ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

FPGA-Based Fault-Tolerant Median Denoising Architecture for Impulse Noise Removal in Digital Images

Bhaskar Rao Palakurthi[1], Rajashekar Kakoju[2], Jyotsna Devi Jonnalagadda[3]

[1]*Associate Professor, Dept. of ECE, Brilliant Grammar School Educational Society's Group of Institutions, Hyderabad, India*

[2]*Assistant Professor, Dept. of ECE, Brilliant Grammar School Educational Society's Group of Institutions, Hyderabad, India*

[3]*Dept. of ECE, Brilliant Grammar School Educational Society's Group of Institutions, Hyderabad, India*

Abstract: *Impulse noise introduced during image acquisition can severely degrade subsequent vision operations, while hardware acceleration of denoising on SRAM-based field-programmable gate arrays (FPGAs) introduces a second reliability problem: configuration-memory single-event upsets can permanently modify the median-filter logic until the device is reconfigured. This paper presents a compact fault-tolerant median denoising architecture that combines a nine-pixel partial sorting network with a dynamic-range consistency checker. For every 3×3 neighborhood, the median is computed together with the closest lower and upper non-median bounds. A filter malfunction is reported whenever the computed median falls outside this dynamically derived interval, after which partial or complete FPGA reconfiguration can restore the correct configuration. The supplied implementation evaluates the design on 8-bit, 128×128 grayscale images and compares it with dual modular redundancy (DMR) and reduced precision redundancy (RPR). The reported results show that the proposed method requires 385 LUTs, corresponding to 35% logic overhead over the unprotected 286-LUT median filter, compared with 61% for RPR and 100% for DMR. Fault-injection experiments further report detection of 91% of corrupted images. The architecture therefore offers a practical balance between denoising capability, configuration-error awareness, and hardware cost for reliability-sensitive FPGA image-processing systems.*

Keywords: Impulse Noise, Median Filter, FPGA, Single-Event Upset, Fault Tolerance, Dynamic-Range Checking, Image Denoising, Reconfiguration.

I. INTRODUCTION

Digital image sensors commonly acquire noise together with the desired scene. Among the most disruptive acquisition artifacts is impulse or salt-and-pepper noise, which appears as isolated abnormally bright or dark pixels. If these pixels are propagated to later stages such as edge detection, feature extraction, segmentation, or object recognition, the resulting decisions can be degraded. Consequently, a denoising stage is often placed immediately after image capture [1].

The median filter is particularly suitable for impulse-noise suppression because it replaces a pixel by the median of a local neighborhood rather than by a linear weighted average. A 3×3 median operator can suppress isolated outliers while retaining object boundaries better than many linear smoothers. For real-time streams, however, the nine samples of every neighborhood must be compared and reordered at pixel rate, which motivates implementation in programmable logic [5].

SRAM-based FPGAs are attractive for this purpose because they provide high parallelism and can be reconfigured in the field. Their configuration memory is nevertheless vulnerable to single-event upsets

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

(SEUs). A bit flip may alter routing or logic functions and can therefore cause the median filter to produce incorrect pixels until the configuration is repaired. In radiation-prone environments, the denoising block must therefore be protected not only against noisy sensor data but also against faults inside the hardware that performs the denoising [2]-[6].

Conventional redundancy schemes provide protection by duplicating or replicating the complete processing core. DMR offers high observability but nearly doubles logic, while RPR reduces cost by checking only a lower-precision replica. The proposed architecture instead exploits a property of the median operation itself: in a correctly operating nine-input median network, the median must remain bounded by the nearest lower and upper non-median outputs. This invariant is used as a low-overhead online error detector.

II. MEDIAN FILTER AND FAULT MODEL

2.1 Impulse-Noise Suppression:

Let W(i,j) denote the 3×3 neighborhood centered at pixel (i,j). For an 8-bit grayscale image, each sample lies in the range 0 to 255. The filtered output is the fifth order statistic of the nine samples:

$$y(i,j) = median\{ x(i+m, j+n) \mid m,n \in \{-1,0,1\} \}$$

In hardware, complete sorting is unnecessary because only the fifth value is required. A partial exchange network therefore offers lower logic cost than a full nine-input sorter. The supplied design uses two input exchange nodes constructed from a comparator and two multiplexers and derives the median together with selected non-median outputs.

