2010 2nd International Conference on Information and Multimedia Technology (ICIMT 2010)
IPCSIT vol. 42 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V42.5

One Simple and Fast Method for the Efficient Removal of High-Density Salt and Pepper Impulse Noise

Zongwei Lu, Zhide Tang, Lin Zhou, Hao Yang and Lisen Lin

State Key Laboratory of Power Transmission Equipment & System Security and New Technology
Chongqing University, Chongqing 400044, China
E-mail: luzongwei@hotmail.com

Abstract—In this paper, one simple and fast method for the removal of high-density salt and pepper impulse noise is proposed. Unlike many methods before, the proposed method in this paper does not need to set thresholds and spend a lot of time in deciding whether each pixel in the corrupted image is noisy or not. In the proposed method, for each salt and pepper pixel in the corrupted image, we only need to calculate the mean value of the non salt and pepper pixels in the window centered at the current salt and pepper pixel. Later, we update the salt and pepper pixel with the mean value at the different steps. Although the principle in the proposed method is quite simple, the simulation results show that the proposed method behaves excellently for all noise ratios, from 10% to 90%. The details of the original image are preserved well in the restored image by the proposed method. Furthermore, the run time for the proposed method is quite small, which makes it possible for the real applications.

Keywords- Median filter;Impulse noise;Image enhancement; Nonlinear filter

1. Introduction

Images are often corrupted by impulse noise due to errors generated in noisy sensors or communication channels. It is important to eliminate noise in the images before some subsequent processing, such as edge detection, image segmentation and object recognition. For this purpose, many methods have been proposed. Among those methods, median filter-based approaches have attracted much attention because of their simplicity and capability of preserving image details [1]. However, because typical median filter is implemented uniformly across the image, it tends to modify both noise pixels and undistributed good pixels. To avoid this problem, the switching scheme is introduced, where impulse detection algorithms are employed before filtering and the detection results are used to control whether a pixel should be modified [2]. The main problem in detection scheme design is the optimization of the tradeoff between noise removal and detail preservation. For that purpose, existing filtering methods employ different optimizing parameters: a threshold in [3], four thresholds in SD-ROM [4] and a set of fuzzy rules and membership functions in [5] etc.

In [6], a tri-state median filter (TSM) was introduced. It incorporates the median and CWM (Center Weighted Median) filters followed by taking on one of the outputs of the identity, median and CWM filters as the new pixel. In [7], an impulse detector PSM (Progressive Switching Median) was devised by estimating the difference between the current pixel intensity and the output of the median filter. The final output is switched between the identity and median filter. In [8], one two-phase algorithm was proposed. In the first phase of this algorithm, an adaptive median filter (AMF) is used to classify corrupted and uncorrupted pixels. In the second phase, specialized regularization method is applied to the noisy pixels to preserve the edges and noise suppression. The main drawback of this that the processing time is very high because it uses a very large window size of 39×39 in both phases to obtain the optimum output. To overcome this problem, one new method is proposed in [9]. The corrupted pixels are replaced by either the median or neighborhood pixels in contrast to AMF and other existing algorithms that use only median values for replacement of corrupted pixels. At higher noise densities, the median value may also be a noisy pixel in which case neighborhood pixels are used for replacement. This provides higher correlation between the corrupted pixel and neighborhood pixel. Higher correlation gives rise to better edge preservation.

Most of the above methods can work well at noise ratios less than 50%. For highly corrupted images (noise ratios greater than 70%), the quality of the restored images by these methods is bad. Some methods can not restore the images at all. When noise ratio is greater than 80%, blurring appears in the restored images.

To restore the images at high noise ratios, we propose one simple and fast method in this short paper. The principle of our method is quite simple and only mean calculation is needed. Two simple steps are included in our method. The salt-and-pepper pixel is updated with the mean value at different steps. Extensive simulations on tested images show that our method can efficiently remove salt and pepper noise and preserve most image details. Especially for high density noise, our method out performs most other proposed methods.

2. Proposed Method

The proposed method includes two simple and non-iterative steps. In the first step, for each salt-and-pepper pixel in the corrupted image, the mean value of the non salt-and-pepper pixels in the window centered at the current pixel is calculated. However, we don't update the salt-and-pepper pixel with its mean values immediately. After the mean value calculation for all the salt-and-pepper pixels in the corrupted image is completed, the corrupted image is updated with the mean values. In the second step, we repeat the same calculation for each salt-and-pepper pixel. And we update the salt-and-pepper pixel with its mean value immediately. In the following, we will describe these two steps.

