International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 11 (2017) pp. 2929-2941
© Research India Publications. http://www.ripublication.com

# MAX Tree Extraction Enabled Area and Energy Efficient Median Filter Design: A VLSI Design Approach

**Mr. Tulasiram.K**[1,*]
*Research Scholar,*
*Jawaharlal Nehru technological*
*Hyderabad, TS, India.*

[1]***ORCID: 0000-0003-2166-9515***

**Dr. Y.Rajasree**[2]
*Professor, Department of Electronics &*
*Communication Engineering,*
*St.Peter's Engineering College,*
*Hyderabad, TS, India.*

**Corresponding Author*

**Dr. K.Anitha Sheela**[3]
*Professor, Department of Electronics*
*& Communication Engineering,*
*Jawaharlal Nehru technological*
*university, Hyderabad, TS, India.*

**Abstract**
The exponential rise in the demands of a cost effective, power and area efficient de-noising filter design for varied image processing and allied application, in this paper a novel MAX-Tree Extraction (MTE) algorithm enriched median filter has been proposed. To achieve an optimal solution for median filter design our proposed model encompasses MTE in conjunction with the key components including token generators, comparator, shifter and noise detector circuit that cumulatively enables low hardware requirement and hence minimum energy and area consumption. Additionally, our proposed MTE extraction model avoids the need of rank generation, rank calculation and median selection. The overall developed model is implemented using VLSI technologies, where its performance is assessed for different data types (image and audio) and noise level. In addition to the area, power, we have examined the proposed VLSI design in terms of peak signal to noise ratio and mean square error, where the overall performance has been found better than the state of art technologies.

**Keywords:** Median Filter Design, Max-Tree Extraction, VLSI Design, Denoising.

## INTRODUCTION

In recent years, the exponential growth in computation techniques and vision based applications has given rise to a new and broad dimension called Image processing that has gained widespread attention across global academia-industries. In fact, digital image processing (DIP) has emerged as a potential research domain because of its vital significance in communication systems, electronic communication, consumer and entertainment electronics, supervision, monitoring and control, biomedical computations, computer aided diagnosis, remote sensing, robotic navigation and control and more importantly computer vision based applications. To perform efficient processes in aforementioned image processing based application efficient image segmentation and object recognition are must. This as a result can provide optimal visual display, accurate decision processes etc to serve major applications such as television, surveillance, monitoring and control and photo-phone, etc. To achieve it the retrieved image is required to be processed for de-noising or de-blurring. Typically, the digital images get corrupted due to the impulse noise which is commonly caused due to error presence during transmission, noise in channel, malfunction in camera sensors, corrupted pixel elements, fault prone memory locations, and analog to digital conversion (ADC) timing errors. One of the key characteristics of such kinds of noise is that only certain fraction of the pixels is noised or corrupted, while the other pixels remain intact and noise-free. Typically, the impulse noise is split into two categories: the fixed-valued impulse noise (FVIN) and the random-valued impulse noise (RVIN), where the first one is also stated as salt-and-pepper noise in which the gray-scale value of a noisy pixel used to be either maximum or the minimum in gray-scale images. Exploring visual perception it can be found that the image contains white and dark dots. Because of this reason the fixed value impulse noise (FVIN) is commonly known as the salt and pepper noise. On the other hand, in case of RVIN, gray-scale values of noisy pixels used to be homogeneously distributed in the range of [0, 255]. Once an image gets corrupted, particularly during image acquisition, transmission through channel and storage or retrieval, it turns out to be inevitable to repress the noise component efficiently without degrading the image quality such as edge information. Maintaining edge information intact plays vital role for various image processing purposes. In addition, it ensures that the filtered or the processed image becomes more significant and useful for display purpose or certain object detection purposes. To enrich image quality noise suppression is must, which is usually performed by median or rank filters or low-pass linear filters. In major image processing tasks or applications, noise suppression or denoising the image is essential before performing edge detection, object or region of interest (ROI) segmentation and object identification, etc.

Considering application specific scenarios, particularly consumer electronics for image acquisition devices (video camera, video recorders, satellite decoders, etc) the acquired images or video data are usually affected by different types of the noise components [1, 2, 3]. Usually for digital video broadcasting [3] or CCD/CMOS cameras [1, 2, 3], White Gaussian distribution is considered for noise modeling. On the contrary, impulsive type of noise usually affects images acquired from satellite TV decoders [2, 3]. In addition, impulsive noise model is applied for faulty bits during coding and transmission. Among various available techniques for noise removal the median filter is the dominating one. It is a nonlinear noise filter that plays vital role in image processing, particularly for signal smoothing, impulse noise suppression, edge preservation etc [4]. Median filter signifies its robustness with the ability that as per the pixel output, it can process within the input window and can move the window over image. In practice, before selecting the output, the sliding window sample is processed for sorting as per the filter operation. Typically, one-dimensional median filter applies bubble sorting to estimate the median value [5]. To perform filtering the sliding window is shifted right to complete the sorting. During this process, incorporating insertion and deletion the subsequent slide window is derived from the current sliding window. To obtain a filter of size $N \times N$, $N$ pairs of insertion and deletion process is needed [6]. In major morphological filters the median filter and the rank order filter are predominantly applied to perform sorting [7].

In function, median filter substitutes a pixel sample with the middle-ranked value among all the samples inside the sample window. Based on the number of samples processed at the same cycle, there can be two distinct architecture types for hardware design. These are, word-level and bit-level architecture. Here, in case of the word-level architecture, the input samples are processed sequentially in word by word manner [8]. In addition, the sample bits are processed in parallel [9-11]. In this approach, the incoming sample is inserted into the correct rank by incorporating two step processes. In first step the samples are moved to the left that as a result eliminates the previous sample from the window. On contrary, in the second step of the process, by comparing the incoming sample with the existing sorted samples, the incoming sample is inserted in the right place [12]. On the other hand, in case of bit-level architecture the samples are processed in parallel, while the sample bits are processed in sequential manner [13-16]. However, the word-level architecture enables low-power median filter design that functions better for real-time applications.

Further exploring in depth it can be found that the median filter architecture can be split into three approaches. These are, sorting network architecture, array architecture and stack based architecture. In case of sorting network architecture, the ranging of the samples is performed before processing sample selection. Because of the exceedingly huge number of compare-swap elements, such approaches can enable higher throughput. In case of array architectures, each window element is assigned its rank. In this process, the ranks are updated with change in the position of the window. The third type of approach, i.e., the stack based architecture applies hamming comparators, and threshold logic that translates filtering into the binary domain [17]. No doubt, the significance of median filter has motivated researchers community to enable more efficient solution; however the existing limitations , such as static filter coefficients etc limits the employability of the generic filter solutions. In addition, numerous existing approaches don’t consider area and power constraints in design. On contrary, the era of miniaturization demands more power and area (i.e., memory) efficient solutions. The high pace rising applications characterizing non-linear features demand more efficient filter solution. To achieve this non-linear adaptive filter has been found a potential candidate. With this motivation in this paper we intend to develop a robust median filter design.

