International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue (4) July, Technovision-2014, ISSN 2249–071X

Technovision-2014: 1st International Conference at SITS, Narhe, Pune on April 5-6, 2014

Impulse Noise Suppression and Edge Preservation of Digital Images using DTBDM Algorithm

Kaveri A. Pokharkar
E &TC, VLSI & Embedded
R.M.D. Sinhgad School of Engineering, Warje, Pune-411058

Prof. K.J. Amrutkar
E &TC , VLSI & Embedded
R.M.D. Sinhgad School of Engineering, Warje, Pune-411058

Abstract **– In the field of digital image processing, two applications of great importance are noise filtering and image enhancement. Image and video signals are often corrupted by Impulse noise in the process of signal acquisition and transmission. To avoid the damage on noise-free pixels, the switching median filters are used which consists of impulse detection and noise filtering .In this paper an efficient denoising technique for removal of impulse noise is presented. This design uses a 3x3 mask on each pixel in the image in order to determine whether it is corrupted by random-valued impulse noise or not. We employ a decision-tree-based impulse noise detector to detect the noise pixels. After noise detection, the algorithm reconstructs the noisy pixel by considering the possible edges existing in the mask. Here we have implemented the design using MATLAB. The experimental results demonstrate that this method achieves excellent performance in terms of image quality.**

Keywords **– Decision Tree, Image Denoising, Impulse Noise.**

I. Introduction

Impulse noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel. Impulse noise can be classified into two types: fixed-valued impulse noise and random-valued impulse noise. The fixed-valued impulse noise is also called salt-and pepper noise where the gray-scale value of a noisy pixel is either minimum or maximum in gray-scale images. The grayscale values of noisy pixels corrupted by random-valued impulse noise are uniformly distributed in the range of [0,255] for gray scale images. There are many works on the restoration of images corrupted by impulse noise. The median filter was once the most popular nonlinear filter for removing impulse noise the main disadvantage of the standard median method is the blurring of the reconstructed image. Different remedies of the median filter have been proposed in the literature, e.g., the adaptive median filter, the multistate median filter and switching filters. In general, the efficient switching median filter consists of two parts: impulse detection and noise filtering. It locates the noisy pixel with an impulse detector, and then filters them rather than the whole pixels of an image to avoid the damage on noise -free pixels.

In addition to median filter, there are other methods like ATMBM and DWM to carry out impulse noise. Based on above basic concepts we present an adaptive decision tree based denoising method (DTBDM) for removing random-valued impulse noise. To enhance the effects of removal of impulse noise, the results of reconstructed pixels are adaptively written back as a part of input data The rest of the paper outlines the proposed decision tree based impulse detector and edge preserving image filter in section II. Implementation of the proposed method is given in section III and finally we discuss the results in section IV.

II. Decision Tree Based Impulse Detector

Here, the window size for the denoising process is 3x3. Assume the pixel to be denoised is located at coordinate (i,j) and denoted as pi,j , and its luminance value is fi,j. The mask under consideration is shown in Fig.1. We have divided the ther pixels in the window as Top Half and Bottom Half. The overall design architecture of the proposed method is shown in Fig.2. Several methods are employed in the literature for the impulse detection. Based on the existing designs we have designed three modules namely, Isolation Module (IM), Fringe Module(FM), and Similarity Module (SM). Three concatenating decisions of these modules make a decision tree.

A. Isolation Module

We use Isolation Module to make a decision whether the pixel value is in a smooth region. If the result is negative, we conclude that the pixel under consideration belongs to a noisy free area. Otherwise if the result is positive, it indicates that the pixel under consideration can be a noisy pixel or it is just situated on an edge of the object in the image. The Dataflow of the different components in Isolation Module is given in Fig. 3, Fig. 4 and Fig. 5. In order to avoid the complexity of the Design, the 3 x 3 window under consideration is divided into two different regions. Window Top Half and Window Bottom Half Where,

$$W_{Top\ Half} = \{a,b,c,d\} \quad (1)$$

$$W_{Bottom\ Half} = \{e,f,g,h\} \quad (2)$$

	j-1	j	j+1
i-1	a	b	c
i	d	f i,j	e
i+1	f	g	h

Fig1. Window (3x3 mask)

International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue (4) July, Technovision-2014, ISSN 2249–071X

Technovision-2014: 1st International Conference at SITS, Narhe, Pune on April 5-6, 2014

