Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A DECISION BASED UNSYMMETRICAL TRIMMED MIDPOINT ALGORITHM FOR THE REMOVAL OF HIGH DENSITY SALT AND PEPPER NOISE

K.VASANTH[1], V.JAWAHAR SENTHILKUMAR[2]
[1]Research Scholar, [2]Research Guide
[1]Sathyabama University, Rajiv Gandhi Salai, Chennai, Tamilnadu, India
[2]Anna University, Chennai, Tamilnadu, India
E-mail: [1]vasanthecek@gmail.com , [2]veerajawahar@annauniv.edu

ABSTRACT

A New Decision based Unsymmetrical Trimmed Midpoint algorithm, which uses asymmetric trimmed midpoint rather than median for the restoration of gray scale and color images that are heavily corrupted by salt and pepper noise is proposed. The processed pixel is checked for 0 or 255; if examined pixel is equal to 0 or 255, then it is considered as noisy pixel else not noisy. The noisy pixel is replaced by midpoint of the asymmetric trimmed array. The non noisy pixel is left unaltered. Under high noise densities, the proposed algorithm fairs well by preserving fine details of the image. The Rank ordering is done by using a modified snake like algorithm with reduced number of comparators. The proposed algorithm shows excellent results quantitatively and qualitatively when compared to existing and recent filters. The proposed algorithm is tested against different grayscale and color images which gives higher Peak Signal-to-Noise Ratio (PSNR) and good Image Enhancement Factor (IEF) with edge preservation capabilities even at very high noise densities.

Keywords: *Decision Based Filter, Unsymmetrical Elimination Filter, Salt And Pepper Noise, Mean Filter*

1. INTRODUCTION

Digital images are often corrupted by impulse noise, due to faulty camera sensors, transmission of images over faulty channels . Impulse noise is of two types, salt and pepper noise and random valued impulse noise. The intensity of salt and pepper noise always takes relatively high or low gray level. Filtering the corrupted digital image by preserving its details is very important part of image processing [1]. Various non linear filtering techniques have been formulated. Standard median filters (SMF) removed salt and pepper noise effectively by preserving the edges but flatters at high noise densities [1]. An adaptive median filter eliminated the above drawback, but owing to its increasing window size lead to blurring of images [2]. Over the years many Threshold based median and related impulse noise filters were proposed in recent years such as Detail preserving filter (DPF)[3], Pixel-wise MAD (PWMAD) [4] filter, Signal dependent rank order mean (SD-ROM) filter[5], Switched median filters [6]-[7] were proposed. These filters do not have a strong decision or does not consider the local statistics. Hence they fail at heavy noise levels without preserving the fine image details. To elude the flaw, Decision based filter [8] was proposed. This filter identifies the processed pixel as noisy, if the pixel value is either 0 or 255; else it is considered as not noisy. Under High noisy environment the DBA filter replaces the noisy pixel with neighborhood pixel. In spite of repeated replacement of neighborhood pixel results in streaks in restored image. To avoid streaks in images an improved DBA (DBUTMF) [9] is proposed with replacement of median of unsymmetrical trimmed output, but under high noise densities all the pixel inside the current would take all 0's or all 255's or combination of both 0 and 255. Replacement of trimmed median did not fare well for above case. Hence Modified decision based un-symmetric trimmed median filter (MDBUTMF) [10] is proposed. The above cause is eliminated by replacing the mean of the current window. When the noise densities scale greater than 80% the Smudging of edges occurs. The proposed Decision Based Modified Mean filter (DBMROMF) uses basic Rank ordered modified mean logistics to eliminate the above stated flaws by preserving This letter is organized as follows. Section II deals exclusively with analogy of decision based modified mean filter. Section III gives the qualitative and quantitative comparison of proposed filter with existing filters. Section IV gives the concluding remarks for this letter.

Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. DECISION BASED MODIFIED MEAN FILTER

2.1 UNSYMMETRICAL TRIMMED FILTER

The crux behind the above filter is to eliminate the outliers inside the current window. Certain type of non linear filters such as Alpha trimmed mean filter (ATMF), Alpha trimmed midpoint (ATMP) etc., works on the above principle. These filters use a parameter called "α" which decides the number of pixels to be eliminated. It was found that when "α" is increased, the filter fared well. For high noise densities it does not preserve the image information due to the elimination of outlier values. So to overcome the drawback of fine detail preservation and removal of impulse noise during heavy noise conditions Decision based midpoint (DBMP) is proposed.

2.2 SNAKE LIKE IMPROVED SHEAR SORTING

Over the years sorting algorithm is a basic operation behind all the median filters. All the existing sorting algorithms require more comparators as shown in table 1. In this paper a new snake like improved shear sorting algorithm is proposed for ordering the entire array of processed pixels as shown in figure 1. Let D be an m x n matrix which is mapped with linear integer sequence W. Sorting the sequence W is then equivalent to sorting the elements of D in some Pre determined indexing scheme. The proposed Snake like modified algorithm consists of three basic operations row sorting, column sorting and semi diagonal sorting. The algorithm of the proposed snake like improved shear sorting algorithm is as follows.

Step1: The considered 2D processing window as shown in figure 1.a

Step2: Sort the 1th and 3rd rows of the 2D array in ascending order and 2nd row in descending order independently .The sorted sequence is fed to step3 as shown in figure1.b.

Step3: Sort the three columns of the 2D array in ascending order .The sorted sequence is fed to step4 as shown in figure 1.c. Step4: Repeat step 2 and 3 once again as shown in figure1.d and e.

Step5: Now Sort the upper semi diagonal of the semi sorted 2D array in ascending order as shown in figure1.e.

Step6: Sort the Lower semi diagonal sorted array in ascending order as shown in figure1. Resulting array is sorted in a snake like order. The procedure is repeated for the other windows of the image.

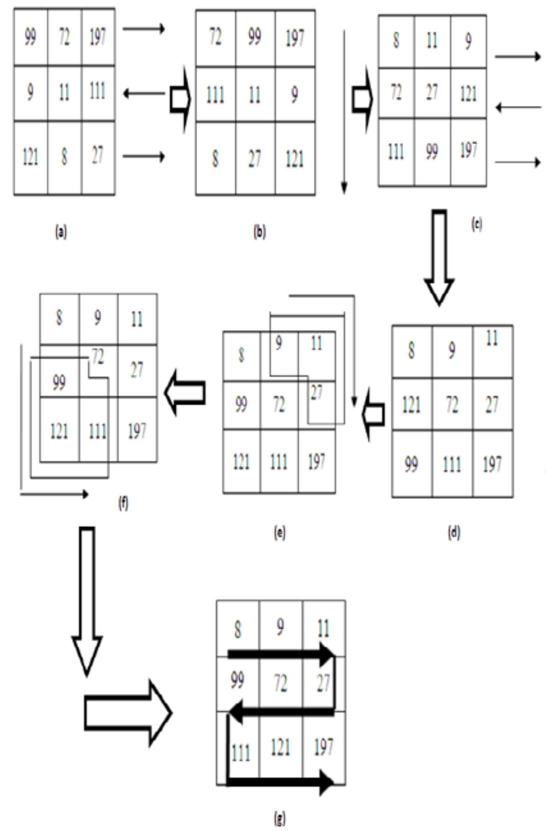


Figure 1 illustration of the proposed sorting methodology

TABLE I Complexities of various sorting algorithm

SORTING TECHNIQUES	Number of Comparisons required to sort the entire 3x3 window (worst case)
Bubble sorting	$0(n^2)$
Insertion sorting $0(n^2)$	$0(n^2)$
Selection sort$0(n^2)$	$0(n^2)$
Merge Sort	0(n log n)
Heap Sort	0(n log n)
Quick Sort	$0(n^2)$
Shear sorting	0(2*(3n))
Snake like Modified shear sorting (proposed Sorting Algorithm)	$0(4n+2\sqrt{(n)})$

2.3 PROPOSED ALGORITHM

The Decision based modified Rank ordered mean filter (DBMROMF) initially detects impulse and corrects it subsequently. All the pixels of an image lie between the dynamic ranges [0,255]. If the processed pixel holds minimum (0) or maximum (255), pixel is considered as noisy and processed by DBMMF else as not noisy and the pixel is

Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

unaltered. The brief illustration of the algorithm is as follows.