Realizing the current technologies and emerging trends very large scale integrated circuits (VLSI) technologies have gained global attention, which has motivated researchers to achieve more efficient computing solutions. The inevitable demands of vision based computations, image processing etc have also motivated academia-industries to exploit more efficient denoising or deblurring approaches. Considering it as motivation, in this paper we have developed and implemented an enhanced and robust median filter design using VLSI technology. Considering the prevalence of the median filter, in our proposed VLSI design model the emphasis is made on developing a sorting algorithm fulfilling the demand of required minimum number of hardware and at high operating frequency [18]. Here, to enhance the sorting algorithm we have considered concurrent design model where deletion as well as addition of the elements takes place simultaneously. Similar to the generic function our design comprises two arrays, window array and sorted array. Here, window array contains the input element while sorted array contains the output element [19] [20]. Unlike traditional approaches, in our proposed model the overall architecture of the median filter is implemented with the help of MAX tree extractor algorithm. In addition, we propose adaptive denoising approach to perform signal image de-noising purpose. In the proposed approach, image window has been de-noised if it has noise. The overall developed Max tree based median filter design has been implemented over Field Programmable Gate Array (FPGA) hardware platform, where the amalgamation of the proposed enhanced approaches has yield better performance in terms of low area, power and total cost.

## RELATED WORK

This section presents a critical review made for different approaches and techniques derived for median filter design for image processing.

Impulse detection has the direct impact on the efficiency of noise removal filters. Considering it as motivation, Aizenberg et al. [21] proposed differential rank impulse detector (DRID) on the basis of the comparison technique. Their proposed comparison model compares the signal samples within a narrow rank window by both rank as well as absolute value. Similarly, Zhang et al. [22] in their research effort developed a new impulse detector (NID) to develop a switching median filter design. In their filter design, NID applied the minimum absolute value of four convolutions which are retrieved by means of a 1-D Laplacian operator that detects noisy pixels in the sample image. Luo et al. [23] developed an impulse noise removal technique using fuzzy impulse detection approach. Deepa et al. [24] applied two line buffers, register banks, impulse noise detector, edge oriented noise filter and impulse arbiter to design a low cost VLSI architecture so as to enable edge preserving impulse denoising model. One of the key novelties of their proposed VLSI design was the need of a two line buffer that avoided any requirement of a full frame memory. Furthermore, their proposed design comprises only defined or fixed size window. This approach enabled reduction in memory requirements and computational complexity. In addition, the use of impulse noise detector turns off the remaining circuit architecture in case the current pixel is noise free. By doing so, it reduced energy consumption significantly. In addition, authors applied a four stage pipeline model that enhanced computational speed significantly. Their proposed approach exhibited better edge preservation that resulted into better image quality. Luo et al. [25] developed an alpha-trimmed mean based method (ATMBM) where they applied alpha trimmed mean for impulse detection, which was then followed by the substitution of the noisy pixel value by a linear combination of its original value and the median of its local window. Srinivasan et al. [26] developed a decision-based algorithm (DBA) for corrupted pixel removal. Authors applied median pixel value to remove noise pixels where median was estimated using their proposed DBA algorithm. Yu et al. [27] designed a low cost VLSI implementation design for fixed valued impulse noise removal. Chen et al. [28] developed a random valued impulse noise removal technique. T reconstruct the corrupted or noise image pixels, authors applied decision tree based impulse detector and an edge preserving filter. However, their approach is too complicate and difficult in implementation. To alleviate such limitations, authors suggested an edge-preserving filter in conjunction with a modified conditional median filter to substitute traditional filter design. To remove impulse noise from sample image, Pei-Yin et al. [29] emphasized on developing a VLSI implementation design for high quality image filter. Authors proposed Simple Edge Preserved De-noising (SEPD) method for reducing the impulse noise. In later stag of research effort, Pei-Yin et al. [30] developed a low cost image scaling mechanism where the Edge Oriented – Area based Pixel Scaling was used to preserve image quality. Chih-Yuan et al. [31] applied decision tree based impulse noise detector and edge preserving filter to remove impulse noise from sample image. To detect noisy pixels authors applied decision tree, this was then processed for edge oriented noise filtering using edge preserving filter. The quantitative analysis of the results affirmed that their approach exhibited better for image quality retrieval. Emin et al. [32] too made effort for impulse noise removal where they emphasized on minimizing the blur. To achieve it author applied Neuro fuzzy based impulse noise removal. In this approach, the Neuro–fuzzy model exhibited noise detection in image by executing artificial training image. On the basis of the training result the decision making detector performed impulse noise detection in the original image and thus performed noises removal by using proposed filters. Shih-Lun et al. [33] emphasized on developing a low cost VLSI implementation model for high quality image scaling processor, where authors developed a bilinear interpolation technique for image smoothing (after resizing). Thus, their proposed filter model in conjunction with the pre filter (such as clamp filter and sharpening filter) was used to alleviate the aliasing and blurring issue.

Considering the significance of an area, speed and power efficient median filter design, Vasanth et al. [34] used a carry select comparator to compare and swap functions. Their approach reduced the sorting related complexity significantly. As design components, Carry select comparator applied one half subractor, 7 full subractor, and multiplexers along with a few inverters. Yang et al. [35] focused on accuracy of the filter design and efficient hardware implementation. To achieve it authors developed a stereo matching model by exploiting guided image filter (GIF) that itself is an edge-preserving filter. Their applied filter design enhances the ease of implementation of the adaptive support window algorithm. Authors estimated the GIF coefficients using their proposed mean filter tree structure (MFTS). Noticeably, MFTS stores hardware resources by sharing huge additions among filter operations. However, the overall filter design was enhanced by means of Laplacian Filter that significantly enhances accuracy, especially for the discontinuous disparity regions. Hsia et al. [36] developed an adaptive digital signal processing approach for denoising. The parallel structure and pipelining processing based simulation results confirmed that their approach could minimize impulse noise even in highly corrupted images. Hiasat et al. [37] developed a new bit-level model to design median filters by exploiting the concept of majority. Authors found their approach suitable for VLSI implementation where it exhibited an area complexity of O (N(N+w)) with the complexity of O(w). In comparison to the other state-of-art techniques, they found their approach better to perform high speed computation while ensuring minimal hardware requirements. Authors applied their proposed median filter design for electrocardiograms (ECG), where they used it to restore the baseline of an ECG signal that usually undergoes noisy due to patient's breathing and movement. Caroline et al. [38] stated that as linear filters tend to blur an image and therefore can’t be used more often. On the other hand, non-linear filters enable efficient results than the linear filters. Considering it as motivation, they [38] designed two filter named Adaptive Rank Order Filter (AROF) and Adaptive Median Filter (AMF) and assessed respective performance with VLSI implementation. Their proposed non-linear adaptive filter functions on the basis of the level of (image) noise intensity. Furthermore, to enable swift and efficient processing authors considered pipelining with parallel processing over the proposed VLSI architecture for AROF and AMF. Tapu et al. [39] proposed a scale space median filter design employing min-max objective function where authors focused on enhancing the relation within each graph cut so as to eliminate noise, particularly introduced because of large scale displacement and camera movement. Vasanth et al. [40] applied modified decomposition method to design an area efficient median filter design. Their proposed modified decomposition algorithm alleviates the complexity issues of the existing threshold decomposition approaches, like complex comparators. Their proposed design exhibited overall task in two sequential phases, decomposition and recombination. The prime novelty of their effort was the minimum slice and look-up table requirements for the VLSI implementation. KalaiPriya et al. [41] emphasized on a real-time FPGA implementation of a high pass filter design using a simple nonlinear median filter algorithm. The overall design comprises a median filter along with a high pass filter, where the signal image to be filtered is given as the input to the median filter. The output of the median filter was given as the input of the high pass filter that recognizes and distinguishes the high frequency components by performing time-ordered window sliding over the output samples of the median filter. Burian et al. [42] developed a new median type filter design by exploiting median cost function and implemented it with VLSI-suitable hardware platform. Furthermore, authors developed a more efficient median cost function reduction approach to achieve fast and area efficient median filter design. Hwang et al. [43] applied adaptive median filter that blur the sample image as both noise-free as well as noisy pixels het modified. Their proposed adaptive median filter eliminates positive as well as negative impulse noise concurrently. An enhanced effort was made where authors [44] developed a progressive switching median filter (PSMF) design to remove impulse noise from highly corrupted images. Their proposed PSMF model performs denoising of the sample image corrupted by salt-and-pepper type noise. A similar effort was made by Abreu et al. [45] where authors developed a denoising model to remove impulse noise from highly corrupted images.