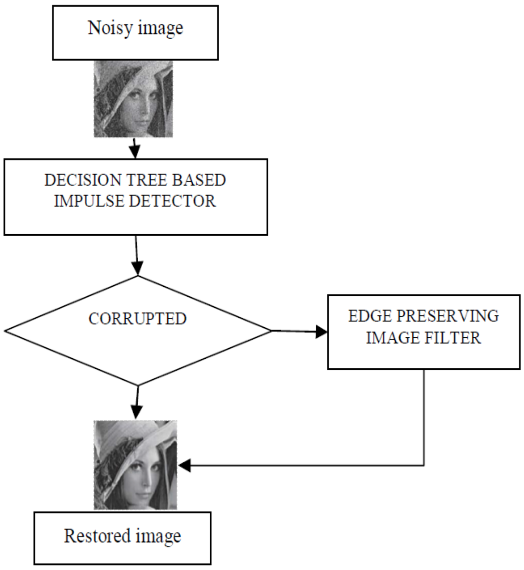

Fig.2. The dataflow of DTBDM

In the given dataflow diagrams, *THD, BHD, TH Max, TH Min, BH Max, BH Min* denotes *Top Half_difference, Bottom Half_difference, Top Half_Max, Top Half_Min, BottomHalf_Max, Bottom Half_Min* respectively.
Where,
THD=TH Max – TH Min (3)
BHD=BH Max – BH Min (4)

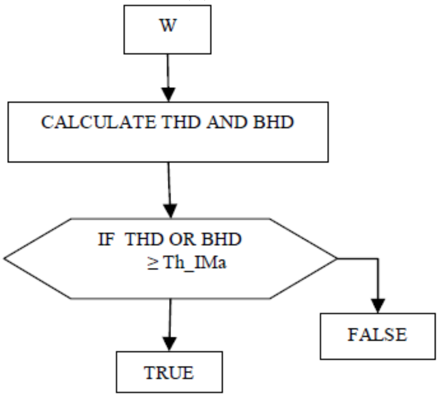

Fig.3. Dataflow of decision I in IM

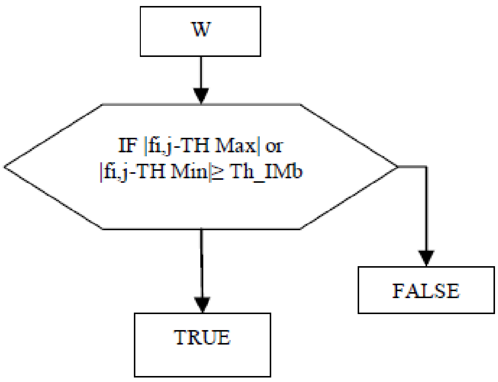

Fig.4. Dataflow of IM_Th in decision I of IM

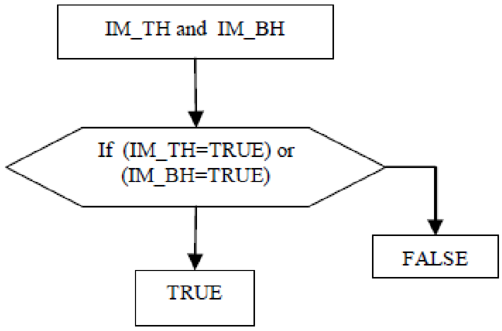

Fig.5. Dataflow of Decision II in IM

B. Fringe Module

In some cases, if the pixel is situated at the edge the Isolation Module may detect it as noisy pixel, In order to deal with this scenario; we define four directions, from E1 to E4, as shown in Fig. 6. By calculating the absolute difference between fi,j and the other two pixel values along the same direction, we can determine whether there is an edge or not.

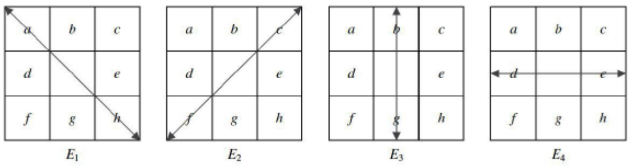

Fig.6. The Directions in The Fringe Module

The Dataflow of the Fringe Module is given in Fig. 7 and Fig. 8.

