Design and VLSI Implementation of High Performance Median Filter

Antima Bajpai[1], Shweta Agrawal[2] and Prashant Badal[2]
[1]Research scholar,[2] Assistant Professor,
Dept of Electronics and Comm., SRCEM Banmore, Morena, India

***Abstract*— The exponential use of the portable devices exhibiting multimedia applications is the result of advancement in the VLSI technology. Image and video processing applications are the primarily used in these devices which demands higher performance processing units. As the noise may introduced anywhere within image, a filter is the essential part of the image processing units. As the Salt and pepper noise is occurred due to incorrect computations with there is special kind of filter used to remove this noise is known as median filter. In this paper different algorithms for the median filters are explored and analyzed. A novel median filter algorithm is proposed in this paper that reduces the complexity of the filter significantly. The proposed filters is implemented and simulated with benchmark inputs. The simulation results show that proposed median filter reduces area and delay by 5.9% and 48.8% respectively over the conventional median filter architecture.**

***Keywords*— Digital Signal Processing (DSP), Median Filter, Image Processing, Integrated Circuits, VLSI, Low Power Design.**

I. INTRODUCTION

With the advancement in the very large scale integration (VLSI) technology, there is the exponential usage of the portable devices exhibiting multimedia application, in the recent years. This integration of huge functionality is imposing several challenges to the VLSI designer as increasing functionality increases the complexity which in turn increases failure probability. Further as area, power and delay are main area of concern and at the same time it is very difficult to achieve optimal value of these parameters. Image processing application is the prime component of the multimedia applications. The image processing tasks are the area and performance inefficient and demands architecture that exhibits low computational complexity. Thus, a high performance image processing is the requirement of modern devices to work with the real-time applications.

In the VLSI design the area, power and delay are the three primary parameter that designer wants to improve. But these parameters form a tradeoff triangle i.e. improving one parameter damages the other. The conventional approach of VLSI design provides accurate results i.e. follow the given specification. However, in real scenario it is not always required. There are many applications where minor error can be tolerated called as error tolerant applications.

The multimedia applications such as image/video processing are error tolerant applications. In these applications, small error is tolerable as these applications produce output for human consumption [2] as human have limited visual perception. Along with the multimedia applications, several other applications such as that exhibit probabilistic computations and iterative computation also exhibits error tolerance. Thus, the accurate designs for these applications are the waste of power/area and performance. For these applications, accuracy can be seen as the new design parameter that can be traded to improve all design parameters.

Filtering is the process that is commonly used most of the image processing cores as noise may occur anywhere in the image. From different types of noise, Salt and pepper (SAP) noise is the noise that occurs due to incorrect computations. Median filter is the most commonly used filter for reducing the SAP noise. In median filtering noisy pixel is replaced by the median of its neighbour. As the noise density increases, the conventional median filter fails to provide image of good quality. Further large size image sub-matrix is required that increases the computational complexity. There are different efficient algorithms for median filtering is developed such as modified decision based trimmed median filter. Although, these algorithms remove high density SAP noise, their complexity is large. There is demand of low complexity median filter design.

This paper proposes a novel low complexity median filter algorithm is developed. The existing and proposed median filter designs are implemented and then evaluated using design and error metrics and compared with the well-known existing architectures. The simulation results shows that proposed median filter provides significantly improved design metrics and simultaneously provides acceptable quality.

II. LITERATURE REVIEW

Noise is unwanted signal which corrupts the desired information in the image. The process of denoising is nothing but eliminating the effect of noise within image to improve the quality. There are several types of noises that may occur in an image. The noises are may have uniform, Gaussian distribution or it have impulsive nature. An example of the impulse kind of noise is salt and pepper noise. Moreover the noise can be classified based on their effect on the information data i.e. it can be either multiplicative or additive. An additive noise can be represented as

$$In(x,y) = I(x,y) + n(x,y)$$

Whereas, multiplicative noise can be modelled as

$$In(x,y) = I(x,y) * n(x,y)$$

Where I(x,y), n(x,y), In(x,y) and (x,y) represents the original image signal, noise signal, corrupted image signal and coordinate at which the noise is introduced respectively. The most of the noise affects the image signal at the pixel level and therefore signal manipulation is done at pixel level and image morphing is application where it is commonly employed. The noise mainly affects the brightness of the original signal. The following subsections provide detailed discussion on different noises where it starts with the Gaussian noise followed by “Salt and pepper”, Speckle and Brownian noise.

