IMPULSE NOISE REDUCTION BASED ON SIMILARITY AND CONNECTIVITY OF PIXELS

Thou-Ho (Chou-Ho) Chen
Department of Electronic Engineering, National Kaohsiung University of Applied Sciences
thouho@cc.kuas.edu.tw

Chou-Yu Chen
Department of Electronic Engineering, National Kaohsiung University of Applied Sciences
1093320122@cc.kuas.edu.tw

ABSTRACT

In this paper, we propose an effective noise reduction method for image corrupted by impulse noise in highly noise ratio. Preserving edges and details in the process of impulsive noise filtering is an important issue. Image quality is determined by human eye. As a result, the basic strategy of the proposed filter is to exploit the similarity and connectivity between the central pixel and its neighbors. We calculate the number of these pixels for judging whether the central pixel is corrupted by noise or not. Then, we use the local image features to estimate the original pixel value of the noisy pixel. In our experimental results, we shall show the proposed noise reduction method is better than other method in highly corrupted images.

Keyword: Impulse noise.

1. INTRODUCTION

Images are often contaminated by impulse noise due to the errors generated in communication channels or sensory devices. Impulsive noise is very noticeable by human eyes and it can cause serious errors in some image processing applications. Therefore, noise elimination is a main concern in computer vision and image processing.

One of the most popular nonlinear filters for impulse removal is the median filter. The median filter is computationally efficient, suppresses impulse noise, and preserves edges. Therefore, it is widely used in image processing applications. However, the median filter suffers the drawback of removing important image details, thereby causing a number of artifacts including edge jitter and streaking. Adaptive variants of the median filter which still retain the rank order structure have been proposed to overcome these disadvantages. Basically, the task is to decide when to apply the median filter and when to keep pixels unchanged [1], [3] and [6]. Another ways to avoid this situation are to incorporate some noise detections [2], [4], [5], [7] and [8]. At each pixel location, it is to detect whether the current pixel is contaminated. However, for some image details like a fine straight line, they will probably be detected as noise.

An optimal impulsive noise filter must smooth dissimilarities of pixels in homogeneous regions, preserve edge information and not alter natural information [5]. Image quality is subjective for human visual perception. As a result, we use the local image features and some templates to judge whether the pixel is affected by noise or not. In this paper, our method is divided into two steps for reducing impulsive noise. First, we utilize the similarity and connectivity for detecting noise. Second, we use the local image feature to recover the noisy pixel. In particular, our method can effectively preserve thin lines and other image details. This paper is outlined as follows. The impulse noise model assumed in this work is first described in Section 2. In Section 3, the proposed filter is depicted in detail. Section 4 provides a few evaluation results to demonstrate the performance of the proposed approach. Finally, conclusions are given in Section 5.

2. IMPULSE NOISE MODEL

Impulsive noise is the typical noise, which arises due to the failures of imaging sensors, faulty memory locations, the error of record during digitization or influenced due to transmission. It can divide into two parts which are fixed-valued impulsive noise and random-valued impulsive noise.

2.1. Fixed-valued impulsive noise

Fixed-valued impulsive noise is also called salt-and-pepper noise. Its gray value is either minimal (0) or maximal (255). The probability density function of impulse noise is given by Eq. (1)

$$p(z) = \begin{cases} P_a \text{ , for } z = a \\ P_b \text{ , for } z = b \end{cases} \qquad (1)$$

2.2. Random-valued impulsive noise

The gray-scale value of random-valued impulsive noise is uniform distribution in the range of [0, 255] for gray-scale images. The other features are similar to fixed-value impulse noise.

3. METHODOLOGY

In natural images, the edges of most objects appear seldom horizontal, vertical or diagonal. Consequently, we exploit the similarity and connectivity between the central pixel and its neighbors to judge whether the central pixel is influenced by noise or not. After noise detection, we exploit the local image features to restoration the noisy pixel.

