Volume 6, Issue 1, January 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Nonlinear Filter Based Restoration of Compressed Medical Image Using AMF

Dr. K. Bikshalu
Assistant Professor, Dept. of ECE, KUCE&T,
KU, Warangal, Telangana, India

R. Srikanth
Assistant Professor, Dept. of ECE, KITS,
Warangal, Telangana, India

Abstract-- This paper proposes a scheme for removing salt-and-pepper impulse noise of medical which are compressed then corrupted by salt and pepper noise. Compression is required before storing the medical images to reduce the memory requirements. In the first phase, an adaptive median filter is used to identify pixels which are likely to be contaminated by noise (noise candidates). In the second phase, the image is restored using a specialized method that applies only to those elected noise candidates. Our restored images show a significant improvement compared to those restored by using just nonlinear filters or regularization methods only like median filter. Our scheme can remove salt-and-pepper-noise with noise level as high as 70% with high level of compression. Digital image processing is an ever elaborating and dynamic area with applications reaching out into our everyday life such as medicine, space exploration, surveillance, authentication, automated industry inspection and many more areas. Applications such as these involve different processes like image enhancement and image restoration, median filtering is a powerful instrument used in image processing. The traditional median filtering algorithm, without any modifications gives good results. There are many variations to the classical algorithm, aimed at reducing computational cost or to achieve additional properties. Median filters are used mainly to remove salt-and pepper noise. The filter logic is implemented on a novel reconfigurable fabric. In this paper look into an efficient architecture for non-linear modified Adaptive median filter implementation.

Keywords-- Salt and Pepper noise, Median, Adaptive Median Filter (AMF), compression, Jpeg.

I. INTRODUCTION

In past years, linear filters became the most popular filters in image signal processing. The reason of their popularity is caused by the existence of robust mathematical models which can be used for their analysis and design. However, there exist many areas in which the nonlinear filters provide significantly better results. The advantage of nonlinear filters lies in their ability to preserve edges and suppress the noise without loss of details. The success of nonlinear filters is caused by the fact that image signals as well as existing noise types are usually nonlinear. Due to the imperfections of image sensors, images are often corrupted by noise. The impulse noise is the most frequently referred type of noise. In most cases, impulse noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or errors in the data transmission. We distinguish two common types of impulse noise[1]; the salt-and-pepper noise (commonly referred to as intensity spikes or speckle) and the random-valued shot noise. For images corrupted by salt-and-pepper noise, the noisy pixels can take only the maximum or minimum values. In case of the random-valued shot noise, the noisy pixels have an arbitrary value. It is very difficult to remove this type of noise using linear filters because they tend to smudge resulting images. For instance a CT brain is considered in to verify the efficiency of adaptive median filter.

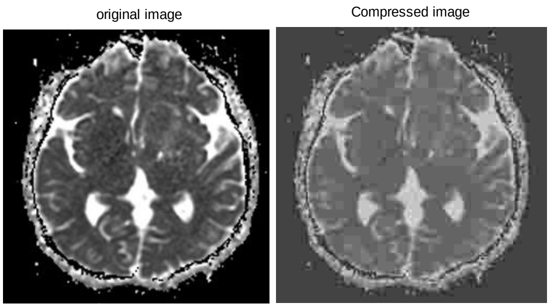


Fig.1: CT brain image (right) and its compressed image (left)

Traditionally, the impulse noise is removed by a median filter which is the most popular nonlinear filter. Its hardware implementation is straightforward and does not require many resources. However, the standard median filter gives a poor performance for images corrupted by impulse noise with higher intensity. A simple median utilizing 3 x 3 or 5 x 5-pixel window is sufficient only when the noise intensity is less than approx. 10-20%. When the intensity of noise is increasing, a simple median filter remains many shots unfiltered [1]. Thus more advanced techniques have to be utilized. In order to overcome this shortage, various approaches [5] were proposed in the recent years). Among the most known techniques we can include: switching median filters, weighted median filters, weighted order statistic filters and adaptive median filters. Apart from median filters, some better algorithms exist. However, because they do not use the concept of a small filtering window, their hardware.

