International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.6, Nov-Dec. 2012 pp-4543-4546 ISSN: 2249-6645

Image Denoising Using Non Linear Filter

Mr. Vijay R. Tripathi
Assistant Professor IN College of Engg & Technology, Akola-444004.

Abstract: *Noise in an image is a serious problem In this project, the various noise conditions are studied which are: Additive white Gaussian noise (AWGN), Bipolar fixed-valued impulse noise, also called salt and pepper noise (SPN), Random-valued impulse noise (RVIN), Mixed noise (MN). Digital images are often corrupted by impulse noise during the acquisition or transmission through communication channels the developed filters are meant for online and real-time applications. In this paper, the following activities are taken up to draw the results: Study of various impulse noise types and their effect on digital images; Study and implementation of various efficient nonlinear digital image filters available in the literature and their relative performance comparison;*

I. Introduction

Today digital imaging is required in many applications e.g., object recognition, satellite imaginary, biomedical instrumentation, digital entertainment media, internet etc. The quality of image degrades due to contamination of various types of noise. Noise corrupts the image during the process of acquisition, transmission, storage etc[1]. For a meaningful and useful processing such as image segmentation and object recognition, and to have very good visual display in applications like television, photo-phone, etc., the acquired image signal must be noise free and made deblurred. The noise suppression (filtering) and deblurring come under a common class of image processing tasks known as image restoration.

In common use the word noise means unwanted signal. In electronics noise can refer to the electronic signal corresponding to acoustic noise (in an audio system) or the electronic signal corresponding to the (visual) noise commonly seen as 'snow' on a degraded television or video image. In signal processing or computing it can be considered data without meaning; that is, data that is not being used to transmit a signal, but is simply produced as an unwanted by-product of other activities. In Information Theory, however, noise is still considered to be information. In a broader sense, film grain or even advertisements in web pages can be considered noise.

In early days, linear filters were the primary tools in signal and image processing However; linear filters have poor performance in the presence of noise that is not additive as well as in systems where system nonlinearities or non-Gaussian statistics are encountered. Linear filters tend to blur edges, do not remove impulsive noise effectively, and do not perform well in the presence of signal dependent noise. To overcome these shortcomings, various types of nonlinear filters have been proposed in the literature.

II. Median Based Filters

In order to effectively remove impulse noise as described in while preserving image details, ideally the filtering should be applied only to the corrupted pixels, and the noise-free pixels should be kept unchanged. This can be achieved by determining whether the current pixel is corrupted, prior to possibly replacing it with a new value. Decision-based filters correspond to a well-known class of filters that appear to be particularly efficient to reduced impulse noise. In this work, we propose an impulse detection scheme by successfully combining the SM filter with CWM filter.

PROGRESSIVE SWITCHING MEDIAN FILTER

A median-based filter, progressive switching median (PSM) filter, is implemented to restore images corrupted by salt–pepper impulse noise. The algorithm is developed by the following two main points: 1) switching scheme—an impulse detection algorithm is used before filtering, thus only a proportion of all the pixels will be filtered and 2) progressive methods—both the impulse detection and the noise filtering procedures are progressively applied through several iterations. The noise pixels processed in the current iteration are used to help the process of the other pixels in the subsequent iterations. A main advantage of such a method is that some impulse pixels located in the middle of large noise blotches can also be properly detected and filtered. Therefore, better restoration results are expected, especially for the cases where the images are highly corrupted.

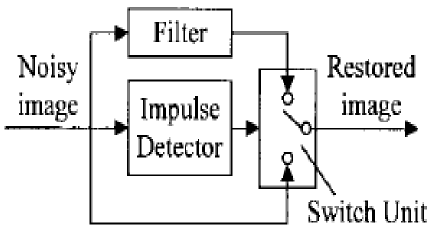


Fig. 1. A general framework of switching scheme-based image filters.