3.1. Noise detection

In the step which judges whether the central pixel and its neighbors are similar and connective, we employ a 5×5 filter window, as shown in Fig. 1(a), which is used to process each pixel. The 5×5 window is divided into three shells of one pixel thick, shown in Fig. 1(b). These are the innermost shell, which consists of the central pixel only, denoted as S0, the middle shell S1 and the outermost shell S2 [4]

First, we check whether the pixels in S1 are similar to the central pixel. The decision rule of the similarity is described as follows :

$$\textit{if } \left| x_{0,0} - x_{i,j} \right| \le T \textit{ then } x_{0,0} \text{ and } x_{i,j} \text{ are similar} \qquad (2)$$

where T is a threshold value, $(i, j) \in W$ and $(i, j) \neq (0,0)$.

Then, we check whether the pixels in S2 are similar to $x_{0,0}$. The method is to compare the similarity of each pixel in S1 with the corresponding three outer adjacent pixels in S2 (e.g. outer adjacent pixels of $x_{1,-1}$ are $x_{1,-2}$, $x_{2,-1}$ and $x_{2,-2}$). But there is a prerequisite that the pixel of S1 should be similar to $x_{0,0}$ [4].

After finishing judging whether the central pixel and its neighbors are similar and connective, we calculate the number of those similar and connective pixels. Then, we judge whether the central pixel is an impulse noise or not. The algorithm of noise detection is described as follows, and the flowchart is shown in Fig. 3:

Step 1. If the number is greater than a threshold T_1, the central pixel is regarded as signal and stop the algorithm. If not, go to Step 2.

Step 2. If the central pixel is on the thin line, the pixel is regarded as signal. If not, the central pixel is regarded as noise.

We use those masks, as shown in Fig. 2, to judge whether the central pixel is on the thin line or not.

The decision rules of judging a pixel is on the thin line or not are described as follows:

$$\left\{ \left| x_{0,0} - x_{i,j} \right| < T \text{ or } (i, j) = \{(a+m, b+n\} \right\} = 4$$

$\Rightarrow$ the central pixel is not influenced by noise.

mask 1 $\Rightarrow$ (a, b) = (0,-2) , m=0 , $0 \leqq n \leqq 4$ and $n \neq 2$

mask 2 $\Rightarrow$ (a, b) = (-2,0) , $0 \leqq m \leqq 4$ and $m \neq 2$, n=0

mask 3 $\Rightarrow$ (a, b) = (-2,-2) , $0 \leqq m \leqq 4$ and $m \neq 2$, $0 \leqq n \leqq 4$ and $n \neq 2$

mask 3 $\Rightarrow$ (a, b) = (-2,2) , $0 \leqq m \leqq 4$ and $m \neq 2$, $0 \leqq n \leqq 4$ and $n \neq -2$

3.2. Restoration procedure

At low noise density level, small window size is desirable as it is capable of removing impulse noise without causing noticeable blurring effect. On the contrary, large window size is more effective in removing impulse noise at high noise density situation but result in much serious blurring side effect [2]. As a result, we study the local image feature to judge the noise ratio. Then, we use various methods to restore noisy pixels. We expect to obtain the better performance for reducing noise.

After explaining how to detect noise, we will interpret how to estimate the original pixel value of the noisy pixel. The steps of the algorithm are described as follows, and the flowchart is shown in Fig. 4 :

Step 1. To check whether the pixels in S1 are similar between each other, and calculate the number of the similar pixels.

Step 2. If the result is greater than a, we exploit the mean of the similar pixels to restore and stop the algorithm. If not, we go to Step 3.

Step 3. To check whether the pixels in S1 are similar to the central pixel, and calculate the number of the similar pixels.

Step 4. If the result is greater than b, we exploit the 3×3 median filter and stop the algorithm. If not, we go to Step 5.

Step 5. We utilize the 5×5 median filter.

4. EXPERIMENTAL RESULTS

To check the efficiency of the proposed method, the test images were artificially corrupted by impulsive noise with the corruption rates of 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45% and 50%, respectively. In all cases, we used uniformly distributed impulsive noise with uniform distribution of its values between the minimal (0) and maximal (255) possible signal values.

The performance of the proposed filter has been evaluated and compared with those of existing median based filters for image noise removal. In our simulations, a set of test images corrupted with random-valued impulses of various noise rations are used. Restoration performances are quantitatively measured by the peak signal-to-noise ratio (PSNR), which is defined as

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\sum_i \sum_j 255^2}{\sum_i \sum_j \left(S_{ij} - \hat{X}_{ij}\right)^2} \right)$$

In the section, the proposed method is considered to restore images corrupted by random-value impulse noise. For our simulation, two 8-bit images, “Lena” and “Pepper”, shown in Fig. 5 are corrupted with 30% random-value impulse noise.

