using Multiplexer based Data Comparator

Yamuna,
PG Student ECE,
Malla Reddy Engineering College
(Jawaharlal Nahru Technological University)
Hyderabad, India

Dr M Jagadeesh Chandra Prasad
Professor and Head ECE
Malla Reddy Engineering College
(Jawaharlal Nehru Technological University)
Hyderabad, India

***Abstract*—Filters are used to remove the different types of noises including salt and pepper, gaussian, and random noises from image. Therefore, the VLSI oriented hardware implementation of filters plays the crucial role in real time applications. However, the conventional hardware-based filters are failed to reduce the look-up-table (LUT)s, path delays, and power consumption. Therefore, this work is focused on implementation of Hybrid Median Filter (HMF) using Data Comparator (DC) logic. Initially, the multiplexer selection logic-based data comparator is used to identify the high and low values from two numbers. Then, data comparator is repeated for multiple number of times for nine pixels combinations, which identifies the median value from nine pixels. The subjective and objective evaluation shows that the proposed HMF-DC resulted in superior performance in terms reduced noise, hardware metrics like LUTs, delay, and power consumption as compared to state of art approaches.**

Keywords— Hybrid Median Filter, Data Comparator, multiplexer selection logic, look-up-tables, power consumption.

I. INTRODUCTION

Noise is undesirable information which degrades image quality [1]. The image can be noisy because of dust present on the lens, electronic noise in camera, imperfection present in the image sensor or can be introduced when image data is transmitted over communication channel. The motive of image processing is to get rid of noise from a digital image while keeping its features unaltered. Image filter is the key blog of Image processing system. An impulsive noise can be added when image data transmitted over an insecure communication channel. [2]. It causes small size dots or dark/black spot on an image. Impulse noise is uniformly distributed and the most often mentioned noise in digital images. Further, Impulse noise can be divided into two parts. The first one is salt and pepper noise which is a type of impulse noise having noisy pixel intensity either 0 (minimum) or 255 (maximum) in the case of gray scale images. It appears as randomly scattered black or white dots over the images [3]. The second one is the random-valued shot noise which has arbitrary valued noisy pixels. To remove these noises, it is necessary that the acquired image must pass through an image pre-processing stage defined as a filter [4]. Spatial and frequency domain are two categories of the filtering operation. Generally, filters are implemented by MATLAB, OCTAVE software's in real time systems [5]. As it is a well-known fact that software implementation offers less processing speed in comparison to hardware implementation [6]. Hardware implementation has become better alternative after the boost in the VLSI technology. To reduce the power consumption in the systems, more cooling devices have to be incorporated results in the costly system. Keeping the same functional capabilities with the reduction in power factors are heavily demanding. Yet in that context, battery and power optimizing technology have not matured up to that target. Most of these products include embedded microprocessors, DSPs and ASICs [7]. It is a provoking undertaking to accomplish low force plan of any VLSI circuit. There are various degrees of advancement in VLSI configuration measure for low force applications. For battery operated portable products [8], power has been the main concern. As System-on-Chip (SoC) developing with more power transistors, it requires less power consumption. Power consumption reduction in highly integrated SoC cut down the heating problem. It reduces the cost of expensive packing and cooling mechanism [9]. In this work, VLSI architecture for noise reduction in different imaging applications is proposed to deal with the above issues of power and cost reduction, respectively. To achieve low resources, this work mainly focusing on Verilog based coding mechanisms with FPGA prototype [10]. Then, the subjective and objective image statistics are measured by using MATLAB environment. The major contributions of this work are as follows:

- Implementation of data comparator for identifying the high, low values using multiplexer selection logic.
- Implementation of multi-level network for selection of median value from nine pixels in a window.
- Implementation of HMF-DC for removal of different types of noises from image using hybrid switching of data blocks.

Rest of the article is organized as follows: section 2 delas with literature survey, section 3 delas with the proposed HMF-DC implementation, section 4 delas with analysis of results with performance comparison, section 5 concludes the article with possible future directions.

