DECISION TREE BASED EFFICIENT IMAGE DENOISING ARCHITECTURE

Azhar.F[1], Rafeek.S[2]

[1]PG Scholar, Dept. of Electronics and Communication Engg, TKM Institute of Technology, Kollam, Kerala, India.azharkdl@gmail.com

[2]Assistant professor, Dept. of Electronics and Communication Engg, TKM Institute of Technology, Kollam, Kerala, India

***Abstract*- The principle objective of Image enhancement is to process an image so that result is more specific than original image for definite application. Digital image enhancement techniques provide choices for improving the visual quality of images. Implementing image enhancement applications on a general purpose processor may be easier, but not very time efficient due to additional constraints on memory and other peripheral devices. With the advancement in the Very Large Scale of Integration (VLSI) technology hardware implementation has become an competing alternative. Images are often corrupted by impulse noise and are recovered by a decision-tree-based impulse noise detector to detect the noisy pixels, and an edge-preserving filter to reconstruct the intensity values of noisy pixels. The efficiency of a denoising architecture comes to proper detection of impulse noise with less miss and hit count which is wrong detection of pixels. Also an adaptive technology is used to enhance the effects of removal of impulse noise. The design requires only low computational complexity and two line memory buffers when compared to other systems. Its hardware cost is low and suitable to be applied to many real-time application .**

Keywords*:*DTBDM,denoising,reconstruction,fpga

1. INTRODUCTION

Image processing is commonly used in many fields, such as medical imaging, scanning techniques, printing, license plate recognition, face recognition, and so on. It is also common that the acquired image to be affected by noise. The noise may affect the performance of the system depending on the application. So an efficient image denoising architecture is required to remove. An image may be defined as a two-dimensional function, where x and y are spatial (plane) coordinates, and the amplitude of 'f' at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the intensity values of 'f' are all finite, discrete quantities, it is a digital image. The field of digital image processing refers to processing digital images by means of a digital computer.

Impulse noise can be classified into two categories according to their nature,(a)fixed-valued impulse noise random-valued impulse noise. There have been many methods for removing salt-and-pepper noise, and some of them perform very well. The first one is also known as salt-and-pepper noise because the pixel value of a noisy pixel is either minimum or maximum value in grayscale images [0, 255]. There have been many methods for removing salt-and-pepper noise, and some of them perform very well, but it is very difficult to remove random valued impulse noise. are many algorithms proposed to correct random valued noise but some are highly complex and also slow on the basis of computation.

The efficiency of an random value noise removal architecture is based purely on the noise detection not on noise correction since there are many methods to reconstruct corrupted values. An approach is median filter but it blurs the image since both noisy and noise-free pixels are modified. To skip the damage on noise-free pixels, a switching strategy has been proposed. Impulse noise detection using different methods mainly concentrates on the clarity or efficiency of the reconstructed image, buffer size, iteration times, mask size, miss-hit count etc. The different methods uses different architecture and they vary in terms of these. The main focus of the proposed architecture is to identify the corrupted pixel value and apply the reconstruction on the same.

2. DECISION TREE BASED IMPULSE DETECTOR

The noise considered in this paper is random-valued impulse noise with uniform distribution. Here a 3×3 mask is used for image denoising. Assume the pixel to be denoised is located at coordinate (i, j) and its luminance value is named as fi,j, as shown in Figure. 1. According to the input sequence of image denoising process, it can divide other eight pixel values into two sets named as : $W_{TopHalf}$ and $W_{BottomHalf}$. They are given as

$$W_{TopHalf} = \{a,b,c,d\}$$

$$W_{BottomHalf} = \{e,f,g,h\}$$

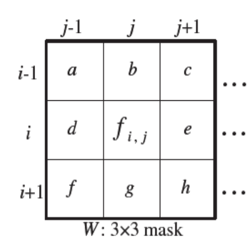

Figure 1 3x3 mask centered on f_{ij}

In order to determine whether f_{ij} is a noisy pixel, the correlations between $f_{i,j}$ and its neighboring pixels are considered. Correlation is based on degree of isolation at current pixel, determining whether the current pixel is on a fringe or comparing the similarity between current pixel and its neighboring pixels. Therefore, in the decision-tree-based impulse detector, it is necessary to design three modules named isolation module, fringe module and similarity module.