When impulse noise density becomes large, a 3×3 median filter can lose edge detail because several samples inside a neighborhood may themselves be corrupted. The present work therefore focuses on the common low-to-moderate impulse-noise case and on ensuring that the hardware implementation does not introduce additional corruption.

2.2 Configuration-Memory Upset Model:

The median network is implemented as combinational logic in an SRAM-based FPGA. An SEU in configuration memory can change a look-up-table truth table, a multiplexer setting, or routing connectivity. Such a fault can produce two broad effects: the median value itself may be miscalculated, or internal non-median values may be altered while the nominal median path remains intact. Since the configuration bit remains corrupted, the resulting behavior persists until the configuration is rewritten.

The protection objective is therefore to identify corrupted output as early as possible and assert an error indication that can trigger partial or complete reconfiguration. Unlike register-level transient masking, the proposed scheme does not attempt to hide a permanent configuration error indefinitely; it detects the malfunction and supports recovery.

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

III. PROPOSED FAULT-TOLERANT DENOISING ARCHITECTURE

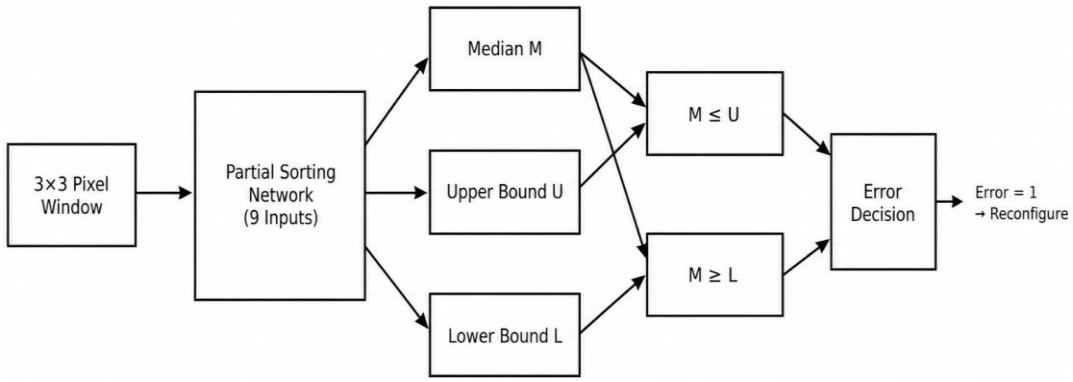

Figure 1: Functional organization of the proposed fault-tolerant median denoising architecture

The architecture accepts nine 8-bit pixels from a 3×3 neighborhood. The partial sorting network produces the median M and two bounds derived from the remaining values. The upper bound U is the smallest value among the four samples greater than the median, while the lower bound L is the largest value among the four samples lower than the median. For a correctly operating network, the following consistency relation must hold:

$$L \leq M \leq U$$

Two 8-bit comparators evaluate the upper and lower inequalities. The error flag is asserted if either condition is violated:

$$E = \neg(M \leq U) \vee \neg(M \geq L)$$

This checker reuses information naturally available from the sorting process and therefore avoids a full duplicate median filter. A useful numerical case from the supplied design contains the ordered neighborhood values 10, 20, 50, 53, 65, 75, 90, 92, and 95. Here M=65, L=53, and U=75; hence the output is valid only while the computed median stays within [53,75].

Not all faults are detectable. If a configuration upset changes the median to another value that still lies inside the dynamic interval, the checker may not assert. The experiments supplied indicate that the impact of these in-range errors is comparatively low. The architecture therefore targets high image-level protection at substantially lower logic cost than complete duplication.