For impartiality of statistics, we execute 100 times with corrupted image with the same ratio impulse noise to pick the mean of PSNR for comparison, as shown in Table 1.

From the statistics in Table 1, Fig. 6, Fig. 7 ,Table 2 , Fig.8 and Fig. 9, we select the mean of PSNR for the two images, “Lena” and “Peppers”, and every noise ratio of images execute repeatedly 100 times to obtain the mean of PSNR. The experimental results verify the method that we presented can really be used for attenuate the image noise effectively.

Fig. 7 and Fig. 9 show that the visual comparisons of our method and other methods because image quality is subjective for human visual perception.

5. CONCLUSIONS

In this paper, we make use of the connectivity and similarity to judge whether the central pixel is influenced by noise. Then, we use local image features for restoring the original pixel value of a noisy pixel. As a result, our method can effectively detect noise and noise-free pixels. In particular, it prevents the removal of fine details such as thin lines from the images and thus improved impulse detection ability. The filtered image is adjusted to human visual perception as much as possible. But the performance of the proposed filter is worse than some methods in low noise ratio.

6. REFERENCES

[1] G. Pok, Jyh-Charn Liu, and A. S., " Selective Removal of Impulse Noise Based on Homogeneity Level Information ", IEEE Trans. Image Processing, Vol. 12, Pages: 85 - 92, Jan. 2003.

[2] How-Lung Eng and Kai-Kuang Ma, " Noise adaptive soft-switching median filter ", IEEE Transactions on Image Processing, Vol. 10, Issue 2, Page(s):242 – 251, Feb. 2001.

[3] I. Aizenberg and C. Butakoff, " Effective Impulse Detector Based on Rank-Order Criteria ", IEEE Signal Processing Lett., Vol. 11, Pages: 363 - 366, Mar. 2004.

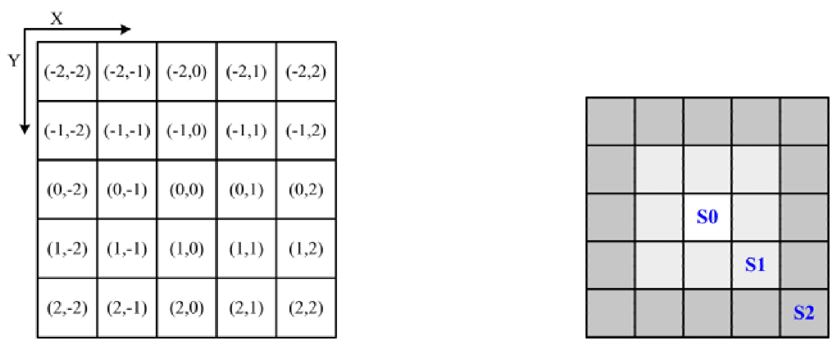
[4] J. Y. F. Ho, "Peer Region Determination Based Impulsive Noise Detection", IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '03, Vol. 3, Pages: III - 713-16, April 6-10, 2003.

[5] Piotr S. Windyga, " Fast Impulsive Noise Removal ", IEEE Transactions on Image Processing, Vol. 10, No. 1, Jan. 2001.

[6] Tao Chen and Hong Ren Wu, " Space Variant Median Filters for the Restoration of Impulse Noise Corrupted Images ", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 48, Issue 8, Pages: 784 - 789, Aug. 2001.

[7] X. D. Jiang, " Image detail-preserving filter for impulsive noise attenuation ", IEE Proceeding-Vis. Image Signal Process, Vol. 150, June 2003.

[8] X. Xu, E. L. Miller, Dongbin Chen, and M. Sarhadi, " Adaptive two-pass rank order filter to remove impulse noise in highly corrupted images ", IEEE Trans. Image Processing, Vol. 13, Pages: 238 - 247, Feb. 2004.



