $3/4^{th}$ or more pixels in the selected 3x3 window including processing pixel are noisy (i.e., 255 or 0 pixel value):

0	0	255
255	**255**	0
0	255	93

As more than $3/4^{th}$ pixels in the selected 3x3 window are noisy, select 5x5 window.

99	113	0	0	255
255	0	0	255	165
190	255	**255**	0	0
0	0	255	93	255
56	255	77	0	255

Now eliminate the salt and pepper noise from the selected 5x5 window. The 1D array of the above matrix is [99 113 0 0 255 255 0 0 255 165 190 255 255 0 0 0 0 255 93 255 56 255 77 0 255]. After elimination of 0's and 255's, the pixel values in the selected window will be [99 113 165 190 93 56 77]. Here the median value is 99. Hence replace the processing pixel P_{ij} by 91.

Case iv) If less than $3/4^{th}$ pixels in the selected window including processing pixel are noisy (i.e., 255 or 0 pixel value):

0	78	255
255	**0**	0
75	255	80

Now eliminate the salt and pepper noise from the selected 3x3 window. The 1D array of the above matrix is [0 78 255 255 0 0 75 255 80]. After elimination of 0's and 255's, the pixel values in the selected window will be [78 75 80]. Here the median value is 78. Hence replace the processing pixel P_{ij} by 78.

V. SIMULATION RESULTS

Among the commonly tested 8-bit grayscale and color images, Lena, Baboon and Cameraman are selected for simulations. In the simulations, images are corrupted by "salt" (with value 255) and "pepper" (with value 0) noise. Also a wide range of noise levels varied from 10% to 90% with increments of 10% is tested. Restoration performances are quantitatively measured by Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR) and Image Enhancement Factor (IEF) as defined in (1), (2) and (3) respectively:

TABLE I
COMPARISON OF PSNR VALUES OF DIFFERENT ALGORITHMS FOR LENA IMAGE (GRAY-SCALE 256x256) AT DIFFERENT NOISE DENSITIES

Noise in %	PSNR in dB						
	MF	AMF	PSMF	DBA	MDBA	MDBUTMF	MDBUTAMF
10	26.33	28.85	30.85	36.84	36.90	37.34	38.08
20	25.63	27.84	28.73	32.37	32.33	34.12	34.44
30	21.99	26.36	25.83	30.45	30.85	32.67	32.02
40	18.36	24.85	22.23	28.88	28.66	30.98	30.39
50	15.85	23.73	19.10	26.41	26.46	28.32	28.89
60	11.28	20.83	12.83	24.99	24.85	26.81	27.35
70	9.46	15.83	9.02	22.11	22.05	24.32	26.09
80	8.84	10.93	8.90	20.67	20.77	21.28	24.81
90	6.37	7.12	6.00	17.89	17.92	18.45	22.65

TABLE II
COMPARISON OF IEF VALUES OF DIFFERENT ALGORITHMS FOR LENA IMAGE (GRAY-SCALE 256x256) AT DIFFERENT NOISE DENSITIES

Noise in %	PSNR in dB						
	MF	AMF	PSMF	DBA	MDBA	MDBUTMF	MDBUTAMF
10	10.18	23.85	171.89	390.8	421.89	648.29	537.14
20	28.76	37.16	207.27	358.4	372.19	566.98	461.90
30	30.18	42.85	190.85	322.3	324.91	591.01	410.02
40	23.98	40.44	143.66	268.0	275.02	422.82	374.75
50	11.01	36.59	62.90	208.9	217.57	365.22	336.39
60	6.67	25.50	6.01	190.5	175.66	267.92	282.63
70	3.21	7.22	3.91	128.0	129.25	171.90	244.71
80	2.99	2.10	1.78	67.01	73.62	101.11	207.48
90	1.06	1.93	1.32	33.56	33.91	34.56	142.81

$$zMSE = \frac{\sum_i \sum_j (Y(i,j) - \hat{Y}(i,j))^2}{M \times \mathrm{N}} \quad (1)$$