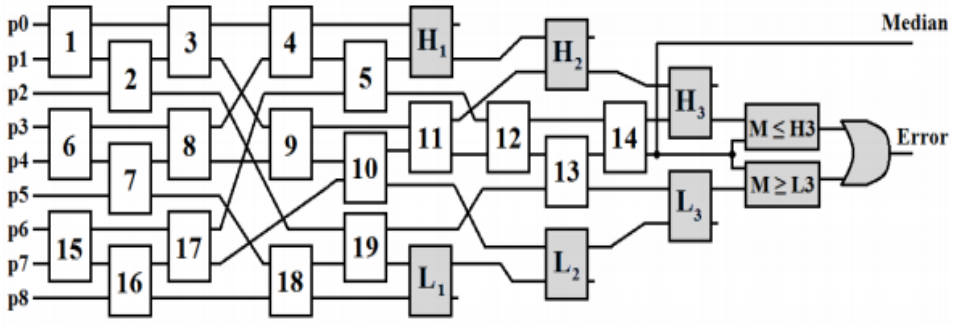

Figure 2: Exchange-network realization of the dynamic-range checking concept from the supplied implementation

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

IV. FPGA IMPLEMENTATION AND EVALUATION METHOD

The supplied design evaluates an 8-bit median filter on 128×128 grayscale images and implements the protection logic in the SRAM-based FPGA fabric of a Xilinx Zynq-7000 SoC. Three protected configurations are considered: DMR, RPR, and the proposed dynamic-range checker. The original unprotected median filter is used as the area baseline.

DMR uses two complete median-filter copies and compares their outputs. RPR uses a reduced-precision copy in which only the four most significant bits of each 8-bit pixel are processed; therefore, changes confined to the least significant bits can escape detection. The proposed design processes the full 8-bit median but adds only the boundary-generation and checking logic.

4.1 Exhaustive Configuration-Bit Fault Injection:

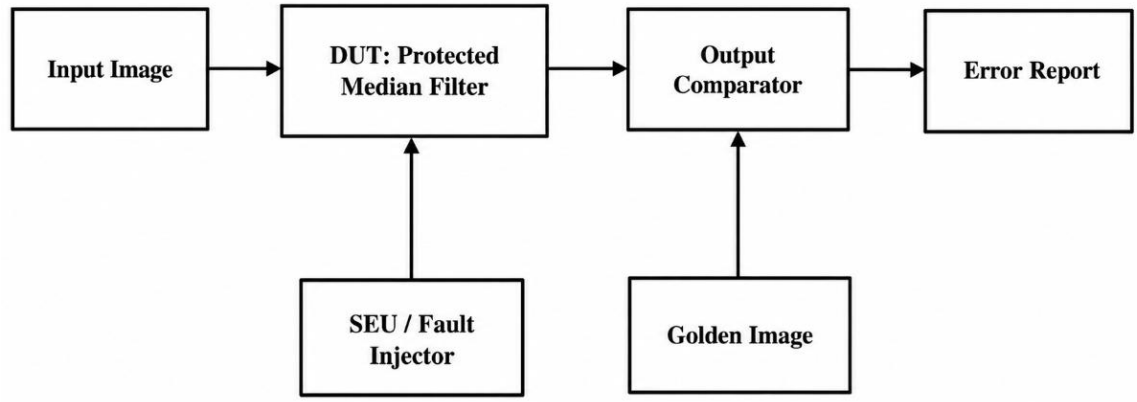

Figure 3: Fault-injection and classification framework used for reliability evaluation

Golden filtered images are first generated with an error-free configuration. The Xilinx Soft Error Mitigation (SEM) controller is then used through the internal configuration access port to flip a design-under-test configuration bit. The image is processed, the output is compared with the golden image, and the result is classified as detected error, undetected error, false positive, or normal operation. The injected bit is then corrected, and the procedure is repeated over the DUT-related configuration space.

Evaluation is performed at two levels. The image-level report records whether a corrupted image is detected before it is forwarded to the next processing stage. The pixel-level report measures the number of corrupted pixels and provides insight into the severity of undetected faults. Mean-squared error (MSE) is used in the supplied analysis to compare the residual damage in undetected images.