PSM Filter

1. Impulse Detection

Similar to other impulse detection algorithms, this impulse detector is implemented by prior information on natural images, i.e., a noise-free image should be locally smoothly varying, and is separated by edges [4]. The noise considered for this algorithm is only salt–pepper impulsive noise which means: 1) only a proportion of all the image pixels are corrupted while other pixels are noise-free and 2) a noise pixel takes either a very large value as a positive impulse or a very small value as a negative impulse. In this chapter, we use noise ratio $R(0 \le R \le 1)$ to represent how much an image is corrupted. For example, if an image is corrupted by R = 30% impulse noise, then 15% of the

www.ijmer.com Vol.2, Issue.6, Nov-Dec. 2012 pp-4543-4546 ISSN: 2249-6645

pixels in the image are corrupted by positive impulses and 15% of the pixels by negative impulses.

Two image sequences are generated during the impulse detection procedure. The first is a sequence of gray scale images, $\{x^{(0)}_{(i,j)}, x^{(1)}_{(i,j)}, x^{(2)}_{(i,j)}, \ldots x^{(n)}_{(i,j)} \ldots\}$, where the initial image $x^{(0)}_{(i,j)}$ is noisy image itself , (i, j) is position of pixel in image, it can be $1 \le i \le M$, $1 \le j \le N$ where M and N are the number of the pixel in horizontal and vertical direction respectively, and $x^{(n)}_{(i,j)}$ is image after n^{th} iteration.

The second is a binary flag image sequence, $\{f^{(0)}_{(i,j)}, f^{(1)}_{(i,j)}, f^{(2)}_{(i,j)}, \ldots f^{(n)}_{(i,j)}\}$ where the binary flag $f^{(n)}_{(i,j)}$ is used to indicate whether the pixel at *(i, j)* in noisy image detected as noisy or noise-free after n^{th} iteration. If $f^{(n)}_{(i,j)}=0$ means pixel at *(i, j)* has been found as noise-free after n^{th} iteration and if $f^{(n)}_{(i,j)}=1$ means pixel at *(i, j)* has been found as noisy after n^{th} iteration. Before the first iteration, we assume that all the image pixels are good, i.e. $f^{(0)}_{(i,j)}=0$ for all *(i, j)*.

In the nth iteration (n= 1, 2, 3…) for each pixel $x^{(n-1)}_{(i,j)}$ we first find out the median value of the samples in a $W_D \times W_D$ (W_D is an odd integer not smaller than 3) window centered about it. To represent the set of the pixels within a $W_D \times W_D$ window centered about $x^{(n-1)}_{(i,j)}$ is $x^{(n-1)}_{(i+k,j+l)}$ where $-W \le k \le W, -W \le l \le W$ $k \le$ W, -W$\le$ l $\le$ W and W$\ge$1, then we have median value of this window $m^{(n-1)}_{(i,j)}$ is

$$m^{(n-1)}_{(i,j)} = \text{median}\,(x^{(n-1)}_{(i+k,j+l)})$$

The difference between $m^{(n-1)}_{(i,j)}$ and $x^{(n-1)}_{(i,j)}$ provides us with a simple measurement to detect impulses

$$f^{(n)}_{(i,j)} = \begin{cases} f^{(n-1)}_{(i,j)}, & if \left|x^{(n-1)}_{(i,j)} - m^{(n-1)}_{(i,j)}\right| < T \\ 1, & otherwise \end{cases}$$

Where T is a predefined threshold value. Once a pixel *(i, j)* is detected as an impulse, the value of $x^{(n)}_{(i,j)}$ is subsequently modified

$$x^{(n)}_{(i,j)} = \begin{cases} m^{(n-1)}_{(i,j)}, & if\ f^{(n)}_{(i,j)} \ne f^{(n-1)}_{(i,j)} \\ x^{(n-1)}_{(i,j)}, & if\ f^{(n)}_{(i,j)} = f^{(n-1)}_{(i,j)} \end{cases}$$

(4.3)

Suppose the impulse detection procedure is stopped after the N_Dth iteration, then two output images- $x^{(N_D)}_{(i,j)}$ and $f^{(N_D)}_{(i,j)}$ are obtained, but only $f^{(N_D)}_{(i,j)}$ is useful for our noise filtering algorithm.