Implementations do not bring any benefits for a reasonable cost. Almost all alternatives to median filters have already been implemented in hardware. There is one exception, adaptive median filter, whose efficient hardware implementation will be described in this paper. In comparison with other approaches, the adaptive median filter provides significantly better results (in terms of visual quality of filtered images) especially for images corrupted with high noise intensity. The main advantage of the adaptive median filter is that it modifies only corrupted pixels [6]. Standard median filters modify almost all pixels of the image. In addition to filtering, adaptive median filters can be also used as detectors of corrupted pixels (detection statistics). This paper deals with a highly pipeline hardware realization of the adaptive median filter which is optimized for high throughput. Proposed solution is implemented in an FPGA, and its performance is compared against various filters on a set of test images. The goal is to provide a filter suitable for a high performance real-time processing of images corrupted by shot noise of various intensities.

A) Median Filter
Median filtering is a non-linear, low-pass filtering method, which you use to remove "speckle" noise from an image. 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. This paper is focused on developing hardware implementations of image processing algorithm for use in an FPGA -based image processing system. The rank order filter is a particularly common algorithm in image processing systems. For every pixel in an image, the window of neighboring pixels is found. Then the pixel values are sorted in ascending, or rank, order [1]. Next, the pixel in the output image corresponding to the origin pixel in the input image is replaced with the value specified by the filter order. The VHDL code can be simulated to verify its functionality.

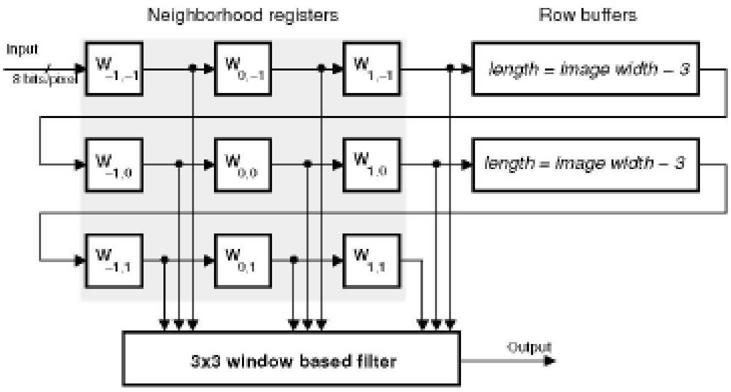


Fig. 2 Hardware for median filter

B) Adaptive Median Filter
The Adaptive Median Filter is designed to eliminate the problems faced with the standard median filter. The basic difference between the two filters is that, in the Adaptive Median Filter, the size of the window surrounding each pixel is variable. This variation depends on the median of the pixels in the present window. If the median value is an impulse, then the size of the window is expanded. Otherwise, further processing is done on the part of the image within the current window specifications. 'Processing' the image basically entails the following: The center pixel of the window is evaluated to verify whether it is an impulse or not. If it is an impulse, then the new value of that pixel in the filtered image will be the median value of the pixels in that window. If, however, the center pixel is not an impulse, then the value of the center pixel is retained in the filtered image. Thus, unless the pixel being considered is an impulse, the gray-scale value of the pixel in the filtered image is the same as that of the input image. Thus, the Adaptive Median Filter solves the dual purpose of removing the impulse noise from the image and reducing distortion in the image. Adaptive Median Filtering can handle the filtering operation of an image corrupted with impulse noise of probability greater than 0.2. This filter also smoothens out other types of noise, thus, giving a much better output image than the standard median filter.

II. MOVING WINDOW ARCHITECTURE
In order to implement a moving window system in VHDL, a design was devised that took advantage of certain features of FPGAs. FPGAs generally handle flip -flops quite easily, but instantiation of memory on chip is more difficult.