First, for each salt-and-pepper pixel $x_{i,j}$ （$x_{i,j} = 0 or 255$）, we define one 3×3 window $S_{i,j}^3$:

$$S_{i,j}^3 = \{(k.l) : |k-i| \le 1 and \quad |j-l| \le 1,\ (k,\, l) \ne (i,j)\} \tag{1}$$

We calculate the mean value for the pixel $x_{i,j}$ as follows.

$$m_{\ i,j} = mean(x_{k,l}, x_{k,l} \in \Omega_{i,j}^3) \tag{2}$$

where, $mean$ denotes the mean calculation and

$$\Omega_{i,j}^3 = \{x_{k,l} \,|\, x_{k,l} \in S_{i,j}^3, x_{k,l} \ne 0 or 255\} \tag{3}$$

Here, if $\Omega_{i,j}^3$ is empty, we don't need to calculate the mean value.

It should be pointed out that we don't update the salt-and-pepper pixel with its mean value immediately in this step. After the calculation for all the salt-and-pepper pixels is completed, we update them with their mean values.

For example, in the following image, the pixel located at the circle is salt-and-pepper. There are three other salt-and-pepper pixels 255 in the window. According to equation (2), the mean value for the current pixel is 138.

$$\begin{bmatrix} 255 & 255 & 255 & 255 & 0 \\ 137 & 138 & 255 & 138 & 135 \\ 255 & 139 & 139 & 138 & 135 \\ 138 & 0 & 138 & 137 & 0 \\ 138 & 137 & 136 & 0 & 0 \end{bmatrix} \quad mean(138,138,139,139,138) = 138$$

while for the following image, since all the pixels in the window are salt-and-pepper, we don't have to calculate the mean value for it.

$$\begin{bmatrix} 255 & 255 & 255 & 255 & 0 \\ 137 & 0 & 0 & 255 & 135 \\ 255 & 255 & 0 & 0 & 255 \\ 0 & 255 & 0 & 137 & 136 \\ 138 & 255 & 255 & 255 & 0 \end{bmatrix}$$

In the second step, we repeat the calculation according to equation (2) for each salt-and-pepper pixel. However, we update the salt-and-pepper pixel with its mean value immediately here. Then, after the calculations and updates for all the pixels are completed, the restored image is available.

The flowchart for our method is shown in Fig.1.

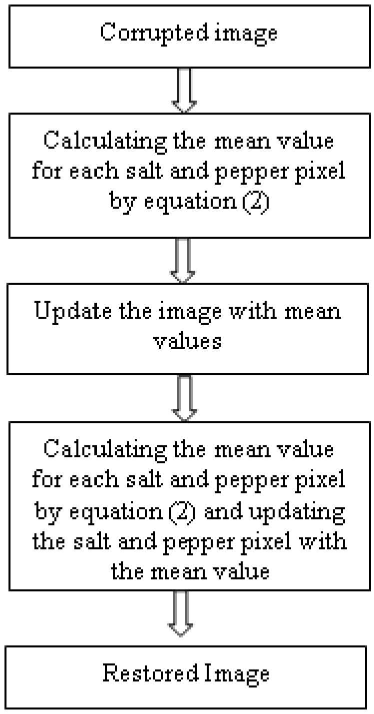


Fig.1 Flowchart for our method

Obviously, there is no one need for thresholds in our method. Furthermore, our method is non-iterative and does not have to spend a lot of time in deciding whether one pixel is noisy or not.

3. Simulation Results

To demonstrate the performance of our method, we compare it with several methods, including Standard Median Filter (SMF) [1], AMF [10], TDF [4], Chan [8] and Sri [9]. Five images used in the simulation are of size 512×512 pixels. The window size used in all methods is 3×3. In the simulation, images will be corrupted by impulse noise (salt and pepper), where 255 represents "salt" and "0" represents "pepper" noise with equal probability. The noise levels are varied from 10% to 90% with increments of 10%. The peak signal-to-noise ratio (PSNR) is used as an objective measurement of the restored image quality.

