Detectors of the Impulsive Noise and new Effective Filters for the Impulsive Noise Reduction

Igor Aizenberg[a], Jaakko Astola[a], Taras Bregin, Constantine Butakoff[a], Karen Egiazarian[a] and Dmitriy Paliy[a]

[a]Tampere University of Technology

ABSTRACT

As it is known, the impulsive noise appears on the image in the form of randomly distributed pixels of random brightness. Impulses themselves usually differ much from the surrounding pixels in brightness. The main topic of the paper is the introduction of the new impulse detection criteria, and their application to such filters as median, rank-order and cellular neural Boolean. Three impulse detectors are considered. The Rank Impulse Detector uses such property of impulse that its rank in variation series is usually quite different from rank of the median. Exponential Median Detector uses the exponent of the difference between the local median and the value of pixel to detect the impulse. Combination of these two detectors forms the Enhanced Rank Impulse Detector and integrates advantages of both of them. In combination with filter it allows iterative filtering without further image destruction.

Keywords: Impulsive Noise, Impulse Detection, Median Filter

1. INTRODUCTION

Impulsive noise filtering is an important field in image processing. As a rule, impulsive noise can significantly damage an image. The acquisition or transmission of digital images through sensor or communication channel is often interfered by impulsive noise. If a majority of the signal values do not change at all or change slightly, while some of the signal values change "dramatically" then the change is clearly visible. In this case it is possible to say that the impulsive noise corrupts the image.

Impulses may have different or the same amplitude values. Common is the appearance of noise as black and/or white spots on an image, i.e., the noisy pixels have a very small or a very large value of brightness. This type of noise is often called salt-and-pepper noise because one can create it by sprinkling salt and pepper all over an image. Pure salt-and-pepper is quite easy to remove from the image because the extremal values occur rarely in actual images. Thus, just checking whether the pixel has a maximal or minimal value, reveals if it is corrupted or not. A more realistic noise is implied by the bit errors in the signal values. The typical sources for this kind of noise are the channel errors in communications or storage errors.

A classical approach to impulsive noise filtering is the simple median filter, which replaces the pixel of interest by the local median. As a result, the noise disposal leads also to the total smoothing of the image. In other words, a median filtering leads to the detail loss. The explanation of this fact is simple. The simple median filter does not recognize, if it is necessary to correct a current pixel or not. It corrects the pixel anyway.

The weighted median filters were proposed to correct such problems as unnecessary blurring. In the simple median filter all samples inside the filter window have same influence on the filter output, while in the weighted median filter this influence can be altered. So, more emphasis can be given to the pixels at some specific locations inside the filter window. Usually it is a central pixel – the pixel, whose destiny is being considered at this moment. But the weighted median filter still does not recognize if it is necessary to correct the value of the pixel or not.[1] The pixel is changed anyway, though new value may be much closer to the expected one.

Further author information: (Send correspondence to DP)
IA: E-mail: igora@netvision.net.il JA: E-mail: jta@cs.tut.fi
TB: E-mail: karen@cs.tut.fi CB: E-mail: cbutakoff@yahoo.com
KE: E-mail: karen@cs.tut.fi DP: E-Mail: paliy@cs.tut.fi

Another interesting approach (NASM) was proposed recently.[2] The multi-pass filter is based both on the global image statistics and the statistics of the filter window. This filter consists of several passes. The first pass is a detection of a type for the sample. This algorithm separates all samples into three groups: "no filtering", edges and noisy. The data collected on the first pass, are used as parameters for the next one. The second pass is the weighted median filtering. But NASM was developed for the filtering of salt-and-pepper noise, and it is not so effective in the case of bit error impulsive noise. Anyway it is worth to note that it is computationally intensive. There are also some other impulse detection schemes such as ones based on the local statistics[3,4,5] or the global statistics,[6] the ones based on the difference between the original and filtered images[7,8]. Here we want to introduce a few very simple detectors, that provide quite an efficient impulse detection with a very small number of misdetections and their application to the filtering.

It is known that the median filter is a sliding window filter with a window $N \times N$, where N is odd. It builds variation series (pixels from the filter window in ascending order) and replaces the filtered pixel with the median. As it is known, the impulses usually differ much from the surrounding pixels in brightness. So the value of the pixel, if it is an impulse, usually belongs to one of the ends of the variation series.

Taking into account the mentioned disadvantages, it is a good idea to use some nonlinear mean for the impulse detection before any filtering. Three detectors are described in the paper. The Rank Impulse Detector analyzes the rank of the pixel of interest. Depending on the rank, it is possible to estimate whether the pixel is corrupted or not. If the rank is close to zero or to the length of the variation series (or, in other words, if the pixel is near the one of the ends of the variation series), then the considered pixel is most probably corrupted and must be filtered.