4.2 Hardware-Cost Metrics:

For a protection technique using Np LUTs compared with an unprotected design using N0 LUTs, the logic overhead is expressed as:

$$Overhead\ (\%) = \left(\frac{(Np\ -\ N0)}{N0}\right) \times\ 100$$

 DOI: 10.5281/zenodo.22773697

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

The supplied project reports LUT utilization for all four architectures, enabling a direct hardware-cost comparison.

V. RESULTS AND DISCUSSION

Table 1: FPGA Logic-utilization comparison reported in the supplied implementation

Architecture	LUTs	Logic Overhead
Original median filter	286	0%
RPR	461	61%
DMR	572	100%
Proposed checker	385	35%

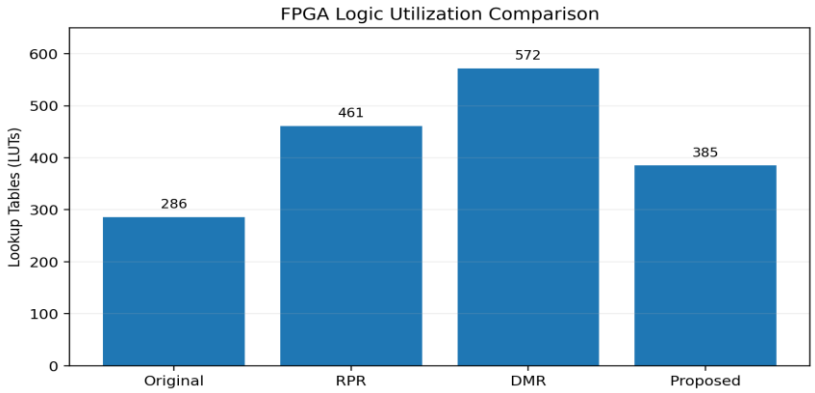

Figure 4: LUT utilization of the unprotected, RPR, DMR, and proposed architectures

The proposed design adds 99 LUTs to the 286-LUT unprotected filter, corresponding to 35% overhead. RPR requires 461 LUTs (61% overhead), while DMR reaches 572 LUTs (100% overhead). The main advantage of the dynamic-range method is therefore the ability to observe a large class of functional failures without replicating the complete filter.

At image level, the supplied fault-injection campaign reports that 91% of corrupted images are detected. Consequently, only 9% of corrupted images can pass without an error indication. The source further reports that these undetected images exhibit lower MSE than the corresponding DMR-undetected cases, indicating that the remaining errors tend to have reduced visual impact and lower propagation severity.

Pixel-level observations show that images containing larger homogeneous regions are less susceptible to visible corruption when a configuration fault occurs. Repeated or similar neighborhood values reduce the probability that an internal sorting fault changes the final fifth-order statistic. The reported tests also indicate that RPR sensitivity changes with image content, whereas the proposed checker maintains a more stable detection behavior because it derives its limits directly from each full-precision neighborhood.

