Removal of High Density Salt & Pepper Noise in Noisy Images Using Decision Based UnSymmetric Trimmed Median Filter (DBUTM)

Dodda Shekar[#1], Rangu Srikanth[*2]

[#1] *M.Tech in VLSI Engineering*
Jayamukhi Institute of Technology and Science
Narsampet, Warangal, AP, India
[*2] *Asst. Professor in Dept of ECE*
Jayamukhi Institute of Technology and Science
Narsampet, Warangal, AP, India.

Abstract-
In the Transmission of images over channels, Images are corrupted by salt and pepper noise, due to faulty communications. Salt and Pepper noise is also referred to as Impulse noise. The objective of filtering is to remove the impulses so that the noise free image is fully recovered with minimum signal distortion. The best-known and most widely used non-linear digital filters, based on order statistics are median filters. Median filters are known for their capability to remove impulse noise without damaging the edges. Median filters are known for their capability to remove impulse noise as well as preserve the edges. The effective removal of impulse often leads to images with blurred and distorted features. Ideally, the filtering should be applied only to corrupted pixels while leaving uncorrupted pixels intact. Applying median filter unconditionally across the entire image as practiced in the conventional schemes would inevitably alter the intensities and remove the signal details of uncorrupted pixels. Therefore, a noise-detection process to discriminate between uncorrupted pixels and the corrupted pixels prior to applying nonlinear filtering is highly desirable. Adaptive Median is a "decision-based" or "switching" filter that first identifies possible noisy pixels and then replaces them using the median filter or its variants, while leaving all other pixels unchanged. This filter is good at detecting noise even at a high noise level. The adaptive structure of this filter ensures that most of the impulse noises are detected even at a high noise level provided that the Window size is large enough. The existing non-linear filter like Standard Median Filter (SMF), Adaptive Median Filter (AMF), Decision Based Algorithm (DBA) and Robust Estimation Algorithm (REA) shows better results at low and medium noise densities. At high noise densities, their performance is poor. A new algorithm to remove high-density salt and pepper noise using modified sheer sorting method and Decision Based UnSymmetric Trimmed Median Filter (DBUTM) is proposed

Keywords:

Median filter, Midpoint filter, trimmed filter, shear sort.

I. INTRODUCTION

Images are often corrupted by impulse noises during acquisition and transmission [5, 6]. Based on the noise values, the noise can be classified as the easier-to-restore salt-and-pepper noise and the more difficult random valued impulse noise [2]. Among all the methods for removal of impulse noise, the median filter [3, 4] is used widely because of its effective noise suppression capability and high computational efficiency [7].non-linear digital filters, based on order statistics are median filters. Median filters are known for their capability to remove impulse noise without damaging the edges. Median filters are known for their capability to remove impulse noise as well as preserve the edges. The main drawback of a standard median filter (SMF) [4], [5] is that it is effective only for low noise densities. At high noise densities, SMFs often exhibit blurring for large window sizes and insufficient noise suppression for small window sizes. However, most of the median filters operate uniformly across the image and thus tend to modify both noise and noise-free pixels. Consequently, the effective removal of impulse often leads to images with blurred and distorted features. Ideally, the filtering should be applied only to corrupted pixels while leaving uncorrupted pixels intact [1]. Applying median filter unconditionally across the entire image as practiced in the conventional schemes would inevitably alter the intensities and remove the signal details of uncorrupted pixels. Therefore, a noise-detection process to discriminate between uncorrupted pixels and the corrupted pixels prior to applying nonlinear filtering is highly desirable Adaptive Median is a “decision-based” or “switching” filter that first identifies possible noisy pixels and then replaces them using the median filter or its variants, while leaving all other pixels unchanged.

This filter is good at detecting noise even at a high noise level. The adaptive structure of this filter ensures that most of the impulse noises are detected even at a high noise level provided that the window size is large enough. The performance of AMF is good at lower noise density levels, due to the fact that there are only fewer corrupted pixels that are replaced by the median values [6], [7], [8]. At higher noise densities, the number of replacements of corrupted pixel increases considerably; increasing window size will provide better noise removal performance; however, the corrupted pixel values and replaced median pixel values are less correlated. As a consequence, the edges are smeared significantly recently. The DBA processes the corrupted image by first detecting the impulse noise. The detection of noisy and noise-free pixels is decided by checking whether the value of a processed pixel element lies between the maximum and minimum values that occur inside the selected window [9], [10]. This is because the impulse noise pixels can take the maximum and minimum values in the dynamic range (0, 255). If the value of the pixel processed is within the range, then it is an uncorrupted pixel and left unchanged. If the value does not lie within this range, then it is a noisy pixel and is replaced by the median value of the window or by its neighborhood values. At higher noise densities, the median value may also be a noisy pixel in which case neighborhood pixels are used for replacement; this provides higher correlation between the corrupted pixel and neighborhood pixel. Higher correlation gives rise to better edge preservation. In addition, the DBA uses simple fixed length window of size 3X3, and hence, it requires significantly lower processing time compared with AMF and other algorithms. The main drawback of decision-based algorithm is that streaking occurs at higher noise densities due to replacement with the neighborhood pixel values. Hence, details and edges are not recovered satisfactorily, especially when the noise level is high.