Step 1: Choose 2-D window of size 3x3. The processed pixel in current window is assumed as p_{xy}.

Step 2: Check for the condition $0 < p_{xy} < 255$, if the condition is true then pixel is considered as not noisy and left unaltered.

Step 3: If the processed pixel p_{xy} holds 0 or 255 i.e. (p_{xy}=0 or p_{xy} =255) then pixel p_{xy} is considered as corrupted pixel. Convert 2D array into 1D array. Sort the 1D array which is assumed as S_{xy}.

Step 4: Initialize two counters, forward counter (F) and reverse counter (L) with 1 and 9 respectively. When a 0 or 255 are encountered inside the window F is increased by 1 or L is decremented by 1 respectively. When pixel is noisy there happens to be two possible cases.

Case I: If the processing pixel is noisy and the current Processed window contains few 0'^s and 255'^s. So check for 0 or 255 in sorted array S_{xy}, simultaneously counters would propagate along the S_{xy} array thereby eliminating outliers retaining only the pixel that hold values other than 0 and 255. After checking all the pixels F and L would hold a particular value indicating the number of outliers eliminated on either sides. The noisy pixel is replaced by the midpoint of the sorted array.

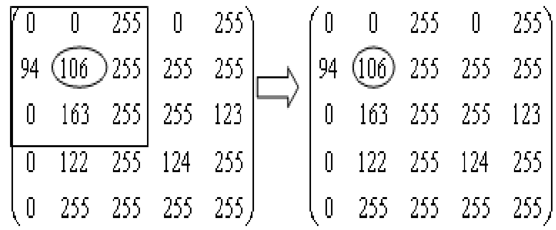
Case II: If every pixels that reside inside the kernel is the combination of 0 or 255. Even this condition is addressed by the case I operation. There by making the algorithm simple. When all the pixel elements hold 0 or 255 then the values are retained, assuming it as texture of the image.

Step 5: Steps 1 to 4 is repeated until all pixels of the entire image is processed.

2.4 INSIGHT OF THE PROPOSED METHODOLOGY

The salt and pepper noise is initially detected by comparing the processed pixel with 0 or 255. This process is done on entire pixels in the image. The bigger matrix refers to image and values enclosed inside a rectangle is considered to be the current processing window. The element encircled refers to processed pixel. Step 2 is illustrated in case (1). Step 3 and 4 are visualized along with the case I in case (2) .Case II is briefed in case (3).

Case (1): In the illustration given below, check the processed pixel for $0 < p_{xy} < 255$. In this case the processed pixel is 106. Hence processed pixel is not 0 or 255. So pixel is considered as noise free and pixel is unaltered.

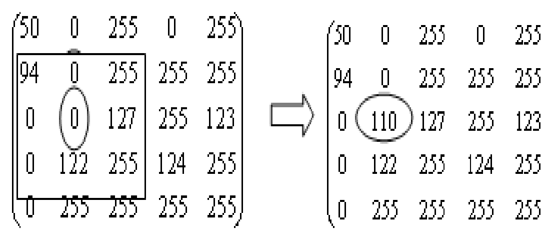


Corrupted image segment Restored image segment

Case(2): In the selected window the processed pixel holds 0 (or 255). So the processed pixel is considered as noisy. Initialize forward counter F=1 and reverse counter L=9. Convert the 2D array into 1D array and sort the converted array. F and L counter moves in forward and reverse directions respectively.