## PROPOSED METHOD

Taking into consideration of the existing at hand techniques for median filter design, it is revealed that majority of the approaches have primarily given preference to image quality retrieval. Unavoidably, maintaining optimal image quality after denoising sample image is must. However the rising demand of miniaturization and compact VLSI design often put question over existing approaches due to their limited efficiency to meet area and power efficient design. On the other hand, a major fraction of approaches are found confined because of their ability to process noise removal or denoising, especially with non-linear noise patterns. To alleviate such limitations and enable a low cost design solution, median-type filters are often considered as a potential candidate. However, it demands architecture as well as functional optimization measures to ensure optimal performance (i.e., area and power efficient VLSI design). Considering it as motivation, in this research paper a robust Max-Tree Extractor Algorithm (MTEA) is developed to design robust median filter VLSI architecture. A brief discussion of the proposed MTEA based median filter design is given as follows:

To design our proposed enhanced median filter design for VLSI implementation, we have emphasized on incorporating MTEA algorithm to reduce area and power consumption. The overall architecture of the proposed MTEA based median filter design is presented in Fig.1. As depicted in Fig. 1, our proposed median filter design contains window register, max tree extractor (MTE), shift enable signal and token generator. As depicted in Fig. 1, the input value is given to the window array that contains $X_0, X_1, X_2, X_3, X_{4,} \ldots, X_n$ . In proposed architecture, these values can be varied depending upon different number of inputs. Unlike generic approaches, in our proposed median filter architecture, all input values are given as input to the MTE, which performs the MAX value operation. Among these input values, MTE extracts the maximum value to be located. Now, based on the input values, varied types of the index values $I_0, I_1, I_2, I_3$ and $I_{4,}$ are generated. Once retrieving the index values, these are then linked with a token generator (Fig. 1) that holds different token values for further computation. In the proposed approach, at first all the input values hold the initial token value assigned with zero (say, token initialization). In our proposed model, we have introduced a 5 bit bus to hold all the index values of the inputs and in such manner the token generator output connects all the input values (Fig. 1). In the next stage or cycle, the remaining input values are loaded by means of the right shift operation and above mentioned process is repeated iteratively till the maximum value is obtained.

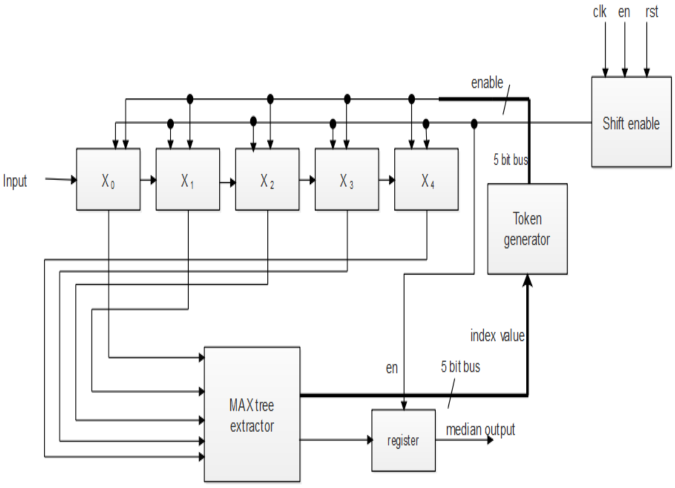

**Figure 1.** Proposed block diagram of median filter architecture

For illustration, let the input values be $X_0, X_1, X_2, X_3$ and $X_{4,}$ such as 10, 30, 5, 60, and 90. As stated, initially, all the input values hold the index values as zero ($\mathrm{i.e.,}\ I_0 = 0, I_1 = 0,\ I_2 =$

$0, \mathrm{I}_3 = 0$ and $\mathrm{I}_{4,} = 0)$. Here, window register takes 5 clock cycles to load the input values to the respective window, which are then feed as input to the MTE. Among these five inputs, 90 is the maximum value, which holds the index value I0 and thus the output values are stored in sorted window. Then, all the index values, $\mathrm{I}_0, \mathrm{I}_1$, $\mathrm{I}_2, \mathrm{I}_3$ and $\mathrm{I}_{4,}$ are given as the input of the token generator. Thus, from that 6th clock cycle, the max value 90 is placed in $\mathrm{I}_{4,}$ index value. So the output of the token generator is should contain this value like i.e., $\mathrm{I}_0 = 0, \mathrm{I}_1 = 0$, $\mathrm{I}_2 = 0, \mathrm{I}_3 = 0$ and $\mathrm{I}_{4,} = 1$, which refers that the sorted output index can be changed at 1 from 0. In subsequent phase, the input values perform the right shift operation. The new input value occupied in $\mathrm{X}_0$ window, but it is not enabled for the next clock cycle. Here, it should be noted that at 7th clock cycle, among the values of 10, 30, 5, and 60 only the same operation can be performed that as a result provides the sorted output 60, which is stored in $\mathrm{Y}_0$ index of the sorted window. Once completing this process, the final sorted window contains the output such as 90, 60, 30, 10, and 5. In the proposed architecture, a total of 8 clock cycles are needed to find the median value (i.e., 30). Once the new input value is changed to enable the process, it executes further processing otherwise maintains stability. Considering the overall objective to design a low cost and memory efficient VLSI architecture, our proposed model requires minimal hardware and the number of clock cycles to produce the median values from the different input values.

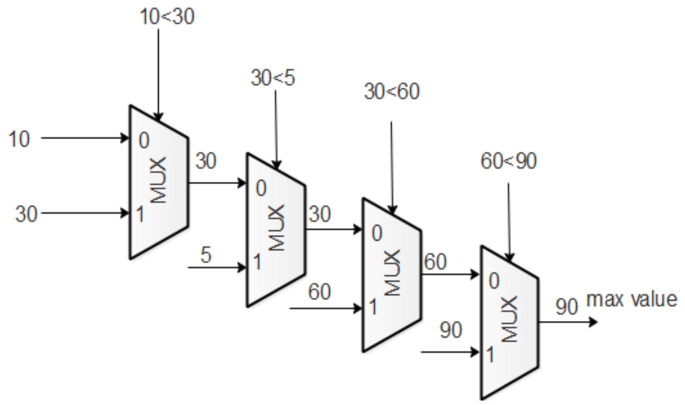

**Figure 2.** Comparator circuit for 5 windows

Fig. 2 presents the applied comparator circuit, where MUX plays a vital role in estimating the maximum value from the given input. Here, as depicted in the circuit input 10 and 30 is applied to the first MUX. Here, the selection line is set to the condition of 10<30 that enables output from the different inputs. In case the condition is satisfied, input line 1 connects the output or else input line 0 connects the output. Similarly the remaining MUX operations take place to estimate the maximum value. Now, considering the above example, the maximum value 90 is obtained. In case of 9 windows, the same procedure is performed that eventually causes increase in the number of MUX.

