An improved algorithm for removing impulse noise based on long-range correlation in an image

Yik-Hing Fung and Yuk-Hee Chan

Center for Multimedia Signal Processing
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hong Kong

ABSTRACT

In this paper, an algorithm for removing impulse noise is proposed. The proposed algorithm does not only use the information of neighboring pixels in a local region, but uses the long-range correlation within a natural image as well. In the proposed algorithm, a searching criterion based on the weighted transformed contents of regions is used to look for a region which is highly correlated to the region of interest and then the center of that region is selected to replace the corrupted pixel accordingly. The method was found to be very effective in removing impulse noise from corrupted images, both in terms of the objective distortion measure and subjective visual assessment.

1. INTRODUCTION

Images are often contaminated by impulse noise due to the errors generated in noisy sensors or transmission channels. Conventional window-based filtering algorithms such as median filters are well known for being able to remove impulse noise[1-4]. However, when these algorithms are used, it often exhibits blurring for large window sizes, or insufficient noise suppression for small window sizes. Another drawback is that it only uses the local information and does not make use of the long-range correlation within natural images. Recently, Wang and Zhang proposed an effective algorithm to remove impulse noise by making use of the long-range correlation between the region of interest and some remote regions of the image [5]. The performance of this algorithm has been proved to be superior to many other existing algorithms. In this paper, an algorithm based on this algorithm is proposed to further improve the performance of the algorithm.

2. PROPOSED ALGORITHM

Similar to the algorithm proposed in [5], the proposed algorithm is a two-stage algorithm. The first stage is impulse noise detection and the second stage is impulse noise removal. For the first stage, most of the reported impulse noise detection algorithms such as [2], [3], [6] and [7] can be used to determine whether a pixel is corrupted. The output of the impulse noise detector can then be used as some information for impulse noise removal in the second stage.

Only pixels that are classified as corrupted will undergo the noise-removal process in the second stage. Let B be a binary map each of whose element $b_{i,j}$ indicates that image pixel $x_{i,j}$ is corrupted by assigning a value 0 or else 1. For each corrupted pixel, a local window of size $(2l+1)\times(2l+1)$ centered at the corrupted pixel is defined. The contents of the window are compared on a pixel-by-pixel basis with the contents of a floating window of the same size in the corrupted image. The best match is searched subject to a pre-defined criterion and its center is used to replace the corrupted pixel. During the search, only those floating windows whose centers are good pixels are considered. Here, a good pixel means a pixel detected to be uncorrupted.

In the algorithm proposed in [5], a simple searching criteria given as follows is used.

$$\bar{x}_{bm} = \arg\min_{\bar{x}_f \in S} \| M(\bar{x}_l - \bar{x}_f) \|^2 \qquad (1)$$

where $\bar{x}_{bm}$, $\bar{x}_l$ and $\bar{x}_f$ are, respectively, the lexicographically ordered contents of the best match, the local window and the floating window. S is the set of all candidate windows to be searched. It defines a searching area that confines the position of the floating window during the search. M is a diagonal matrix defined as $M = M_l M_f$, where M_l and M_f are both binary diagonal matrices. The i[th] diagonal element of M_l (M_f) equals to 1 if the i[th] element of $\bar{x}_l$ ($\bar{x}_f$) is a good pixel or 0 else.

This simple approach works effectively and was proved to be superior to some other existing algorithms such as [1-4]. However, we found that the performance can further be improved by using a better criterion. In our proposed algorithm, the searching criterion is modified based on some observations we have had.

The use of a best match to estimate the original

The work described in this paper was substantially supported by a grant from the Center for Multimedia Signal Processing, HKPolyU.

0-7803-7304-9/02/$17.00 C2002 IEEE

intensity value of the corrupted pixel is based on an idea that there is some long-range correlation between the local region and a remote region. However, correlation among pixels is generally not block-based. In other words, a block which best matches the block of interest does not imply that their center pixels are close in terms of intensity value. This introduces an issue of the so-called optimal window size. If the window size is too large, the searching result may be incorrect due to the aforementioned reason. If the window size is too small, it is very difficult to judge if the two blocks are really a match pair as there are too few good pixels for one to ensure it is. The compromise given in [5] is 7x7.