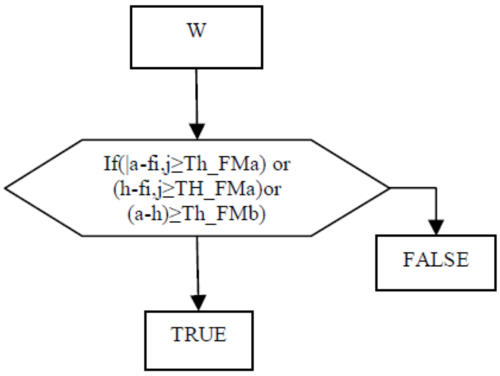

Fig.7. Dataflow of FM_E1 in IM

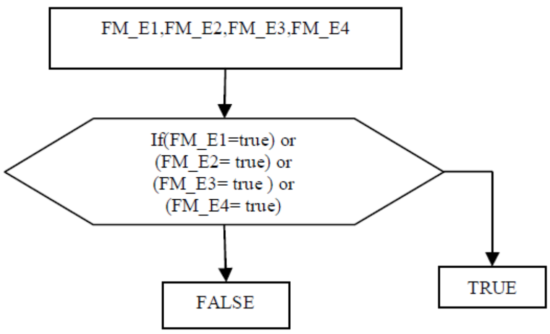

Fig.8. Dataflow of Decision III in FM

International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue (4) July, Technovision-2014, ISSN 2249–071X

Technovision-2014: 1st International Conference at SITS, Narhe, Pune on April 5-6, 2014

C. Similarity Module

The last step in impulse detection is the Similarity Module. The luminance values in mask W positioned in a noisy free area might be close. The median is always positioned in the center of the variation series, whereas the impulse is frequently located near one of its ends. Hence, if there are extreme big or small values, that shows the chance of noisy signals. Based on this perception, we sort nine values in ascending order and obtain the 4th, 5th and 6th values which are close to the median in mask W. In order to perform the operation, we need to define the following variables.

Maxij =6th in Wij+ Th_SMa (5)

Minij =4th in Wij- Th_SMa (6)

The Dataflow diagram of the Similarity module is given in Fig. 9 and Fig. 10. Calculation of Nmin is same as Nmax with some minor modifications.

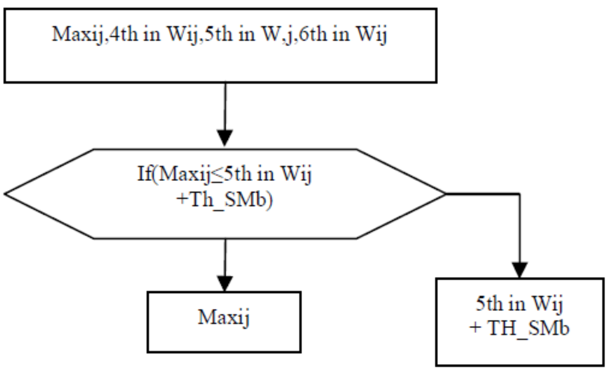

Fig.9. Dataflow of Nmax in Decision IV

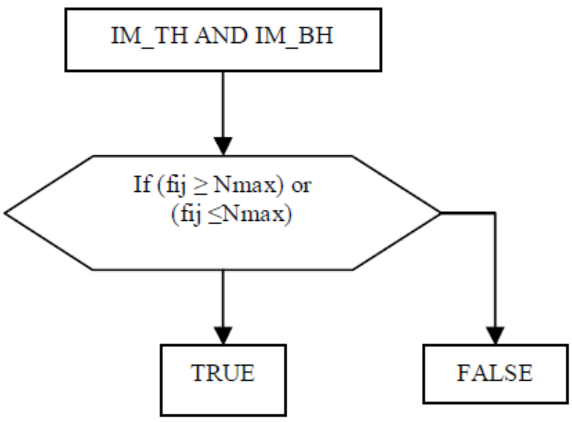

Fig.10. Dataflow of Decision IV

Threshold affects the performance of the denoising algorithms. The fixed values of the Thresholds make the algorithm simple and suitable for Hardware implementation. As per the experimental analysis and literature review, the values of *Th_IMa, Th_IMb, Th_FMa, Th_FMb, Th_SMa, Th_SMb* are 20, 25, 40, 80, 15 and 60.