The next approach to the impulse detection is the Exponential Median Detector that uses an exponent of the difference between the median value and the value of the pixel of interest. This detector allows to make a more precise adjustment. Its combination with the median filter allows filtering of heavy impulsive noise with 20% corruption rate, although it does not preserve borders as Rank Impulse Detector does. The Exponential Detector can be reduced to linear detector, which checks only the difference between median and pixel's value, without calculation of exponent, which is mathematically proven in the paper. This can be used to simplify a filtering implementation.

The combination of Rank Impulse Detector and Exponential Median Detector gives another interesting detector. It is similar to the Rank Impulse Detector, but allows iterative filtering. First, a pixel of interest is checked if it is far from the median in the variation series, and then its value is checked against the value of the median. The combination of this detector with the median filter and its (filter's) iterative application shows excellent results – it removes the impulsive noise from the images even in the case of high corruption rate and ensures a good preservation of image boundaries. On the other hand, any combination of the different filters with the described noise detectors does not increase the computational complexity of the filters too much. Mentioned noise detection criteria can be used with widest variety of filters, not only with those that filter impulsive noise, but also with those that must exclude impulses from consideration. In the following chapters we will consider their combination with the median filter, rank-order filter, Cellular Neural Boolean filter.[9] Section 2 is completely devoted to the description of the mentioned detectors.

Some of the rank-order filters show very good results removing the impulsive noise from the image.[10] On the other hand, the level of image smoothing is sometimes high, though filtering preserves image boundaries more carefully than the median filter. The application of proposed detection criteria together with rank-order filters enhances the filter efficiency. The implementation for this case is the following. The result of rank-order filtering is used as input for the impulsive noise detector. If some of the resulting pixels are still recognized as impulses, then even the simple median filter can be used to correct these pixels. The implementation of impulse detection in combination with the rank-order filters will be considered in section 4.

Section 3 presents another interesting implementation. It is the application of described criteria to Cellular Neural Boolean Filter (CNBF). The impulsive noise detector has to be used on the first stage of filtering. Only if the pixel is detected as impulse, the CNBF is used for its correction. This approach was developed specially for the images with a low corruption rate. It is especially important, when the preservation of details and boundaries is as important as the removal of noise.

Section 5 presents the results of application of the mentioned solutions to the noisy images. The proposed solutions are compared to the commonly used techniques and to each other.

2. THE IMPULSIVE NOISE DETECTION CRITERIA

As it was mentioned in the introduction, it is always a good idea to detect impulses prior to filtering. The prior detection prevents from the filtering of all the pixels, which otherwise may lead to significant smoothing of an image. Since the impulsive noise is a noise, which dramatically changes the brightness value of a pixel, it can be identified by the height of the brightness jump in comparison to the surrounding pixels.

In this article we would like to introduce three impulse detection criteria.

The first one is a simple detector that comes from the fact that the difference between the ranks of the impulse and the median is usually large. In other words, if the variation series (pixels from the filter window arranged by their value in ascending order) for the filter window is considered, then the median is positioned in the center of the series, and an impulse is usually positioned near one of its ends. This gives a nice idea for impulse detection, which can be described by the following formula:

$$(rank\left(X_{ij}\right) \leq s) \vee (rank\left(X_{ij}\right) \geq N - s) \tag{1}$$

where X_{ij} is a pixel of interest, $s > 0$ is a threshold and N is the length of the variation series. Rank is the function that returns the rank of the element in the variation series. So, if for some value of X_{ij} the equation (1) is true then we classify X_{ij} as corrupted. For further reference we will call this criterion a *Rank Impulse Detector* (RID).

The next criterion is the *Exponential Median Detector* (EMD). It has been specially developed for images that have very high corruption rate. It can be described by the following equation:

$$\exp\left|X_{ij} - MED\left(X_{ij}\right)\right| \geq \Theta \tag{2}$$

where Θ is a threshold. As you may notice (2) simply checks the difference between the value of the median and pixel of interest. This criterion is good for very dense impulsive noise, where criterion (1) fails.

Let us define the following function:

$$f\left(x, a_x, a_y\right) = \begin{cases} \exp\left(\frac{x \cdot \ln\left(a_y\right)}{a_x}\right) & \text{, if } x \in [0, a_x) \\ x + (a_y - a_x) & \text{, if } x \in [a_x, +\infty) \end{cases} \tag{3}$$

where a_x, a_y are the parameters that influence the behavior of the function and, geometrically, represent the point where exponential curve joins the straight line (Fig. 1), thus giving control over the steepness of the exponential curve.