Still, compared with the other option, off-chip memory, the choice using on-chip memory was clear. It was determined that the output of the architecture should be vectors for pixels in the window, along with a data -valid signal, which is used to inform an algorithm using the window generation unit as to when the data is ready for processing. Since it was deemed necessary to achieve maximum performance in a relatively small space [3], FIFO Units specific to the target FPGA were used. Importantly though, to the algorithms using the window generation architecture, the output of the window generation units is exactly the same as shown in Fig.3. This useful feature allows algorithm interchangeability between the two architectures, which helped significantly, cut down algorithm development time. A window size was chosen because it was small enough to be easily fit onto the target FPGAs, and is considered large enough to be effective for most commonly used image sizes. With larger window sizes, more FIFOs and flip -flops must be used, which increases the FPGA resources used significantly. The Fig.3 shows a graphic representation of the FIFO and flip -flop architecture used for this design for a given output pixel window.

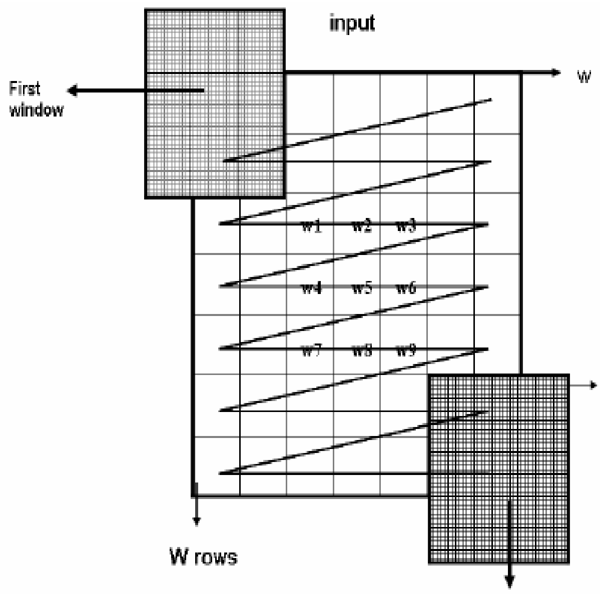
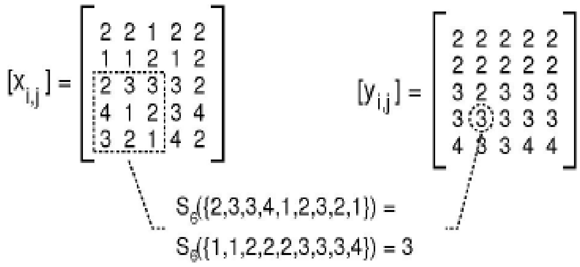


Fig.3 Moving window

III. ALGORITHM OF ADAPTIVE MEDIAN FILTER

Implementation and testing the adaptive filter works on a rectangular region Sxy. The adaptive median filter changes the size of Sxy during the filtering operation depending on certain criteria as listed below. The output of the filter is a single value which the replaces the current pixel value at (x, y), the point on which Sxy is centered at the time. The following notation is adapted from the book and is reintroduced here:



Zmin = Minimum gray level value in Sxy.
Zmax = Maximum gray level value in Sxy
Zmed = Median of gray levels in Sxy
Zxy = gray level at coordinates (x, y)
Smax = Maximum allowed size of Sxy

The adaptive median filter works in two levels denoted Level A and Level B as follows:

Level A: A1= Zmed -Zmin
A2= Zmed - Zmax
If A1 > 0 AND A2 < 0, Go to level B
Else increase the window size
If window size <= Smax repeat level A
Else output Zxy.

Level B: B1 = Zxy – Zmin
B2 = Zxy – Zmin
If B1 > 0 And B2 < 0 output Zxy
Else output Zmed.

The algorithm has three main purposes:
1) To remove 'Salt and Pepper' noise
2) To smoothen any non impulsive noise
3) To reduce excessive distortions such as too much thinning or thickening of object boundaries.

