IJIRMET ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering and Technology
Vol. 6, Issue 4, April 2021

A Low-Power Architecture For The Design Of A One-Dimensional Median Filter

[1] S. Lakshmi,[2] P. Boopathi,[3] B.Karthikeyan

[1] Head Of The Department, Department Of Electronics And Communication Engineering, Thirumalai Engineering College, Kilambi, Kanchipuram- 631551

[2] Assistant Professor, Department Of Electronics And Communication Engineering, Thirumalai Engineering College, Kilambi, Kanchipuram- 631551

[3] M.E Vlsi Design, Department Of Electronics And Communication Engineering, Thirumalai Engineering College, Kilambi, Kanchipuram- 631551

Abstract: The median of a set of samples in the word-level sorting network is often computed by first sorting the input samples and then selecting the middle value. The power consumption is reduced by decreasing the number of signal transitions in the circuit. This can be done by keeping the stored samples immobile in the window through the use of a token ring in our architecture.

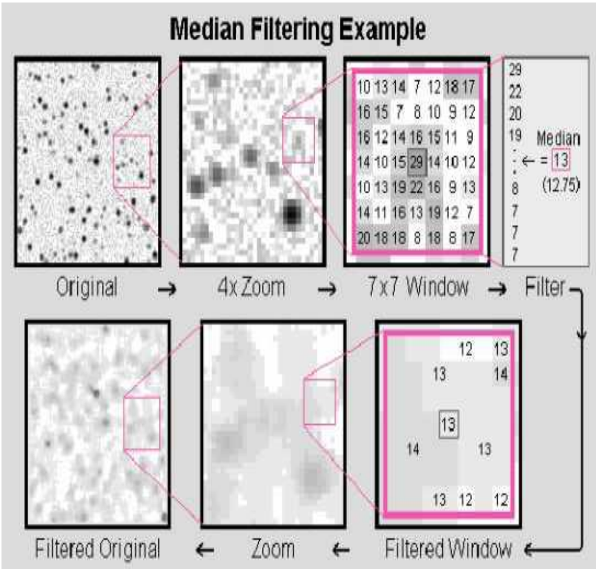
The experimental results have shown that, at the expense of some additional area cost, the power consumption can be successfully reduced. This paper proposes new architecture which is implemented as a two-stage pipeline, the median output, which is the sample with median rank, will also be generated at each cycle. The improvement in power consumption is achieved by utilizing a token ring in our architecture. Since the stored samples in the window are immobile, our architecture is suitable for low-power applications.

INTRODUCTION

To understand what adaptive median filtering is all about, one first needs to understand what a median filter is and what it does. In many different kinds of digital image processing, the basic operation is as follows: at each pixel in a digital image we place a neighborhood around that point, analyze the values of all the pixels in the neighborhood according to some algorithm, and then replace the original pixel's value
with one based on the analysis performed on the pixels in the neighborhood. The neighborhood then moves successively over every pixel in the image, repeating the process.

Median filter

Median filtering follows this basic prescription. The median filters normally used to reduce noise in an image, somewhat like the mean filter. However, it often does a better job than the mean filter of preserving useful detail in the image. This class of filter belongs to the class of edge preserving smoothing filters which are non-linear filters.
This means that for two images A(x) and B(x):These filters smoothes the data while keeping the small and sharp details. The median is just the middle value of all the values of the pixels5in the neighborhood. Note that this is not the same as the average (or mean); instead, the median has half the values in the neighborhood larger and half smaller. The median is a stronger "central indicator" than the average. In particular, the median is hardly affected by a small number of discrepant values among the pixels in the neighborhood. Consequently, median filtering is very effective at removing various kinds of noise. Figure 1 illustrates an example of median filtering
.



Like the mean filter, the median filter considers each pixel in the image in turn and looks at its nearby neighbors to decide whether or not it is representative of its surroundings. Instead of simply replacing the pixel value with the mean of neighboring pixel values, it replaces it with the median of those values. The median is calculated by first sorting all the pixel values from the surrounding neighborhood into numerical order and

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

Calculating the median value of a pixel neighborhood. As can be seen, the central pixel value of 150 is rather unrepresentative of the surrounding pixels and is replaced with the median value: 124. A 3×3square neighborhood is used here --- larger neighborhoods will produce more severe smoothing.

What is noise?

Noise is any undesirable signal. Noise is everywhere and thus we have to learn to live with it. Noise gets introduced into the data via any electrical system used for storage, transmission, and/or processing. In addition, nature will always plays a "noisy" trick or two with the data under observation. When encountering an image corrupted with noise you will want to improve its appearance for a specific application. The techniques applied are application-oriented. Also, the different7procedures are related to the types of noise introduced to the image. Some examples of noise are: Gaussian or White, Rayleigh, Shot or Impulse, periodic, sinusoidal or coherent, uncorrelated, and granular.