Unlike traditional approaches, in our proposed model token generator circuit is applied for 5 windows (Fig. 3). Once estimating the maximum value, the obtained maximum value index is feed as input to the token generators, a, b and c (Fig. 3).

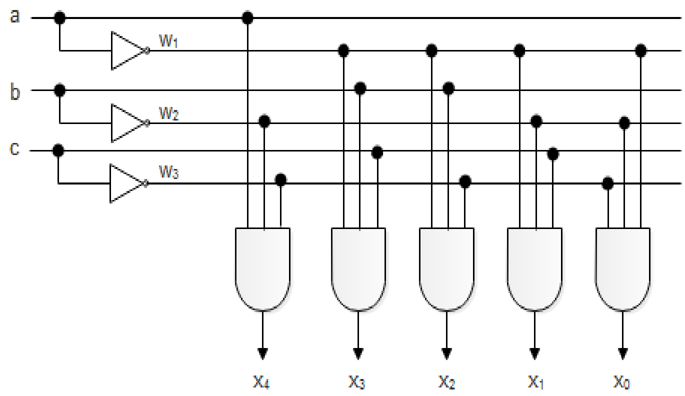

**Figure 3.** Token generator circuit for 5 window

As depicted in Fig. 3, our proposed design incorporates 5 window-operations that require only 3 bits to generate the token. As stated above, initially the entire token applied are set to zero or initialized with zero. For example, among all the inputs, 90 can be placed in zero level that holds the binary value 000. In this way, the token generator a, b and c holds 0 value. Now applying an inverter in succession of the token generator the value $\mathrm{w}_1$, $\mathrm{w}_2$ and $\mathrm{w}_3$ holds '1'. Thus the final value are connected to the AND gate that generates output as 1 in the terminal of $\mathrm{X}_0$. In this way $\mathrm{X}_0$ holds the token 1, while other values would have the zero. In this way, the maximum value generated from the MTE can hold the token value as 1.

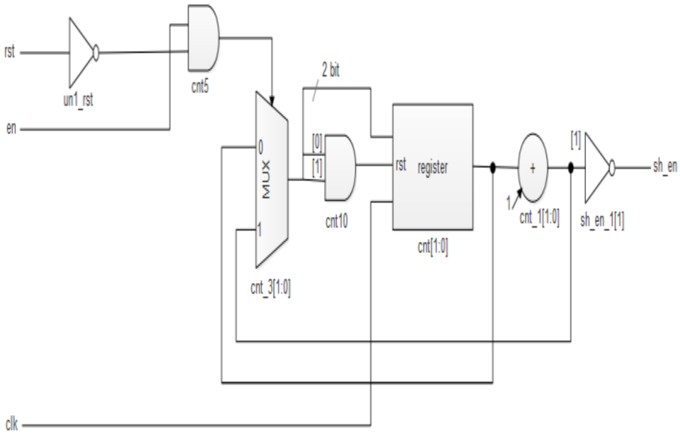

**Figure 4.** Shifter circuit

Fig. 4 presents the shifting circuit, where the initial register output and adder output is considered binary (i.e., 0 and 1). Let the initial reset be 0 then once performing inverter task it would become 1 and thus makes the enable signal as 1. Noticeably, this output (i.e., enable signal) is given as the input of the AND gate (Fig. 4). The output of AND gate is 1, which is further given to the selector line of MUX. In case the selection line is 1, the output of the adder is fed to the MUX output else register output connects the MUX output. Here, considering the binary representation of 1 as “01”, it is (i.e., 01) given as the input of AND gate and the two bits are connected to the register input directly (Fig. 4). In this manner, the output value of the register is 1, which is further added with the adder output to produce 2. Meanwhile, the MUX operation is performed. When the adder value is 3, the AND gate value set to be as 1. By doing so at 3rd clock cycle, reset gets enabled and register reset the values and thus the circuit functions as a counter. Now considering the binary representation of the value 3 as “011”, when given as the input to the inverter gives result as “100”. From this it signifies that only after 3rd clock cycle the input value can be filled in respective window and thus the median value is stored in output terminal. In our proposed architecture, the shifting operation of input window value has been performed by enabling the shifter circuit and at the same time the median value is produced in the output terminal. Once estimating the median value, we emphasize on performing noise detection and filtering. Fig. 5 presents the noise detection circuit. The detailed and internal architecture of the noise detection circuit is given in Fig. 6.

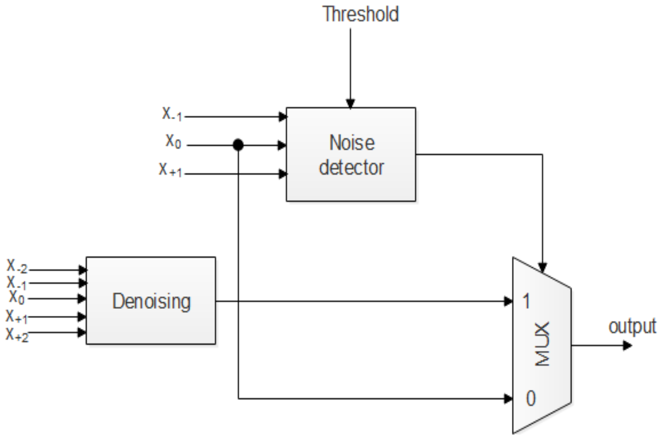

**Figure 5.** Noise detection circuit

As depicted in Fig. 5, in our proposed noise detection model values $X_0$, $X_{-1}$ and $X_{+1}$ signifying the middle, left and right value from the window are fed to the noise detector. In our proposed model the noise detector unit compares the input with the threshold, whose output act as selector line of the MUX (Fig. 5). Meanwhile, all the input values can be given to the de-noising block. In case of selection line as 1, the window can be affected by noise and this specific value is given to the de-noising block that generates the output of the de-noising circuit. It should be noted that here MUX provides the median value output. On the contrary, if the selection line is 0, it signifies the absence of noise in the window and thus the middle value can be delivered directly to the MUX output. This overall mechanism is referred as the adaptive median circuit.

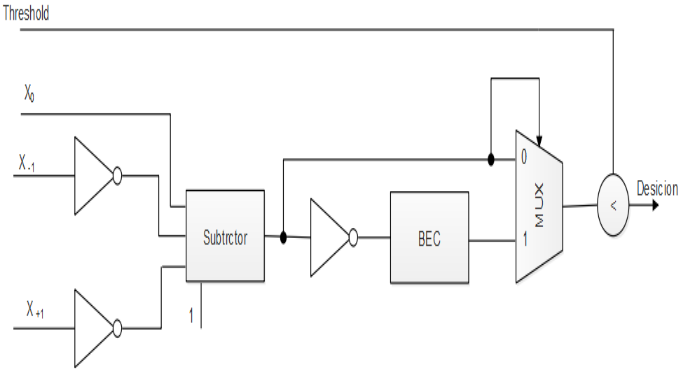

**Figure 6.** Internal architecture of the noise detection circuit

Fig. 6 presents the internal architecture of the noise detection circuit. In the proposed noise detection circuit a subtractor unit is applied that as per the input value performs subtraction of the middle value to the left and the right values. In this way, the output of the subtractor performs inverting operation, which is further processed with Binary to Excess-1 Code Converter (BEC) component. In our proposed design, BEC is obtained by performing or adding one to the binary value (i.e., 2’s component). Here, in the proposed design the BEC takes n input and thus finally generates n + 1 output so as to facilitate the carry output as the selection input of the next stage MUX (Fig. 6). Here, in case the selector line is ‘1’, the BEC output connects MUX output; else the subtractor output connects MUX output. In our proposed architecture the output of the MUX is compared with a threshold value that generates the decision output and thus the result obtained functions as the selector line of MUX in noise detector circuit. Thus, the developed VLSI architecture has been examined over FPGA platform to assess its effectiveness for noise removal efficacy, area and power performance. The results obtained are discussed in the next section.