2.1 Gaussian Noise

This is a noise which exhibits Gaussian distribution over the location at which noise is introduced therefore it is called as Gaussian noise. The pixel at which noise is cantered will be affected more as compared to the farther pixels noise intensity reduces exponentially. Its distribution of probability can be represented by a bell shaped graph. The mathematical model of the Gaussian noise can be represented by the Equation 1 given below.

$$F(g) = \frac{1}{\sqrt{2\pi\sigma^2}} \varepsilon^{-(g-m)^2 / 2\sigma^2} \qquad (1)$$

Where, m, g and σ represent the mean or average, gray level and the standard deviation of the noise respectively.

2.2 Salt and Pepper Noise

The Salt and pepper noise is kind of noise which exhibits maximum error in the original signal i.e. it either makes the pixel value to 0 or 255. This kind of error occurred in the image due to error in the data transmission. Let the probability of the error is a and b as shown in Fig. 1. This minimum or maximum setting of input image value is called as Salt and pepper noise due the fact that maximum value will be represented as white point while the minimum value will be represented by the black pixel. The SAP noise may also occurs due to the faulty memory locations or improper digitization process. The image containing SAP noise is shown in Fig. 2.

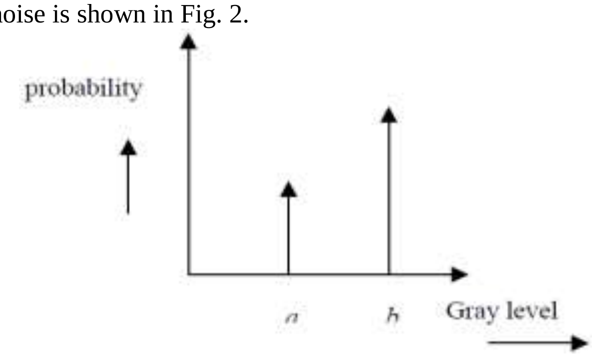


Figure 1: PDF for salt and pepper noise

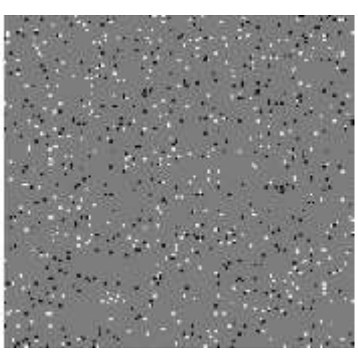


Figure 2: Salt and pepper noise

2.3 Speckle Noise

It is multiplicative in nature i.e. it affects the signal value severely. The prime sources of this noise the coherent returns of randomness in the coherent system such as synthetic aperture radar imagery, LASER and acoustic etc. In a fully developed speckle noise, properties of the multiplicative noise are present.

2.4 Brownian Noise

Brownian noise also called flicker noise exhibits inverse relationship with the frequency i.e. its magnitude is severe at lower frequencies compared to the higher frequencies. The Brownian motion represents the behaviour of the 1/f noise. A stochastic process which is non-stationary and exhibiting normal distribution is the Brownian noise. Mathematically the 1/f noise can be achieved by integrating the while noise.

2.5 Median filter: principle and performance

A median filter is classified under nonlinear filters dissimilar to the mean filter. It follows same moving window principle as mean filter. In this filter kernel size of 3x3, 5x5, or 7x7 are consider at once and median of the sub-image matrix of kernel size is taken as the output. In other words, the pixels of the selected window sizes are sorted and the central element is considered as the output of the median filter as compared to mean filter where the mean of the selected window is taken as output. The example shown in Fig. 3 illustrates that window size of 3x3 exhibits median value of 124 which will replace the central noise pixel (red colour) of value 255. Thus the median filter can easily eliminate the effect of noise as the noisy pixels will always fall on the boundaries.