2. Noise Filtering

Like the impulse detection procedure, the noise filtering procedure also generates a gray scale image sequence, $\{y^{(0)}_{(i,j)}, y^{(1)}_{(i,j)}, y^{(2)}_{(i,j)}, \ldots y^{(n)}_{(i,j)} \ldots\}$ and a binary flag image sequence $\{g^{(0)}_{(i,j)}, g^{(1)}_{(i,j)}, \ldots g^{(n)}_{(i,j)} \ldots\}$. In the gray scale image sequence, we still use $y^{(0)}_{(i,j)}$ to denote the pixel value at position *(i, j)* in the noisy image to be filtered and use $y^{(n)}_{(i,j)}$ to represent the pixel value at position *(i, j)* in the image after the *n*th iteration. In a binary flag image $g^{(n)}_{(i,j)}$, the value $g^{(n)}_{(i,j)}=0$ means the pixel *(i, j)* is good and $g^{(n)}_{(i,j)} = 1$ means it is an impulse that should be filtered. A difference between the impulse detection and noise-filtering procedures is that the initial flag image $g^{(0)}_{(i,j)}$ of the noise-filtering procedure is not a blank image, but the impulse detection result $f^{(N_D)}_{(i,j)}$, i.e., $g^{(0)}_{(i,j)} = f^{(N_D)}_{(i,j)}$.

In the *n*th iteration (n = 1; 2; ….), for each pixel $y^{(n-1)}_{(i,j)}$, we also first find its median value $m^{(n-1)}_{(i,j)}$ of a $W_F \times W_F$ (W_F is an odd integer and not smaller than 3) window centered about it. However, unlike that in the impulse detection procedure, the median value here is selected from only good pixels with $g^{(n-1)}_{(i,j)} = 0$ in the window.

Let M denote the number of all the pixels with $g^{(n-1)}_{(i,j)} = 0$ in the $W_F \times W_F$ window. If M is odd, then

$$m^{(n-1)}_{(i,j)} = median\{y^{(n-1)}_{(i,j)} \left| g^{(n-1)}_{(i,j)} = 0, (i,j) \in W_F \times W_F \right.\}$$

The value of $y^{(n)}_{(i,j)}$ is modified only when the pixel *(i, j)* is an impulse and M is greater than 0:

$$y^{(n)}_{(i,j)} = \begin{cases} m^{(n-1)}_{(i,j)}, & if\ g^{(n-1)}_{(i,j)} = 1; M > 0 \\ y^{(n-1)}_{(i,j)}, & else \end{cases}$$

Once an impulse pixel is modified, it is considered as a good pixel in the subsequent iterations

$$g^{(n)}_{(i,j)} = \begin{cases} g^{(n-1)}_{(i,j)}, & if\ y^{(n)}_{(i,j)} = y^{(n-1)}_{(i,j)} \\ 0, & if\ y^{(n)}_{(i,j)} = m^{(n-1)}_{(i,j)} \end{cases}$$

The procedure stops after the N_Fth iteration when all of the impulse pixels have been modified, i.e.,

$$\sum_{(i,j)} g^{N_F}_{(i,j)} = 0$$

Then we obtain the image $\{y^{(N_F)}_{(i,j)}\}$ which is our restored output image.

Table 1.WF(3 * 3) Median Filter,CWM Filter&PWM Filter

Noise	PSNR With Noise	PSNR After Filtering		
		Median Filter	CWM Filter	PWM Filter
10%	**15**	**12**	**13.5**	**29**
20%	**12**	**12**	**13**	**27**
30%	**11**	**12**	**13**	**26**
40%	**9.5**	**12**	**13**	**24**

The Experimental results are shown in Table.2 for WF (5 * 5) PWM Filter

Table 2. WF (5 * 5) PWM Filter

Noise	PSNR with Noisy	PSNR after Filtering
10	15	21
20	12	19
30	11	19
40	9.5	19

The Experimental results are shown in Table.3 for WF (7 * 7) PWM Filter

Table 3. WF (7 * 7) PWM Filter

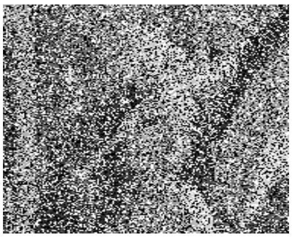
Noise	PSNR with Noisy	PSNR after Filtering
10	15	14
20	12	7. 3
30	11	6
40	9.5	5.9

.