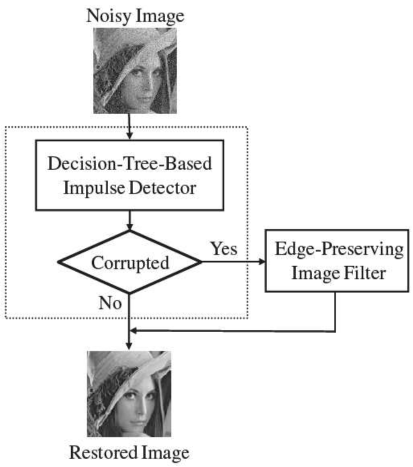

Figure 2 Data flow of DTBDM

The method uses an adaptive technology to improve the quality of reconstructed image. Figure.3 shows the architecture of the algorithm. The reconstructed pixels are adaptively stored into the line buffers. The current pixel to be denoised is fij. The architecture consists of five main blocks: line buffer, register bank, decision tree based impulse detector, edge-preserving image filter and controller. Each of them is described briefly in the following subsections.

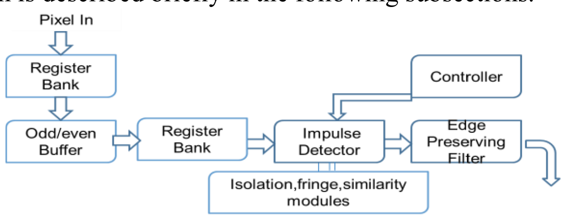

Figure 3 Data flow of DTBDM

A. Line Buffer

DTBDM adopts a 3×3 mask, so three scanning lines are needed. If fi,j are processed, three pixels from row i-1, row i and row i+1, are needed to perform the denoising process. Here, used the concept of ping-pong arrangement. With the help of four crossover multiplexers, realized three scanning lines with two line buffers. Odd-Line Buffer and Even- Line Buffer are designed to store the pixels at odd and even rows respectively, as shown in Figure. 4. To reduce cost and power consumption, the line buffer is implemented with a dual-port SRAM (one port for reading out data and the other for writing back data concurrently) instead of a series of shifter registers.

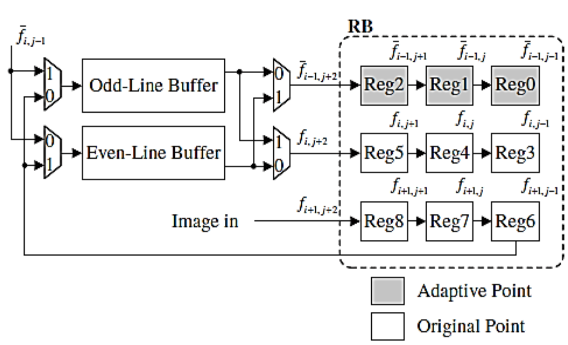

Figure.4 Line Buffer

B. Register Bank

A 3x3 register bank is used to store the pixel window in accordance with the adaptive technology. The image is read from the top left to bottom right through the window. Hence the buffer size is smal when compared to other algorithm.