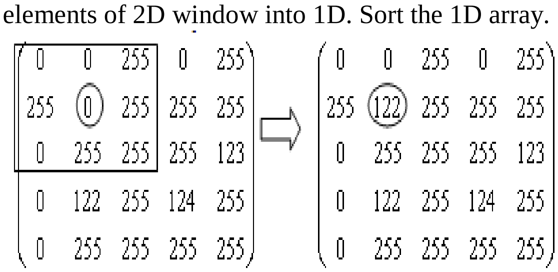
Unsorted array: 94 0 0 0 0 122 255 127 255
Sorted array S_{xy} 0 0 0 0 94 122 127 255 255

Now check for the presence of 0 or 255 in the sorted array. Every time a 0 is detected F is incremented by1 and 255 is detected L is decremented by1. In the above example there are three 0 and two 255. Hence F is incremented by four times and L is decremented by 2 times. Now finally F is holding 5 and L is holding 7. Now the corrupted pixel is replaced with midpoint of the trimmed array i.e. corrupted pixel is replaced by (S(4)+S(7))/2 = (94 +127)/2= 110.



Corrupted image segment Restored image segment

Case (3): This sub case works if the entire pixel inside the current window is either pepper (0) or salt (255). Initialize F=1 and L=9 and convert the elements of 2D window into 1D. Sort the 1D array.



Corrupted image segment Restored image segment

Unsorted 1D array:

0 255 0 0 (0) 255 255 255 255

Sorted 1D array S_{xy}:

Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

0 0 0 0 255 255 255 255 255

Now the F counters propagates forward and L in reverse direction. Finally F and L hold 5 and 4 respectively. Hence the noisy pixel is replaced by (S (5) + S(4))/2= (255+0)/2=122.

3. SIMULATION RESULTS AND DISCUSSIONS

The Quantitative performance of the proposed algorithm is evaluated based on Peak signal to noise ratio (PSNR) ,Mean Square Error (MSE) and Image Enhancement Factor (IEF) which is given in equations 1,2 ,3 respectively.

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

$$MSE = \frac{\sum_i \sum_j (r_{ij} - x_{ij})2}{M \times N} \quad (2)$$

$$IEF = \frac{\left(\sum_i \sum_j n_{ij} - r_{ij}\right)^2}{\left(\sum_i \sum_j x_{ij} - r_{ij}\right)^2} \quad (3)$$

Where *r* refers to Original image, n gives the corrupted image *x* is denotes restored image, M x N is the size of Processed image. The existing algorithms used for the comparison are SMF. AMF, DPF, PSMF, DBA, MDBUTMF. The qualitative performance of the proposed algorithm is tested on various images of various compositions. Quantitative analysis is made by varying noise densities in steps of ten from 10% to 90% on low detail, medium detail and high detail images and comparisons are made in terms of PSNR, IEF, MSE and Time. Results and graphs are given in Table II, III, IV, V and figure 2,3,7,8 respectively. All the simulation is implemented using MATLAB 7.0 (R14) in Intel Pentium (R) dual CPU E2140@1.6Ghz with 1GB RAM capacity. It is vivid from the tables and graphs that the proposed algorithm fairs on par with MDBUTMF at low noise densities. In case of high noise densities the proposed algorithm has excellent noise suppression characteristics. Hence the algorithm has high PSNR, IEF and low MSE when compared to other algorithms. The qualitative aspect of the proposed algorithm is listed against various algorithms for very high noise densities (70%, 80%, and 90%) on Lena (Gray scale image) as shown in figure4. The edge preservation of the proposed algorithm is shown for Baboon image for noise composition of 85% as shown in figure 5. Canny edge detection method is used for the edge detection. The proposed algorithm also fairs well even for color image at very high noise densities as shown in figure 6. It was found that even at 80% noise the proposed algorithm preserved the global and local edges. At high noise densities (above 80% for high detailed image such as baboon) filters such as SMF, AMF, DPF, PSMF fails. The DBA algorithm induces streaking thereby edges are affected. The problem with MDBUTMF causes the image edge to smudge away thereby resulting in poor edge preservation. The proposed algorithm shows excellent noise suppression and edge preservation capabilities. The proposed algorithm effectively removes the salt and pepper noise and preserves edges for low, medium and high detail images. For very high noise densities the proposed algorithm preserves the global edge even for high detail image. It was found that SMF, AMF, PSMF, DPF, fails at very high noise densities. The Streaks caused by DBA is vividly seen on image 4 6^{th} columns. The MDBUTMF fails to preserve edges due to smudging of pixel edges it is given in column 7. Hence it is evident from column 8 that the proposed algorithm eliminates salt and pepper noise at very high noise densities. It is evident qualitatively the proposed algorithm preserves global edge (overall structure of the image) while eliminating salt and pepper noise for the high detailed image. It is found that the time consumed by the algorithm is also optimum from the figure 4. The proposed Mesh based sorting outclasses all the existing Sorting Techniques.