(a)

(b)

(c)



(d)

(e)

<f>

(g)

(h)

<i>

Figure 2. a, b, c, d are noisy images of Lena (512×512) corrupted by salt and pepper noise with noise density of 10%, 20%, 30%, 40% respectively and corresponding restored image by PWM are in e, f, g, h for WF (3 * 3)

III. Conclusion

In this entire dissertation work, two different non-linear filters are implemented and extensive experiments are performed to obtain the results with various parameters to assess the performance of each filter. The plot of PSNR for these two filters is given below. The Table.1, 2 &3 below shows the PSNR value obtain using Lena Image of size 512 x 512.

From the PSNR value mention in the simulation result it is very clear that PWM Filter shows better performance in suppressing impulsive noise compare to above mention filter in suppressing impulse noise when noise exceeds from 10 % to 40 %.

& Secondly extensive experimental result show that if we increase window size i.e. 5 * 5 & 7 * 7,we find that by increasing the window size image get more & more corrupted & filter is not able to suppress impulsive noise effectively compare to when window size in filter was (3 * 3) in filter. Though simulation time required is less, which is given in table below.

Filter Used	**PSM**
Window Size	**Time In Sec**
WF (3 * 3)	**16sec**
WF (5 * 5)	**13sec**
WF (7 * 7)	**10sec**

Table4: Average Run Time in Sec

Therefore from the above table it is very cleared that as window size increases image get more & more blurred & distorted though it requires less simulation time compare to that when window size in filter was (3 * 3). So mostly window size of WF (3 * 3) is preferred compare to that of WF (5 * 5) & WF (7 * 7).

Reference

[1] R.C. Gonzalez and R.E. Woods Digital Image Processing Second Edition.

[2] A. C. Bovik, T. S. Huang, and D. C. Munson, “A generalization of median filtering using Linear combinations of order statistics,” IEEE Tran Acoust., Speech, Signal Process., volASSP-31, no.6, pp.1342–1350, Dec. 1983.

[3] J. Astola and P. Kuosmanen, Fundamental of Nonlinear Filtering, Boca Raton, CRC Press, 1997.

[4] T. Sun and Y. Neuvo, “Detail-preserving median based filters in image processing,” *Pattern Recognit. Lett.*, vol. 15, pp. 341–347, Apr. 1994

[5] I. Pitas and A. N. Venetsanopoulos, Nonlinear Digital Filters: Principles and Applications, Boston, Kluwer, 1990.

[6] B. I. Justus son, “Median filtering: Statistical properties, “Two-Dimensional Digital Signal Processing II, T. S. Huang Ed., New York; Springer Verlag, 1981.

[7] S-J. KO and Y. H. Lee, “Center-weighted median filters and their applications to image enhancement,” IEEE Trans. Circuits and Syst., vol. 38, pp. 984-993, Sept. 1991.

[8] J.-H. Wang, “Rescanned minmax center-weighted filters for image restoration,” Proc. Inst. Elect. Eng., vol. 146, no. 2, pp. 101–107, 1999.

[9] “Tri-State Median Filter for Image Image Denoising”, T Chen, K-K.Ma, and L-H Chen IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 8, NO. 12, DECEMBER 1999.

[10] T. Kasparis, N. S. Tzannes, and Q., “Detail-preserving adaptive conditional median filters,” J. Electron. Imag, vol. 1, no. 14, pp. 358–364, 1992.