C. Isolation module

The isolation module is used to decide whether the pixel value is in a smooth region. If the result is negative, this concludes that the current pixel belongs to noisy-free. Otherwise, if the result is positive, it means that the current pixel might be a noisy pixel or just situated on an edge. The pixel values in a smooth region should be close or locally slightly varying. The differences between its neighboring pixel values are small. If there are noisy values, edges or blocks in this region, the distribution of the values is different. The difference between it and its neighboring pixel value is large .The architecture for bottom half is same as top half. Figure 5 shows architecture of isolation module. The equations are as:

$$\text{TopHalf\_diff} = \text{TopHalf\_max} - \text{TopHalf\_min}$$

$$\text{BottomHalf\_diff} = \text{BottomHalf\_max} - \text{BottomHalf\_min}$$

$$\text{Decision I} = \begin{cases} \text{true}, & \text{if (TopHalf\_diff} \geq \text{Th\_IMa)} \\ & \text{or(BottomHalf\_diff} \geq \text{h\_IMa)} \\ \text{False}, & \text{otherwise} \end{cases}$$

$$\text{IM\_TopHalf} = \begin{cases} \text{True, if (} \lvert f_{ij} - \text{TopHalf\_max} \rvert \geq \text{Th\_IMb)} \\ \quad \text{or(} \lvert f_{ij} - \text{TopHalf\_min} \rvert \geq \text{Th\_IMb)} \\ \text{False, otherwise} \end{cases}$$

$$\text{Decision II} = \begin{cases} \text{True}, & \text{if (IM\_TopHalf = True)} \\ & \text{or (IM\_BottomHalf = True)} \\ \text{False}, & \text{otherwise} \end{cases}$$

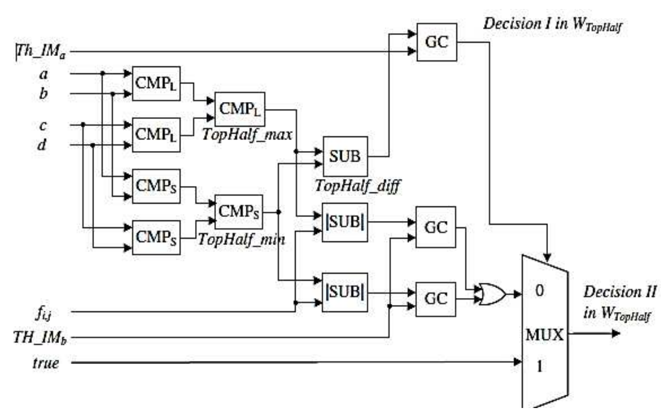

Figure.5 Architecture of IM ($W_{TopHalf}$)

D. Fringe module

The fringe module is used to confirm the result. If the current pixel is situated on an edge, the result of fringe module will be negative (noisy-free); otherwise, the result will be positive. If $f_{i,j}$ has a great difference with neighboring pixels, it might be a noisy pixel or just situated on an edge. In order to deal with this case, define four directions, from E1 to E4, and take direction E1 for example. By calculating the absolute difference between $f_{i,j}$ and the other two pixel values along the same direction respectively, it is easy determine whether there is an edge or not.

The directions are as:

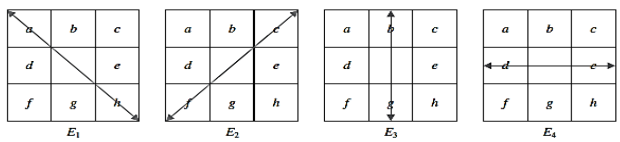

Figure. 6. Four directions in DTBDM

The FM is composed of four small modules, from FM_1 to FM_4, and each of them is used to determine its direction. Since E1 is the direction from a to h (Figure.6), the relation between a, h and $f_{i,j}$ must be referenced. The three |SUB| units are used to determine the absolute differences between them. The GC is described in the above section and the NOR gate is used to generate the result of FM_E1. If the result is positive, confirms that $f_{i,j}$ is on the edge E1 and regard it as noise-free.