The detector based on (3), which can be considered as a modification of the detector (2), looks as following:

$$f\left(\left|X_{ij} - \underset{m \times n}{MED}\left(X_{ij}\right)\right|, a_x, a_y\right) \geq \Theta \tag{4}$$

where X_{ij} is a pixel of interest, $\underset{m \times n}{MED}\left(X_{ij}\right)$ is a median of a $m \times n$ window around X_{ij}, $0 \leq \Theta \leq f\left(K, a_x, a_y\right)$ is a threshold. If the inequality (4) holds, then X_{ij} is considered corrupted and should be corrected.

It is easy to see that (4) is improvement of (3), because in (3) exponent can result in very large numbers that can lead to overflow. Furthermore (4) is computationally faster since it does not calculate exponent for all the differences.

The equation (4) defines the *Exponential Median Detector* (EMD).

Let us consider the following two theorems:

THEOREM 2.1. $\forall \theta > 1\ \exists \delta > 0\ \exists X \geq 0\ \forall x \geq X : e^x \geq \theta \Leftrightarrow x \geq \delta$

Proof. *Necessity.* Let, as theorem states, $\forall\theta > 1\ \exists\delta > 0\ \exists X \ge 0\ \forall x \ge X : e^x \ge \theta$.

Let $\delta = \ln\theta$. Then $e^x \ge \theta \Rightarrow \ln e^x \ge \ln\theta \Rightarrow x \ge \delta$.

Sufficiency. Let, as theorem states, $\forall\theta > 1\ \exists\delta > 0\ \exists X \ge 0\ \forall x \ge X : x \ge \delta$.

$x \ge \delta \Rightarrow x\ln e \ge \delta\ln e \Rightarrow \ln e^x \ge \ln e^\delta \Rightarrow e^x \ge e^\delta \Rightarrow e^x \ge e^{\ln\theta} \Rightarrow e^x \ge \theta$

Theorem is proven. □

THEOREM 2.2. $\forall\theta > 0\ \exists\delta > 1\ \exists X \ge 0\ \forall x \ge X : e^x \ge \delta \Leftrightarrow x \ge \theta$

Proof. Proof is analogous to the previous one. □

The two previous theorems mean that the function (3) can be replaced by simple line $f(x) = x$. Exponential function simply changes reaction to threshold. Thus (2) and (4) can be replaced by the following:

$$\frac{\left|X_{ij} - \underset{m\times n}{MED}(X_{ij})\right|}{N} \ge \Theta \tag{5}$$

This important property will be used to define the *Enhanced Rank Impulse Detector* (ERID), which is described by the following equation:

$$\begin{gathered}[(rank(X_{ij}) \le s) \vee (rank(X_{ij}) \ge N - s)] \wedge \\ \wedge\left(\left|X_{ij} - \underset{m\times n}{MED}(X_{ij})\right| \ge \Theta\right)\end{gathered} \tag{6}$$

If equation (6) holds for some X_{ij} then it is considered corrupted and should be corrected. Criterion (6) can be considered as generalization of (1) and (4). It is easy to see that for $\Theta = 0$ (6) is reduced to (1) and for $s = rank\left[MED(X_{ij})\right]$ it is reduced to (5) which is equivalent to (2) and (4).

The last detector incorporates the edge preservation property of (1), power of (4), and allows impulsive noise filtering with different corruption rates (CR), varying from small to high.

Generally speaking, (1) can be used to remove dense impulsive noise (for example, with corruption rate of 20%), but in this case it usually takes few iterations of the filtering. This leads to significant image destruction, because sometimes, even if the pixel is not an impulse, it is far from the median. As an example, let us consider the possible 3×3 fragment of the image:

$$\begin{bmatrix} 2 & 2 & 2 \\ 2 & 8 & 7 \\ 2 & 7 & 7 \end{bmatrix}$$

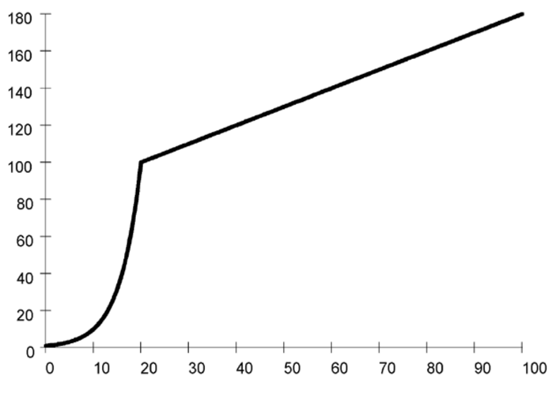

Figure 1. Example of (3), $a_x = 20$, $a_y = 100$

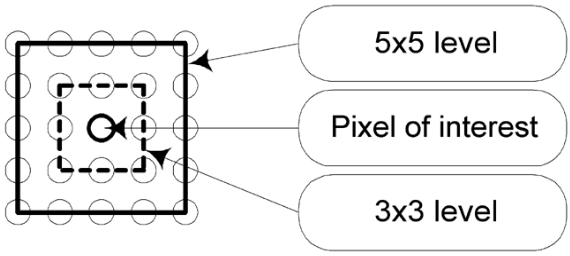

Figure 2. Levels for 5×5 CNBF

Most probably 8 in the middle is not an impulse but some part of the edge on the image. Let us build variation series:

$$(2, 2, 2, 2, 2, 7, 7, 7, 8)$$

If we will use a median filter with the described criteria then according to (1) the central pixel of the fragment must be replaced by median for any value of the parameter s. But according to (2) and (4) it is easy to find a threshold when the pixel will not be filtered, thus the image will be preserved from smoothing. But (4) alone, although it cleans very dense noise, in combination with the median filter is a bit ignorant to the boundaries. Combination of (1) and (4), which is given by (6), gives the best preservation.

Another important property of (6) is that it can be applied iteratively without further smoothing or destruction of the image. This allows the impulsive noise removal from the images even with the high corruption rate and ensures a good preservation of image boundaries.

In the following sections we are going to consider these criteria in combination with the different filters.

3. CELLULAR NEURAL BOOLEAN FILTER

The main idea of the Cellular Neural Boolean Filtering (CNBF) algorithm is that the whole image is directly decomposed into bit planes[9] (e.g. for 8-bit grayscale images we have exactly 8 bit planes). Every bit plane is processed independently of each other. Each pixel inside the bit plane is analyzed using the sliding window. This analysis and the processing are defined by the Boolean function. CNBF with a 3×3 window has been considered earlier.[9] Here we are going to develop the suggested algorithm and introduce CNBF with a 5×5 window, which uses 3×3 and 5×5 levels (Fig. 2) for the noise detection and filtering.

To be more precise, both 3×3 and 5×5 neighborhoods around the analyzed pixel will be considered for design of the Boolean function for the noise analysis and filtering. If the central pixel's value (in some specific bit plane) is opposite to the majority of the pixels' values in the window, then it has to be considered as an impulse and its value has to be inverted. Two special thresholds for a 5×5 level and a 3×3 level can be introduced as the variable parameters. These thresholds represent the average number of the bits that equal to the central bit. This way we can adjust a number of the corrected bits and a level of corrections.

Let X be a matrix of pixels from the filtering window:

$$X = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 & x_9 & x_{10} \\ x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{16} & x_{17} & x_{18} & x_{19} & x_{20} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \end{pmatrix}$$

$$\begin{aligned} L_5 &= \{x_i | x_i = x_{13}, i = 1, \ldots, 6, 10, 11, 15, 16, 20, \ldots, 25\} \\ L_3 &= \{x_i | x_i = x_{13}, i = 7, 8, 9, 12, 14, 17, 18, 19\} \\ L_5' &= \{x_i | x_i \neq x_{13}, i = 1, \ldots, 6, 10, 11, 15, 16, 20, \ldots, 25\} \\ L_3' &= \{x_i | x_i \neq x_{13}, i = 7, 8, 9, 12, 14, 17, 18, 19\} \end{aligned}$$

$x_c^l \in \{0,1\}$ is calculated as:

$$\begin{aligned} x_c^l &= f(X) = \\ &= \begin{cases} \bar{x}_{13} & \text{, if } P(L_5) \le P(L_5') \text{ and } P(L_3) \le P(L_3') \\ x_{13} & \text{, otherwise} \end{cases} \end{aligned}$$

where x_c^l is a central bit of l-th bit plane after the processing; $P(A)$ is a cardinality of the set A. It is possible to replace $P(L_5')$ and $P(L_3')$ by the parameters t_5 and t_3. These parameters could be used as "critical indexes" for the corresponding levels to influence the filtering.

It is clear that we do not want to correct all the pixels of an image. Our target is to correct only those pixels, which are really corrupted by the noise. Let us concentrate on the case of relatively small corruption rate (1-3%). For impulse detection we will use criterion (1). Let $A = \{a_k | k = 1, \ldots, 25\}$ be a variation series built from the elements of X. If $x_{13} \in [a_1, a_s] \cup [a_{25-s}, a_{25}]$, then x_{13} has to be processed according to the algorithm described above.

A special particular case is when x_c^l also satisfies equation (1), i.e. $x_c^l \in [a_1, a_s] \cup [a_{25-s}, a_{25}]$, which means that the impulse will be replaced by another impulse. In this case x_c^l must be replaced by the nearest value:

$$x_c^l = \begin{cases} a_s & \text{, if } x_c^l < a_s \\ a_{25-s} & \text{, if } x_c^l > a_{25-s} \end{cases}$$

to prevent the occurrence the new corrupted pixel instead of the old one. Another solution lies in the replacement of x_c^l by the median calculated from A, i.e. $x_c^l = MED(a_s, \ldots, a_{25-s})$. But this solution leads to the greater distortion of image boundaries than previous one.