(a) (b)

Fig. 1 Working window

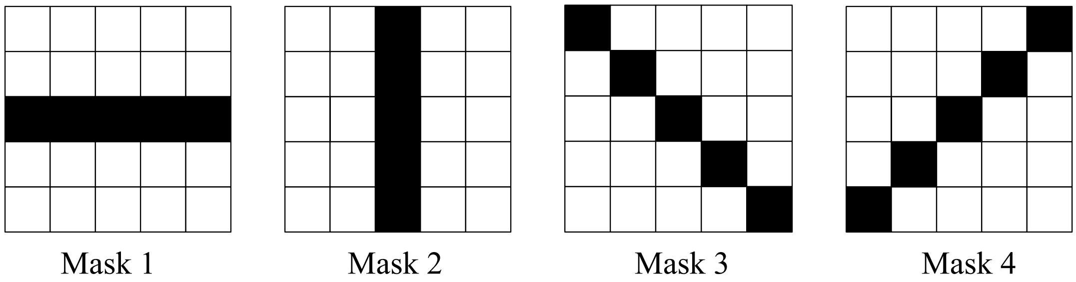


Fig. 2 Masks for judging a pixel
on a thin line.

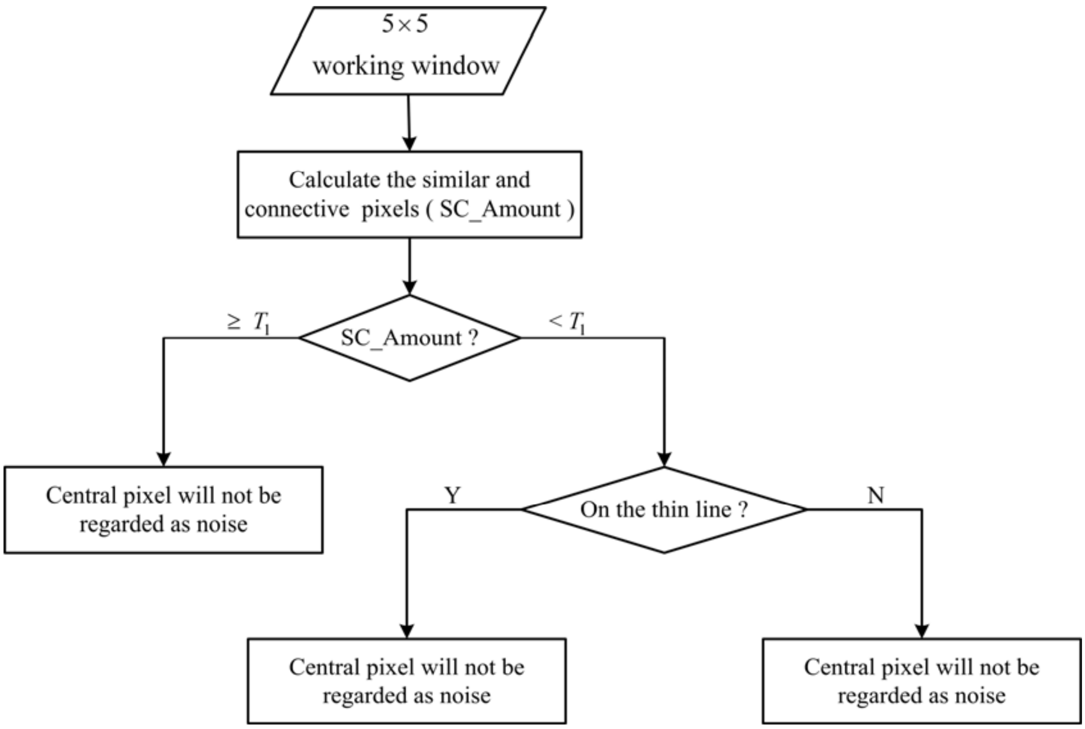


Fig. 3 The flowchart of noise detection

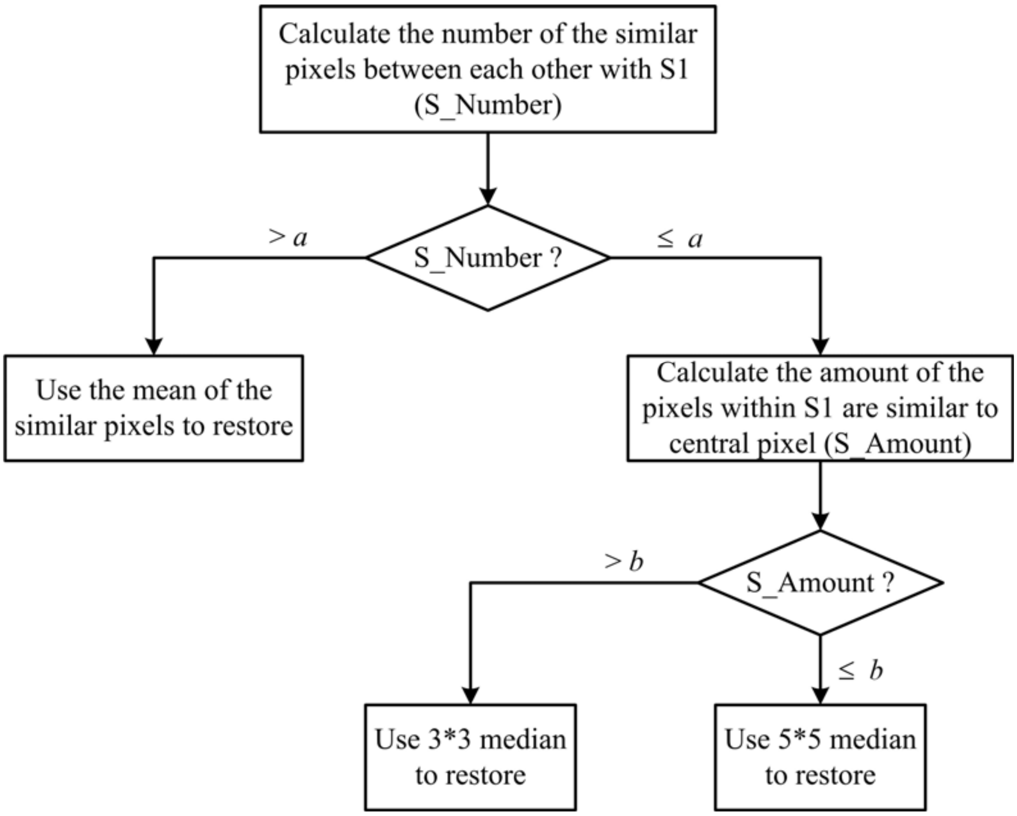
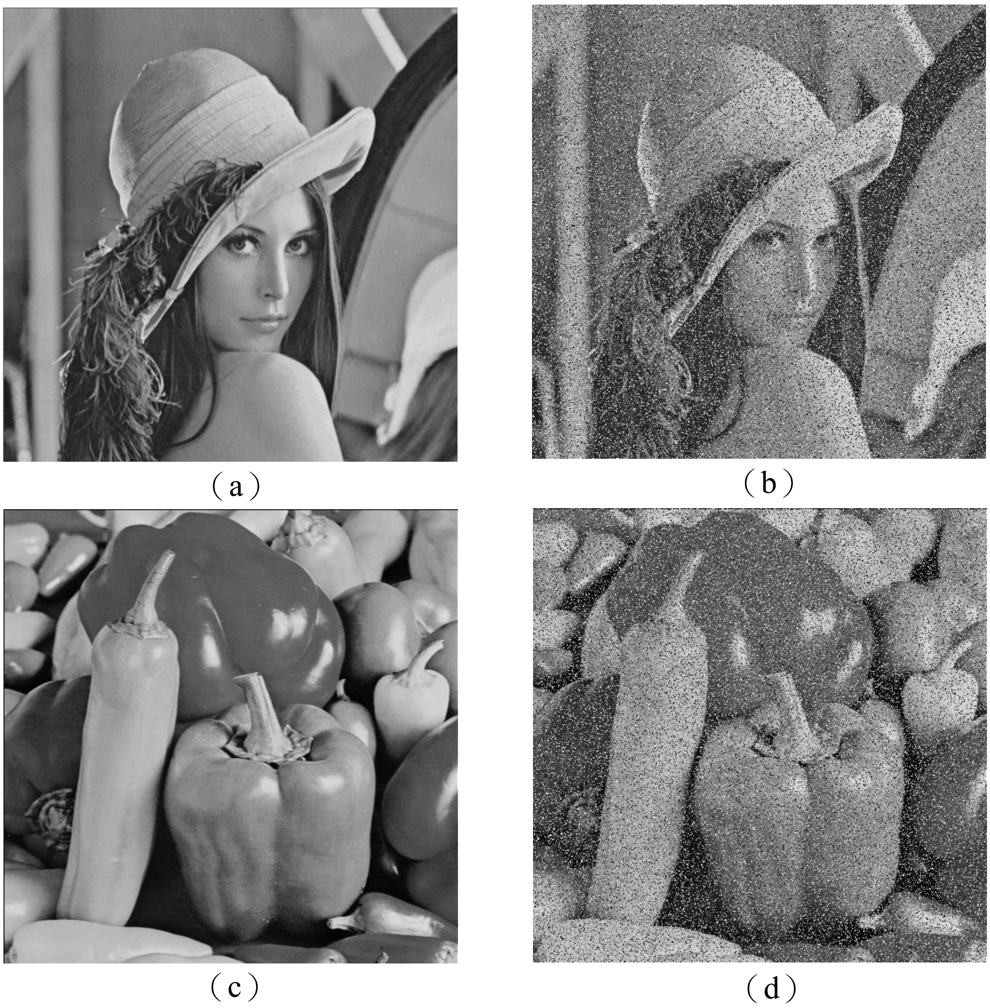