During the search of the best match, the contents of the two windows are compared on a pixel-by-pixel basis. In [5], the differences of all good pixel pairs contribute equally in the computation of the objective function value $\| M(\bar{x}_l - \bar{x}_f) \|^2$. However, in order to have a more reliable decision, the match of a pixel pair which is close to the center is more critical than the match of that is far away from the center. This can be explained by the fact that two pixels are more correlated if they are closer to each other. Accordingly, the difference of a pixel pair should be weighted according to their distance from the center of the window.

The size of the set S confines the number of potential candidates in the search. In [5], it includes all windows in a predefined searching area centered at the pixel of interest. On the one hand, this searching area should not be too large so as to reduce the computation cost. Fortunately it does little harm as pixels far away from each other is generally less correlated and it is not likely to find a distant match. On the other hand, if the searching area is too small, it is also unlikely to find a real match among a few candidates. Note that the best match may not be a real match in practice.

In order to increase the probability of having a real match to the target among a limited number of candidates, we adopt the concept used in fractal coding technique. Contents of the candidates are transformed to make them closer to that of the local window before the search. This makes the competition among candidates more meaningful and a closer match can then be obtained at the end.

Based on the aforementioned observations, in our proposed algorithm, the following searching criterion is used instead.

$$\bar{x}_{bm} = \arg\min_{\bar{x}_f \in S} \frac{1}{K_M} \| M((a\bar{x}_f + c\bar{1}) - \bar{x}_l) \|_W^2 \qquad (2)$$

where a and c are scalar parameters to be determined, $\bar{1}$ is a vector whose elements are all 1, W is a weighting matrix and $K_M = \| M\bar{1} \|_W^2$ is a normalization factor. Here $\| \bullet \|_W^2$ denotes the weighted norm of a vector.

For each candidate, its contents are transformed with a simple linear mapping $(a\bar{x}_f + c\bar{1})$. Parameters a and c are, respectively, the scale factor and the offset. They are determined by minimizing the objective function $J = \frac{1}{K_M} \| M((a\bar{x}_f + c\bar{1}) - \bar{x}_l) \|_W^2$.

In general, a typical image can be modeled as a Markov-I signal. Let ρ be the correlation coefficient of the signal model of the image. By assuming that the correlation of pixels in a row and that in a column are independent, the correlation between two pixels at a relative distance (i,j) from each other is given by $\rho^{|i|+|j|}$. Accordingly, a corresponding weighting matrix for a window is given by $\rho^{|i|+|j|}$ for $-l \le i, j \le l$, where (i,j) is the displacement of the element from the center of the window. For a 7×7 window, we assume $\rho = 0.7$.

The minimization of J with respect to c and a results in

$$\frac{\partial J}{\partial c} = 2\bar{1}^t WM(a\bar{x}_f + c\bar{1} - \bar{x}_l)/K_M = 0 \qquad (3)$$

$$\frac{\partial J}{\partial a} = 2\bar{x}_f^t WM(a\bar{x}_f + c\bar{1} - \bar{x}_l)/K_M = 0 \qquad (4)$$

Parameters a and c can then be obtained by solving the following two equations.

$$c = \frac{\bar{1}^t WM(\bar{x}_l - a\bar{x}_f)}{\bar{1}^t WM\bar{1}} \qquad (5)$$

$$a = \frac{\bar{x}_f^t WM(\bar{x}_l - c\bar{1})}{\bar{x}_f^t WM\bar{x}_f} \qquad (6)$$

The corresponding J can also be determined accordingly.

After obtained the best match with eqn.(2), the rounded value of the center of its transformed contents will be used to replace the center pixel of the local window.

3. SIMULATION

Simulation has been carried out to evaluate the performance of the proposed algorithm on a set of 256 gray-level impulse corrupted images of size 256×256 each. The type of impulse noise is random-valued impulse noise where the impulse values are uniformly distributed between 0 and 255.

Table 1 shows the PSNR performance of various algorithms under different conditions. Here, PSNR is defined as

$$PSNR = 10\log_{10} \frac{255^2 N}{\| \bar{x} - \bar{y} \|^2} \qquad (7)$$

where $\bar{x}$ and $\bar{y}$ are, respectively, the lexicographically ordered corrupted image and original image, and N is the total number of pixels of the original image. The results show that the proposed method is superior to the other algorithms. Table 2 shows the effect of weighting in the proposed algorithm. One can see that, when weighting is on, a measurable improvement in terms of PSNR can be obtained. Note that there is also a measurable improvement as compared with [5] even when weighting is off.