4. CONCLUSION

A new decision based algorithm which used 2D mesh sorting which is simple in approach and used mainly for rank ordering. The proposed algorithm gives excellent noise suppression capabilities in gray scale and color images corrupted by salt and pepper noise for high noise densities. The proposed algorithm also fairs well in preserving the global edge of the high detail images. The proposed algorithm is also very good in eliminating salt and pepper noise in color images. The Proposed algorithm outclasses other classical and existing recent algorithms both quantitatively and qualitatively for very high noise densities. The architecture of the proposed algorithm can also be easily realized.

Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TABLE II
PERFORMANCE OF VARIOUS ALGORITHMS AT DIFFERENT NOISE DENSITIES FOR PSNR

Noise in %	PSNR IN DB						
	SMF	AMF	PSM	DPF	DBA	MD BUTMF	PA
10%	34.9	39.3	38.85	33.81	39.02	43.12	42.5
20%	30.3	36.9	33.41	27.53	36.84	41.22	39.3
30%	23.9	34.6	29.4	23.18	35.84	37.9	37.5
40%	19.0	32.2	25.45	19.76	33.26	36.4	36.1
50%	15.9	27.3	23.39	16.8	31.45	34.3	34.8
60%	12.3	21.6	20.27	14.51	29.63	32.15	33.6
70%	10.0	16.6	9.94	12.5	27.88	29.61	32.3
80%	8.1	12.7	8.1	10.79	25.51	26.83	30.6
90%	6.6	9.86	6.68	9.21	21.82	22.45	27.6

TABLE III
PERFORMANCE OF VARIOUS ALGORITHMS AT DIFFERENT NOISE DENSITIES FOR IEF

Noise in %	IEF						
	SMF	AMF	PSM	DPF	DBA	MD BUTMF	PA
10%	89.0	246.8	219.8	69.1	230.3	630.8	516.8
20%	61.0	281.3	124.9	32.4	276.3	762.7	498.9
30%	21.4	254.4	74.5	17.8	331.1	665.5	474.8
40%	9.1	192.9	40.1	10.7	242.3	521	470.6
50%	4.9	78.3	39.6	6.8	199.9	384.8	431.7
60%	2.9	25	19.1	4.8	157.8	282.1	400.6
70%	2.0	9.1	1.9	3.5	123.0	183.4	346.0
80%	1.4	4.3	1.4	2.7	81.5	110.5	261.7
90%	1.1	2.5	1.1	2.1	39.1	45.5	151.2

TABLE IV
Performance of various algorithms at different noise densities for MSE

Noise in %	PSNR IN DB						
	SMF	AMF	PSM	DPF	DBA	MD BUTMF	PA
10%	20.9	7.4	8.4	27	8.1	2	3.6
20%	60.6	13.1	29.6	114.6	13.4	4.9	7.4
30%	259.3	22.1	74.5	312.1	16.9	8.2	11.3
40%	814.2	38.5	185.2	686.9	30.6	14.2	15.8
50%	1877	118	187.5	1355	46.4	23.9	21.2
60%	3776	443	484.2	1197	70.7	39.6	28.0
70%	6637	1421	600.0	3651	105.9	69.1	37.5
80%	9945	3413	1000	5408	182.6	134.6	56.6
90%	14179	6708	1396	7798	427.1	369.2	112

TABLE V
Performance of various algorithms at different noise densities for Time

Noise in %	PSNR IN DB						
	SMF	AMF	PSM	DPF	DBA	MD BUTMF	PA
10%	2.13	23	3.41	27.38	8.4	15.43	17.3
20%	2.26	22.9	2.12	27.56	8.98	7.64	17.7
30%	1.63	22.5	2.14	27.96	8.13	8.87	17.5
40%	1.52	22.4	2.35	28.37	8.09	9.25	17.6
50%	1.47	22.1	2.43	28.78	8.07	10.21	17.2
60%	1.55	21.7	1.73	29.97	7.99	11.3	17.4
70%	2.51	21.2	1.74	29.74	8.18	11.85	17.2
80%	2.16	20.8	1.64	30.66	8.01	12.99	17.4
90%	1.47	19.9	1.71	30.8	8.31	14.74	17.9