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

5.1 Supplied RTL and Timing Snapshot:

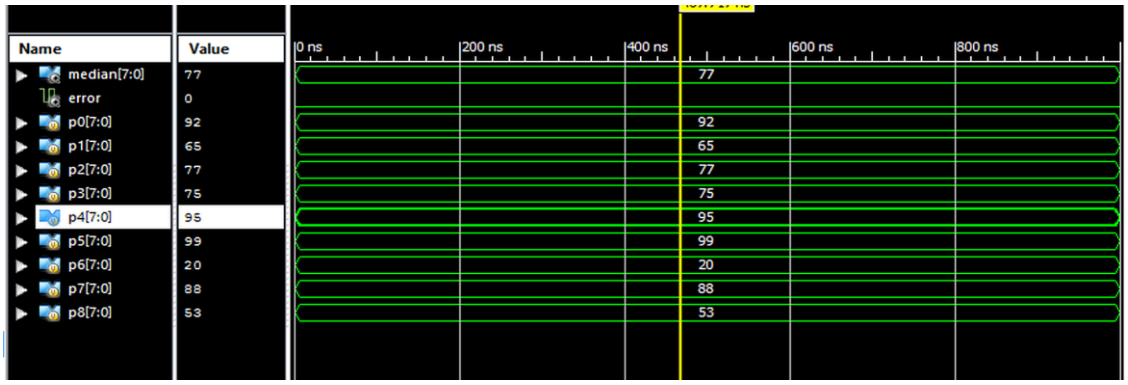

Figure 5: Simulation waveform from the supplied median-filter implementation

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: Device-utilization summary from the supplied FPGA implementation

```
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: Timing-path summary from the supplied FPGA implementation

The implementation screenshots supplied with the project report show a simulation in which the median output is 77 for the displayed nine-pixel input set and the error output remains deasserted. A separate utilization summary reports 437 Slice LUTs and 81 bonded I/O blocks for that synthesized build. The timing report shows a total path delay of 20.375 ns, composed of 1.726 ns logic delay and 18.649 ns routing delay.

The 437-LUT implementation snapshot is higher than the 385-LUT value in the comparative utilization table. Because the supplied document does not explicitly reconcile the tool settings, device mapping, or design revision associated with these two reports, the values are reported separately rather than treated as a direct inconsistency. For publication-quality benchmarking, all compared architectures should be re-synthesized with the same FPGA part, constraints, and tool flow.

ISSN (Online): 2583-648X

IJIRID

International Journal of Ingenious Research, Invention and Development

An International, High Impact Factor, Double-Blind Peer-Reviewed, Open-Access, Multidisciplinary Online Journal

Volume 5 | Issue 4 | August 2026

www.ijirid.in

VI. CONCLUSION

This work presents a compact FPGA-based denoising architecture that combines impulse-noise suppression with online detection of median-filter malfunctions caused by configuration-memory upsets. The key idea is to avoid complete hardware duplication and instead verify an invariant of the median operation: the output median must remain between the closest lower and upper non-median values generated by the partial sorting network.

Using the supplied implementation data, the proposed design requires 385 LUTs, only 35% more logic than the original 286-LUT filter, compared with 61% overhead for RPR and 100% for DMR. The reported fault-injection campaign detects 91% of corrupted images before they are propagated to subsequent processing. Although in-range median changes can remain undetected, the supplied analysis indicates lower residual MSE for the remaining 9% and more stable detection behavior than RPR across different image contents.

The architecture is therefore suitable for reliability-aware image preprocessing where FPGA resources are constrained and rapid recovery by partial or complete reconfiguration is available. Future work should unify all synthesis runs under a single tool flow, evaluate larger image resolutions and higher impulse-noise densities, and investigate adaptive bounds or lightweight temporal checking to improve detection of in-range faults without approaching the hardware cost of full duplication.

REFERENCES

[1] L. A. Aranda, P. Reviriego, and J. A. Maestro, “Error detection technique for a median filter,” IEEE Transactions on Nuclear Science, vol. 64, no. 8, pp. 2219–2226, Aug. 2017, doi: 10.1109/TNS.2017.2666843.

[2] M. Senthil Kumar and M. Javeed, “Design of NoC router with 3PE, double and triple error detection by using improved Hamming code,” ARPN Journal of Engineering and Applied Sciences, vol. 14, no. 16, pp. 2874–2880, Aug. 2019.

[3] L. A. Aranda, P. Reviriego, and J. A. Maestro, “A fault-tolerant implementation of the median filter,” in Proc. European Conference on Radiation and Its Effects on Components and Systems (RADECS), 2016, doi: 10.1109/RADECS.2016.8093153.

[4] M. Javeed and M. Senthil Kumar, “Automatic brain tumour detection using new structure algorithm,” International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, no. 11, pp. 3402–3405, Sep. 2019.

[5] R. C. Baumann, “Soft errors in advanced computer systems,” IEEE Design & Test of Computers, vol. 22, no. 3, pp. 258–266, May/Jun. 2005.

[6] MD. Javeed and F. Jabeen, “FPGA implementation for the linking of cell tracks using new structure algorithm,” International Journal of Engineering and Advanced Technology (IJEAT), vol. 9, no. 1, pp. 6773–6778, Oct. 2019.

[7] MD. Javeed and C. F. Jabeen, “Linking of cell tracks using segmentation,” International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 1, pp. 3020–3026, Nov. 2019.