Figure 1 shows some simulation results for subjective evaluation. The resulting images show that the impulse noise has been successively removed and the details of the images can be preserved.

4. CONCLUSIONS

In this paper, an improved impulse noise removal algorithm based on Wang and Zhang's approach[5] is proposed. The proposed algorithm makes use of the long-range correlation in an image in a different way. In our approach, the correlation between the contents of different windows was explored and a weighting mechanism is introduced accordingly. Besides, contents of a remote window is transformed to increase the extent of match between the contents of two windows by making use of the concept of the fractal coding technique. Simulation results show that this algorithm can provide a better restoration result than [5].

REFERENCES

[1] H. M. Lin and A. N. Willson, "Median filters with adaptive length," IEEE Trans. on Circuits Syst., vol. 35, pp.675-690, Jun. 1988.

[2] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," IEEE Trans. on Image Processing, vol. 3, pp. 192-206, Mar. 1994.

[3] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing." Pattern Recognition Letters, 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. on Image processing, vol. 5, pp. 1012-1025, Jun. 1996.

[5] D. Zhang and Z. Wang, "Restoration of Impulse Noise Corrupted Images Using Long-Range Correlation," IEEE Signal Processing Letters, vol. 5, pp. 4-7, Jan. 1998.

[6] D. Zhang and Z. Wang, "Impulse noise removal using polynomial approximation". Optical Engineering, 37(4), 1275-1282, 1998.

[7] T. Chen and H. R. Wu, "Adaptive Impulse Detection Using Center-Weighted Median Filters," IEEE Signal Processing Letters, vol. 8, pp.1-3, Jan. 2001.

	Lena	Golden Hill	Mandrill	Barb
10% impulse noise corrupted				
Median filter (3x3)	30.36	27.71	23.75	26.92
Median filter (5x5)	27.40	25.74	22.59	26.54
Abreu et al. (M=2) * [4]	31.91	29.89	26.12	29.98
Abreu et al. (M=1296) ** [4]	32.24	30.59	28.29	31.21
Chen & Wu [7]	30.98	28.91	25.08	28.76
Wang & Zhang [5]	38.05	35.58	32.42	39.16
Proposed approach	38.91	36.01	32.62	40.62
20% impulse noise corrupted				
Median filter (3x3)	29.25	27.15	23.41	26.47
Median filter (5x5)	27.05	25.56	22.49	26.28
Abreu et al. (M=2) * [4]	29.72	28.51	25.26	28.58
Abreu et al. (M=1296) ** [4]	30.41	29.32	26.73	29.47
Chen & Wu [7]	29.76	28.00	24.65	27.93
Wang & Zhang [5]	34.46	32.32	29.21	35.67
Proposed approach	35.19	32.84	29.37	36.84
30% impulse noise corrupted				
Median filter (3x3)	27.90	26.25	22.98	25.74
Median filter (5x5)	26.54	25.33	22.36	25.95
Abreu et al. (M=2) * [4]	27.64	26.97	24.27	27.04
Abreu et al. (M=1296) ** [4]	28.58	27.54	24.94	27.50
Chen & Wu [7]	28.25	27.06	24.06	27.07
Wang & Zhang [5]	32.18	30.04	26.78	33.84
Proposed approach	32.62	30.53	27.06	34.75
40% impulse noise corrupted				
Median filter (3x3)	26.63	25.16	22.39	24.92
Median filter (5x5)	26.22	25.06	22.17	25.68
Abreu et al. (M=2) * [4]	26.00	25.40	23.24	25.76
Abreu et al. (M=1296) ** [4]	27.23	26.05	23.59	26.08
Chen & Wu [7]	27.09	26.07	23.46	26.43
Wang & Zhang [5]	30.16	28.36	25.09	31.91
Proposed approach	30.81	28.79	25.24	32.62

* No training

** Training with corrupted Lena (20% impulse noise)

Table 2. PSNR performance of different algorithms under different conditions (unit: dB)

Noise	Weighting	PSNR			
		Lena	Golden Hill	Mandrill	Barb
10%	On	38.91dB	36.01dB	32.62dB	40.62dB
	Off	38.37dB	35.79dB	32.50dB	39.96dB
20%	On	35.19dB	32.84dB	29.37dB	36.84dB
	Off	34.83dB	32.66dB	29.18dB	36.26dB
30%	On	32.62dB	30.53dB	27.06dB	34.75dB
	Off	32.29dB	30.21dB	26.94dB	34.19dB
40%	On	30.81dB	28.79dB	25.24dB	32.62dB
	Off	30.44dB	28.51dB	25.10dB	32.18dB

Table 3. Effect of weighting in the proposed algorithm.