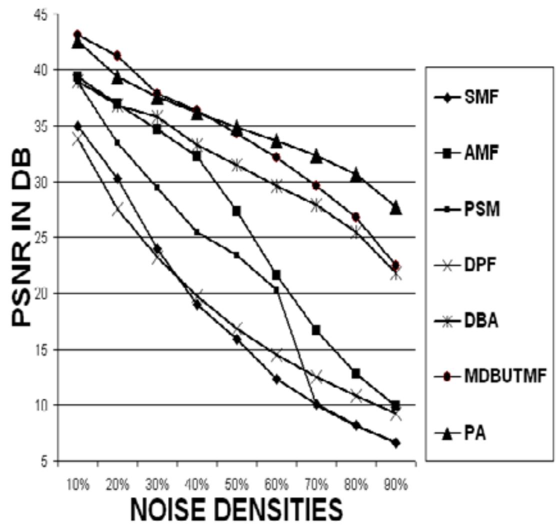


Figure 2 Performance comparison of PSNR at various noise densities for low detail Lena image

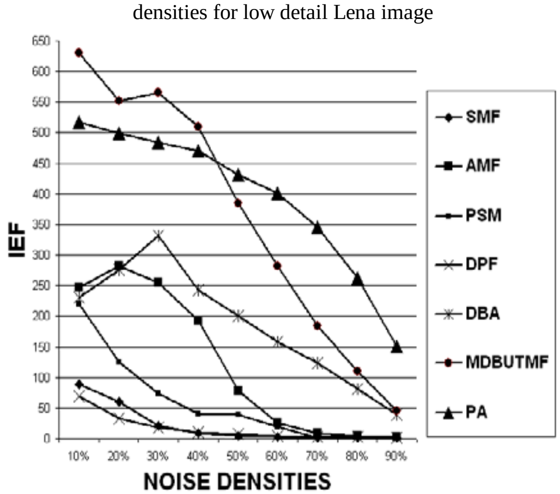
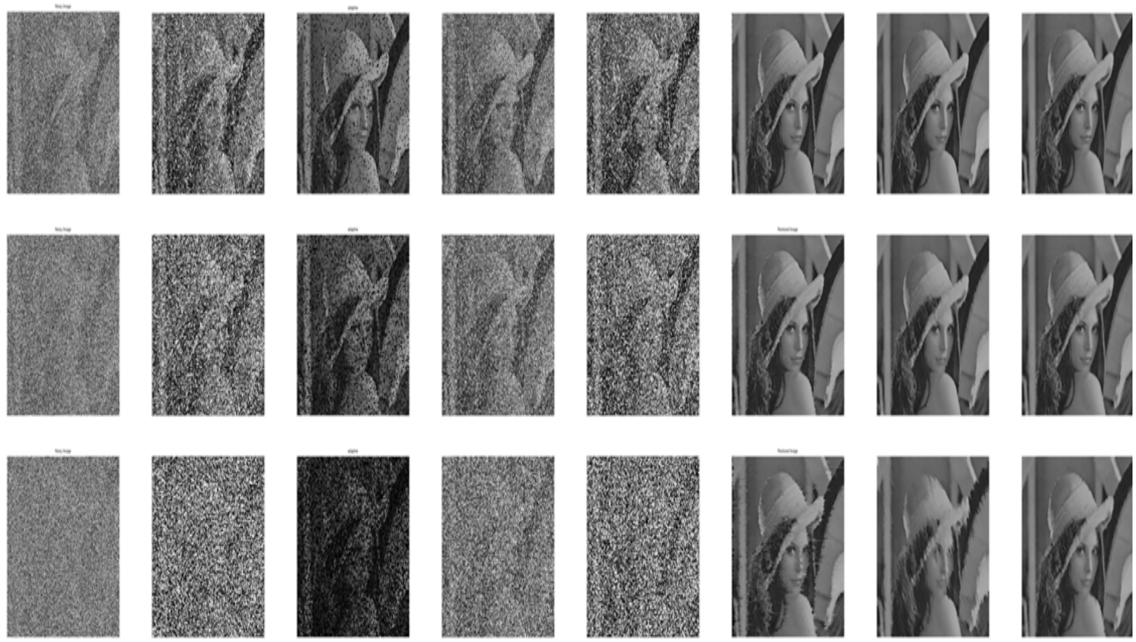