II. REVIEW OF SORTING ALGORITHMS

A) Shear Sorting Algorithm

Sorting is the most important operation used to find the median of a window. There are various sorting algorithms such as binary sort, bubble sort, merge sort, quick sort etc. In the proposed algorithm, shear sorting technique is used since it is based on parallel architecture. In practice the parallel architectures help to reduce the number of logic cells required for its implementation. The illustration of shear sorting is shown in Figure. 1- 4. In the odd phases (1, 3, 5) even rows are sorted in descending order and rows are sorted out in ascending order. In the even phases columns are sorted out independently in ascending order.

13	**17**	**12**
18	**11**	**19**
15	**14**	**16**

Fig 1.Original matrix before sorting

12	**13**	**17**
19	**18**	**11**
14	**15**	**16**

Fig 2.Step 1- Row sorting

12	**13**	**11**
14	**15**	**16**
19	**18**	**17**

Fig 3.Step 3- Column sorting

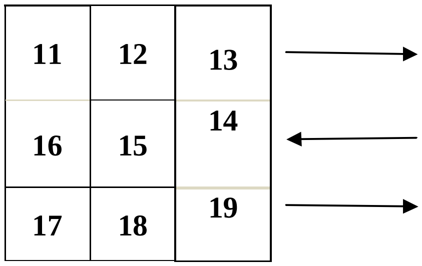

Fig 4.Step 4- Row sorting

B) Modified *Shear Sorting Algorithm*

In order to improve the computational efficiency shear Sorting algorithm is modified as follows:

Step 1) A 2-D Window **"S_{xy}"** of size 3×3 is selected. Assume the pixel to be processed is P(X, Y).

Step 2) the pixel values inside the window are sorted, And **P_{min}, P_{max}** and **P_{med}** determined as follows.

a) The rows of the window are arranged in Ascending order.

b) The columns of the window are arranged Ascending order.

c) The right diagonal of the window is now Arranged in ascending order.

Now the first element of the window is the Minimum value **P_{min}** the last element of the Window is the Maximum value **P_{max}** and the middle element of the window is the median value **P_{med}**

Step 3)

Case 1)

The **P(X, Y)** is an uncorrupted pixel if **P_{min}< P(X, Y) <P_{max}, P_{min}>0** and **P_{max}<255**: the pixel being processed is left unchanged .otherwise **P(X, Y)** is a corrupted pixel

Case 2)

If **P(X, Y)** is a corrupted pixel, it is replaced by its median value if **P_{min}<P_{med}< P_{max}** and **0<P_{med}<255**.

Case 3)

If **P_{min}<P_{med}< P_{max}** is not satisfied or **255<P_{med}=0**,then P_{med} is a noisy pixel .In this case, the **P(X,Y)** is replaced by the value of neighborhood pixel value.

Step 4) Steps 1 to 3 are repeated until the processing is completed for the entire image

DECISION BASED UNSYMMETRIC TRIMMED MEDIAN FILTER(DBUTM)

Decision Based Algorithm (DBA) is a recently proposed algorithm to remove salt and pepper noise. In DBA each Pixel is processed for de noising using a 3 X 3 window. During processing if a pixel is '0' or '255' then it is processed else it is left unchanged. In DBA the corrupted pixel is replaced by the median of the window. At higher noise densities the median itself will be noisy, and, the processing pixel will be replaced by the neighborhood processed pixel. This repeated replacement of neighborhood pixels produces streaking effect. In DBUTM, the corrupted pixels are identified and processed. The DBUTM algorithm checks whether the left and right extreme values of the sorted array obtained from the 3x3 window are impulse values. The corrupted processing pixel is replaced by a median value of the pixels in the 3X3 window after trimming impulse values. The corrupted pixel is replaced by the median of the resulting array.

