2nd Workshop on Signal and Document Processing
March 23-24, 2012 - Hammamet - Tunisia

# Study and Design of Median Filter

Elmoncef Benrhouma
RL-SIIT
National School of Engineering of Tunis, Tunisia
benrhouma.moncef@gmail.com

Meddeb Souad
RL-SIIT
National School of Engineering of Tunis, Tunisia
mmemeddeb@gmail.com

Abdulqadir Alaqeeli
King Abdulaziz City for Science and Technology (Kacst),
Riyadh, KSA
aalaqeeli@kacst.edu.sa

Hamid Amiri
RL-SIIT
National School of Engineering of Tunis, Tunisia
hamidlamiri@yahoo.com

*Abstract*— **Signal Processing is a very advanced which comes very close to its apogee by the undeniable attempts to reach the best performances. This process is noticed by the orientation with the digital signal processing. This article aims to study and design a median filter. Firstly, we highlighted the behavioral study for this filter using MATLAB software. Secondly, we passed to the functional study of the median with MODELSIM software (VHDL).**

***Keywords- Median Filter; Signal Processing; VHDL.***

## I. INTRODUCTION

The digitized image processing has become in recent years a very important area of research both for image restoration, image coding or feature extraction for pattern recognition features. Whatever the sensor used, especially when it comes to a video camera, the quality of acquired images is often affected. In general, the images are degraded by the presence of noise, and the other edges or ridges forms present in some areas may fade. To remedy this situation, many filtering techniques have been proposed. The objective of the digitized image filtering is to eliminate the noise in order to make more uniform gray levels while preserving the main features such as edges or ridges. The techniques commonly used include replacing the gray level of each point of the image processed by the value resulting from an analysis of local neighborhood of this point [1, 2].

This analysis can be predefined, this is the case when using a convolution mask whose size and the coefficients are set at the outset. But, although the noise is eliminated relatively well, it results an attenuation of the transition lines or ridge lines that can be fatal for the extraction of primitives.

In this context, falls the design of median filter. In fact, median filtering is used extensively in smoothing and de-noising applications for images and video [3]. It is a cost-effective solution used predominantly in video pre- and post-processing systems. It is also deployed extensively in real-time vision systems and automatic target recognition (ATR) systems [4].

## II. METHODS

### A. *Behavior study*

The median filter, a sub-class of the rank order filter sorts the pixels in a region by luminance, finds the median value and replaces the central pixel with that value [5,6,7]. Used to remove noise from images, this operation completely eliminates extreme values from the image. Rank order filtering is a class of operators that use neighborhood pixels to perform comparisons and ranking.
Rank operations also include the maximum and minimum operators, which find the brightest or darkest pixels in each neighborhood and place that value into the central pixel. By loose analogy to the erosion and dilation operations on binary images, these are sometimes called grey scale erosion and dilation [8].

One important variable in the use of a rank operator is the size of the neighborhood. Generally, rectangular (for convenience of computation) or circular (to minimize directional effects) shapes are used. As the size of the neighborhood is increased, however, the computational effort in performing the ranking increases rapidly.

In fact, a median filter is a non-linear digital filter which is able to preserve sharp signal changes and is very effective in removing impulse noise (or salt and pepper noise). An impulse noise has a gray level with higher or lower value that is different from the neighborhood point. Linear filters have no ability to remove this type of noise without affecting the distinguishing characteristics of the signal. Median filters have remarkable advantages over linear filters for this type of noise.

Therefore median filter is very widely used in digital signal and image/video processing applications [9].

A standard median operation is implemented by sliding a window of odd size (e.g. 3x3 window) over an image. At each window position the sampled values of signal or image are sorted, and the median value of the samples replaces the sample in the center of the window as shown in Fig.1.

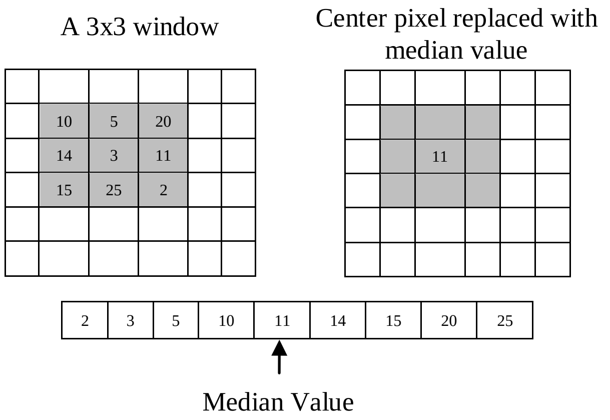


Figure 1. Median Filter.