Figure3 Performance comparison of IEF at various noise densities for low detail Lena image

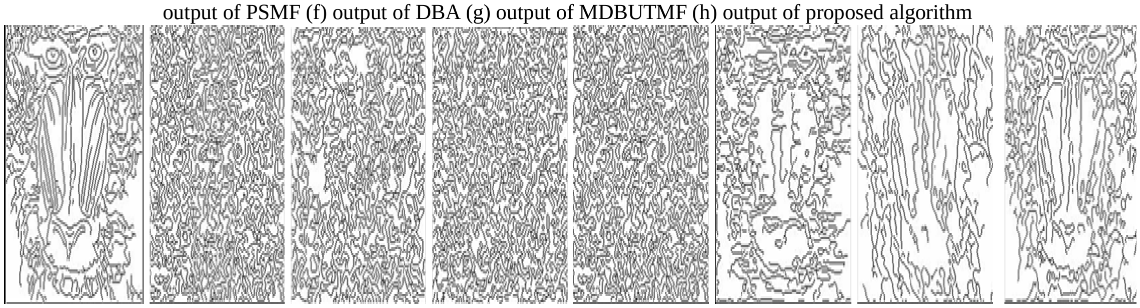
Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



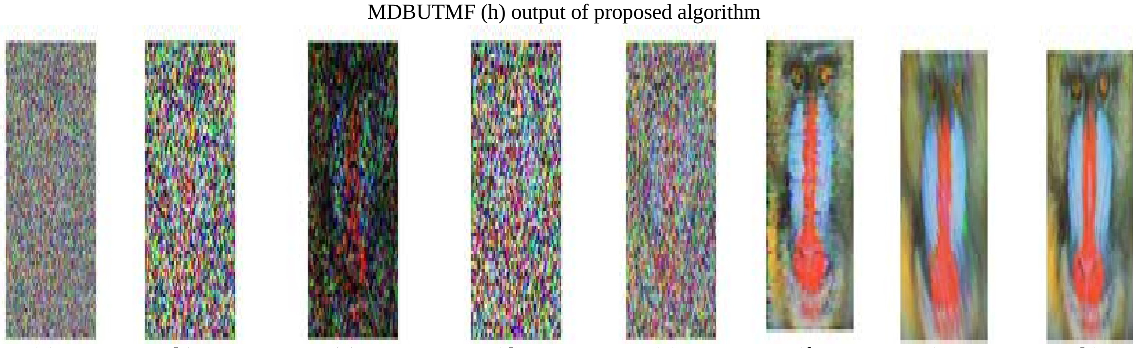
(a) (b) (c) (d) (e) (f) (g) (h)

Figure 4. Performance of different algorithms on Lena image corrupted by 70% salt and pepper noise in row1, 80% salt and pepper noise in row 2, 90% salt and pepper noise in row3 (a) impulse noise (b) output of SMF (c) output of AMF (d) output of DPF (e) output of PSMF (f) output of DBA (g) output of MDBUTMF (h) output of proposed algorithm



(a) (b) (c) (d) (e) (f) (g) (h)

Figure. 5. Edge preservation after the filtering of various algorithms on baboon image for 85% salt and pepper noise (a) edges of the original image (b) output of SMF (c) output of AMF (d) output of DPF (e) output of PSMF (f)output of DBA (g) output of MDBUTMF (h) output of proposed algorithm



(a) (b) (c) (d) (e) (f) (g) (h)