A) Proposed Architecture

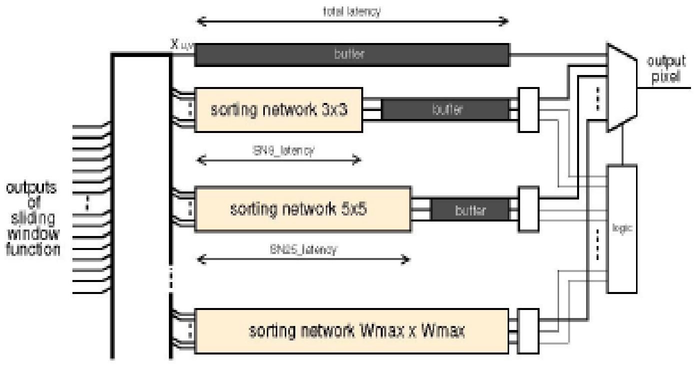


Fig.4: Architecture of AMF

1) Non-Recursive Adaptive Median Filter:
Although the adaptive median filter is defined as an iterative filter, the result can be computed in a two-step process. The idea is to implement a set of sorting networks [2] of different number of inputs (from 3 x 3 to Smax x Smax). The minimum, maximum and median value of each sorting network is utilized. As these sorting networks have different latencies it is necessary to include registers at suitable positions to synchronize the computation. In the second step, the outputs of sorting networks are combined together using simple combination logic.

2) Recursive adaptive median filter
The implementation cost can be reduced if only one Smax X Smax-input sorting network is used. This sorting network is used iteratively for all filtering windows. It is necessary to ensure that unused inputs are initialized to correctly calculate the minimum, maximum and median value for each filtering window. Immediate results have to be stored to registers. However, the throughput of this solution is three times lower (in case that Smax = 7) in comparison to a non-recursive implementation [4].

IV. JPEG COMPRESSION
The JPEG process is a widely used form a lossy image compression that centers around the Discrete Cosine Transform (DCT).The DCT works by separating images into parts of different frequencies. During a step called Quantization, where part of compression actually occurs, the less important frequencies are discarded, hence the use of term lossy. Then only the most important frequencies that remain are used to retrieve the image in the decompression process. As a result reconstruction images contain some distortion, but as we shall soon, these levels of distortion can be adjusted during the compression stage. The JPEG method is used for both color and black and white images, in this paper brain CT scan image is taken and compressed then applied salt and pepper noise finally restored by adaptive median filter.

A) Results
The median filters as well as adaptive median filters were described in Mat lab simulated for CT brain image with compressed by JPEG and corrupted by salt and pepper noise. The parameters for comparison is given as

$$PSNR = 10\log 10\left(\frac{255^2}{MSE}\right) \quad \text{-- (1)}$$

$$MSE = \frac{\sum_{ij}\left(r_{ij} - x_{ij}\right)^2}{MN} \quad \text{--(2)}$$

$$IEF = \frac{\left(\sum_{ij} n_{ij} - r_{ij}\right)^2}{\left(\sum_{ij} x_{ij} - r_{ij}\right)^2} \quad \text{--- (3)}$$

PSNR : pead signal to noise ratio
MSE : mean square error
IEF : image enhancement factor;
n : corrupted image

r : original image
M x N : size of image
x : restored image

Table.1 :Performance results for Median filter

% of Noise	Results of Median filter with un compressed image				Results of Median filter with compressed image			
	MSE	PSNR	IEF	SNR	MSE	PSNR	IEF	SNR
10	31.2169	33.1869	1.0956	6.2959	0.0021	74.8240	33.3101	7.7201
20	31.6331	33.1294	1.6205	4.1804	0.0024	74.3500	32.8789	4.4758
30	32.4012	33.0252	1.8779	3.0964	0.0030	73.4065	17.9377	3.4004
40	33.6959	32.8550	1.9143	2.5112	0.0049	71.2513	8.4913	2.8665
50	36.3583	32.5248	1.7605	2.1716	0.0103	67.9941	4.9118	2.4576
60	41.8731	31.9115	1.6016	1.9958	0.0248	64.1788	2.9970	2.1870
70	52.2290	30.9517	1.4225	1.9297	0.0548	60.7428	2.0051	1.9435
80	60.7313	30.2967	1.2401	2.0017	0.1010	58.0885	1.5051	1.7141
90	73.3860	29.4747	1.1205	2.1973	0.1652	55.9497	1.1914	1.5470
100	72.7503	29.5125	1.0144	2.7294	0.2245	54.6181	1.0112	1.3638