II. LITERATURE SURVEY

In spatial domain filtering, the image pixel values are directly manipulated to achieve the desired result. Available spatial domain filters are mean, order statistics and adaptive filters [11]. Image filters have wide applications in the domain of image processing, satellite, and remote sensing, medical and microscopic imaging, geographic image

surveillance and seismographic analysis. In [12] authors proposed a new method of median filter by sharing Common Boolean Logic (CBL) which replaces RCA-XOR gate and inverter are used for the sum generation. AND gate and OR gate is used for carry generation [13]. Based on the specific carry input given to the multiplexer both the required sum and carry output are generated. An efficient design is proposed by partial sharing of the circuit and by logical simplification. This design leads to a decrease in the transistor count with minimum power dissipation. In [14] authors proposed the efficient Frequency Domain Denoising Filters by using a newly proposed new type of basic full adder. In [15] authors discussed about the implementation of median filter using basic logic gates. The main advantage of using the basic gates is its zero-power dissipation under ideal conditions. Design modifications are performed in the basic gates to reduce the garbage bits and constant inputs.

In [16] authors proposed a proficient Efficient Median Filter which replaces a BEC with normal Boolean logic in conventional CSLA. This work utilizes an effective CSLA by sharing the Common Boolean logic term [17]. One OR gate and one inverter are used for carry and sum generation. Multiplexer is used for selecting the required output based on appropriate carry. Power and delay are reduced with increase in area. In [18] authors proposed a low-cost image denoising standard median filter (SMF) methodology using CSLA without multiplexers. First carry input zero operation is performed followed by BEC adder operation. The circuit is designed such that BEC adder replaces the last MUX arrange utilized in customary methodology. Replacing the MUX stage will lessen the area and delay to give considerably higher execution for the adder. In [19] authors implemented parallel pipeline median finder using ultra low power design in near threshold region. Sub threshold operation is similar to minimum energy operation. This work deals with the energy delay modeling framework that develops in the weak, moderate and strong inversion regions. The operation below the minimum energy point is also discussed [20]. The experimental results show that there is a 20% increase in energy which leads to better performance. This concept is used for comparing adders based on their energy delay characteristics and presents the results for our estimation technique. In [21] authors discuss area, power and delay performances of hybrid sorting-based dynamic median filtering (DMF) by using different CMOS logic styles. A new hybrid style is proposed for designing full adder. Though full adders are used in tree structured arithmetic circuits [22], new hybrid logic is used for simulation which is used in the application environment. A full swing and balanced output are achieved using this logic. An area efficient layout is achieved by this methodology.

In [23] authors designed low power median filter based on the variation in supply voltage for impulse noise suppression. Based on input vector pattern, supp.ly voltage is selected. This method will drastically reduce the power consumption [24]. This methodology is explained with respect to the prototype of 32-bit RCA. Simulation results show that there is a 29% reduction in power requirement when compared with conventional RCA In [25] authors designed adaptive median filter (AMF) with thresholding methods for low power applications. Designing the structure with single supp.ly voltage or comparison based on the gate count is not a suitable method for finding the optimal structures. Therefore, high performance structures should be combined with the supp.ly voltage scaling for obtaining a reduced energy. These technology outlooks the traditional design for low power operation.

III. Proposed Method

Noise is signal-subordinate and is hard to be eliminated without disabling image subtleties. Various sorts of error influence the image, like Gaussian, drive, dot and Rician Noises. In the image denoising measure, data about the sort of error present in the original image assumes a huge part. The image error can be delegated either added substance or multiplicative. The image is a 2-D function $f(x,y)$ of light intensities, where f is amplitude at any spatial coordinate x and y. The beam of light falls on an object and reflected light reaches to eyes. It makes human to see the object. The smallest element of the image is pixels. Each pixel represents intensity value at a particular location. Mathematically, image can be represented as Equation (1).

$$F(x,y) = I(x,y).R(x,y) \quad (1)$$

Here, $I(x,y)$ is intensity of incident light on object, $R(x,y)$ is reflected light from object in intensity and $F(x,y)$ is intensity of resultant image. The image restoration is the process to denoising an image, which has been distorted by prior knowledge of degradation model. Once the degradation model is known, by applying inverse process to recover the desired imagery. image restoration is different than traditional image enhancement techniques. It is a subjective process, which produces more effective results to an observer with and without using degradation model. The image degradation process is shown in Figure 1. Image degradation model in the spatial domain is achieved by performing the convolution between $f(x,y)$ and model function ($h(x,y)$).

$$F(x,y) = h(x,y) * f(x,y) + \eta(x,y) \quad (2)$$

Here, $\eta(x,y)$ represents the speckle noise. Further, degradation model in the frequency domain is achieved by applying the Fourier transform as follows:

$$F(u,v) - h(u,v) * f(u,v) + \eta(u,v) \quad (3)$$

Here, u, v represents the frequency domain coefficients.