4. RANK-ORDER FILTERS

We are going to consider here Rank-Order EV, ER and KNV filters introduced in [10]. Let

$$M_{ev} = \{x_{kl} : |x_{ij} - x_{kl}| \le T_{ev}\} \tag{7}$$

$$M_{er} = \{x_{kl} : |rank(x_{ij}) - rank(x_{kl})| \le T_{er}\} \tag{8}$$

$$M_{knv} = \left\{ v_k : \sum_{k=p}^{p+T_{knv}} |x_{ij} - v_k| = \min_p \right\} \tag{9}$$

$$k \in \left[i - \frac{N-1}{2}, i + \frac{N-1}{2} \right],$$

$$l \in \left[j - \frac{N-1}{2}, j + \frac{N-1}{2} \right]$$

$$V = \{v_k \in X : k < l \Rightarrow v_k < v_l\}, X \subseteq V$$

where X is a $N \times N$ matrix of pixels from filtering window, V is the corresponding variational series, x_{ij} is a filtered pixel (central pixel of the filter window), T_{ev}, T_{er}, T_{knv} are the manually defined parameters. In other words (7) defines a set of pixels whose value differs from the filtered pixel not more than by T_{ev}, (8) defines a set of pixels whose rank differs from the rank of the filtered pixel not more than by T_{er}, (9) defines a set of T_{knv} pixels closest to the filtered pixel by their value.

The rank-order EV, ER and KNV filters are defined by the equation $\tilde{x}_{ij} = F$, where F is one of the following:

$$F(x_{ij}, M) = MEAN(M) \tag{10}$$
$$F(x_{ij}, M) = MED(M) \tag{11}$$
$$F(x_{ij}, M, l, r) = \begin{cases} \tilde{v}_r & \text{, if } x_{ij} > \tilde{v}_r \\ \tilde{v}_l & \text{, if } x_{ij} < \tilde{v}_l \\ x_{ij} & \text{, otherwise} \end{cases} \tag{12}$$
$$\tilde{V} = \{\tilde{v}_k \in M : k < l \Rightarrow \tilde{v}_k < \tilde{v}_l\}, M \subseteq V$$
$$F(X_{ij}, M) = Q\left(\sum_k P(\tilde{v}_k)\right) \tag{13}$$
$$P(y) = \varepsilon^y, Q(z) = \left[\frac{K \cdot \arg(z)}{2\pi}\right]$$

where $\varepsilon = e^{\frac{2\pi}{K}i}$ (i is an imaginary unity), M is either M_{ev} for rank-order EV, M_{er} for rank-order ER or M_{knv} for rank-order KNV; x_{ij} and $\tilde{x}_{ij}$ are the pixels before and after filtering, l and r are called the top and bottom cut-off values; $\tilde{V}$ is a variation series corresponding to M, square brackets in definition of $Q(z)$ in (13) mean the integer part of the expression, K is a maximal number of possible discrete signal values (e.g. for 8 bits per pixel grayscale images $K = 256$).

Equations (10), (11), (12), (13) are called rank-order mean, median, cut-off[10] and multi-valued[11] filters respectively (they are variations of ER, EV and KNV filters).

5. RESULTS

The presented impulsive noise detectors were tested together with the following five filters CNBF, median, Rank-Order ER (RO-ER), Rank-Order EV (RO-EV) and Rank-Order KNV (RO-KNV).

Since the noise on the images was added artificially, it is possible to compare the results to the original clean image. Two criteria were used. One is the PSNR between the clean image and the filtered one. Another criterion is the following. Let us compose an image where the value of each pixel is equal to 1, if this pixel was corrected on the filtered image and 0 otherwise. Let us call it a *filtered noise model*. Then let us take another image, which is the *real model of the noise*. This model of noise also has each pixel equal to 1 if the corresponding pixel on the image contains impulse and 0 otherwise. A standard deviation between these noise models is taken as the second criterion. It shows the precision of impulse detection and it is also quite good to show the level of boundaries' smoothing on the image.

Fig. 4, 3, 5 show some of the filtering techniques applied to the images with the different corruption rates. They also present the differences to the original images to show the quality of detail preservation. The detailed comparison of the filtering by the discussed filters being combined with the introduced detectors are presented in Tables 1, 3, 2, 4. Both testing images were corrupted by the impulsive noise with the 2%, 5% and 15% corruption rates, respectively. Images with 15% corruption rate were not filtered with CNBF because the CNBF performs well only on the images with low corruption rate. The best results for different corruption rates are shown in bold.