The detailed equations are as:

$$FM\_E_1 = \begin{cases} \text{False}, & \text{if } (|a - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|h - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|a - h| \geq Th\_FMb) \\ \text{True}, & \text{otherwise} \end{cases}$$

$$FM\_E_2 = \begin{cases} \text{False}, & \text{if } (|c - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|f - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|c - f| \geq Th\_FMb) \\ \text{True}, & \text{otherwise} \end{cases}$$

$$FM\_E_3 = \begin{cases} \text{False}, & \text{if } (|b - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|g - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|b - g| \geq Th\_FMb) \\ \text{True}, & \text{otherwise} \end{cases}$$

$$FM\_E_4 = \begin{cases} \text{False}, & \text{if } (|d - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|e - f_{ij}| \geq Th\_FMa) \\ & \text{or } (|d - e| \geq Th\_FMb) \\ \text{True}, & \text{otherwise} \end{cases}$$

Figure 7 and Figure 8 shows the architecture of fringe module directional module and fringe module.

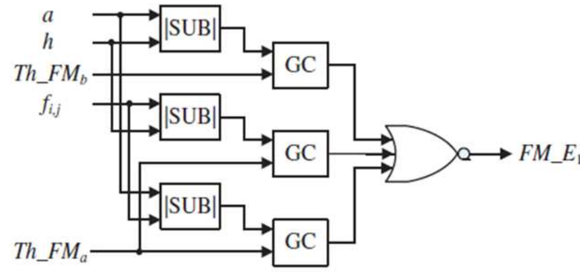

Figure.7. Architecture of FM_1

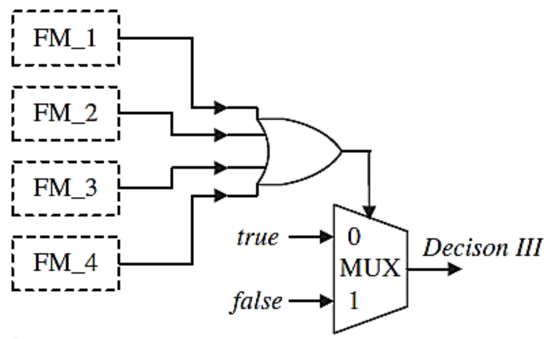

Figure.8. Architecture of FM module

The detailed equation of fringe module is given below

$$\text{Decision III} = \begin{cases} \text{False}, & \text{if } (FM\_E_1) \text{ or } (FM\_E_2) \\ & \text{or } (FM\_E_4) \text{ or } (FM\_E_3) \\ \text{True}, & \text{otherwise} \end{cases}$$

E. Similarity Module

The last module is similarity module. The luminance values in mask W located in a noisy-free area might be close. The median is always located in the center of the variation series, while the impulse is usually located near one of its ends. Hence, if there are extreme big or small values, that implies the possibility of noisy signals. According to this concept, sort nine values in ascending order and obtain the 4^{th}, 5^{th} and 6^{th} values which are close to the median in mask W. The 4^{th}, 5^{th} and 6^{th} values are represented as 4^{th}inWi,j, MedianInWi,j and 6^{th} inWi,j.

Define Maxi,j and Mini,j as

$$Max_{i,j} = 6^{th} \text{ in } W_{ij} + Th\_SM_a$$
$$Min_{ij} = 4^{th} \text{ in } W_{ij} - Th\_SM_a$$

Maxi,j and Mini,j are used to determine the status of pixel fi,j. However, in order to make the decision more precisely, modify N_{max} and N_{min} as.

$$N_{max} = \begin{cases} Max_{i,j}, \text{if}(Max_{i,j} \leq MedianInW_{i,j} + Th\_SMb) \\ MedianInW_{i,j} + Th\_SMb \text{ , otherwise.} \end{cases}$$

$$N_{min} = \begin{cases} Min_{i,j}, \text{if } (Min_{i,j} \geq MedianInW_{i,j} - Th\_SMb) \\ MedianInW_{i,j} + Th\_SMb, \text{ otherwise.} \end{cases}$$

Finally, if fi,j is not between Nmax and Nmin, it concludes that fi,j is a noise pixel. Edge-preserving image filter will be used to build the reconstructed value. Otherwise, the original value fi,j will be the output. Figure. 3.16 show the architecture of SM after obtaining the 4^{th}, 5^{th} and 6^{th} values in mask W. Figure. 10 show the architecture of SM after obtaining the 4^{th}, 5^{th} and 6^{th} values in mask W.