Table.2 :Performance results for Median filter

% of Noise	Results of Adaptive Median filter with un compressed image				Results of Adaptive Median filter with compressed image			
	MSE	PSNR	IEF	SNR	MSE	PSNR	IEF	SNR
10	4.5305	41.5693	3.0717	11.1377	0.0003	83.8499	106.9874	7.8225
20	7.1048	39.6153	3.5936	6.0368	0.0005	81.4325	119.6007	4.5717
30	10.2372	38.0290	3.5299	4.0110	0.0007	79.6026	118.3765	3.4471
40	13.8489	36.7167	3.4382	2.9478	0.0011	77.8386	105.5569	2.8675
50	18.6548	35.4229	3.3719	2.3504	0.0016	76.0574	86.6691	2.5431
60	23.9473	34.3382	3.1414	1.9030	0.0026	73.9065	63.5250	2.2930
70	28.9537	33.5138	2.6979	1.6187	0.0075	69.3851	26.2411	2.1300
80	38.3433	32.2939	2.1274	1.4691	0.0278	63.6938	8.0844	1.9652
90	53.4481	30.8515	1.5231	1.5119	0.0984	58.2017	2.5811	1.6618
100	78.4245	29.1863	0.9952	2.2917	0.2844	53.5918	0.9895	1.1301

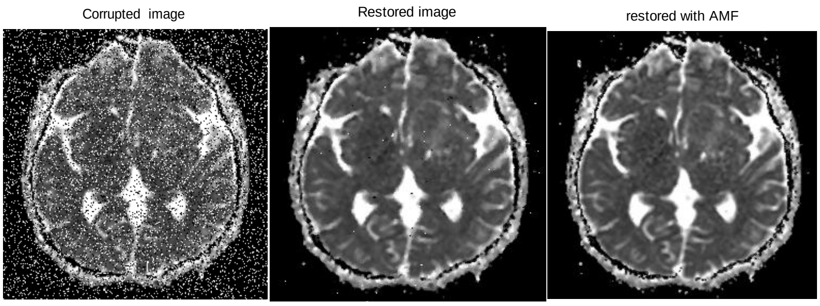


Fig.5: First Colum shows CT brain images corrupted with salt and pepper noise of 20 %, .Second Colum show restoration by median filter and third Colum shows restoration by adaptive median filter.

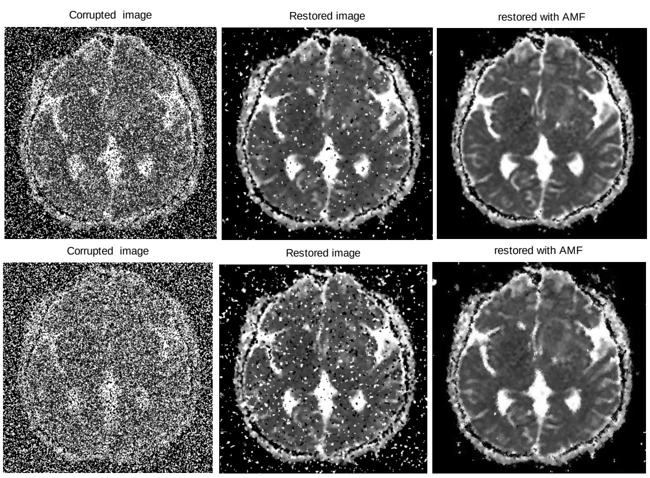


Fig.6: First Colum shows CT brain images corrupted with salt and pepper noise of 40% and 60%.Secod Colum show restoration by median filter and third Colum shows restoration by adaptive median filter.