122	124	125	129	139
121	123	125	126	134
117	119	255	124	133
118	114	148	122	132
110	115	109	119	129

114 119 122 123 124 125 126 148 255

Figure 3: Median filtering concept

The median filter provides better filtering compared to the mean filter as the noisy pixel does not affects the neighbour pixels. Further, in the median filter, the noisy pixel is replaced by the one of its neighbour pixels; it does not create false value which cannot occur in the image. Therefore, the median filter provide image which exhibits good edge characteristics over the mean filter which generates new pixels from the mean of its neighbours.

To do median filtering on an image, a processing tool box is provided in MATLAB i.e. **medfilt2()** function. The function takes input image and the window size as input parameters. The image can be loaded into the variable using **imread()** function. The noise in this image can be added to make noisy image by using inbuilt function **imnoise()**. This noisy image can be filter using median filter function **medfilt2()** existing in the MATLAB toolbox. The function **imnoise()** have feature to allow noise of different mean and standard deviation while **medfilt2()** can filter image with different kernel size. Fig. 4 shows the filtered image using median filter. From the filter image it can be seen that edges in the image are not spoiled and are very sharp.

(a) Input to median filter (b) Output from median filter

Figure 4 Cameraman image with (a) Salt and pepper noise (b) filtered image

The major limitation of the median filter is the poor output image quality in the presence of large noise. The median of a given kernel will always a noisy pixel if the noise is very high. To achieve high image quality even under present of high noise, different decision based median filter algorithm and their architectures are developed in the literature. Following section discussed each of these techniques in detail.

Modified decision based asymmetric trimmed median filter (MDBATMF) algorithm

The MDBATMF algorithm first detects the noisy pixels and if the entire pixels are noisy then the selected window mean will be the resulting pixels otherwise the algorithm eliminates the noisy pixels and performs the median operation on the remaining pixel. For example, consider that all the pixels of the selected window are noisy as shown below. In this case mean of the all elements are computed and the pixel to be processed will be replaced by this mean value.

$$\begin{bmatrix} 0 & 255 & 0 \\ 0 & <255> & 255 \\ 255 & 0 & 255 \end{bmatrix}$$

Where <> arrow represents the pixel to be processed.

It may be possible the pixel to be processed in not noisy while the selected window contains several noisy pixels, in this can, pixel will not be processed and its value will remain same.

$$\begin{bmatrix} 67 & 43 & 70 \\ 55 & <90> & 79 \\ 66 & 85 & 81 \end{bmatrix}$$

It may be possible that pixel is noisy and not all the elements of the selected window are noisy in this can noisy pixels will be eliminated and the median of the noise-free pixels will be calculated that will replace the noisy pixel.

$$\begin{bmatrix} 78 & 90 & 0 \\ 120 & <0> & 255 \\ 97 & 73 & 255 \end{bmatrix}$$

Since the proposed algorithm is eliminating the noisy pixels while computing the median, the output of the median filter will be more accurate over the conventional approach

III. PRAPOSED WORK

The section provides detailed discussion on the proposed energy efficient median filter. It starts with the proposed algorithm followed by the proposed architecture.

Proposed median filter algorithm

The pseudo code of the proposed median filter is given below. The proposed algorithm works as follows: First it eliminates the noisy pixels i.e. the pixels which are having value either 0 or 255 from the given input window. The noise free value is computed from the remaining elements of the window that may fall under to conditions: a) all the elements of the window may be corrupted and the remaining window is the null vector, b) the remaining window has some elements in it. In the first case since all the elements are noisy, the output value is computed by computed median of the all elements of the window which provides nearly average value of the pixels. Whereas in the second case some elements are present in the window are sorted. In the proposed algorithm, the comparison of the elements for sorting is done in an efficient manner by comparing only four MSB bits. This consideration of only four MSB bits significantly reduces the cost of the comparator used in the sorting network. Finally, the median of sorted is computed as the output noise free pixel.