The equations are :

$$\text{Decision IV} = \begin{cases} \text{True}, & \text{if } (F_{ij} \geq N_{max}) \text{ or } (F_{ij} \leq N_{min}) \\ \text{False}, & \text{otherwise} \end{cases}$$

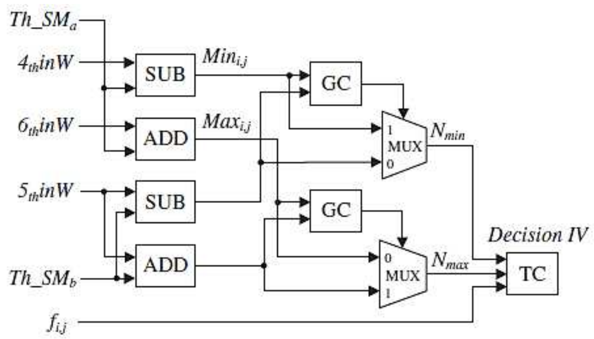

Figure 10. Architecture of Similarity Module

E. 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, consider eight directional differences, D1-D8, for the reconstruction of the noisy pixel in the image .The main technique adopted here is to avoid the pixels, which are already known affected, for the reconstruction of the pixel Fi,j. This is to eliminate 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 likely corrupted respectively.

If d, e, f, g and h are all supposed to be noisy pixels, and no edge can be used, then the estimated value of fi,j is equal to the weighted average of pixel values of three previously denoised pixels and calculated as *(a+b x 2+c)/4*. In other conditions, the edge filter finds the directional differences of the selected directions and locates the smallest one (D_{min}). The architecture of minED generator and average generator is shown in Figure 11 and 12 shows. The following equation shows the directional differences and estimated pixels respectively.

$$\begin{aligned} D1 &= |d-h| + |a-e| \\ D2 &= |a-g| + |b-h| \\ D3 &= |b-g| \times 2 \\ D4 &= |b-f| + |c-g| \\ D5 &= |c-d| + |e-f| \\ D6 &= |d-e| \times 2 \\ D7 &= |a-h| \times 2 \\ D8 &= |c-f| \times 2 \end{aligned}$$

$$\text{New } f_{ij} = \begin{cases} (a+d+e+h)/4, & \text{if Dmin} = D1 \\ (a+b+g+h)/4, & \text{if Dmin} = D2 \\ (b+g)/2, & \text{if Dmin} = D3 \\ (b+c+f+g)/4 & \text{if Dmin} = D4 \\ (c+d+e+f)/4, & \text{if Dmin} = D5 \\ (d+e)/2, & \text{if Dmin} = D6 \\ (a+h)/2, & \text{if Dmin} = D7 \\ (c+f)/2, & \text{if Dmin} = D8 \end{cases}$$