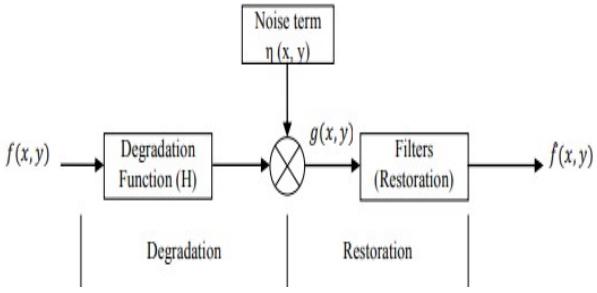

Figure 1: Proposed HMF-DC block diagram.

The HMF-DC is a digital non-linear method used to eliminate noise, similar to that of the medium filter. However, by keeping valuable details in the image, it typically does better than the mean filter. This filter class belongs to the class of filter that preserves the edge. These filters smooth down the data while maintaining the details. The median is only the average of all the pixel values in the area. It doesn't correspond to the mean (or average), but the median is half bigger and half smaller in the neighborhood. The median is a "centre indication" stronger than the

average. Like the medium, every pixel in the image is taken into account by the HMF-DC and its close neighbors are examined to determine if it is typical of their surroundings. It replaces the median value with those values instead of just replacing the pixel value by the mean of the next pixel value. Particularly better than the typical filter is to take away impulsive noise. The HMF-DC eliminates the noise as well as the fine details as the difference between them cannot be identified. Anything that is comparatively tiny in size with the area size will minimize and filter out the median value. In other words, the HMF-DC can differentiate between fine detail and noise.

A. Data comparator

Figure 2 shows the block diagram of data comparator, which is used to perform the selection of highest and lowest values from the given two input data. Further, the data comparator block contains inputs as A, B and outputs are High (H) and Low (L).

Step 1: Initially, A < B condition is verified, if condition is satisfied selection line of multiplexer becomes one, else condition failed selection line becomes zero.

Step 2: Input-A is applied as Data-input-0 and Input-B is applied as Data-input-1 to 2to1 multiplexer. If A value is smaller than B, then selection line becomes one and multiplexer generates H as input-B through selection switching. If A value is higher than B, then selection line becomes zero and multiplexer generates H as input-A through selection switching.

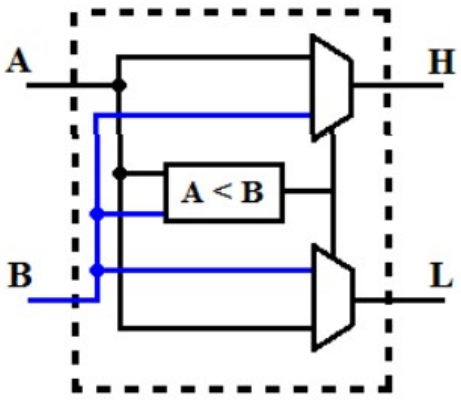

Figure 2. Block diagram of data comparator.

Step 3: Input-B is applied as Data-input-0 and Input-A is applied as Data-input-1 to 2to1 multiplexer. If A value is smaller than B, then selection line becomes one and multiplexer generates L as input-A through selection switching. If A value is higher than B, then selection line becomes zero and multiplexer generates L as input-B through selection switching.

B. Hardware architecture of HMF

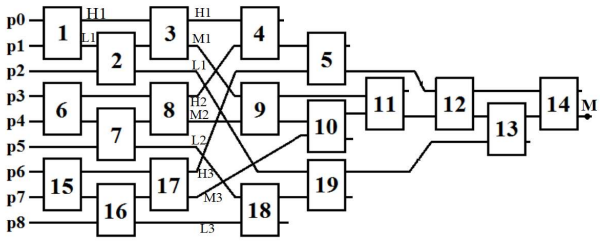

Figure 3. Hardware architecture of HMF-DC.