Fig. 4 The flowchart of the restoration procedure



(a) (b) (c) (d)

Fig. 5 Example of original images and their impulse corrupted images.

(a) Original Lena. (b) Lena with 30% impulse noise.

(c) Original Peppers. (d) Peppers with 30% impulse noise.

Table 1 Compare result in PSNR for various noise ratios on “Lena” image

Noise ratio	ATPMF	Truncation	DRID	CSAM	MSM	Our method
5%	33.03	27.72	31.24	30.61	38.67	35.73
10%	32.55	22.88	26.2	30.35	36.42	34.28
15%	32.03	20.11	23.14	30.05	33.96	33.15
20%	31.46	18.22	20.98	29.7	31.46	32.24
25%	30.79	16.82	19.3	29.29	29.07	31.36
30%	29.96	15.72	17.95	28.81	26.86	30.54
35%	28.9	14.82	16.82	28.23	24.92	29.66
40%	27.58	14.08	15.85	27.49	23.16	28.67
45%	26.07	13.43	15.01	26.53	21.61	27.52
50%	24.46	12.87	14.26	25.34	20.22	26.13

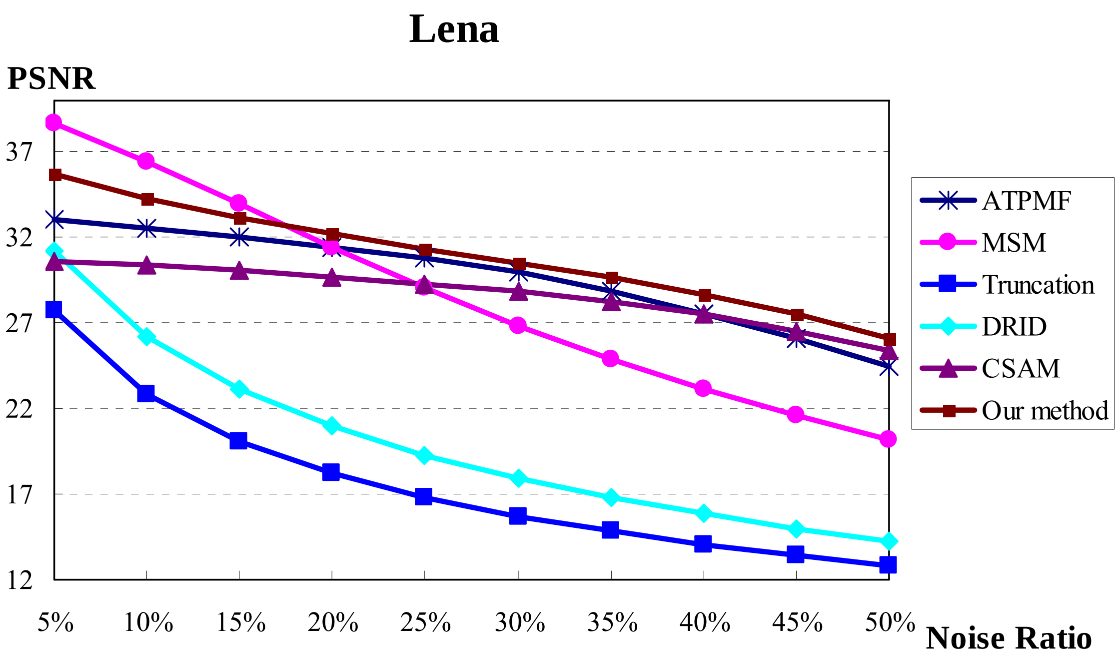


Fig. 6 Comparison of PSNR for various noise ratios on “Lena” image

Fig. 7 Filtered images of 30% impulse corrupted Lena images.

(a) ATPMF filter. (b) Truncation filter. (c) DRID filter.

(d) CSAM filter. (e) MSM filter. (f) the proposed filter.

Table 2 Compare result in PSNR for various noise ratios on "Peppers" image

Noise ratio	ATPMF	Truncation	DRID	CSAM	MSM	Our method
5%	36.25	27.51	31.3	33.85	41.95	38.07
10%	35.41	22.55	26.11	33.29	38.44	36.32
15%	34.57	19.76	22.99	32.66	35.03	35
20%	33.7	17.87	20.78	31.99	31.86	33.9
25%	32.64	16.47	19.07	31.26	29.06	32.86
30%	31.32	15.37	17.71	30.47	26.65	31.84
35%	29.74	14.49	16.57	29.53	24.54	30.69
40%	27.89	13.74	15.59	28.36	22.72	29.41
45%	25.91	13.1	14.73	26.9	21.12	27.86
50%	24.02	12.54	13.97	25.23	19.69	26.11