Noise Models

Noise can be characterized by its:

Probability density function (pdf): Gaussian, uniform, Poisson, etc.

Spatial properties: correlation

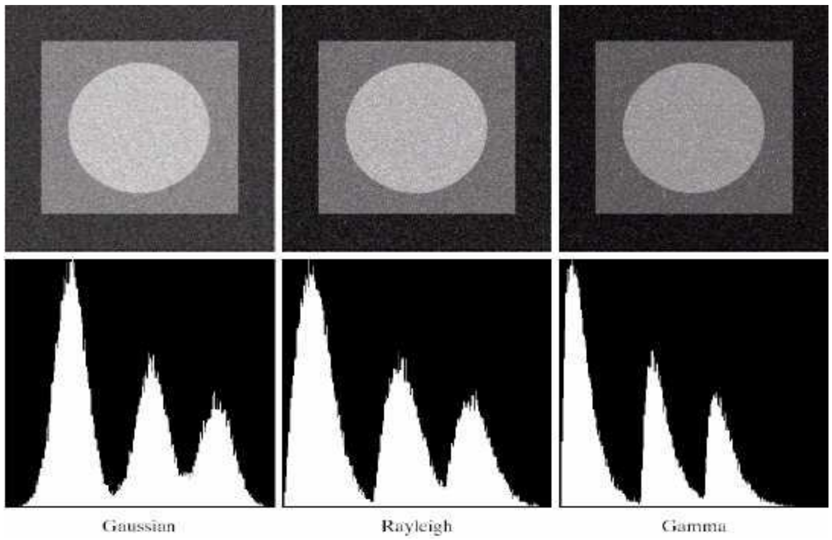


Figure 1.3: Gamma noise to the original image.

2. LITERATURE REVIEW
AUTHOR : Ren-Der Chen, Pei-Yin Chen.
YEAR : 2013
DESCRIPTION:

In this paper they present area-efficient 1-D median filter based on the sorting network for low complexity. It consists of N cascaded blocks, one for each window rank. Each cell block is composed of a data register (Ri), anm-bit (m = log 2 N) age register (Pi),and a control unit for data transfer. Register Ri stores the value of the sample in cell ci, and register Pi keeps the age of this sample, i.e., the number of clock cycles the sample has experienced in the window. All cells are connected to a global input X, through which they receive the new input sample.

At each machine cycle, depending on the value of the input sample, the data transfer control unit determines if each cell will keep its original sample, receive the input sample from the outside, or receive thesample from its adjoining cell on the left or right. The window samplesare stored in descending order from left to right, so that at any time, the maximum sample is at the leftmost register (R1) and the minimum sample is at the rightmost register (RN). The median value of the window will beat the center cell (R(N+1)/2),assuming N is an odd number When an input sample is inserted into the window along with some necessary shift operations, each cell in the window may keep it original sample, receive the input sample, or receive the sample from its adjoining cell on the left or right.17

Advantages:
1.Here by deleting the samples which are not in appropriate position in order to increase the throughput rate.
2. median output is generated at each cycle.
Disadvantages:
1.Cell-based storage components were used leads un-stable behavior in FPGA implantation.

TITLE:"HIGH-THROUGHPUT NEDIMENSIONAL MEDIAN AND WEIGHTED MEDIAN FILTERS ONFPGA"
AUTHOR : S. A. Fahmy.
YEAR : 2009
DESCRIPTION:

Most effort in designing median filter shads focused on two-dimensional filters with small window sizes, used for image processing. However, recent work on novel image processing algorithms, such as the trace transform, has highlighted the need for architectures that can compute the median and weighted median of large one-dimensional windows, to which the optimizations in the aforementioned architectures do not apply. A set of architectures for computing both the median and weighted median of large, flexibly sized windows through parallel cumulative histogram construction is presented. The architecture uses embedded memories to control the highly parallel bank of histogram nodes, and can implicitly determine window sizes formedian and weighted median calculations. The architecture is shown toperform at 72 Msamples, and has been integrated within a trace transform architecture**.**

Advantages:

- Here through parallel cumulative histogram construction fast median computation is achieved.
- For stage selection small finite state machine is used.

Disadvantages:

- If window sizes increase complexity will also get increased. 19

TITLE : "An Efficient Implementation of 1-D Median Filter"
AUTHOR : Vasily G. Moshnyaga.
YEAR : 2009
DESCRIPTION:

An area-efficient 1-D median filter based on the sorting network is presented in this brief. It is a word-level filter, storing the samples in the window in descending order according to their values. When a sample enters the window, the oldest sample is removed, and the new sample is inserted in an appropriate position to preserve the sorting
of samples. To increase the throughput, the deletion and insertion of samples are performed in one clock cycle, so that the median output is generated at each cycle. The experimental results have shown the improved area efficiency of our design in comparison with previous work.