Figure 3 shows the hardware architecture of HMF-DC, which contains the fourteen number of hardware resource blocks. Here, inputs P0, P1, P2, P3, P4, P5, P6, P7, and P8 are applied to HMF-DC, which generates the median value as M. Here, DC-1, DC-2, DC-3 are grouped together and performs the selection of high (H1), low (L1) and median (M1) values. Similarly, DC-6, DC-7, DC-8 and DC-15, DC-16, DC-17 performs the generation of high, low and median values. Further, DC-4 is used to select the lowest value from H1, H2, H3 outcomes. Furthermore, DC-18 is used to select the highest value from L1, L2, L3 outcomes. Similarly, DC-9, DC-10, DC-11 are grouped together and performs the selection of high, low and median values. Like this, the process will continue and generates the median value (M) from DC-14 low outcome.

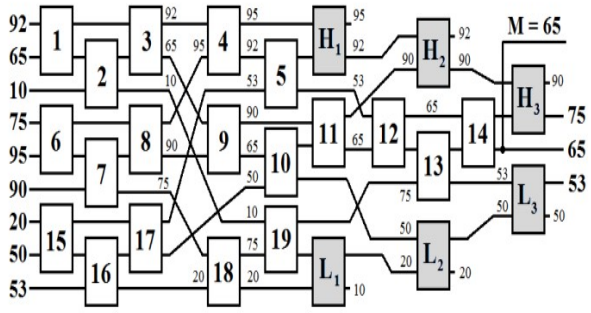

Figure 4. Example of HMF-DC.

Figure 4 provides a numerical example that might help better illustrate how the HMF-DC system works. In this case, the median value is defined by the two non-median outputs that are located the closest to it. As can be seen in this diagram, the H1, H2, and H3 blocks are used to sort the four highest pixel values (95, 92, 90, and 75), which results in 75 being the upper range. At the same time, the three lower ranges (L1, L2, and L3) are used to sort the four lowest values (10, 20, and 50), which results in 53 being the lower range.

IV. Results and Discussion

Xilinx ISE software was used to create all of the HMF-DC designs. This software programmed gives two types of outputs: simulation and synthesis. The simulation results provide a thorough examination of the HMF-DC architecture in terms of input and output byte level combinations. Decoding procedure approximated simply by applying numerous combinations of inputs and monitoring various outputs through simulated study of encoding correctness. The use of area in relation to the LUT count will be accomplished as a result of the synthesis findings. In addition, a time summary will be obtained with regard to various path delays, and a power summary will be prepared

utilizing the static and dynamic power consumption. Further, MatlabR2020a software is used to evaluate the subjective performance of HMF-DC.

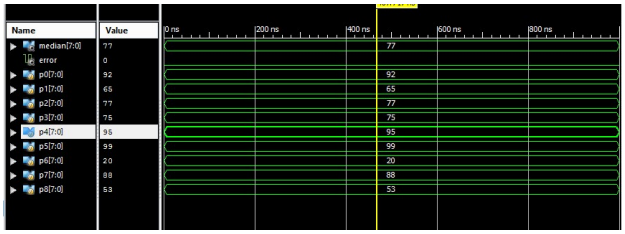

Figure 5. Simulation outcome of HMF-DC.

Figure 5 presents the simulation outcome of HMF-DC. Here, P0, P1, P2, P3, P4, P5, P6, P7, P8 are the inputs to HMF-DC and median is the output value.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice LUTs	437	303600	0%
Number of fully used LUT-FF pairs	0	437	0%
Number of bonded IOBs	81	700	11%