Pseudo code: Proposed median filter

Input: IW % Input window size of 3x3
Output: Po % Output noise free pixel
[1]. IW_{NF} = Eliminate noisy pixels from IW
[2]. If (IW_{NF} == Null)
Po = mean (IW)
[3]. Else
Sort IW_{NF} by comparing only four MSB bits
Po= median of the IW_{NF}

The block diagram of the proposed median filter as shown in Fig. 5 consists of the image sub-matrix extractor that extract sub-image matrix of size given by the window. In the proposed filter window of size 3x3 is considered.

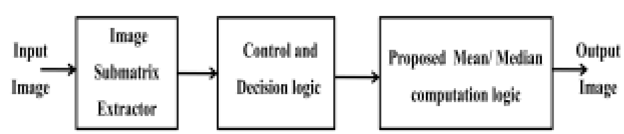


Figure 5: Block diagram of the proposed median filter.

As the four bit comparator is used in the proposed control logic to compare and swap the pixel for sorting, it significantly reduces the area requirement over the existing median filter architectures.

IV. EXPERIME NTAL RESULT & ANALYSIS

In order to evaluate the quality metrics [19, 20], MATLAB tool is to model the proposed and existing architectures of the median filters. These implemented designs on MATLAB. The noisy images are created and then filtered via proposed and existing median filters. The filtered images are extracted and compared. On the other hand, to evaluate the design metrics designs are coded in Verilog HDL, implemented on Xilinx Vertex XC7VX330T and simulated on ModelSim EDA tool. Finally, the design metrics such as area, power and delay are extracted for the proposed and existing designs and compared.

4.1 Quality and Design Metrics

Various quality and design parameters are used to evaluate the design. This subsection introduces the different quality parameters which are used in our design.

4.1.1 Mean Square Error (MSE)

The MSE for an input image I and noisy output image K, is defined by the expression given below.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - k(i,j)]^2 \quad (9)$$

Where, variables m and n represent the number of row and column of the image.

4.1.2 Peak Signal to Noise Ratio (PSNR)

The peak-signal-to-noise-ratio is the parameter used widely in image/video processing applications to quantify the amount of the noise present in the image and it is equal to the maximum signal power to the noise power. The mathematical expression that computes the PSNR in decibel is given by the equation below.

$$PSNR_{db} = 10.\log_{10}(\frac{Sig_I^2}{MSE}) \quad (10)$$

Where, Sig_I reflects the maximum signal value which for an image is 255.

4.2 Simulation results on MATLAB

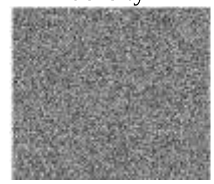
To evaluate the quality metrics, in the original image, salt and pepper noise is first introduced with different noise density using MATLAB inbuilt function. The noisy images having different noise density are shown in Figure 6 while the filtered images using various filters are shown in Figure 7.

(a) Original image

(b) Image with 2% noise density

(c) Image with 20% noise density

(d) Image with 50% noise density



(e) Image with 100% noise density

Figure 6: Original and noisy images with varying noise density

(a) Filtered via CMF with 20% noise

(b) Filtered via CMF with 50% noise

(c) Filtered via MDBTMF with 20% noise

(d) Filtered via MDBT-MF with 50% noise

(e) Filtered via PMF with 20% noise

(f) Filtered via PMF with 50% noise

Figure 7: Filtered Lena images using CMF (a), (b), MDBTMF (c), (d) and proposed MF (e), (f).

The error between the original image and the filtered output image is extracted for different image median filters as shown in Table 1.

Table 1: Comparison error metrics for different median filter architecture

Metrics	CMF		MDBTMF		PMF	
	20%	50%	20%	50%	20%	50%
PSNR	85.81	77.01	88.44	84.44	88.46	85.08
SSIM	0.938	0.314	0.980	0.934	0.980	0.953

The simulation results of the proposed filter show that proposed filter provides improved quality metrics over the existing even at higher noise immunity. It can be observed that proposed MF provides 8.07dB and 0.64dB more PSNR over conventional and MDBTMF respectively at 50% noise density as shown in Figure 5.4. Similarly it can also observed that proposed median filter provides much higher SSIM metrics over the conventional median filter architecture as shown in Figure 8.