6. CONCLUSIONS

Three original detectors of the impulsive noise are considered in the paper: rank impulsive detector (RID), exponential median detector (EMD) and enhanced rank impulsive detector (ERID). ERID (6) proves to be the best choice for all cases. The most important is that ERID allows filtering with high boundaries' and edges' preservation rate. This property is very important for images, which have a large amount of small details.

Table 1. Estimation of corrected pixels quantity for image "Bridge". Standard deviation between the noise models is presented.

Filter	STD		
	2%	**5%**	**15%**
CNBF(RID)	0.194	0.252	-
CNBF(EMD)	0.208	0.242	-
CNBF(ERID)	0.135	0.175	-
Median (RID)	0.348	0.421	0.499
Median (EMD)	0.196	0.215	0.318
Median (ERID)	**0.111**	**0.167**	**0.255**
RO-ER (RID)	0.415	0.505	0.693
RO-ER (EMD)	0.537	0.419	0.532
RO-ER (ERID)	0.168	0.259	0.282
RO-EV (RID)	0.497	0.543	0.627
RO-EV (EMD)	0.369	0.429	0.530
RO-EV (ERID)	0.443	0.446	0.476
RO-KNV(RID)	0.343	0.349	0.537
RO-KNV(EMD)	0.206	0.227	0.299
RO-KNV(ERID)	0.154	0.180	0.270

Table 2. Estimation of corrected pixels quantity for image "Lena". Standard deviation between the noise models is presented.

Filter	STD		
	2%	**5%**	**15%**
CNBF(RID)	0.235	0.243	-
CNBF(EMD)	0.164	0.273	-
CNBF(ERID)	0.111	0.151	-
Median (RID)	0,264	0.319	0.450
Median (EMD)	0,155	0.180	0.232
Median (ERID)	0,107	**0.145**	0.202
RO-ER (RID)	0,449	0.510	0.560
RO-ER (EMD)	0,384	0.428	0.417
RO-ER (ERID)	**0,094**	0.173	**0.201**
RO-EV (RID)	0,477	0.534	0.584
RO-EV (EMD)	0,464	0.473	0.470
RO-EV (ERID)	0,464	0.489	0.499
RO-KNV(RID)	0.321	0.321	0.435
RO-KNV(EMD)	0.177	0.190	0.235
RO-KNV(ERID)	0.130	0.147	0.223

Table 3. Filtering of the image "Bridge". PSNR comparison.

Filter	PSNR		
	2%	**5%**	**15%**
ORIGINAL	24.063	20.828	16.155
CNBF(RID)	29.636	27.309	-
CNBF(EMD)	28.248	26.493	-
CNBF(ERID)	29.921	27.399	-
Median (RID)	29.671	28.652	26.476
Median (EMD)	29.284	28.664	26.224
Median (ERID)	32.796	30.308	**27.257**
RO-ER (RID)	29.513	28.278	25.646
RO-ER (EMD)	32.403	27.920	24.746
RO-ER (ERID)	**33.416**	**30.356**	27.012
RO-EV (RID)	29.513	28.403	25.923
RO-EV (EMD)	28.631	28.848	24.844
RO-EV (ERID)	30.786	28.875	26.563
RO-KNV(RID)	30.241	28.741	25.918
RO-KNV(EMD)	28.739	28.041	25.511
RO-KNV(ERID)	30.726	29.573	26.581

Table 4. Filtering of the image "Lena". PSNR comparison.

Filter	PSNR		
	2%	**5%**	**15%**
ORIGINAL	24.915	21.755	17.183
CNBF(RID)	32.514	30.542	-
CNBF(EMD)	30.457	28.085	-
CNBF(ERID)	32.911	30.131	-
Median (RID)	33,348	33.459	30.604
Median (EMD)	35,989	33.251	30.434
Median (ERID)	36,279	34.515	**31.008**
RO-ER (RID)	33,683	32.448	30.024
RO-ER (EMD)	36,051	32.163	30.176
RO-ER (ERID)	**37,750**	**34.934**	30.624
RO-EV (RID)	35,335	33.410	30.315
RO-EV (EMD)	35,595	32.353	30.306
RO-EV (ERID)	35,965	33.861	30.788
RO-KNV(RID)	34.171	33.397	30.281
RO-KNV(EMD)	32.929	32.377	30.319
RO-KNV(ERID)	34.969	33.966	30.780