Figure 6. Design summary.

```
LUT6:I0->O            1    0.043   0.350  c14/a[7]_b[7]_LessThan_1_o22
LUT6:I5->O           14    0.043   0.422  c14/a[7]_b[7]_LessThan_1_o24
LUT4:I3->O            3    0.043   0.507  c14/a[7]_b[7]_LessThan_1_o25
LUT6:I3->O            1    0.043   0.613  H3/a[7]_b[7]_LessThan_1_o3 (H:
LUT6:I0->O            1    0.043   0.405  H3/a[7]_b[7]_LessThan_1_o4 (H:
LUT3:I1->O            2    0.043   0.410  H3/a[7]_b[7]_LessThan_1_o1_SW:
LUT5:I3->O            1    0.043   0.000  H3/a[7]_b[7]_LessThan_1_o1_G
MUXF7:I1->O           1    0.178   0.405  H3/a[7]_b[7]_LessThan_1_o1 (H:
LUT5:I3->O            6    0.043   0.631  H3/a[7]_b[7]_LessThan_1_o21 (I
LUT5:I0->O            1    0.043   0.613  H3/Mmux_13 (h31<2>)
LUT6:I0->O            1    0.043   0.405  median[7]_h31[7]_LessThan_1_o:
LUT5:I3->O            1    0.043   0.613  median[7]_h31[7]_LessThan_1_o:
LUT6:I0->O            1    0.043   0.339  error5 (error_OBUF)
OBUF:I->O                  0.000          error_OBUF (error)
----------------------------------------
Total                     20.375ns (1.726ns logic, 18.649ns route)
                                   (8.5% logic, 91.5% route)
```

Figure 7. Time summary.

Figure 6 shows the design (area) summary of proposed method. Here, the proposed method utilizes the low area in terms of slice LUTs i.e., 437 out of available 303600. Figure 7 shows the time summary of proposed method. Here, the proposed method consumed total 20.375ns of time delay, where 1.726ns is logical delay, and 18.649ns is route delay. Figure 8 shows the power consumption report of proposed HFE-DC. Here, the proposed HFE-DC consumed power as 32.83 milli watts.

```
2.  Summary
2.1.  On-Chip Power Summary
-------------------------------------------------------------------------
|                        On-Chip Power Summary                          |
-------------------------------------------------------------------------
|       On-Chip       | Power (mW) |  Used  | Available | Utilization (%) |
-------------------------------------------------------------------------
| Clocks              |       1.30 |      3 |    ---    |       ---       |
| Logic               |       0.00 |     10 |     11776 |               0 |
| Signals             |       0.00 |     20 |    ---    |       ---       |
| IOs                 |       0.00 |     20 |       372 |               5 |
| Quiescent           |      31.52 |        |           |                 |
| Total               |      32.83 |        |           |                 |
-------------------------------------------------------------------------
```

Figure 8. Power summary.

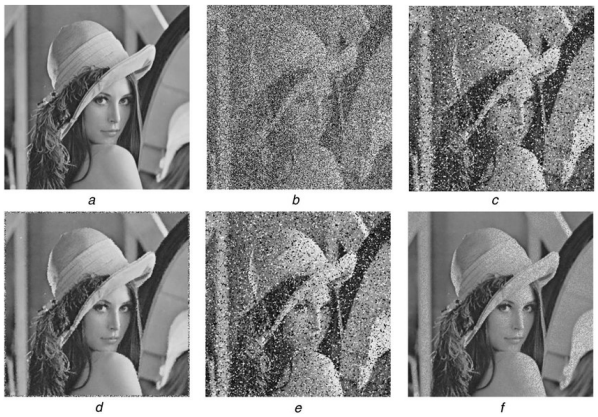

Figure 9. Visual performance of HMF-DC. (a) original image, (b) noisy image, (c) SMF [18], (d) DMF [21], (e) AMF [25], (d) proposed HMF-DC.

Figure 9 shows the filtering performance of various methods like SMF [18], DMF [21], AMF [25], and proposed HMF-DC. Here, SMF [18] and AMF [25] methods resulted outcome still contains the higher noises, DMF [21] method outcome contains the low level noises. But, the proposed HMF-DC method resulted outcome is looks similar to the original image. Table 1 compares the performance evaluation of proposed HMF-DC method. Here, the proposed HMF-DC resulted in superior (reduced) hardware performance in terms of LUTs, time-delay, and power consumption as compared to conventional approaches such as SMF [18], DMF [21], and AMF [25]. Further, the proposed HMF-DC resulted in improved subjective performance in terms of peak signal to noise ratio (PSNR), stctural similarity index metric (SSIM) as compared to conventional approaches such as SMF [18], DMF [21], and AMF [25]. Further, the graphical representation of performance comparison is presented in Figure 10.