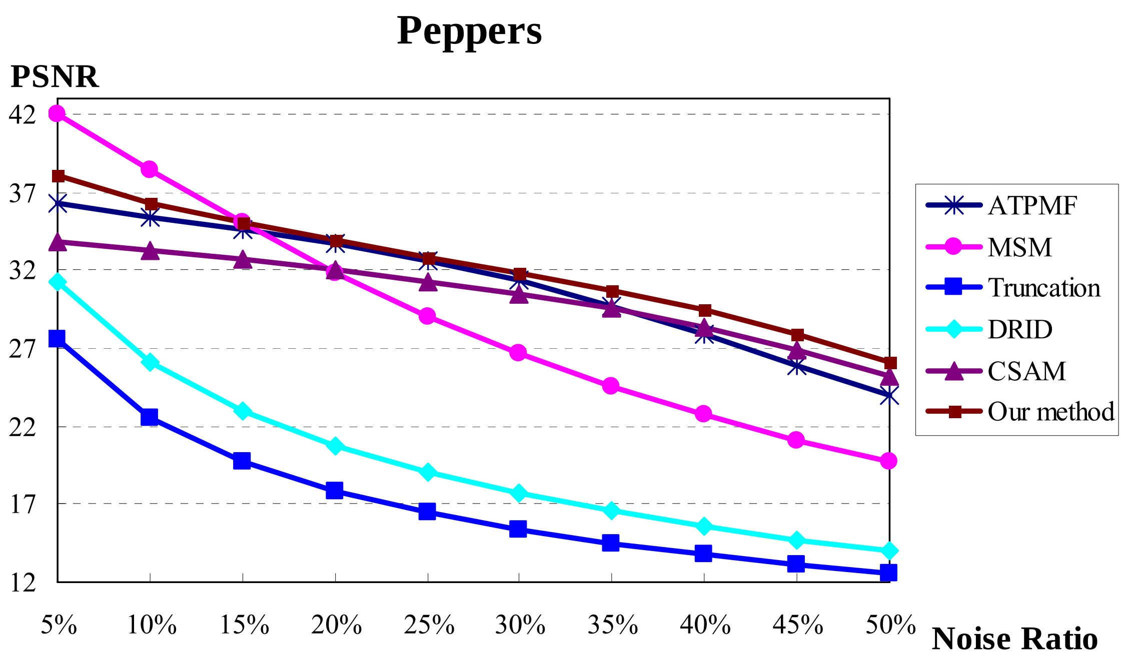


Fig. 8 Comparison of PSNR for various noise ratios on "Peppers" image

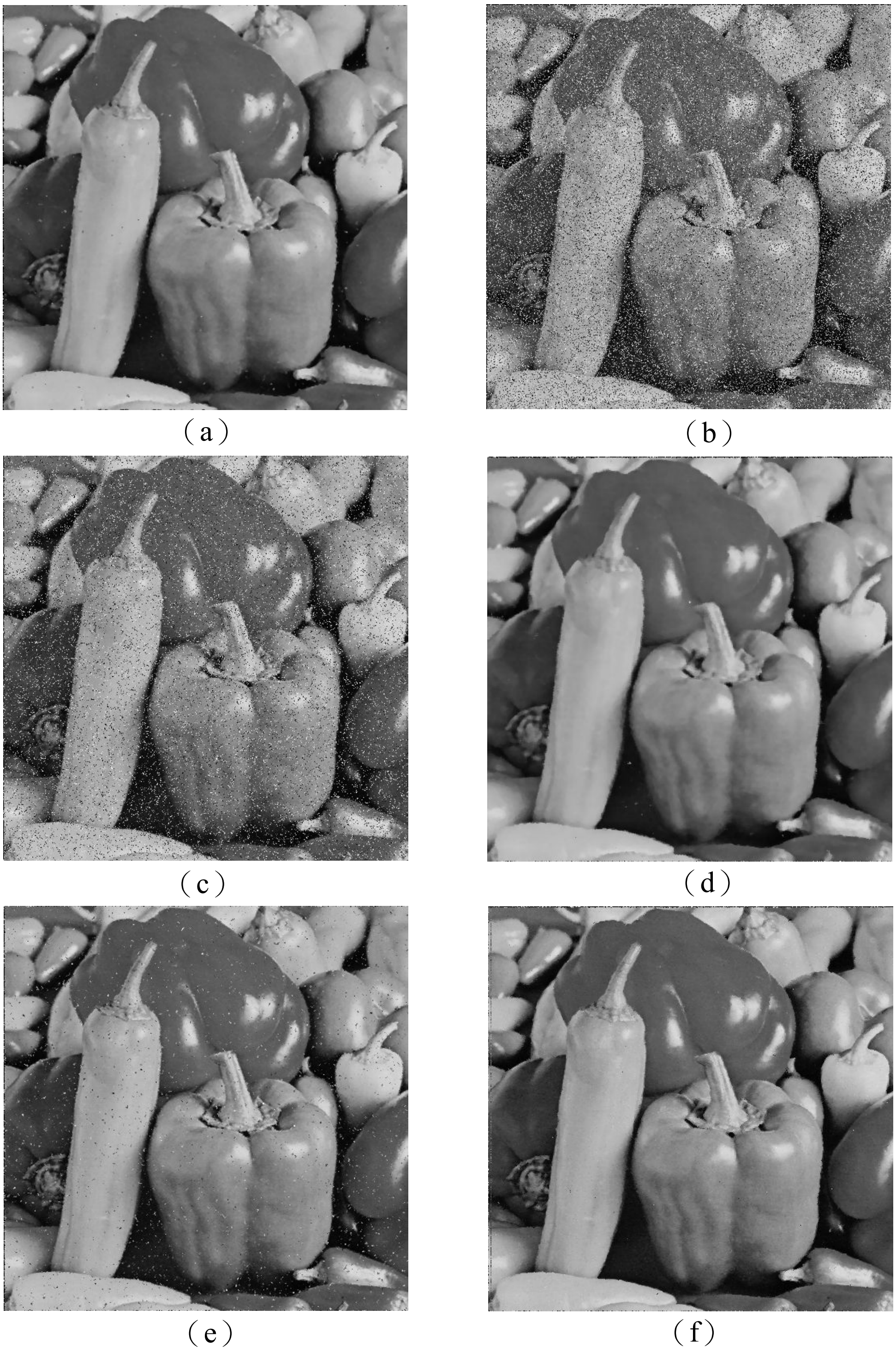


Fig. 9 Filtered images of 30% impulse corrupted Peppers images.

(a) ATPMF filter. (b) Truncation filter. (c) DRID filter.

(d) CSAM filter. (e) MSM filter. (f) the proposed filter.