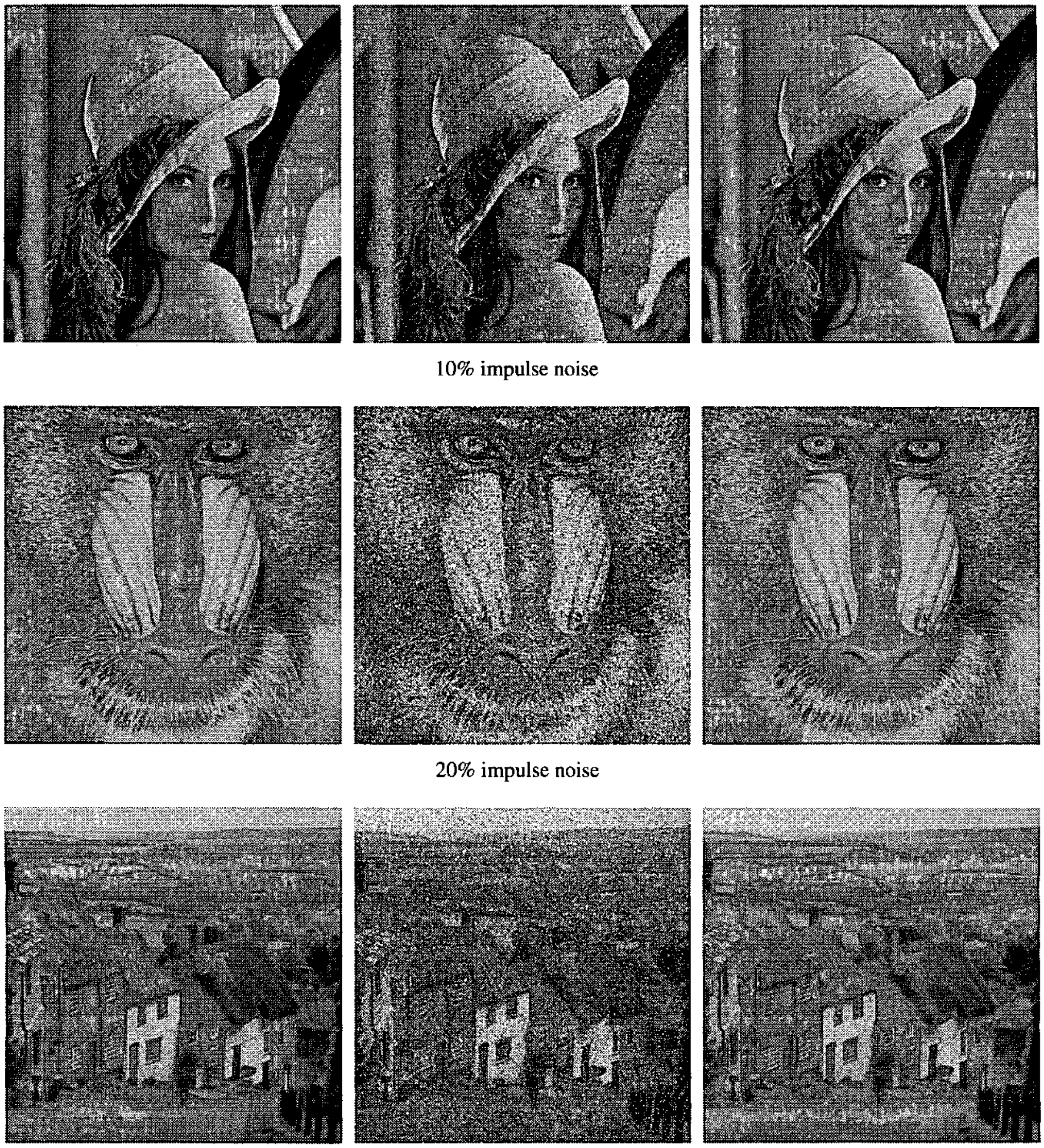

Figure 1. Restoration performance of the proposed algorithm