Table 1. Performance evaluation.

Metric	**SMF [18]**	**DMF [21]**	**AMF [25]**	**Proposed HMF-DC**
LUTs	767	655	542	437
Time delay (ns)	51.927	43.837	32.735	20.37
Power consumption (mw)	82.61	73.41	58.26	32.83
PSNR (dB)	37.34	42.45	48.38	54.53
SSIM	0.827	0.893	0.927	0.992

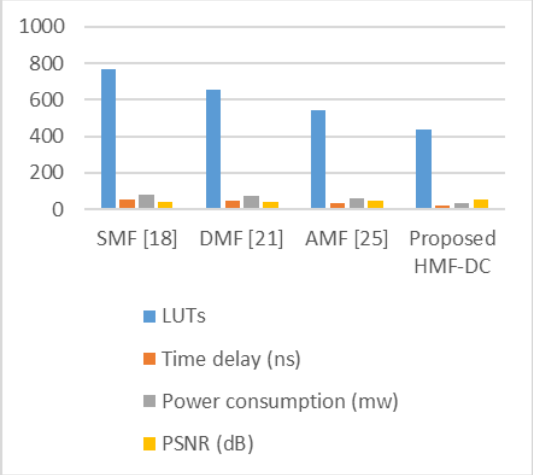

Figure 10. Graphical representation of performance.

V. Conclusion

The development of a Hybrid Median Filter by making use of Data Comparator logic is the primary emphasis of this study. In the beginning, a multiplexer selection logic-based data comparator is used in order to determine which of two numbers have high and low values. After then, the data comparator is carried out a number of times for the nine different possible combinations of pixels, which determines the median value for all nine of those values. The subjective and objective evaluations both reveal that the proposed HMF-DC resulted in greater performance when compared to the state-of-the-art techniques in terms of decreased noise, latency, and power consumption. Hardware metrics such as LUTs were also reduced and software metrics such as PSNR, SSIM are improved using the proposed HMF-DC approach. Further, this work can be extended with the hybrid adaptive filters for improved PSNR performance.

References

[1] Goel, Anish, M. Omair Ahmad, and M. N. S. Swamy. "Design of a 2D median filter with a high throughput FPGA implementation." *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2019.

[2] Ngo, Dat, Gi-Dong Lee, and Bongsoon Kang. "A 4K-Capable FPGA Implementation of Single Image Haze Removal Using Hazy Particle Maps." *Applied Sciences* 9.17 (2019): 3443.

[3] Rahnama, O., Cavallari, T., Golodetz, S., Tonioni, A., Joy, T., Di Stefano, L., ... & Torr, P. H. (2019). Real-time highly accurate dense depth on a power budget using an fpga-cpu hybrid soc. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *66*(5), 773-777.

[4] Sambamurthy, N., & Kamaraju, M. (2020). Power optimised hybrid sorting-based median filtering. *International Journal of Digital Signals and Smart Systems*, *4*(1-3), 80-86.

[5] Arnal, J., & Súcar, L. (2019). Hybrid filter based on fuzzy techniques for mixed noise reduction in color images. *Applied Sciences*, *10*(1), 243.

[6] Ahmad, Rais, and Som Pal Gangwar. "Rain Streaks Elimination Using Hybrid Median Filter and Contrast Stretching." *Advances in VLSI, Communication, and Signal Processing*. Springer, Singapore, 2021. 157-178.

[7] Rini, C., Perumal, B., & Rajasekaran, M. P. (2019). Eradication of Rician Noise in Orthopedic Knee MR Images Using Local Mean-Based Hybrid Median Filter. In *Proceedings of the 2nd International Conference on Data Engineering and Communication Technology* (pp. 693-702). Springer, Singapore.

[8] Banerjee, S., Sinha Chaudhuri, S., Mehra, R., & Misra, A. (2021). A Comprehensive Review of Median Filter, Hybrid Median Filter and Their Proposed Variants. *Advances in Smart Communication Technology and Information Processing*, 393-406.