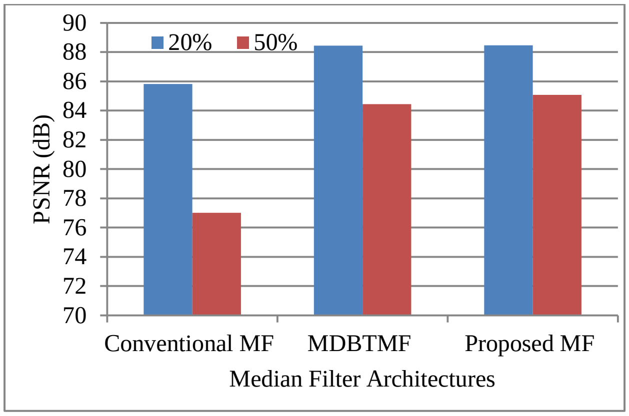


Figure 8: Comparison of PSNR for various MF at different noise density

It can be concluded from the quality metrics and the image obtained after filtering is that the proposed median filter performs better noise filter over the existing median filter architectures.

5.5 Simulation Results on FPGA

To compute the hardware results of the proposed median filter, all the designs are implemented in Verilog and processed through Xilinx ISE tool chain. These designs are simulated for functional verification. Further, the designs are synthesized and are implemented in FPGA (Spartan 6). The post-synthesis results for the proposed and existing median filters are tabulated in Table 2. It can be observed that proposed median filter requires small number of Slice for implementation over the existing architectures.

Table 2: Illustration of design metrics for different median filters

Metrics		**CMF**	**MDBTMF**	**PMF**
Area	**#Slice**	94	136	88
	#LUTs	2725	1297	1220
Delay (nS)		20.6	27.5	14.08
Frequency (MHz)		48.51	36.36	71.02

It can be observed that proposed median filter requires 35.29% reduced slices and 5.9% reduced LUTs over the MDBUTM. Further the proposed median filter design also reduces delay by 48.8% over the best known median filter architecture as shown in Figure 9.

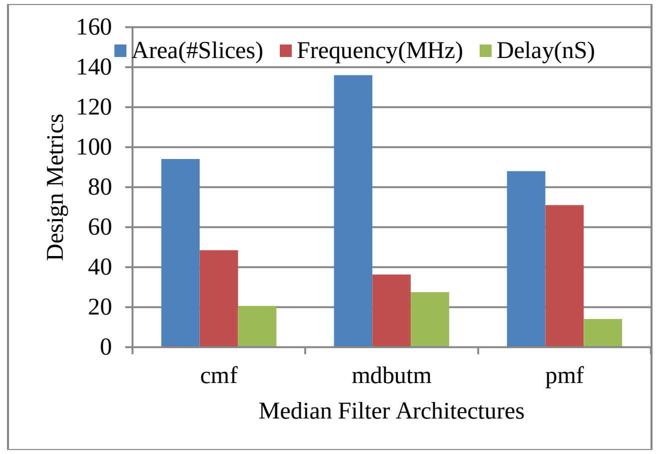


Figure 9: Comparison of design metrics for various median filter architectures

It can be observed from quality and design metrics that proposed median outperform over the existing well known median filter architectures.

V. CONCLUSION

The median filter is commonly used to reduce the salt and pepper noise compared to the mean filter. In this work existing techniques of the median filters are reviewed and analyzed. A novel low complexity median filter algorithm is proposed in this paper. The simulation results show that proposed median filter algorithm not only provides good quality metrics but also exhibits improved design metrics over the existing architectures. The simulation results on FPGA shows that the proposed median filter provides more than **5.9%** reduction in area over the existing. Further the proposed filter also shows reduces delay by **48.8%** over the best-known designs.