A) Algorithm (For Image)

The Actual Block Diagram for removing the noise from noisy image

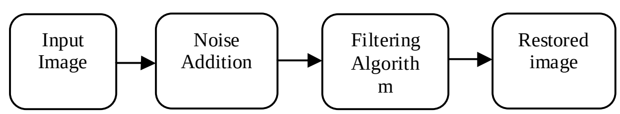

Fig 5. Flowchart for processing an image Sequence by DBUTM

The algorithm for DBUTM is as follows:

Step 1) A 2-D window **"S_{xy}"** of size 3x3 is selected.

Step 2) the pixel values in the window are sorted in ascending order, and stored in a 1-D array.

Step 3) if the pixel value in the array is either **'0' or '255'**, The corresponding pixel values are trimmed (eliminated), And the median of remaining values is calculated.

Step 4) the pixel being processed is replaced by the median value calculated.

Move the window by one step, and repeat from Step 2 to Step 4.

The above steps are repeated, until the processing is completed for entire image.

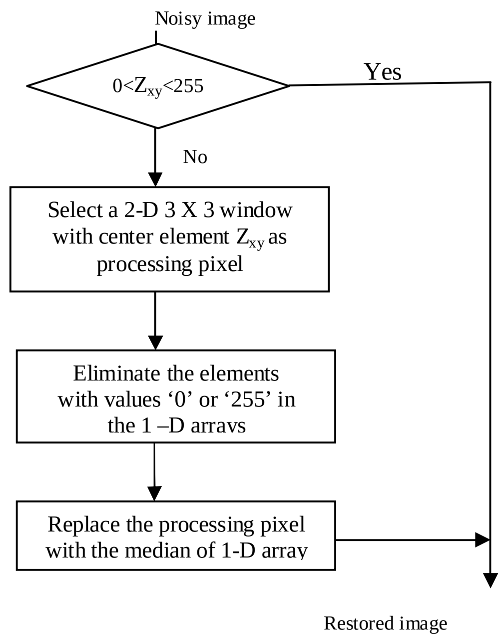

Fig 6. Flow Chart for DBUTM

B. Implementation for video sequence

The video sequence is first converted into frames and frames into images. Then DBUTM algorithm is applied to the images which are separated from frames. After the filtering process, the frames are converted back to the original movie.

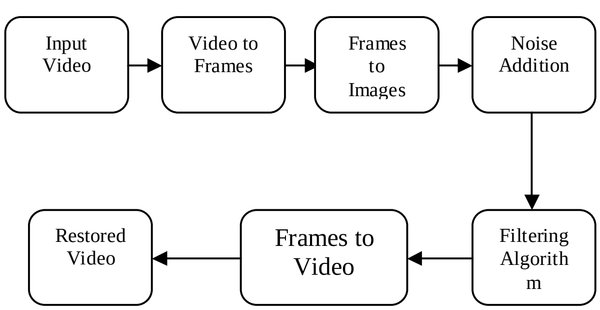

Fig 7. Flowchart for processing a Video Sequence by DBUTM

Algorithm (For Video)

1. Video to frames: The noisy video sequence containing Impulse noise is converted into avi format, which is an Uncompressed format and frames are extracted from the Video.
2. Frames to images: Frames are then converted to images For further processing.

3. Filtering method: The noisy images are de noised using DBUTM algorithm.
4. Frames to movie: After completing the entire process, the processed frames are finally converted back into original movie.

III. EXPERIMENTAL RESULTS

The developed algorithms are tested using 512X512, 8-bits/pixel image Lena (Gray), Parrot (color), Barbara (color). The performance of the proposed algorithm is tested for various levels of noise corruption and compared with standard filters namely standard median filter (SMF), Adaptive median filter (AMF) and decision based algorithm (DBA). Each time the test image is corrupted by salt and pepper noise of different density ranging from 10 to 90 with an increment of 10 and it will be applied to various filters. In addition to the visual quality, the performance of the developed algorithm and other standard algorithms are quantitatively measured by the following parameters such as peak signal-to-noise ratio (PSNR), Mean square error (MSE) and Image Enhancement Factor (IEF). All the filters are implemented in MATLAB 7.5 on a PC equipped with 2.4 GHz CPU and 1 GB RAM memory for the evaluation of computation time of all algorithms.