The above Fig .5 shows the noise removing capability of adaptive median filter as capered to median filter. The Fig.6 shows the comparison of median and adaptive median for compressed image.

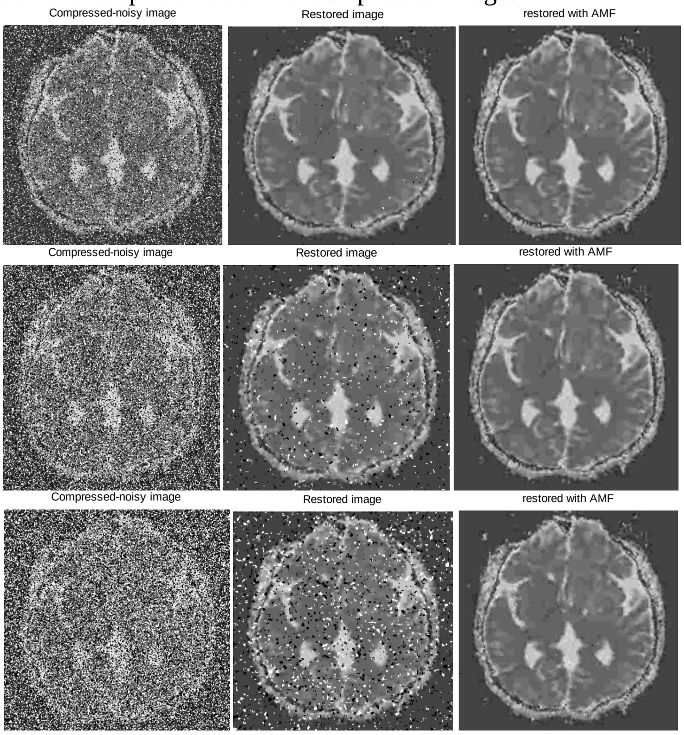


Fig.7: First Colum shows compressed CT brain images corrupted with salt and pepper noise of 20%, 40% and 60%.Secod Colum show restoration by median filter and third Colum shows restoration by adaptive median filter.

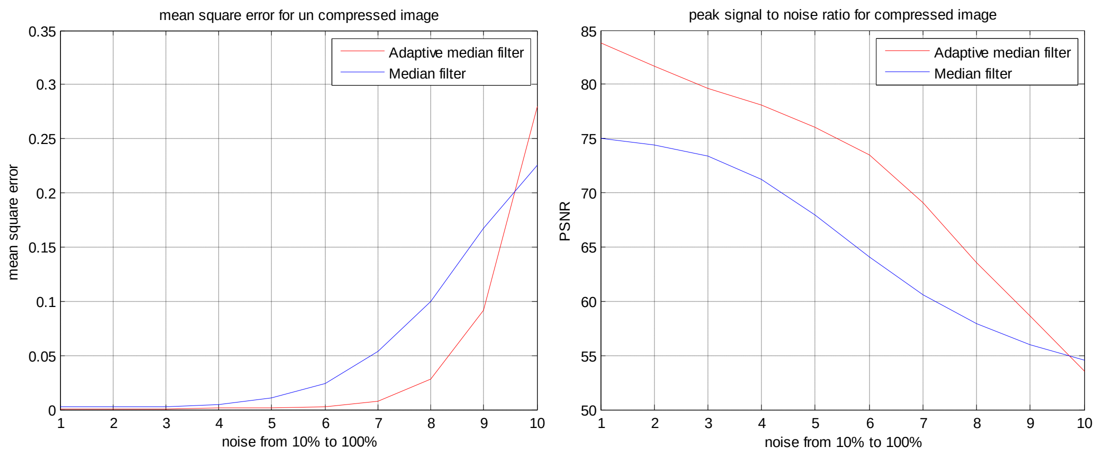


Fig.8: Compare Mean Square error (Colum 1) and peak signal to noise ratio (Colum 2) for compressed image red line represents adaptive median filter and blue line represents median filter