REFERENCES

[1] V. Caselles, J.-M. Morel, and C. Sbert, “An axiomatic approach to image interpolation,” in Image Processing, 1997. Proceedings, Int. Conference on, vol. 3, Oct 1997, pp. 376–379.

[2] Karaman, Mustafa, Levent Onural, and Abdullah Atalar. "Design and implementation of a general-purpose median filter unit in CMOS VLSI." *Solid-State Circuits, IEEE Journal of* 25.2 (1990): 505-513.

[3] Arce, Gonzalo R., Peter J. Warter, and Russell E. Foster. "Theory and VLSI implementation of multilevel median filters." *Circuits and Systems, 1988., IEEE International Symposium on*, 1988.

[4] Diaz-Sanchez, Alejandro, and Jaime Ramirez-Angulo. "Analog VLSI weighted median filters." *Circuits and Systems, 2000. Proceedings of the 43rd IEEE Midwest Symposium on*. Vol. 1. IEEE, 2000.

[5] Hiasat, Ahmad A. "Semi-custom VLSI chip implementation of a new two-dimensional separable median filtering algorithm." *Electronics, Circuits and Systems, 2001. ICECS 2001. The 8th IEEE International Conference on*. Vol. 2. IEEE, 2001.

[6] Oflazer, Kemal. "Design and implementation of a single-chip 1-D median filter." *Acoustics, Speech and Signal Processing, IEEE Transactions on*31.5 (1983): 1164-1168.

[7] Karaman, Mustafa, Levent Onural, and Abdullah Atalar. "Design and implementation of a general purpose VLSI

median filter unit and its applications." *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*. IEEE, 1989.

[8] Chen, Chun Te, et al. "Design and VLSI implementation of real-time weighted median filters." *Circuits and Systems, 1994. APCCAS'94, IEEE Asia-Pacific Conference on*. IEEE, 1994.

[9] Hiasat, Ahmad A. "Semi-custom VLSI chip implementation of a new two-dimensional separable median filtering algorithm." *Electronics, Circuits and Systems, ICECS 2001. The 8th IEEE International Conference on*. Vol. 2 IEEE, 2001.

[10] Burian, Adrian, and Jarmo Takala. "VLSI-efficient implementation of full adder-based median filter" *IEEE International Symposium on Circuits and Systems*. 2004.

[11] Fahmy, Suhaib A., Peter YK Cheung, and Wayne Luk. "Novel FPGA-based implementation of median and weighted median filters for image processing."*Field Programmable Logic and Applications, 2005. International Conference on*. IEEE, 2005.

[12] Esakkirajan, S., et al. "Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter." *Signal Processing Letters, IEEE* 18.5 (2011): 287-290.

[13] Lu, Ching-Ta, and Tzu-Chun Chou. "Denoising of salt-and-pepper noise corrupted image using modified directional-weighted-median filter." *Pattern Recognition Letters* 33.10 (2012): 1287-1295.

[14] Wan, Yi, Jiafa Zhu, and Qiqiang Chen. "On the nature of variational salt-and-pepper noise removal and its fast approximation." *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012.

[15] Kalali, Ercan, and Ilker Hamzaoglu. "A low energy 2D adaptive median filters hardware" *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015*. IEEE, 2015.

[16] Monajati, M., S. M. Fakhraie, and E. Kabir. "Approximate arithmetic for low-power image median filtering." *Circuits, Systems, and Signal Processing*34.10 (2015): 3191-3219.

[17] Hosseini, Hossein, Farzad Hessar, and Farokh Marvasti. "Real-Time Impulse Noise Suppression from Images Using an Efficient Weighted-Average Filtering." *Signal Processing Letters, IEEE* 22.8 (2015): 1050-1054.

[18] Chen, Ren-Der, Pei-Yin Chen, and Chun-HsienYeh. "A low power architecture for the design of a one-dimensional median filter." *Circuits and Systems II: Express Briefs, IEEE Transactions on* 62.3 (2015): 266-270.

[19] Burian, Adrian, and Jarmo Takala. "VLSI-efficient implementation of full adder-based median filter." *IEEE International Symposium on Circuits and Systems*. 2004.