## EXPERIMENTAL SETUP

This section briefs about the research model or VLSI architecture developed, simulation, and performance assessment in terms of various parameters. The overall developed architecture is simulated in Modelsim SE 10.1c, where the algorithms are developed using Verilog programming platform. Furthermore, the developed VLSI architecture was simulated over Windows 7 OS platform with I5 processor armored with 8 GB RAM. To assess the developed model in terms of area, power and delay, Cadence 180nm technology and RTL compiler have been taken into consideration. The RTL schematic of the developed model for windows 5 is presented in Fig. 7.

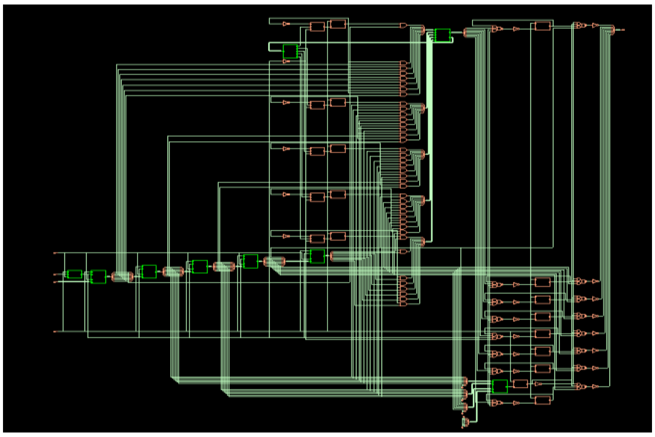

**Figure 7.** RTL schematic for window 5 - (8 bit sample width)

The overall proposed model can be considered as a calibrated optimization measure where eventual target is made on achieving power, area and delay efficient median filter design. To assess the performance of our proposed MTE enabled median filter design, we have considered the work by Tulasiram et al. [47], where authors have emphasized on developing a power efficient median filter design. However, authors [47] could not address the significant parameters such as latency or delay that possesses direct impact or relationship with a real-world system performance. Similar to the existing work [47] (here onwards we state reference [47] as the existing system), our proposed VLSI design considers one dimensional median filter design; however the use of Max Tree Extractor (MTE) in conjunction with token generation strengthen our proposed model to exhibit area and power efficient VLSI design while ensuring low power and low area consumption. This exhibits robustness of our proposed median filter design over existing state of art techniques.

The following tables (Table 1 and Table 2) present the performance assessment of our proposed median filter design. Here to examine the efficacy of our proposed MTE enabled median filter design a reference work done by Tulasiram et al [47] has been considered. Our proposed VLSI implementation model of the median filter design is developed for window size 5 as well as window size 9 and the respective performance assessment is done in terms of area, power and latency. The proposed model is simulated with different image benchmark data, such as Lena, baboon and Peppers. Observing the results for 8 bit sample width (Table I), it can be found that the proposed MTE enabled median filter design exhibits better performance in terms of area and power. However, higher latency remains the motivation for further research efforts and initiatives. Unlike Table 1, where standard Image data were used as benchmark data, Table 2 presents the simulation output with Audio data. A total of three audio samples were taken into consideration for performance assessment. Now, exploring Table 2, where the simulation results for 16 bit sample width is depicted, it can be found that our proposed median filter design outperforms existing one-dimensional median filter approach [47] in terms of area and power efficiency.

**TABLE I.** SIMULATION RESULTS FOR 8-BIT SAMPLE WIDTH

| Design | Throughput (#median outputs/clock) | Latency (#clock cycles) | Window size | 8- bit sample width | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | EPS(nJ) | | | |
| | | | | Area (um2) | Power (nW) | Delay (ps) | Lena | Peppers | Baboon | Average |
| Existing [47] | 1 | w | 5 | 17989 | 1325922 | 1907 | 39.28 | 40.02 | 41.71 | 40.33 |
| | | | 9 | 34931 | 2694208 | 2728.9 | 79.81 | 81.33 | 84.75 | 81.96 |
| Proposed | 1 | w | 5 | 17743 | 1104987 | 5531.1 | 32.73 | 33.35 | 34.76 | 33.61 |
| | | | 9 | 29721 | 1765880 | 10864 | 52.31 | 53.30 | 55.55 | 53.72 |

**TABLE II.** EXPERIMENTS RESULTS FOR 16-BIT

| Design | Throughput (#median outputs/clock) | Latency (#clock cycles) | Window size | 16- bit sample width | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | EPS(nJ) | | | |
| | | | | Area (um2) | Power (nW) | Delay (ps) | Audio 1 | Audio 2 | Audio 3 | Average |
| Existing [47] | 1 | w | 5 | 28710 | 2076690 | 1998 | 75.77 | 70.22 | 107.66 | 84.55 |
| | | | 9 | 54272 | 3917452 | 3234 | 142.93 | 132.46 | 203.09 | 159.4 |
| Proposed | 1 | w | 5 | 26129 | 1653820 | 10065 | 60.24 | 55.92 | 85.73 | 67.29 |
| | | | 9 | 53630 | 3638506 | 19586 | 132.75 | 123.03 | 188.63 | 148.1 |

The graphical depiction of the proposed MTE enabled media filter design and existing work (i.e., Tulasiram et al. [47]) is given as follows:

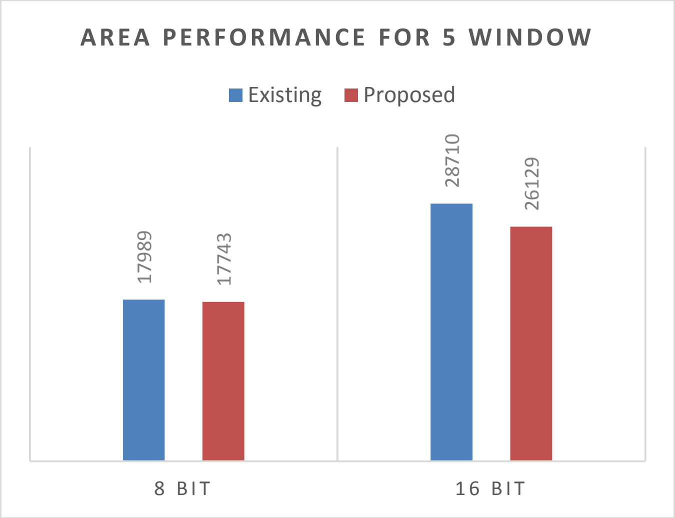

**Figure 8.** Comparison of Area performance for window size 5

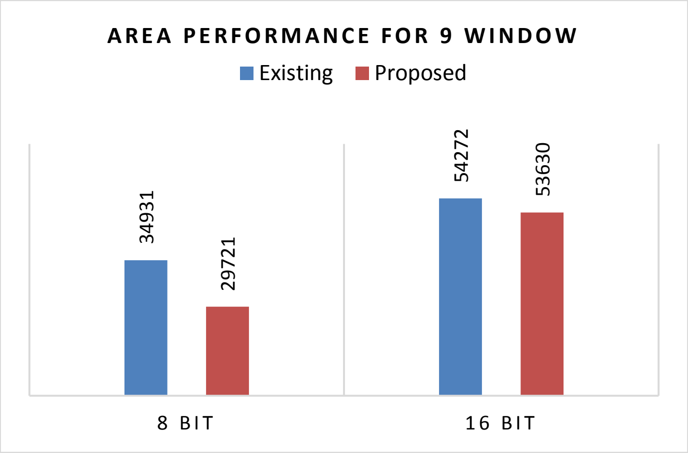

**Figure 9.** Comparison of Area performance for window size 9

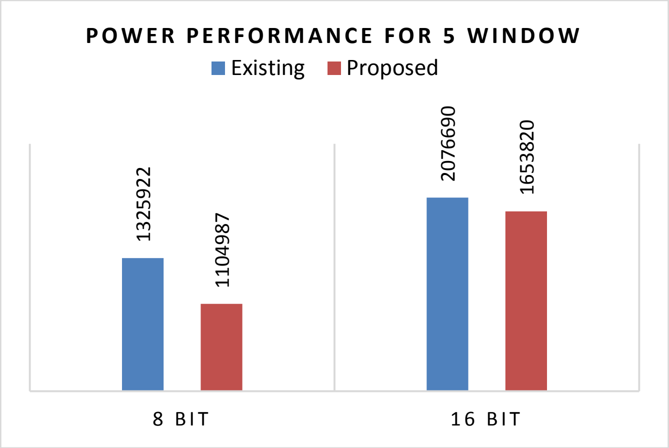

**Figure 10.** Comparison of Power performance for window size 5

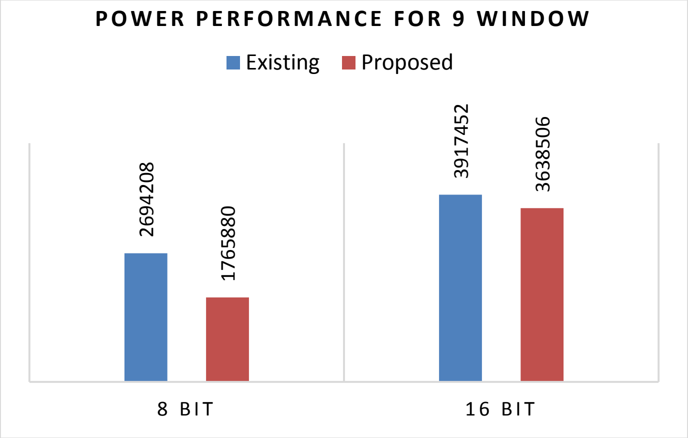

**Figure 11.** Comparison of Power performance for 9 window

In present day applications maintaining Quality of Service (QoS) and Quality of Experience (QoE) is of paramount significance. Considering present study for multimedia communication or processing where enabling or retaining optimal peak signal to noise ratio (PSNR) is vital, we have assessed the proposed system in terms of PSNR and mean square error (MSE). Table III presents the PSNR of our proposed MTE enabled median filter design and the reference work [47]. Observing overall results, it can be found that the proposed median filter design outperforms state of art technique in terms of higher PSNR and low MSE. This result affirms that the proposed system can be effective for retaining better QoE and QoS requirements.

**TABLE III.** COMPARISON OF PSNR AND MSE VALUES FOR LENA, PEPPER, AND BABOON IMAGES

| Design | Performance | Lena | Pepper | Baboon |
|---|---|---|---|---|
| Existing [47] | PSNR | 22.7 | 22.25 | 23.59 |
| | MSE | 4686.16 | 4667.76 | 4728.664 |
| Proposed | PSNR | 35.85 | 39.41 | 33.54 |
| | MSE | 3843.02 | 8722.34 | 2260.14 |

The visual analysis of the proposed model for different data types is given as follows:

Fig.12. (a). Lena Input image

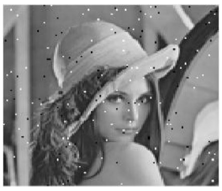

Fig. 12. (b). Lena Noisy image

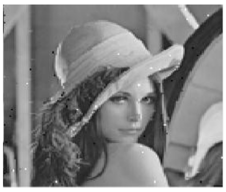

Fig. 12 (c). Lena De-noising output

Fig. 13. (a). Pepper Input image

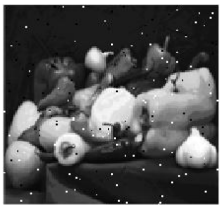

Fig. 13. (b). Pepper Noisy image

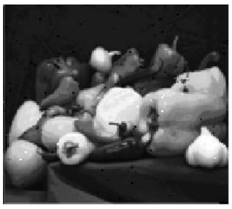

Fig. 13. (c). Pepper De-noising output

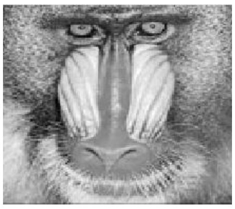

Fig. 14. (a). Baboon Input image

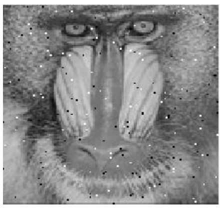

Fig. 14. (b). Baboon Noisy image

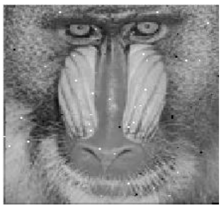

Fig. 14. (c). Baboon De-noising output

Realizing the emergence of computer vision based computer aided diagnosis systems, providing a potential technique for denoising can be of utmost significance. With this motivation, we have examined our proposed model with different biomedical (Magnetic Resonance Imaging (MRI)) data for study. Here, we have considered MRI, X-ray and CT scan data for assessment.

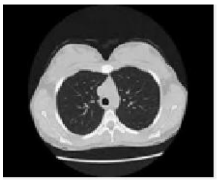
Fig. 15. (a). Input image

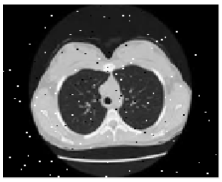
Fig. 15. (b). Noisy image

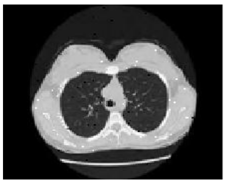
Fig. 15. (c). De-noising output

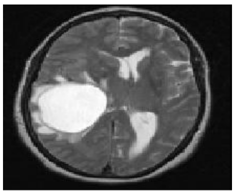
Fig.16. (a). Input image

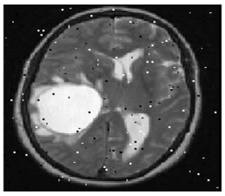
Fig.16. (b). Noisy image

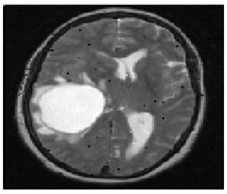
Fig. 16. (c). De-noising output

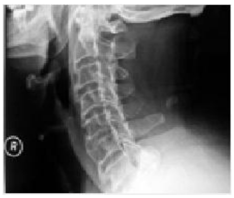
Fig. 17. (a). X-ray Input image

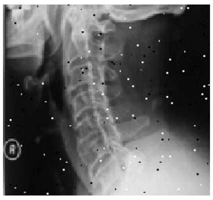
Fig. 17 (b). X-ray Noisy image

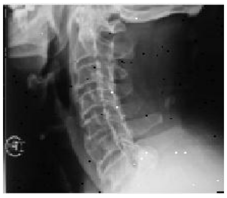
Fig. 17 (c). X-ray De-noising output

The performance exhibited by the proposed median filter design and its comparison with the existing system is given in Table IV.