TABLE 1. RESULT FOR IMAGES LENA

Noise	PSNR for image 'Lena' (dB)				
	SMF	AMF	TDF	Sri	Ours
10%	22.75	29.48	35.53	38.43	41.56
20%	18.75	28.30	33.25	37.36	38.24
30%	15.30	27.10	31.73	35.92	36.25
40%	13.18	25.55	30.65	34.12	34.51
50%	11.82	24.04	29.78	32.21	32.82
60%	11.00	21.07	26.12	30.43	31.37
70%	10.72	16.10	24.67	28.62	29.59
80%	9.08	11.60	22.51	26.23	27.88
90%	8.25	8.02	20.04	23.94	25.66

TABLE 2 RESULT FOR IMAGES GIRL

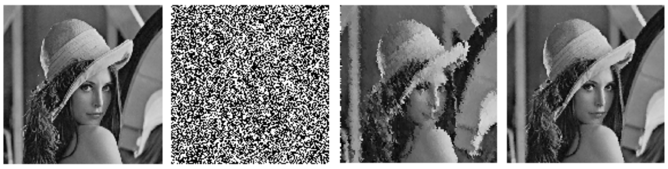
Noise	PSNR for image ‘Girl’ (dB)				
	SMF	AMF	TDF	Sri	Ours
10%	28.29	28.32	31.81	38.29	40.99
20%	26.34	27.93	29.75	36.22	38.71
30%	22.34	27.14	27.99	34.33	36.85
40%	18.34	26.03	25.90	32.51	35.27
50%	14.67	23.60	23.61	30.93	33.87
60%	11.94	19.09	20.18	29.52	32.42
70%	9.49	14.13	16.52	27.88	30.89
80%	7.68	10.97	12.21	26.17	28.14
90%	6.19	7.38	8.52	24.99	25.74

In Table 1 and 2, we show the result for two images Lena and Girl. It is seen from the table that our method has the best result for the two images. Sri’s method has better results for these two images too. However, when the noise ratio is greater than 70%, we notice that some image details are lost in the images restored by Sri’s method.

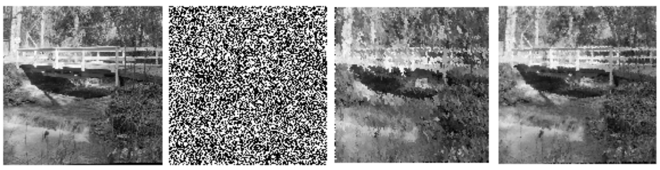
To further demonstrate the performance of our method, we test another additional images Bridge, Cameraman and Goldhill. The PSNR values for the three images by Sri’s and our method is shown in Table3. It is seen from the table that the PSNR values calculated by our method are 10-20% higher than those by Sri’s method. In Fig.2, we show the restored images by Sri’s and our method. From the left to right, we show the original image, corrupted image, restored image by Sri’s method and our method. We can easily see from the restored images that some details are lost both by Sri’s and our method. However, the quality of the restored images by our method looks better than those by Sri’s method.

TABLE 3 RESULT FOR IMAGES BRIDGE, CAMERAMAN AND GOLDHILL

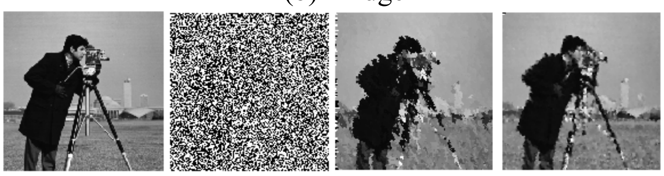
Noise	PSNR for image ‘Bridge’ (dB)		PSNR for image ‘Cameraman’ (dB)		PSNR for image ‘Goldhill’ (dB)	
	Sri	Ours	Sri	Ours	Sri	Ours
10%	32.94	34.62	34.41	34.57	35.13	36.43
20%	30.15	31.64	30.65	31.96	32.19	33.25
30%	27.93	29.64	27.75	29.61	29.73	31.34
40%	26.25	28.22	26.03	27.97	27.90	29.75
50%	24.46	26.89	24.21	26.71	26.34	28.46
60%	22.81	25.52	22.56	25.39	24.51	27.19
70%	21.14	24.39	20.73	24.00	22.78	26.10
80%	20.26	22.86	19.10	22.63	20.72	24.64
90%	19.90	21.13	17.53	20.82	18.30	23.13



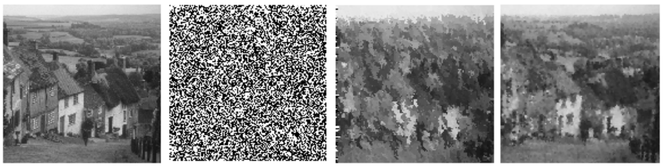
(a) Lena



(b) Bridge



(c) Cameraman



(d) Goldhill

Fig.2 Images restored by Sri's and our method for noise ratio 90%

The runtime for our method is shown in Table 4. Matlab 6.5 (R13) on a PC equipped with 1.8 GHz CPU and 256 MB of RAM memory is employed for the evaluation of the methods.