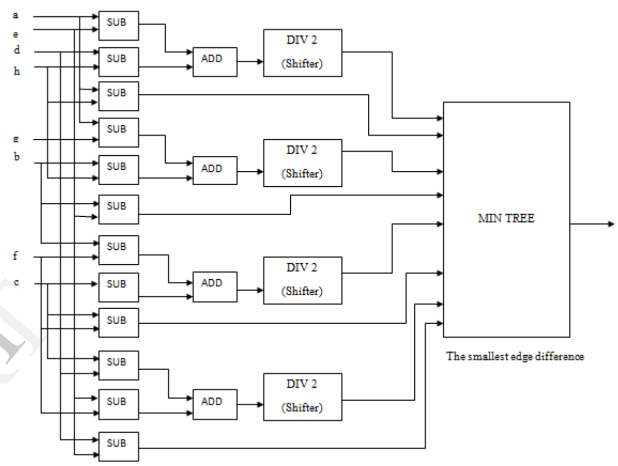

Figure 11. Architecture of minED Module

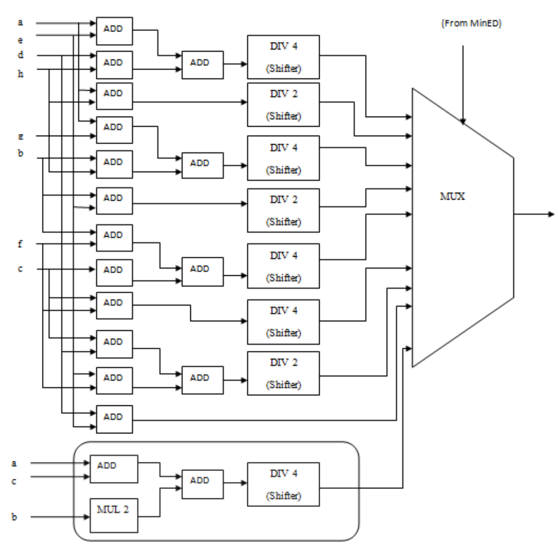

Figure 12. Architecture of average generator

Also a controller is needed to synchronize the whole operation from image input to the reconstructed pixel output. The controller controls the buffer and register bank and also pass the pixel value to each module for processing.

3. SIMULATION RESULTS

The design entry is modeled using VHDL in Xilinx ISE Design Suite 12.1 and the simulation of the design is performed using ISim SE to validate the functionality of the design. The output of each module is focused on the decision of the module which in turn decides whether pixel is noisy or not.

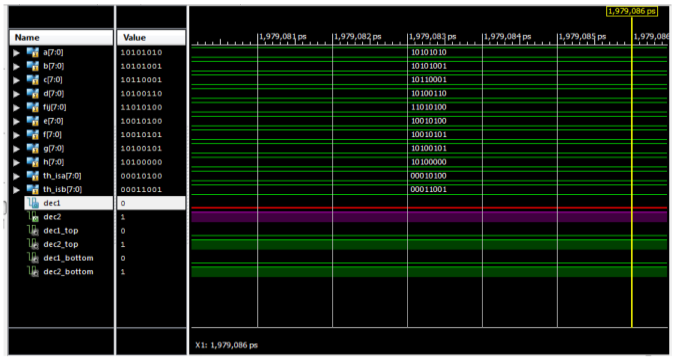

Figure: 13 Isolation module.

Figure 13 shows the output of isolation module which shows that decision 2 is high even when decision 1 is low.This shows that there may be noisy pixel or it is an edge.Fringe module is used to find an edge.

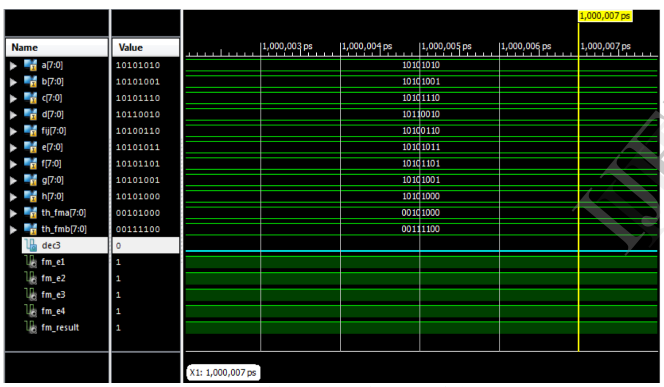

Figure: 14 Fringe module

Figure 14 shows the output of fringe module which shows that decision 3 is low .Thus the current pixel is an edge and is free from noise. If the result is low then next modules are skipped.