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

$$IEF = \frac{(\sum_i \sum_j |\eta(i,j) - Y(i,j)|)^2}{(\sum_i \sum_j |\hat{Y}(i,j) - Y(i,j)|)^2} \quad (3)$$

where, $M \times N$ is size of the image, Y represents the original image, $\hat{Y}$ denotes the restored image, η represents the noisy image. The PSNR and IEF values of the proposed algorithm are compared against the existing algorithms by varying the noise density from 10% to 90% and are shown in

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Table I and Table II. From the Tables I and II, it is observed that the performance of the proposed algorithm (MDBUTAMF) is better than the existing algorithms. A plot of PSNR and IEF against noise densities for Lena image is shown in Fig. 2.

The qualitative analysis of the proposed algorithm against the existing algorithms at different noise densities for Baboon image is shown in Fig. 3. In this figure, the first column represents the processed image using MF at 80% and 90% noise densities. Subsequent columns represent the processed images for AMF, PSMF, DBA, MDBA, MDBUTMF and

TABLE III
COMPARISON OF PSNR VALUES OF DIFFERENT TEST IMAGES (GRAY-SCALE) AT NOISE DENSITY OF 70%

Test images	PSNR in dB						
	MF	AMF	PSMF	DBA	MDBA	MDBUTMF	MDBUTAMF
Camera-man	9.64	13.39	9.74	20.48	19.79	22.25	23.51
Lena	9.46	15.83	9.02	22.11	22.05	24.32	26.09

MDBUTAMF. The PSNR values of images (Lena and Cameraman) at 70% noise density using different algorithms are given in Table III. From the table, it is clear that the MDBUTAMF gives better PSNR values irrespective of the nature of the input image.

The MDBUTAMF is also used to process the color images that are corrupted by salt and pepper noise. The color image taken into account is Baboon. In Fig. 4, the first column represents the processed image using MF at 80% and 90% noise densities. Subsequent columns represent the processed images for AMF, PSMF, DBA, MDBA, MDBUTMF and MDBUTAMF. From the figure, it is possible to observe that the quality of the restored image using proposed algorithm is better than the quality of the restored image using existing algorithms.

VI. CONCLUSION

The performance of the algorithm has been tested under a wide range (from 10% to 90%) of noise densities on both gray-scale and color images. Results reveal that the proposed algorithm exhibits better performance in comparison with other existing algorithms in terms of MSE, PSNR and IEF. Even at high noise density levels, proposed filter gives better results both visually and quantitatively. This restoration of images is very likely to find potential applications in a number of different areas such as medical diagnostics, archaeology, satellite imaging, etc.

REFERENCES

[1] Pratt William, "*Digital Image Processing*," John Wiley & Sons, Fourth Edition, 2007.

[2] Gonzalez and Woods, "*Digital Image Processing*," Pearson Education, Second Edition, 2005.

[3] 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.

[4] 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.

[5] K. S. Srinivasan and D. Ebenezer, "*A new fast and efficient decision based algorithm for removal of high density impulse noise*," IEEE Signal Processing Letters, vol. 14, no. 3, pp. 189–192, Mar. 2007.

[6] S. Esakkirajan, T. Veerakumar, Adabala N. Subramanyam and C. H. PremChand, "*Removal of high density salt and pepper noise throughmodified decision based unsymmetric trimmed median filter*," IEEE Signal Processing Letters, vol. 18, no. 5, pp. 287-290, May 2011.