Figure. 6. Performance of various algorithms on baboon image corrupted by 90% salt and Pepper noise on color image (a) Noisy image (b) output of SMF (c) output of AMF (d) output of PSMF (e) output of DPF (f)output of DBA (g) output of MDBUTMF (h) output of proposed algorithm

Journal of Theoretical and Applied Information Technology
31 August 2012. Vol. 42 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

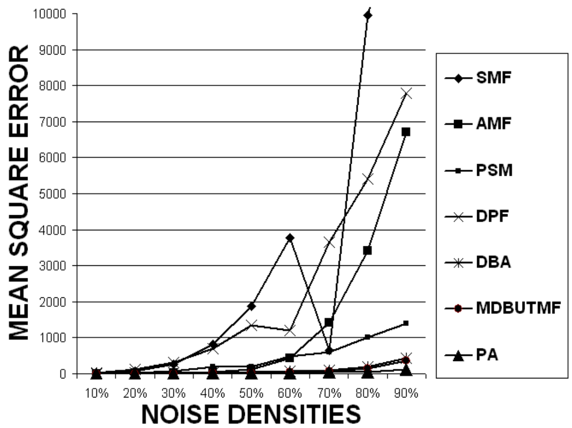


Figure 7. Performance comparison of MSE at various noise densities for low detail Lena image

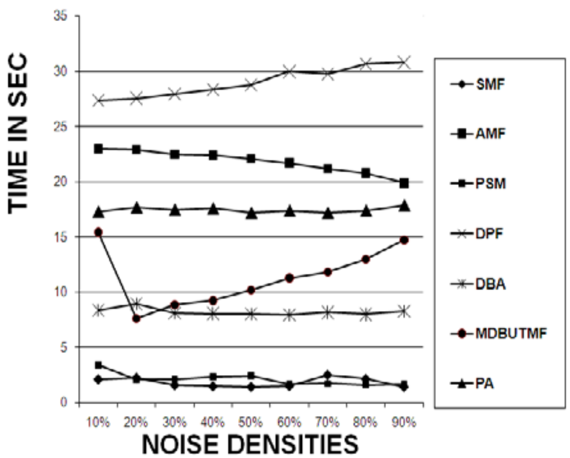


Figure 8. Performance comparison of Time at various noise densities for low detail Lena image

REFERENCES:

[1] J. Astola and P. Kuosmaneen, Fundamentals of non linear Digital Filtering. Boca Raton, FL: CRC, 1997.

[2] H. Hwang and R. A.Hadded, “Adaptive median filter: New algorithms and results,” IEEE transaction on image processing., vol 4,no 4,pp 499-502

[3] Naif Alajlan, Mohamed Kamel, Ed Jernigan, “Detail Preserving impulse noise removal”, International journal on Signal processing: image communication, pages 993-1003, Vol 19, 2004.

[4] BV. Crnojevi,V. Senkand and Z. Trpovski, “Advanced impulse detection based on pixel-wise MAD”, IEEE signal Processing Letters,2004,vol.11, pp.589-592.

[5] E. Abreu, M. Lightstone, S.K. Mitra, “ An efficient approach for the removal of impulse noise from highly corrupted images,” IEEE transactions on image Processing, 1996, vol 5 pp.1012-1025.

[6] P. E. Ng and K. K. Ma, “A switching median filter with boundary discriminative noise detection for extremely corrupted images,” IEEE Transactions on image processing, vol.15, no.6, pp. 1506-1516, June 2006.

[7] S. Zhang and M.A. Karim, “A new impulse detector for switching median filters,” IEEE Signal processing letters, vol.9, no.11, pp. 360-363, November 2002.

[8] K.S. Srinivasan and D.Ebenezer, “A new fast and efficient decision based algorithm for the removal of high density impulse noise,” IEEE Signal processing letters, vol.14, no.3, pp.189-192, March 2007.

[9] 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 international conference on computer modeling and simulation, 2010, pp.409-413.

[10] S. Esakkirajan, T. Veerakumar, Adabala N. Subramanyam and C.H. Prem Chand, “Removal of high density Salt and pepper noise through modified decision based Unsymmetrical trimmed median filter.” IEEE Signal processing letters, Vol. 18,no.5, May 2011.