TABLE 4 RUNTIME FOR SRI'S AND OUR METHOD

Noise	Run time (seconds) for image 'Lena'		Run time (seconds) for image 'Girl' (dB)	
	Sri	Ours	Sri	Ours
10%	36.26	7.52	40.01	8.51
20%	36.29	9.69	41.30	10.82
30%	36.30	12.08	40.83	13.63
40%	36.15	14.29	40.23	15.91
50%	36.40	16.26	41.47	18.36
60%	36.32	17.94	40.70	20.32
70%	36.10	20.28	41.88	22.73
80%	36.42	23.29	40.73	26.17
90%	36.25	27.69	40.55	30.46

It is seen that our method runs faster than Sir's method. Notice that the runtime for Sri's method is almost constant for all noise ratios, from 10% to 90%, because same operations are completed for each pixel in Sri's method. So, the runtime for Sri's method does not depend on the noise ratio. However, since only salt and pepper pixels are considered in our method, the runtime of our method depends on the noise ratio. In other words, the runtime for our method increases linearly with the incensement of noise ratio.

Finally, we notice that Chan's method is effective in removing high density salt and pepper noise [8]. However, the runtime for Chan's method is unacceptable, especially for high density noise. In Table 4, we show the PSNR and runtime for Chan's method. Obviously, when the noise ratio is higher than 70%, the runtime is beyond 0.5 hour which is not tolerated in real applications. Additionally, it is noticed that for higher noise ratio, our method has the same performance as Chan's method.

TABLE 5 RESULT FOR CHAN'S METHOD

Method	Noise Density=70%		Noise Density=90%	
	PSNR	CPU Time	PSNR	CPU Time
SMF	10.72	1.1820	8.2513	1.1700
Chan	29.3	2009	25.4	6917
Sri	28.6253	36.10	23.9471	36.25
Ours	29.59	20.28	25.66	27.69

4. Conclusions

One simple and fast method for the removal of the high density salt and pepper noise is proposed in this short paper. In our method, only mean calculation is needed. So, our method can run faster than most other proposed methods. Although the principle of our method is quite simple, simulation result shows that our method exhibits excellent performances for all noise ratios, from 10% to 90%. It is shown also through extensive simulations on many tested images that most details of the original images can be preserved well. Additionally, there is no threshold in our method which makes it possible for the automation of image processing. In other words, our method doesn't depend on the contents of different images.

5. Acknowledgement

This paper is supported by the Fundamental Research Funds for the Central Universities (No. CDJZR10 15 00 15)

6. References

[1] D.R.K.Brownrigg. The weighted median filter. Commun.Ass.Comput.Mach., Vol.27, No.8, pp.807-818, Aug, 1984.

[2] Z.Wang and D.Zhang. Progressive switching median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Circuits Syst., Vol.46, pp.78-80, Jan 1999.

[3] T.Sun and Y.Neuvo. Detail-preserving median based filters in image processing. Pattern Recognit. Lett., Vol.15, pp.341-347, Apr 1994.

[4] E.Abreu, M.Lightstone, S.K.Mitra and K.Arakawa. A new efficient approach for the removal of impulse noise from highly corrupted images. IEEE Trans. Image Processing, Vol.5, pp.1012-1025, June 1996.

[5] F.Russo and G.Ramponi. A fuzzy filter for images corrupted by impulse noise. IEEE Signal Processing Lett., Vol.3, pp.168-170, June 1996.

[6] T.Chen, K. –K.Ma and L.-H.Chen. Tri-state median filter for image denosing. IEEE Trans. Image Processing, Vol.8, pp.1834-1838, Dec 1999.

[7] Z.Wang and D.Zhang. Progressive switching median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Circuits Syst., Vol.46, pp.78-80, Jan 1999.

[8] R.H.Chan, C.-W.Ho and M.Nikolova. Salt and pepper noise removal by median type noise detectors and detail preserving regularization. IEEE Trans. Image Processing, Vol.14, No.10, pp.1479-1485, Oct 2005.

[9] K.S.Srinivasan and D.Ebenezer. A new fast and efficient decision-based algorithm for removal of high-density impulse noise. IEEE Signal Processing Letters, Vol.14, No.3, pp.189-192, Mar 2007.

[10] H.Huang and R.A.Hadad. Adaptive median filters: new algorithms and results. IEEE Trans. Image Process, Vol.4, No.4, pp.499-502, Apr 1995.