Median filters remove isolated pixels, whether they are bright or dark. Prior to any hardware design, the software versions of the algorithms are created in MATLAB. Using MATLAB procedural routines to operate on images represented as matrix data, these software algorithms were designed to resemble the hardware algorithms as closely as possible. While a hardware system and a matrix- manipulating software program are fundamentally different, they can produce identical results, provided that care is taken in development. This approach was taken because it speeds understanding of the algorithm design. In addition, this approach facilitates comparison of the software and synthesized hardware algorithm outputs [10].

### *B. Functional study*

In this part, we used two strategy of median filter, the parallel sorting strategy and the minimum exchange strategy.

*a) Parallel Sorting Strategy :* The parallel strategy leads to a significant reduction compared to the wave sorter approach.

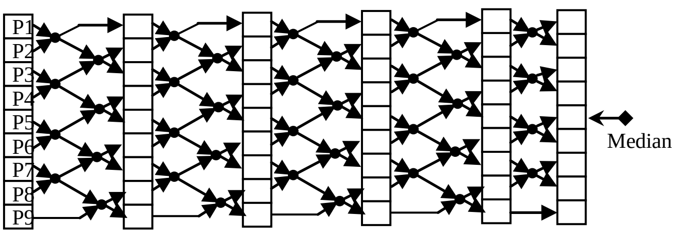


Figure 2. Structure of the parallel sorting strategy.

Furthermore, in additional sorts the necessary number of steps for sorting is equal to the number of characters in the biggest group of identical characters divided by 2 (remember that an additional sorting is implied if groups of identical adjacent characters appear in the array). This implies that in practice, it is possible to reduce more than the number of steps to solve the suffix problem.

*b) Minimum Exchange Strategy:* The graph of Fig.3 shows the minimum exchange network required to produce a median from nine input pixels by performing a partial sort. Each node is a two element sort, with the lower input exiting the node on the left, the higher input leaving on the right.

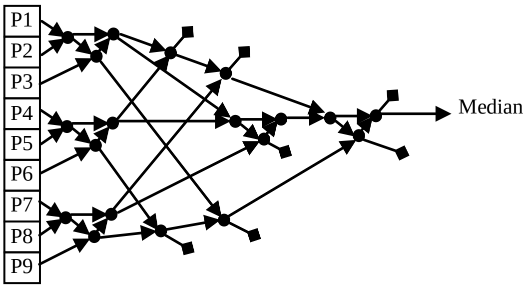


Figure 3. Structure of the Minimum Exchange Strategy.

In this functional study, the two strategies are based on principal element. It is the comparison module (CM). The figure Fig.4 shows this module.

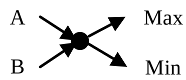


Figure 4. Comparison Module.

Firstly, in order to explain the functioning of this module, A and B are two 8 bits pixels. We will compare every time two bits so we need to 8 modules unitary comparison module (UCM). MCU is based to the following logic table:

TABLE I. LOGIC TABLE OF MCU

| $Te_i$ | $a_i$ | $b_i$ | $rmx_i$ | $re_i$ |
|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 1 |
| **1** | 0 | 1 | 0 | 0 |
| **1** | 1 | 0 | 1 | 0 |
| **1** | 1 | 1 | 0 | 1 |

We will compare two bits $a_i$ and bi if the bits $a_i$+1 and $b_i$+1 are equal (It is mean that $te_i$= $re_i$+1 = 1). For that, we supposed that the values of $te_i$ are "1" in this table.

In addition, $rmx_i$=1 only if $a_i$>$b_i$. Based on the logic table, we have these equations:

$$rmx_i = te_i \bullet a_i \bullet b_i \quad (1)$$

And

$$re_i = te_i \bullet \left(a_i \bullet b_i + \overline{a_i} \bullet \overline{b_i}\right) = te_i \bullet \left(a_i \odot b_i\right) \quad (2)$$

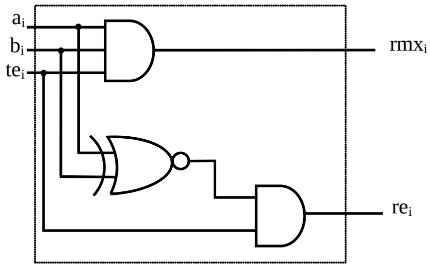


Figure 5. Logic Schematic of de UCM.