A) Calculations

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

$$MSE = \frac{\sum_{ij}\left(r_{ij} - x_{ij}\right)^2}{MN}$$ And

$$IEF = \frac{\left(\sum_{ij} n_{ij} - r_{ij}\right)^2}{\left(\sum_{ij} x_{ij} - r_{ij}\right)^2}$$

Where

MSE **mean square error;**
IEF **image enhancement factor;**
n **corrupted image;**
r **original image;**
M x N **size of image;**
x **restored image;**

a b c d e f

Fig 8. Results for 50% noise corrupted Barbara image (a) original image (b) 30% noise corrupted image. Restoration results of (c) SMF (d) AMF (e) DBA (f) DBUTM

Table 1
Quantitative results of various filters for 10% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	54.34	50.01	38.5	4.56
PSNR	31.78	32.54	34.52	38.2
IEF	34.5	65.54	68.2	87.8
Time	1.9	64.15	6.53	5.7

Table 2
Quantitative results of various filters for 20% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	54.34	50.01	38.5	10.5
PSNR	30.78	31.54	33.52	37.2
IEF	22.5	60.54	61.2	79.8
Time	1.9	64.15	6.53	5.7

Table 3
Quantitative results of various filters for 30% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	63.54	55.66	45.55	17.12
PSNR	29.78	30.54	31.52	36.2
IEF	21.5	57.56	59.8	78.4
Time	2.1	66.15	6.56	5.9

Table 4
Quantitative results of various filters for 40% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	72.34	60.56	50.1	30.4
PSNR	27.34	29.7	30.5	34.72
IEF	20.53	51.45	50.5	76.7
Time	2.5	67.5	7.59	6.9

Table 5
Quantitative results of various filters for 70% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	89.34	82.5	70.71	71.5
PSNR	25.89	28.56	28.65	32.78
IEF	19.53	42.45	39.5	70.8
Time	2.9	68.5	8.33	6.12

Table 6
Quantitative results of various filters for 80% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	289.5	244	124.2	118.9
PSNR	26.34	28.7	29.5	30.72
IEF	19.53	42.45	39.5	70.8
Time	2.9	68.5	8.33	6.12

Table 7
Quantitative results of various filters for 90% corrupted Barbara image

Parameter	SMF	AMF	DBA	DBUTM
MSE	8024.4	1045.2	894.24	134
PSNR	12.0	15.2	19.4	22.5
IEF	2.4	15.68	31.21	67.54
Time	2.98	69.43	9.12	6.78

IV. CONCLUSION

An efficient non-linear algorithm to remove high-density salt and pepper noise is proposed. The modified sheer sorting architecture reduces the computational time required for finding the median. This increases the efficiency of the system. The algorithm removes noise even at higher noise densities and preserves the edges and fine details. The performance of the algorithm is better when compared to the other architecture of this type.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their constructive comments that helped to improve the quality of this work

REFERENCES

[1]. W.K.Pratt, Digital Image Processing (New York: Wiley 1978).

[2]. Pitas and A.N. Venetsanopoulous, Non-linear Digital Filter principles and Applications, (Boston: Kluwer Academic publishers, 1990).

[3]. J. Astola and P. Kuosmanen, Fundamentals of Non- Linear Digital Filtering, BocRaton, CRC, 1997

[4]. N.C.Gallagher, Jr. and G.L.Wise, “A Theoretical Analysis of the Properties of Median Filters,” IEEE Trans.Acoust, Speech and Signal, vol.ASSP-29, pp.1136-1141, Dec.1981.

[5]. T.A.Nodes and N.C.Gallagher, “Median Filters: Some Modifications and their properties,” IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-30, pp.739-746, April 1987.

[6]. S.Manikandan, O.Uma Maheswari, D.Ebenezer, "An Adaptive Recursive Weighted Median Filter with Improved Performance in Impulsive Noisy Environment", WSEAS Transactions on Electronics, Issue 3, Vol.1, July2004.

[7]. Ho-Ming Lin, "Median Filters with Adaptive Length", IEEE Transactions on Circuits and Systems, Vol.5, No. 6, June 1988

[8]. T.S. Huang, G.J. Yang, and G.Y. Tang, "Fast two dimensional Median Filtering algorithm" , IEEE Transactions on Acoustics, Speech and Signal Processing,1(1995), pp. 499-502.

[9]. Pok, G. Jyh-Charn Liu, "Decision based median filter improved by predictions", ICIP 99(1999) proceedings vol 2 pp 410-413.

[10]. Srinivasan K. S. and Ebenezer D., "A New Fast and Efficient Decision- Based Algorithm for Removal of High-Density Impulse Noises", IEEE signal processing letters, Vol. 14, No. 3, pp.189 - 192,