**TABLE IV.** COMPARISON OF PSNR AND MSE VALUES FOR MRI, CT AND X-RAYMEDICAL IMAGES

| Design | Performance | MRI | CT | X-ray |
|---|---|---|---|---|
| Existing [47] | PSNR | 22.81 | 23.2 | 22.52 |
| | MSE | 4690.8 | 4708.92 | 4642.02 |
| Proposed | PSNR | 36.1668 | 37.69 | 39.51 |
| | MSE | 4136.91 | 5871.23 | 8936.76 |

Exploring inside, it can be found that the proposed MTE enabled median filter design exhibits better than the state of art technique proposed by Tulasiram et al [47]. Here, it can be found that our proposed model exhibits higher PNSR and low MSE that cumulatively affirms suitability of the proposed system for real-world applications. In addition, the performance of our proposed system is compared with different level of noise. No doubt, the extent of noise presence directly influences the efficacy of a denoising technique. With this motivation, we have examined the proposed approach as well as the existing median filter design with varying noise level (0.02, 0.03, 0.04 and 0.05). The noise dependent performance comparison is presented in Table V.

**TABLE V.** COMPARISON OF PSNR AND MSE VALUES FOR DIFFERENT NOISE LEVEL FOR LENA AND MRI IMAGES

| Noise level | | Performance | Lena | MRI |
|---|---|---|---|---|
| 0.02 | Existing | PSNR | 22.68 | 22.88 |
| | | MSE | 4685.5 | 4694.27 |
| | Proposed | PSNR | 35.88 | 36.19 |
| | | MSE | 3876.27 | 4157.92 |
| 0.03 | Existing | PSNR | 22.81 | 22.77 |
| | | MSE | 4691.02 | 4689.04 |
| | Proposed | PSNR | 35.92 | 36.21 |
| | | MSE | 3905.6 | 4179.27 |
| 0.04 | Existing | PSNR | 22.93 | 22.85 |
| | | MSE | 4696.55 | 4692.74 |
| | Proposed | PSNR | 35.99 | 36.28 |
| | | MSE | 3968.53 | 4245.52 |
| 0.05 | Existing | PSNR | 23.12 | 22.87 |
| | | MSE | 4704.94 | 4693.51 |
| | Proposed | PSNR | 36.03 | 36.32 |
| | | MSE | 4013.02 | 4284.02 |

The conclusion of the overall developed system with respective significances is discussed in the next section.

## CONCLUSION

In recent years, the high pace emergence of image processing techniques and associated applications have been witnessed globally. To ensure quality of service, quality of experience and reliable processing or decision maintaining optimal image or equivalent data signal quality is must. Realizing the need of a robust denoising technique for data processing, in this paper a highly robust median filter design was proposed. The proposed one dimensional median filter design comprised MAX tree extractor algorithm followed by significant components such as token generators, comparator, shifter and noise detector circuit that cumulatively provided an area and power efficient filter design for VLSI implementation. The proposed MTE extraction model avoids the need of rank generation, rank calculation and median selection. This as a result achieves optimal design specifications. The overall developed VLSI implementation model has been examined with different data types, including image and audio affirms that the proposed approach exhibits better in terms of peak signal to noise ratio and mean square error. Similarly, the performance assessment for biomedical data, where reliability and intact feature of image (i.e., biomedical images for CAD purpose) is must, the proposed system has exhibited better in terms of PSNR and MSE. Observing overall results, it can be found that the proposed Max-Tree Extraction (MTE), followed by token generation base median filter design exhibits better than the existing state of art techniques. The assessment of proposed median filter design with different types of data types (i.e., image and audio data), noise level etc also confirms robustness of the proposed system. The proposed work emphasized over a 1-D filter design while targeting on area and power consumption. However, in future 2D filter design could be explored and can be enhanced for latency as well as cost effectiveness.

## REFERENCES

[1] G. E. Healey and R. Kondepudy, “Radiometric CCD camera calibration and noise estimation,” IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 16, no. 3, pp. 267–276, 1994.

[2] L. Tenze, S. Carrato, C. Alessandretti, and S. Olivieri, “Design and real-time implementation of a low cost noise reduction video system,” in Proc. COST 254-Workshop on Intelligent Communication Technologies and Applications, pp. 36–40, May 1999.

[3] A. Amer and H. Schr¨oder, “A new video noise reduction algorithm using spatial subbands,” in Proc. IEEE Conference on Electronics, Circuits and Systems, vol. 1, pp. 45–48, October 1996.

[4] D. S. Richards, “VLSI median filters,” IEEE Trans. Acoust., Speech, Signal Process., vol. 38, no. 1, pp. 145–153, Jan. 1990.

[5] Benkrid, Khaled, Danny Crookes, and AbdsamadBenkrid., “Design and implementation of a novel algorithm for general purpose median filtering on FPGAs”, Circuits and Systems, IEEE International Symposium,vol.4, no. 6, pp. 425-428, 2002.

[6] C. T. Chen, L. G. Chen, and J. H. Hsiao, “VLSI implementation of a selective median filter”, IEEE transactions on consumer electronics, vol.42, no. 1, pp. 33-42, 1996.

[7] Gasteratos, Antonios, IoannisAndreadis, and PhTsalides., “Realization of rank order filters based on majority gate”, Pattern Recognition,vol.30, no. 9, pp. 1571-1576, 1997.

[8] Prokin, Dragana, and Milan Prokin, “Low hardware complexity pipelined rank filter”, IEEE Transactions on Circuits and Systems II: Express Briefs, vol.57, no. 6, pp. 446-450, 2010.

[9] R. D. Chen, P. Y. Chen, and C. H. Yeh, “Design of an area-efficient one dimensional median filter,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 10, pp. 662–666, Oct. 2013.

[10] S. A. Fahmy, P. Y. K. Cheung, and W. Luk, “High-throughput one dimensional median and weighted median filters on FPGA,” IET Comput. Digit. Tech., vol. 3, no. 4, pp. 384–394, Jul. 2009.

[11] V. G. Moshnyaga and K. Hashimoto, “An efficient implementation of 1-D median filter,” in Proc. 52nd IEEE Int. MWSCAS, pp. 451–454, 2009.

[12] Q. Fang, W. Zhang, Z. Pang, D.Chen, and Z. Wang, “A proposed fast word level sequential scheme and parallel architecture for bit plane coding of EBCOT used in JPEG2000”, In Multimedia Technology (ICMT),vol.7, no. 2, pp. 1-4, 2010.

[13] J. Cadenas, G. M. Megson, R. S. Sherratt, and P. Huerta, “Fast median calculation method,” Electron. Lett., vol. 48, no. 10, pp. 558–560, May 2012.

[14] C. Choo and P. Verma, “A real-time bit-serial rank filter implementation using Xilinx FPGA,” in Proc. SPIE Real-Time Image Process., vol. 6811, pp. 68110F-1–68110F-8, 2008.