The figure Fig.6 shows the logic schematic of CM.

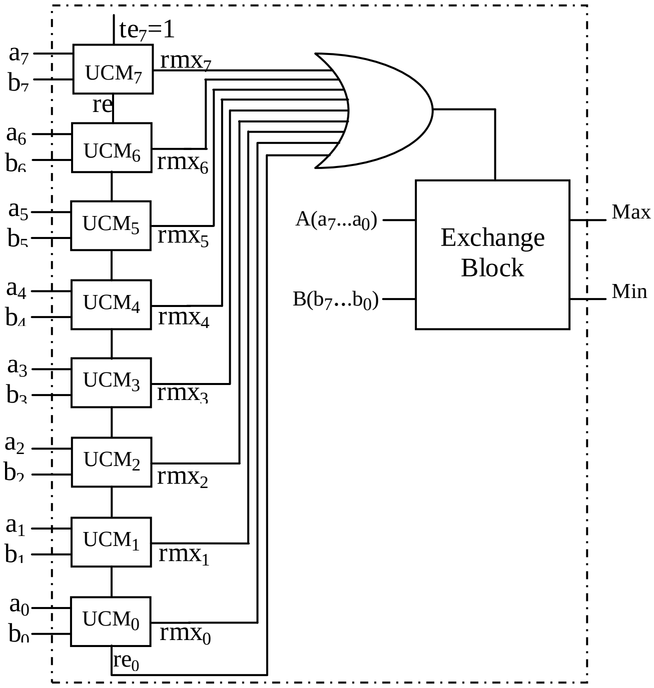


Figure 6. Structure of CM.

## III. RESULTS

Following the behavioral study and functional study, we had results and simulations by using MATALB and MODELSIM.

### *A. Results of Behavior Study*

The behavioral study is very important; it shows the operation of this filter and presents its effect on a noisy image. That is why we have taken any image, and then we have the noisy image. Figure Fig.1 shows the original image and the noisy image.

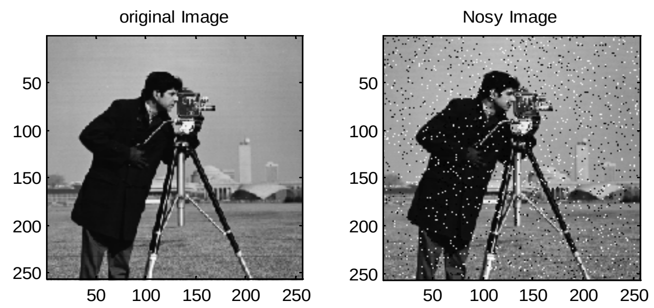


Figure 7. The original image and the noisy image.

Noise "pepper-and-salt" used here is to put, randomly, several pixels to the values 0 or 255 (range interval of gray levels). This type of impulse noise may appear by scanning an image or during a transmission. Figure Fig.2 shows the image obtained using the median filter.


Figure 8. Filtered image by the median filter.

Impulse noise is visibly reduced. Median filtering of a pixel P, on a neighborhood V(P) of size (MxN), directs the pixel values of V(P) in ascending order, and assigned the median value at the output of this neighborhood to the pixel P (non-linear operation).

Using the previous example: the pixel values are arranged in ascending order: 0, 8, 8, 8, 8, 8, 8, 8, 255. The median is 8. To this non-linear operation, the pulses 0 and 255 did not affect the median. Median filtering is therefore suitable for reducing impulse noise.

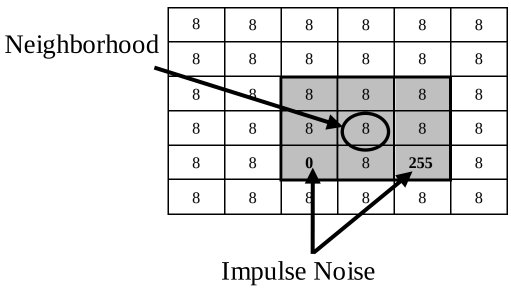


Figure 9. Example of a matrix of a noisy image by the noise "pepper and salt".

By increasing the filter size, the image becomes blurred. The following figure illustrates four images filtered by various median filters.