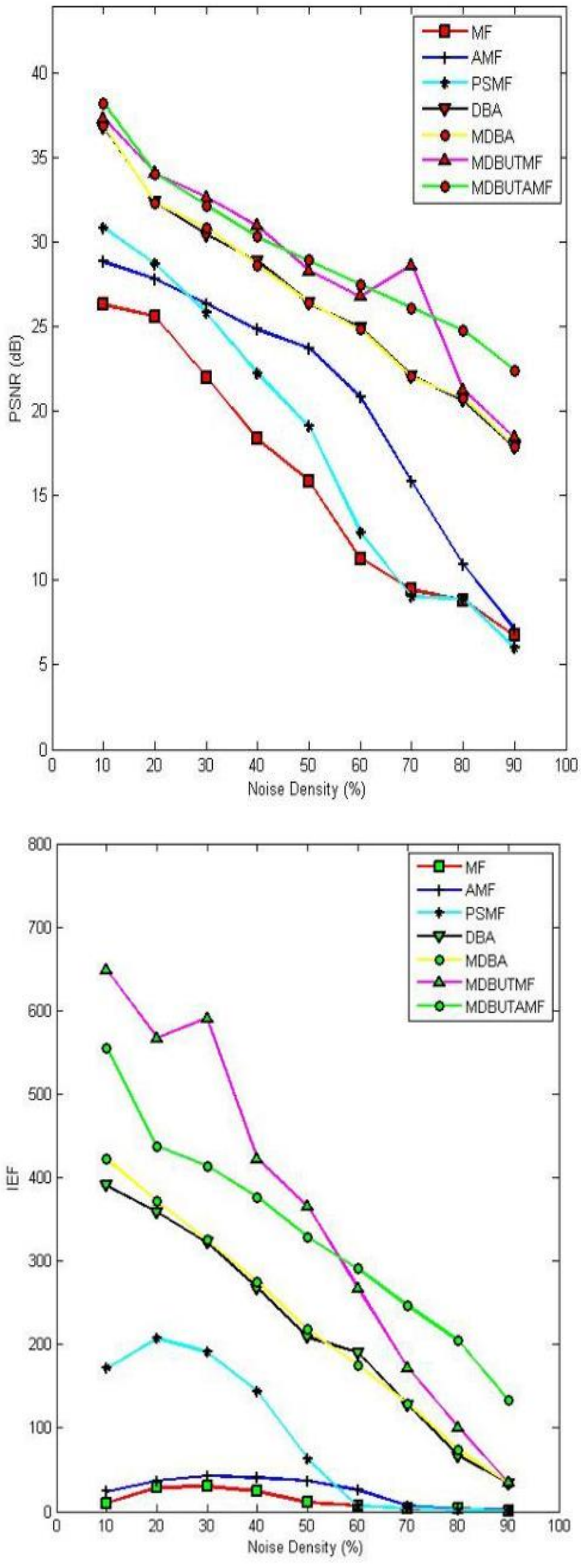

Fig. 2 Comparison graph of PSNR and IEF at different noise densities for Lena image