III. Proposed Edge Preserving Image Filter

Edge preservation is one of the important consideration in Denoising algorithms. The similarity of the reconstructed image with the Original image mainly depends on the Edges in the Image. Here, we consider eight directional differences, D1-D8, for the reconstruction of the Noisy pixel in the image. The main idea adopted here is to avoid the pixels, which are already known affected, for the reconstruction of the pixel *Pi,j*. This is to avoid the possible misdetection of the edges. This is accomplished by using *Maxi,j* and *Mini,j*, defined in similarity module (SM), to determine whether the values of *d, e, f, g* and *h* are likel y corrupted respectively. If *d, e, f, g* and *h* are all suspected to be noisy pixels, and no edge can be processed, then the estimated value of *Pi,j* is equal to the weighted average of luminance values of three previously denoised pixels and calculated as *(a+b x 2+c)/4*. In other conditions, the edge filter calculates the directional differences of the chosen directions and locates the smallest one (*Dmin*) among them. The overall Dataflow of the Edge Preserving Image Filter is given in the Fig.11.

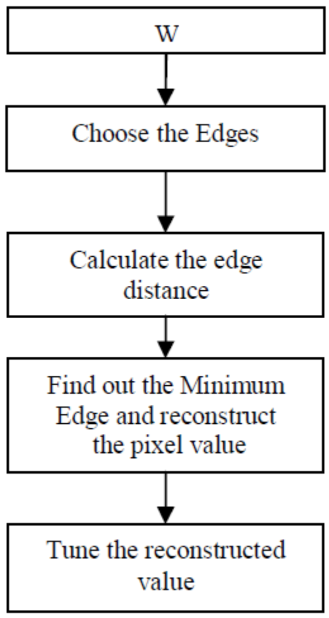

Fig.11. Edge Preserving Filter Algorithm

IV. Simulation Results

To verify the characteristics and the quality of denoised images of various denoising algorithms, a variety of simulations are carried out on the well-known 512x512 8-bit gray-scale test image: Lena and Boat. For a single test image, the corrupted versions of it are generated in Matlab environment with random-valued impulse noise at various noise densities. Then, we employ different approaches to detect impulse noise and restore the corrupted image.

Thus, we can easily compare the restored images with the source image for various denoising methods. Totally, 2 denoising methods ACWM and our method (DTBDM) are compared in terms of objective testing (quantitative evaluation) and subjective testing (visual quality) where the parameters or thresholds of these methods are set as suggested.

Table I. Comparative Results in PSNR (dB) of Images Corrupted by 5 Percent Impulses

	Comparative results	
Method/Images	Lena	Boat
ACWM	21.7450	21.3130
DTBDM	29.3952	27.3282

Table II: Comparative Results in PSNR (dB) of Images Corrupted by 10 Percent Impulses

	Comparative results	
Method/Images	Lena	Boat
ACWM	18.6770	18.2872
DTBDM	25.8969	24.4458

V. Conclusion

In this project, a new denoising algorithm for removing salt and pepper noise is presented in the paper. It can detect the impulse noise efficiently while preserving the edges very well. Although the technique has very low computational complexity, it provides superior quality of results in terms of PSNR and image quality. Currently we are working on implementations of the different directional filter methods.

References

[1] R. C. Gonzalez and R. E. Woods, Digital Image Processing. Pearson Education, Upper Saddle River, New Jersey, 2007
[2] W. K. Pratt, Digital Image Processing. New York: Wiley-Interscience, 1991.
[3] Chih-Yuan Lien, Chien-Chuan Huang, Pei-Yin Chen, Member, IEEE, and Yi-Fan Lin"An Efficient Denoising Architecture for Removal of Impulse Noise in Images" IEEE TRANSACTIONS ON COMPUTERS, VOL. 62, NO. 4, APRIL 2013
[4] S.-J. Ko and Y.-H. Lee, "Center weighted median filters and their applications to image enhancement," IEEE Trans. Circuits Syst., vol. 38, no. 9,pp. 984- 993,Sept. 1991.
[5] T. Chen and H. R. Wu, "Adaptive impulse detection using center-weighted median filters," IEEE Signal Process. Lett., vol. 8, no. 1, pp. 1–3, Jan.2001.
[6] T. Chen and H. R.Wu, "Space variant median filters for the restoration of impulse noise corrupted images," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol. 48, no. 8, pp. 784–789, Aug. 2001.
[7] W. Luo, "An efficient detail-preserving approach for removing impulse noise in images," IEEE Signal Process.Lett., vol. 13, no. 7, pp. 413-416, 2006
[8] Y. Dong and S. Xu, "A new directional weighted median filter for removal of random-valued impulse noise," IEEE Signal Process.Lett., vol. 14,2007
[9] T. Nodes and N. Gallagher, "Median filters: some modifications and their properties," IEEE Trans. ASSP, vol. ASSP-30, no. 5, pp. 739-746, Oct.1982