Advantages:
• In order to process the input samples sequentially non-recursive
sorting network architectures is proposed.
• Can able to achieve high throughput rate with minimal latency
Disadvantages:
• Embedded data-transfer control leads synchronization problems.
• Complexity is high due to use of tree based non-recursive for
sorting.

3. EXISTING SYSTEM:

- **In the existing system, there is no autonomous system to know the free space through online**
- **A smart car parking system that will assist users to solve the issue of finding a parking space and to minimize the time spent in searching for the nearest available car park**

DRAWBACKS OF EXISTING
- **Does not have auto Booking System**
- **No online monitoring system**
- **Less efficiency**

IJIRMET ISSN (Online): 2456-0448
International Journal of Innovative Research in Management, Engineering, and Technology
Vol. 6, Issue 4, April 2021

4. PROPOSED SYSTEM

low-power median filter architecture with window size N. It consists of a circular array of N identical cells and three auxiliary modules: rank calculation (Rank Cal),rank selection (Rank Sel), and median selection (Median Sel). All the cell sare also connected to a global input register X, through which they receive the incoming sample and the median is stored in the output register Y
The architecture is implemented as a two-stage pipeline, where the registers in all the cells serve as the internal pipeline registers. All the registers in the architecture are synchronized by the rising edge of a global clock. Each cell block ci is composed of a rank generation (RankGen) module, a comparator module "==," and three registers: an mbit (m = log 2 N) rank register (Pi), a data register (Ri), and a 1-bit tokenregister (Ti). Register Ri stores the value of the sample in cell ci, register26Pi keeps the rank of this sample, and the enable signal (en) of Ri is stored in register Ti.

All the samples in the window are ranked according to their values, regardless of their physical locations in the window. In our design, a cell with a greater sample value will be associated with a greater rank. However, for two cells ci and cj, whose sample values are equal, ci will be given a greater rank if Ri is newer than Rj (or Rj is older than Ri); i.e., the sample in cj enters the window earlier than the sample in ci. The rank is hence unique for each cell. For a window with size N, the rank starts from 1 for a cell with the least sample value, and ends with N for a cell with the greatest sample value. The median of the window can then be obtained from the sample value Ri of a cell ci whose rank Pi is equal to (N + 1)/2, assuming N is an odd number. In the architecture, the input sample enters the window in a FIFO manner. After it is queued, it will not be moved and is simply de-queued in place. A token, which is represented as logic state 1 in the token register of some cell, is used to control the de-queuing of old sample and queuing of new input sample at the same time.

After the token is used, it will be passed to the next cell at a newcycle. All the token registers form a token ring with exactly one token. Itserves as a state machine in the circuit, where the location of the token defines the state of the machine. Since the data in each register R is immobile, our architecture has the potential for low-power applications.

At the initial state of the architecture, to make the first incoming sample be stored in the first cell c1, the token is assumed to exist in the last cell c N, shown as the shadowed circle at the output of register TN.

Overall System Operation

Therefore the adaptive median filtering has been applied widely as an advanced method compared with standard median filtering. The11Adaptive Median Filter performs spatial processing to determine which pixels in an image have been affected by impulse noise. The Adaptive Median Filter classifies pixels as noise by comparing each pixel in the image to its surrounding neighbor pixels. The size of the neighborhood is adjustable, as well as the threshold for the comparison. A pixel that is different from a majority of its neighbors, as well as being not structurally aligned with those pixels to which it is similar, is labeled as impulse noise. These noise pixels are then replaced by the median pixel value of the pixels in the neighborhood that have passed the noise labeling test.

Purpose

1). Remove impulse noise
2). Smoothing of other noise
3). Reduce distortion, like excessive thinning or thickening of object boundaries

1.2.1 FUNDAMENTAL STEPS IN DIGITAL IMAGE

PROCESSING

There are basically so many steps involved in processing of digital image some of them are given in below.

1) Image acquisition is the first process where in it involves preprocessing, such as scaling of images.
2) Image enhancement is the process manipulating an image so that the result is more suitable than the original for specific application the word specific is important because it establish at the outset that enhancement technique are problem oriented.
3) Image restoration is an area that also deals with improving the appearance of an image unlike enhancement, which is subjective, image restoration, is objective, in the sense that restoration technique tends to be based mathematical or probabilistic models of image degradation.

4) Image compression deals with technique with reducing for storage required in saving an image, or the bandwidth required to transmit it. image compression can be done using joint photographic experts group(JPEG) and moving pictures experts group(MPEG).
5) Image segmentation procedures partition an image into its constituent parts or objects. Segmentation techniques' are of different types such as autonomous, rugged, weak or erratic.