[9] Elsharif, R. I., Ibraheem, B. A., Mustafa, Z. A., & Abass, S. K. (2018). Wavelet decomposition–based speckle reduction method for ultrasound images by using speckle-reducing anisotropic diffusion and hybrid median. *Journal of Clinical Engineering*, *43*(4), 163-170.

[10] Kannan, L. M., & Deepa, D. (2021). Low power very large scale integration (VLSI) design of finite impulse response (FIR) filter for biomedical imaging application. *DYNA-Ingeniería e Industria*, *96*(5).

[11] Yang, C. H., & Lin, C. H. (2020, October). Combined Architectures of Denoising Filter and Local Binary Patterns Feature Extraction. In *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)* (pp. 152-153). IEEE.

[12] Goel, A., Ahmad, M. O., & Swamy, M. N. S. (2019, August). Design of a 2D median filter with a high throughput FPGA implementation. In *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)* (pp. 1073-1076). IEEE.

[13] Pashaeifar, Masoud, Mehdi Kamal, Ali Afzali-Kusha, and Massoud Pedram. "Approximate reverse carry propagate adder for energy-efficient DSP applications." *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 26, no. 11 (2018): 2530-2541.

[14] Srinu, B., Bevara, S., & Kumar, M. N. (2019). FPGA Based Implementation of Median Filter using Compare and Exchange Unit. *i-Manager's Journal on Digital Signal Processing*, *7*(1), 33.

[15] Allwin Devaraj, S., Jacob, S., Darling Rosita, J., & Vijay, M. M. (2021). VLSI Implementation of Color Demosaicing Algorithm for Real-Time Image Applications. In *2nd EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing* (pp. 317-329). Springer, Cham.

[16] Kattekola, N., Jawale, A., Nath, P. K., & Majumdar, S. (2021). Efficient partial product reduction for image processing application using approximate 4: 2 compressor. *Circuit World*.

[17] Sindhana Devi, M., and M. Soranamageswari. "Efficient impulse noise removal using hybrid neuro-fuzzy filter with optimized intelligent water drop technique." *International Journal of Imaging Systems and Technology* 29.4 (2019): 465-475.

[18] Bose, Sudipta, Arijit De, and Indrajit Chakrabarti. "Area-Delay-Power Efficient VLSI Architecture of FIR Filter for Processing Seismic Signal." *IEEE Transactions on Circuits and Systems II: Express Briefs* 68.11 (2021): 3451-3455.

[19] Abubakar, A., Zhao, X., Takruri, M., Bastaki, E., & Bermak, A. (2020). A hybrid denoising algorithm of BM3D and KSVD for Gaussian noise in DoFP polarization images. *IEEE Access*, *8*, 57451-57459.

[20] Arjun, Jawale Amol, and Shubhankar Majumdar. "Development of approximate compressor based hybrid dadda multiplier for image de-noising applications." *2019 IEEE 16th India Council International Conference (INDICON)*. IEEE, 2019.

[21] Chen, Wei-Ting, Pei-Yin Chen, Yu-Che Hsiao, and Shih-Hsiang Lin. "A low-cost design of 2D median filter." *IEEE Access* 7 (2019): 150623-150629.

[22] Kumar, Rahul, et al. "A hybrid dehazing method and its hardware implementation for image sensors." *IEEE Sensors Journal* 21.22 (2021): 25931-25940.

[23] Gollamandala, Udaykiran Bhargava, Vasujadevi Midasala, and Vallabhuni Rajeev Ratna. "FPGA implementation of hybrid recursive reversable box filter-based fast adaptive bilateral filter for image denoising." *Microprocessors and Microsystems* 90 (2022): 104520.

[24] Lin, C., Chen, W. T., Chou, Y. C., & Chen, P. Y. (2019). A novel comparison-free 1-D median filter. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *67*(7), 1329-1333.

[25] Najafi, M. H., Lilja, D. J., Riedel, M. D., & Bazargan, K. (2018). Low-cost sorting network circuits using unary processing. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *26*(8), 1471-1480.