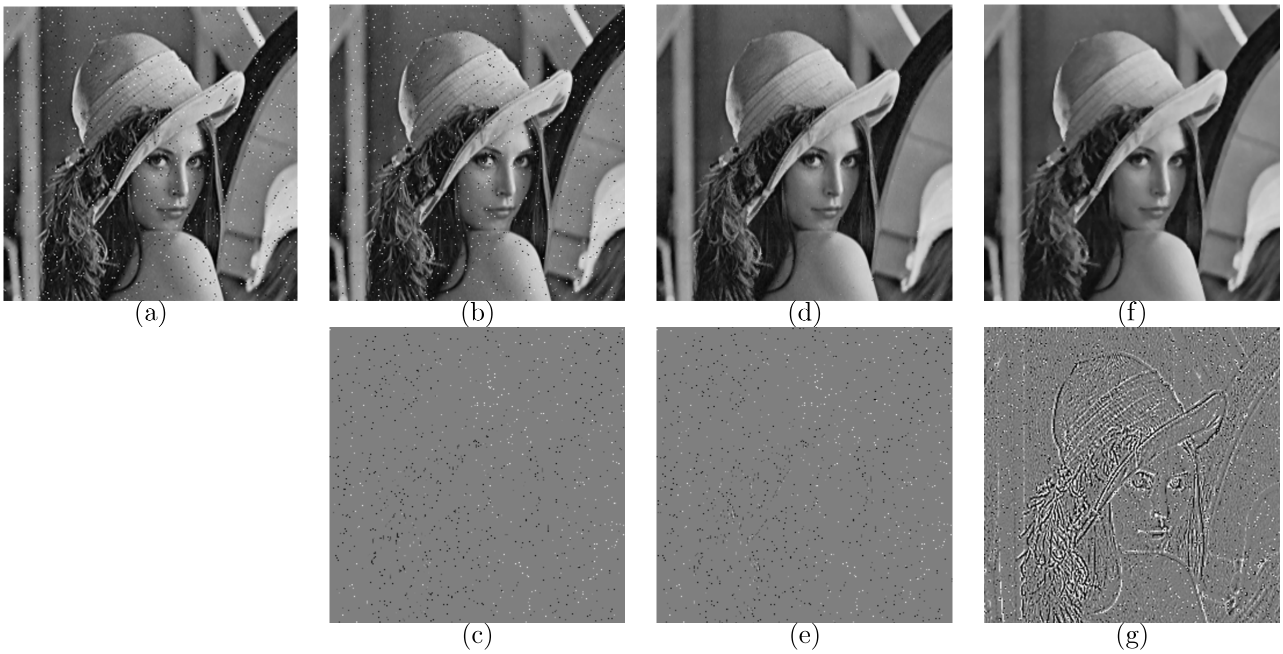

Figure 3. Filtering with the preliminary noise detection for image with 2% corruption rate. (a) The original noisy image "Lena" with CR=2%; (b) CNBF with 5×5 window with ERID, $t_5 = 0, t_3 = 5, s = 8, \Theta = 5$, 4 iterations; (c) Enhanced difference between (a) and (b); (d) Median filter with ERID, $s = 1, \Theta = 30$, 2 iterations; (e) Enhanced difference between (a) and (d); (f) Classical median filter; (g) Enhanced difference between (a) and (f);