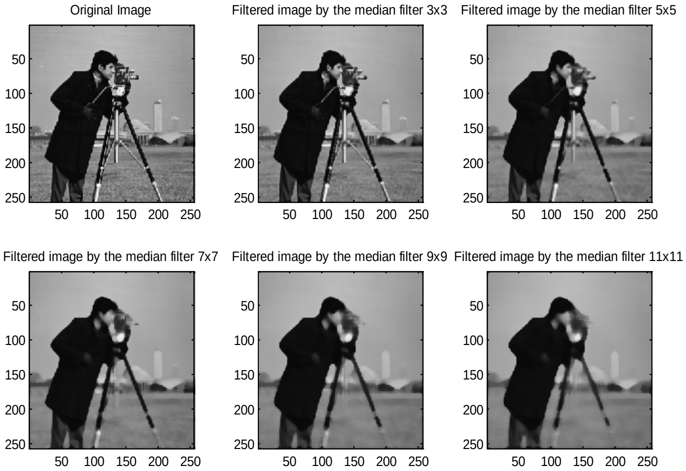


Figure 10. Filtered images by the various median filters.

### B. *Results of Functional Study*

We simulated each time a strategy. Indeed, we used an arbitrary matrix (3x3) and we applied our filter on this matrix and we had the following results using MODELSIM using VHDL.

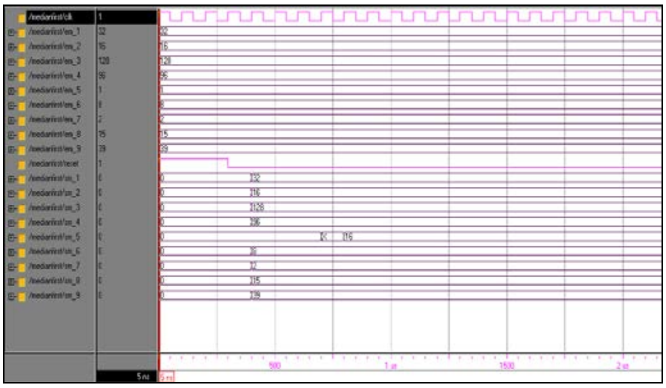

Figure 11. Simulation of the 1st strategy.

Figure 12. Simulation of the 2nd strategy.

## IV. Conclusion

In this paper, we have approached the study of architecture very important in the field of signal processing. This architecture called Median Filter, finds its application in various fields. In this study, we have achieved a behavioral study of this filter which allowed us to determine the suitable settings and the proper functioning of this filter. Based on these parameters established, we simulated the architecture designed by MODELSIM.

As perspective, we aim to optimize the architecture of our filter and study its performances. Then, choose the optimal and efficient structure. Finally, implement this architecture on an FPGA board.

## References

[1] W. K. PRATT, Digital Image Processing, Wiley, New York, 1978.

[2] A. ROSENFELD et A . C . KAK, Digital Picture Processing, Acad. Press. NY, 1976.

[3] V. Fischer, R. Lukac, and K. Martin, Cost-Effective Video Filtering Solution for Real-Time Vision Systems, EURASIP Journal on Applied Signal Processing 2005:13, 2036-2042.

[4] J. B. Wilburn, A Statistical Methodology for Automatic Target Recognition in Satellite Imagery.

[5] R. Roncella, R. Saletti, P. Terreni, 70-MHz 2-mm CMOS Bit-Level Systolic Array Median Filter, IEEE Journal of Solid State Circuits, vol 28, 1993.

[6] L. Chang and J. Lin, Bit Level Systolic Arrays for Real Time Median Filters, International Conference on Acoustics, Speech and Signal Processing, 1990.

[7] B. K. Kar and D. K. Pradhan, A New Algorithm for Order Statistic and Sorting, IEEE Transactions on Signal Processing, vol 41, August 1993.

[8] H. Heijmans (1991). Theoretic Aspects Of Grey-Scale Morphology. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 13(6):568-582.

[9] D.V. Rao, S. Patil, N. A. Babu and V. Muthukumar, Implementation and Evaluation of Image Processing Algorithms on Reconfigurable Architecture using C-based Hardware Descriptive Languages, International Journal of Theoretical and Applied Computer Sciences Vol.1 Number 1 (2006) pp. 9–34

[10] D.Dhanasekaran and K.B. Bagan, High Speed Pipelined Architecture for Adaptive Median Filter, European Journal of Scientific Research ISSN 1450-216X Vol.29 No.4 (2009), pp. 454-460