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

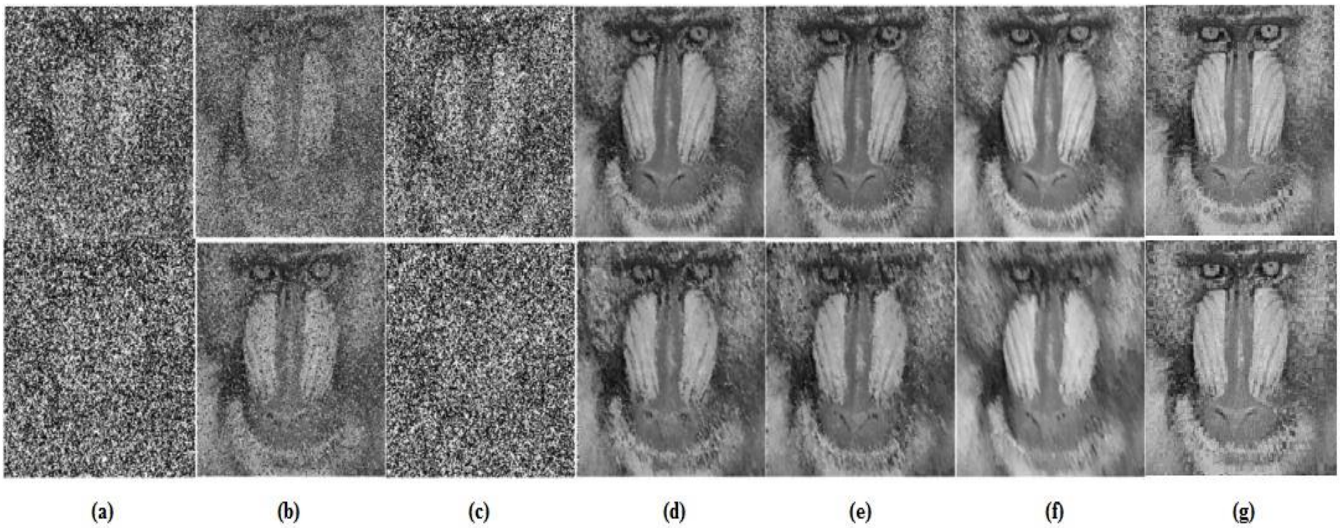

Fig 3. Results of different algorithms for Baboon image. (a) Output of MF. (b) Output of AMF. (c) Output of PSMF. (d) Output of DBA. (e) Output of MDBA. (f) Output of MDBUTMF. (g) Output of MDBUTAMF. Row 1 and Row 2 show processed results of various algorithms for image corrupted by 80% and 90% noise densities, respectively.

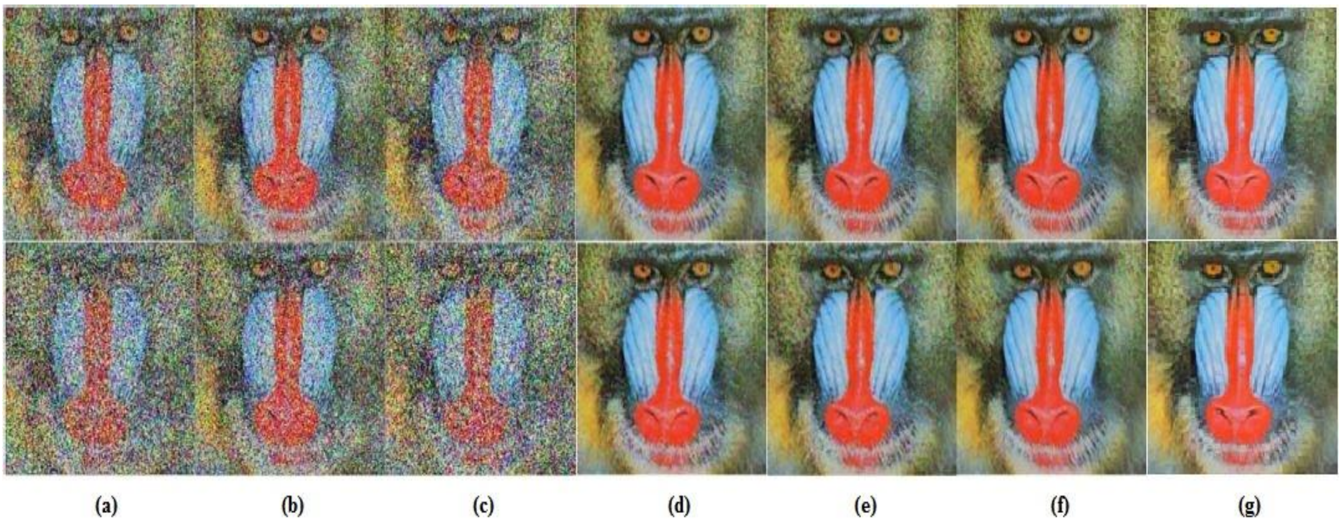

Fig 4. Results of different algorithms for Color Baboon image. (a) Output of MF. (b) Output of AMF. (c) Output of PSMF. (d) Output of DBA. (e) Output of MDBA. (f) Output of MDBUTMF. (g) Output of MDBUTAMF. Row 1 and Row 2 show processed results of various algorithms for image corrupted by 70% and 80% noise densities, respectively.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)