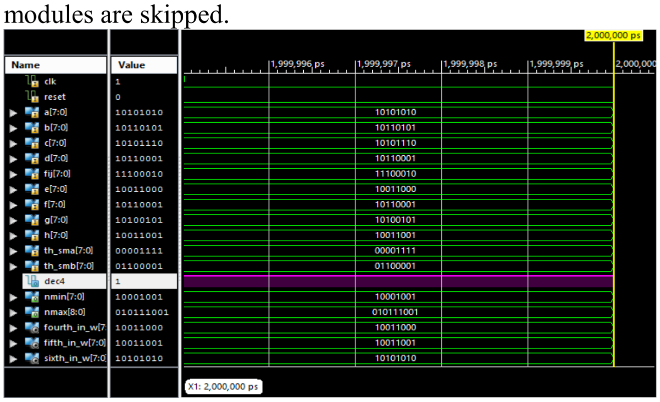

Figure: 15 Similarity module.

Figure 15 shows the output of similarity module which shows that decision 4 is high .Thus the current pixel is an noise. If the result is high then next modules are activated and starts image reconstruction for the corresponding pixel.

REFERENCES

[1] Chih-Yuan Lien, Chien-Chuan Huang and Yi-Fan Lin, "An Efficient Denoising Architecture for Removal of Impulse Noise in Images," *IEEE Transactions On Computers*, vol. 62.no 4, pp. 631-643,April 2013.

[2] Sherin N R, Riyas A N "Denoising Techniques In VLSI For Impulse Noise Removal"*International Journal of Engineering Research & Technology (IJERT),*Vol. 2 Issue 4, April – 2013.

[3] Tripti Jain, Prashant bansod, C. B. Singh Kushwah and mayenk Mewara,"ReconFigureurable Hardware for Median Filtering for image Processing Application", 3rd *International Conference on Emerging Trends in Engineering and Technology,*IEEE,2012.

[4] P.Neelavathi .P,' A New Detection Statistics for Removal of Impulsive Noise' *International journal of computer application,* issue2, volume 1,2012

[5] Vinoth Kannan. A, Priyameenakshi .K, "A Low-Power VLSI Implementation for Efficient Removal of Impulse Noise using CLA"*International Journal of Electronics and Computer Science Engineering*,ISSN- 2277-1956,2012.

[6] Irphan Ali Shaik, Mirza shafi shahsavar, K.J.Silva Lorraine, Ajesh kumar vishwanadham, "Impulse Noise Detection and Filtering Based On Adaptive Weighted Median Filter", International Journal of Engineering and Science ISBN: 2319-6483, Vol. 1, Issue 8, 2012.

[7] Geoffrine Judith.M.C1 and N.Kumarasabapathy,"study and analysis of impulse noise reduction filters", *An International Journal(SIPIJ),* Vol.2, No.1, March 2011.

[8] Mandal J. K and Somnath Mukhopadhyay 'Image filtering using all neighbor directional weighted pixels:optimization using particle swarm optimization Signal & Image Processing : *An International Journal (SIPIJ)* Vol.2, No.4, December 2011.

[9] J.Harikiran, B.Saichandana, B.Divakar, "Impulse Noise Removal in Digital Images,"*International Journal of Computer Applications, Volume 10– No.8, November 2010.*

[10]Behrooz ghandeharian, Hadi Sadoghi Yazdi and Faranak Homayouni, 'Modified adaptive center eighted median filter for uppressingimpulsive noise in images' *International Journal of Research and Reviews in Applied Sciences,* ISSN: 2076-734X, EISSN: 2076-7366, Volume 1, Issue 3, 2009 .

[11]P. Y. Chen and C. Y. Lien, "An Efficient Edge-Preserving Algorithm for Removal of Salt-and-Pepper Noise," *IEEE Signal Process. Lett.*, vol. 15, pp. 833-836, Dec. 2008.

[12]Nemanja I. Petrovi´c and Vladimir S. Crnojevi , 'Evolutionary Tree-Structured Filter for Impulse Noise Removal' *International conference on Advanced Concepts For Intelligent Vision Systems*, ISBN:3-540-3,978-3-540-44630-9, 2006.

[13]P.E. Ng and K.K. Ma, "A switching median filter with boundary discriminativenoise detection for extremely corrupted images," *IEEE Trans. Image Process.,*vol. 15, no. 6, pp. 1506–1516, Jun. 2006.