[15] D. Prokin and M. Prokin, “Low hardware complexity pipelined rank filter,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 6, pp. 446– 450, Jun. 2010.

[16] Z. Vasicek and L. Sekanina, “Novel hardware implementation of adaptive median filters,” in Proc. 11th IEEE Workshop DDECS, pp. 1–6, 2008.

[17] V. G. Moshnyaga, and K. Hashimoto, “An efficient implementation of 1-D median filter”, IEEE International Midwest Symposium on Circuits and Systems, vol.12, no. 6, pp. 451-454, 2009.

[18] Sanny, Andrea, and Viktor K. Prasanna., “Energy-efficient median filter on FPGA”, International Conference on Reconfigurable Computing and FPGAs (ReConFig), IEEE, vol.15, no. 8, pp. 1-8, 2013.

[19] Teja, V. V. Ravi., "High throughput VLSI architecture for

one dimensional median filter", Signal Processing, Communications and Networking, ICSCN'08, InternationalConference, IEEE, vol.15, no. 8, pp. 339-344, 2008.

[20] Oflazer, Kemal., "Design and implementation of a single-chip 1-D median filter", IEEE transactions on acoustics, speech, and signal processing, vol.31, no. 5, pp. 1164-1168, 1983.

[21] I. Aizenberg and C. Butakoff, "Effective impulse detector based on rank-order criteria," IEEE Signal Process. Lett., vol. 11, no. 3, pp. 363–366, Mar. 2004.

[22] S. Zhang and M. A. Karim, "A new impulse detector for switching median filter," IEEE Signal Process. Lett., vol. 9, no. 11, pp. 360–363, Nov. 2002.

[23] W. Luo, "Efficient removal of impulse noise from digital images," IEEE Trans. Consum. Electron., vol. 52, no. 2, pp. 523–527, May 2006.

[24] P. Deepa and C. Vasanthanayaki, "VLSI Implementation of Enhanced Edge Preserving Impulse Noise Removal Technique," 26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems, Pune, pp. 98-102, 2013.

[25] W. Luo, "An efficient detail-preserving approach for removing impulse noise in images," IEEE Signal Process. Lett., vol. 13, no. 7, pp. 413–416, Jul. 2006.

[26] K. S. Srinivasan and D. Ebenezer, "A new fast and efficient decision based algorithm for removal of high-density impulse noises," IEEE Signal Process. Lett., vol. 14, no. 3, pp. 189–192, Mar. 2007.

[27] H. Yu, L. Zhao, and H. Wang, "An Efficient Procedure for Removing Random-Valued Impulse Noise in Images," IEEE Signal Processing Letters, vol. 15, pp. 922-925, 2008.

[28] P. Y. Chen, C. Y. Lien, and H. M. Chuang, "A Low-Cost VLSI Implementation for Efficient Removal of Impulse Noise," IEEE Trans. Very Large Scale Integration Systems, vol. 18, no. 3, pp. 473- 481, Mar. 2010.

[29] Y. C. Pei, "A low-cost VLSI implementation for efficient removal of impulse noise", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 18, no. 3, 2010.

[30] Y. C. Pei, "VLSI implementation of an edge-oriented image scaling processor", IEEE Transactions on very Large Scale Integration (VLSI) Systems, vol. 17, no. 9, 2010.

[31] Y. L. Chih, C. H. Chien, and Y. C. Pei, "An efficient denoising architecture for removal of impulse noise in images", IEEE Transactions on Computers, Vol. 6, no. 4, 2013.

[32] M. Y. Emin, and B. Erkan, "A simple neuro-fuzzy impulse detector for efficient blur reduction of impulse noise removal operators for digital images", IEEE Transactions on Fuzzy Systems, vol. 12, no. 6, 2004.

[33] L. C. Shih, "VLSI implementation of a low-cost high-quality image scaling processor", IEEE Transactions on Circuits and Systems, Express Brief, vol. 60, no. 1, 2013.

[34] K. Vasanth, S. N. Raj, S. Karthik and P. P. Mol, "Fpga implementation of optimized sorting network algorithm for median filters," INTERACT-2010, Chennai, pp. 224-229, 2010.

[35] C. Yang, Y. Li, W. Zhong and S. Chen, "Real-time hardware stereo matching using guided image filter," 2016 International Great Lakes Symposium on VLSI (GLSVLSI), Boston, MA, pp. 105-108, 2016.

[36] Shih-Chang Hsia, "Parallel VLSI design for a real-time video-impulse noise-reduction processor," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 11, no. 4, pp. 651-658, Aug. 2003.

[37] A. A. Hiasat, M. M. Al-Ibrahim and K. M. Gharaibeh, "Design and implementation of a new efficient median filtering algorithm," in IEE Proceedings - Vision, Image and Signal Processing, vol. 146, no. 5, pp. 273-278, Oct 1999.

[38] B. E. Caroline, G. Sheeba, J. Jeyarani and F. Salma Rosline Mary, "VLSI implementation and performance evaluation of adaptive filters for impulse noise removal," 2012 International Conference on Emerging Trends in Science, Engineering and Technology (INCOSET), Tiruchirappalli, Tamilnadu, India, pp. 294-299, 2012.

[39] R. Tapu, B. Mocanu, M. Raducanu and T. Petrescu, "Multiresolution median filtering based video temporal segmentation," ISSCS 2011 - International Symposium on Signals, Circuits and Systems, lasi, pp. 1-4, 2011.

[40] K. Vasanth and S. Karthik, "FPGA implementation of modified decomposition filter," 2010 International Conference on Signal and Image Processing, Chennai, pp. 526-530, 2010.

[41] O. KalaiPriya, S. Ramasamy and D. Ebenezer, "Vlsi implementation of nonlinear variable cutoff high pass filter algorithm," 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, pp. 275-278, 2011.

[42] A. Burian, J. Takala and M. Ylinen, "Design and implementation of a median filtering algorithm using the median cost function," 3rd International Symposium on Image and Signal Processing and Analysis, 2003. ISPA 2003. Proceedings of the, Vol. 2, pp. 961-965, 2003.

[43] H. Hwang and R.A. Haddad, "Adaptive Median Filters: New Algorithms and Results," IEEE Trans. Image Processing, vol. 4, no. 4, pp. 499-502, Apr. 1995.

[44] Z. Wang and D. Zhang, "Progressive Switching Median Filter for the Removal of Impulse Noise from Highly Corrupted Images," IEEE Trans. Circuits Systems II, Analog Digital Signal Processing, vol. 46, no. 1, pp. 78-80, Jan. 1999.

[45] E. Abreu, M. Lightstone, S.K. Mitra, and K. Arakawa, "A New Efficient Approach for the Removal of Impulse Noise from Highly Corrupted Images," IEEE Trans. Image Processing, vol. 5, no. 6, pp. 1012-1025, June

1996.

[46] Ren-Der Chen, Pei-Yin Chen, Chun-Hsien Yeh "A Low-Power Architecture for the Design of a One-Dimensional Median Filter" IEEE transactions on circuits and systems—ii: express briefs, vol. 62, No. 3, PP. 266-270, 2015.

[47] K.Tulasiram, Y.Rajasree rao, K.Anithasheela "Area Efficient Implementation of One-Dimensional Median Filter using BEC CSLA"International journal of control theory and applications,vol.10, no.22,pp 177-187, 2017