TYPES OF NOISES:

Amplifier noise (Gaussian noise):
Amplifier noise is a major part of the "read noise" of an image sensor, that is, of the constant noise level in dark areas of the image. In color cameras where more amplification is used in the blue color channel than in the green or red channel, there can be more noise in the blue channel.

Salt-and-pepper noise:

Fat-tail distributed or "impulsive" noise is sometimes called salt-and-pepper noise or spike noise. An image containing salt-and pepper noise will have dark pixels in bright regions and bright pixels in dark regions.
This type of noise can be caused by dead pixels, analog-to digital converter errors, bit errors in transmission, etc. This can be eliminated in large part by using dark frame subtraction and by interpolating around dark/bright pixels.

Shot noise:

The dominant noise in the lighter parts of an image from an image sensor is typically that caused by statistical quantum fluctuations, that is, variation in the number of photons sensed at a given exposure level; this noise is known as photon shot noise.[3]
Quantization noise (uniform noise):
The noise caused by quantizing the pixels of a sensed image to a number of discrete levels is known as quantization noise; it has anapproximately uniform distribution.

Film grain:

The grain of photographic film is a signal-dependent noise, related to shot noise. That is, if film grains are uniformly distributed (equal number per area), and if each grain has an equal and independent probability of developing to a dark silver grain after absorbing photons, then the number of such dark grains in an area will be random with a binomial distribution.

Impulse Noise (IN):

Impulse Noise (IN) is a general term for single-pixel bright or dark spots that are not authentic imagery. This artifact can have several different causes, each with a slightly different appearance. Impulse noise is always independent and uncorrelated to the image pixels and is randomly distributed over the image. Impulse noise corrupted image all the image pixels are not noisy, a number of image pixels will be noisy and the rest of pixels will be noise free.

NOISE GENERATION

In testing, we consider sets of images containing various amounts of artificial noise. Impulse noise represents random spikes of energy that happen during the data transfer of an image. To generate noise, a percentage of the image is damaged by changing a randomly selected point channel to a random value from 0 to 255.I is the original image, Ir, Ig, and Ib represent the original red, green, and blue component intensities of the original image, x,y= [0,1] are continuous uniform random numbers, z= [0,255] is a discrete uniform random number and p = [0,1] is a parameter which represents the probability of noise in image

5. SOFTWARES USED

We have used mat lab, Model sim, and Quarters II. Let us see in brief.

FUTURE WORK

To extend this median computation to reduce the number of register by using time driven registering method and to prove the efficiency of new approach in terms of high throughput rate and complexity reduction. To implement the proposed method to remove the noise with edge preserved pre-processing method.

6. CONCLUSION

A method to remover salt-and-pepper noise is proposed. This increases the efficiency of the system. The algorithm removes noise by finding median for each macro blocks with minimum number of transitions. The performance of the proposed median filter is better when compared to the other filter of this type. In proposed method to reduce the complexity median is computed with minimum number of registers. Here we implement median filter on a noisy image blocks. The proposed method is successfully synthesized using QUARTUS II EDA tool.

REFERENCES

1. J. Cadenas, P. Huerta, G. M. Megson and R. S. Sherratt, "Fast median calculation method," Electron. Lett., vol. 48, no. 10, pp. 558–560, May 2012.
2. T. Chelcea and S. M. Nowick, "Robust interfaces for mixed-timing systems," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 12, no. 8, pp. 857–873, Aug. 2004.
3. O. T.-C. Chen, S. Wang, and Y.-W. Wu, "Minimization of switching activities of partial products for designing low-power multipliers," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 11, no. 3, pp. 418–433, Jun. 2003.
4. 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.
5. C. Choo and P. Verma, "A real-time bit-serial rank filter implementation using Xilinx FPGA," in Proc. SPIE Real-Time Image Process., 2008, vol. 6811, pp. 68110F-1–68110F-8.
6. S. A. Fahmy, P. Y. K. Cheung, and W. Luk, "High-throughput onedimensional medianandweightedmedianfilters onFPGA," IET Comput. Digit. Tech., vol. 3, no. 4, pp. 384–394, Jul. 2009.
7. I. M. Panades and A. Greiner, "Bi-synchronous FIFO for synchronous 60 circuit communication well suited for network-on-chip in GALS architectures," in Proc. 1st Int. Symp. NOCS, 2007, pp. 83–94.
8. V. G. Moshnyaga and K. Hashimoto, "An efficient implementation of 1-D median filter," in Proc. 52nd IEEE Int. MWSCAS, 2009, pp. 451–454.
9. M. Mottaghi-Dastjerdi, A. Afzali-Kusha, and M. Pedram, "BZ-FAD: Alow-powerlow-areamultiplierbasedonshift-and add architecture," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 17, no. 2, pp. 302–306, Feb. 2009.
10. 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.