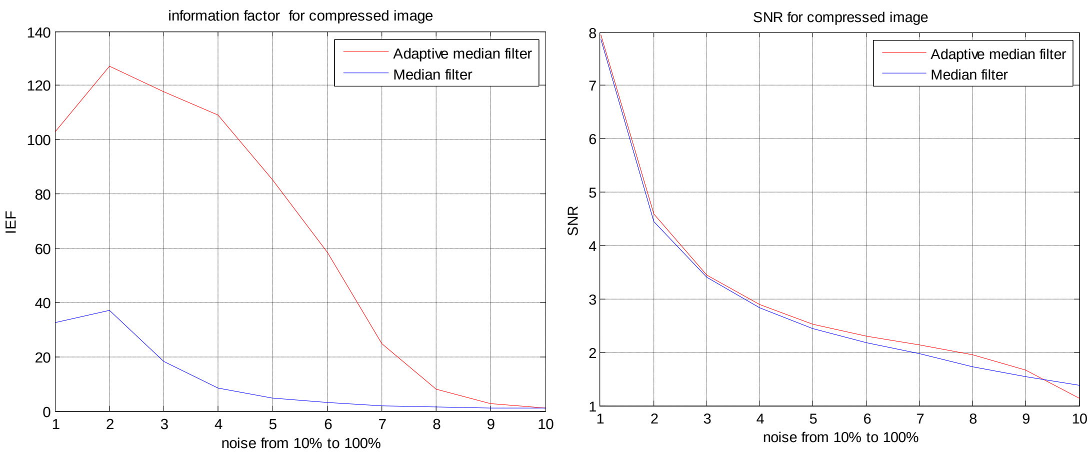


Fig.9: Fig.8: Compare Information Enhance factor (Colum 1) and SNR (Colum 2) for compressed image red line represents adaptive median filter and blue line represents median filter

V. CONCLUSION

In this paper, we propose a decision-based, details preserving restoration method. It is the ultimate filter for removing salt-and-pepper noise. Experimental results show that our method performs much better than median-based filters or the edge-preserving regularization methods. Even at a very high noise level and compression it gives good results. One can further improve our results by using different noise detectors and regularization functional that are tailored to different types of noises, such as the random-valued impulse noise or impulse-plus-Gaussian noise. These extensions together with fast solvers will be given in our forthcoming papers.

REFERENCES

[1] Salt-and-Pepper Noise Removal by Median-type Noise Detectors and Detail-preserving Regularization Raymond H. Chan, Chung-Wa Ho, and Mila Nikolova'

[2] Hardware Implementation for Design of Modified Adaptive Median Filter for Image Processing Ch.Ravi Kumar Research Scholar, Bharath University, Chennai, India

[3] Novel Hardware Implementation of Adaptive Median Filters Zdenek Vasicek , Faculty of Information Technology Brno University of Technology Bozetechova 2, 612 66 Brno, Czech Republic

[4] High Speed Pipelined Architecture for Adaptive Median Filter ,D.Dhanasekaran Assistant Professor, ECE dept., Sri Venkataswara College Engg. College Pennalur, Sriperumbudur, Anna University,Sriperumbudur-602105

[5] A.Sri Nagesh, Dr.G.P.Saradhi Varma, Dr.A.Govardhan & Dr.B.Raveendra Babu, "An Analysis and Comparison of Quality Index Using Clustering Techniques for Spot Detection in Noisy Microarray Images". International Journal of Image Processing (IJIP), pp 504-511, Volume (5): Issue (4): 2011, CSC Journals, Kuala Lumpur, Malaysia, ISSN: 1985-2304.

[6] Madhu S.Nair,K.Revathy and Rao Tatavarti "Removal of salt-and pepper noise in images: A new Decision Based Algorithm" Proceedings of International MultiConference of Engineers and Computer Scientists 2008 Volume 1 IMECS 2008,19-21 March,2008,Hong Kong.