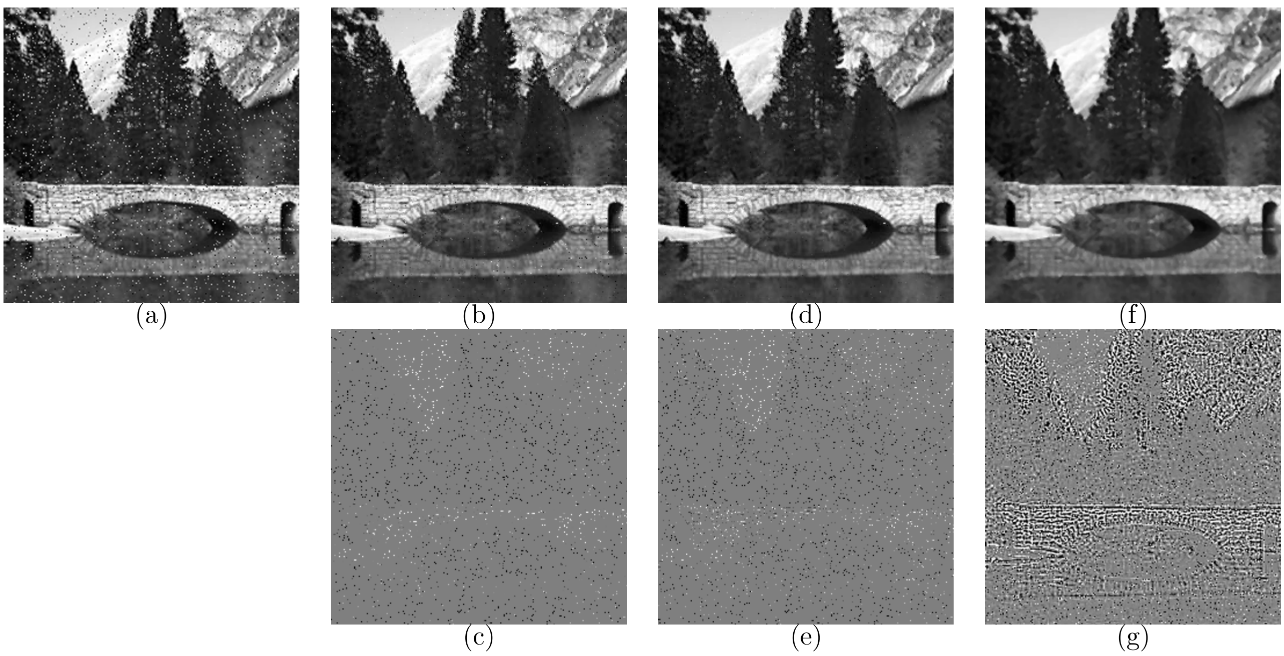

Figure 4. Filtering with the preliminary noise detection for image with 5% corruption rate. (a) Image "Bridge" corrupted by impulsive noise, CR=5%; (b) CNBF with 5×5 window with ERID, $t_5 = 0, t_3 = 6, s = 8, \Theta = 5$, 3 iterations; (c) Enhanced difference between (a) and (b); (d) Median filter with 3×3 window with ERID, $s = 1, \Theta = 30$, 2 iterations; (e) Enhanced difference between (a) and (d); (f) Classical median filter; (g) Enhanced difference between (a) and (f);

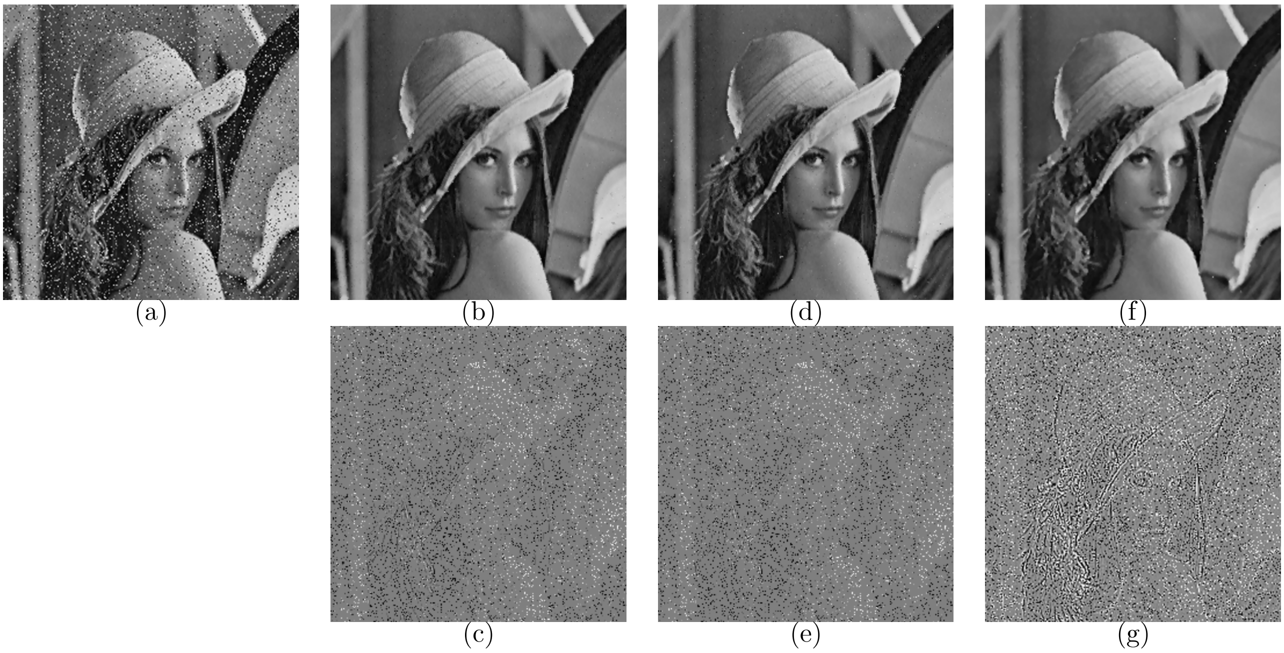

Figure 5. Filtering with the preliminary noise detection for image with 15% corruption rate. (a) Noisy image "Lena", CR=15%; (b) Rank-Order KNV with 3×3 window with ERID, $KNV = 1, s = 2, \Theta = 20$, 2 iterations; (c) Difference between (a) and (b); (d) Median filter with 3×3 window with ERID, $s = 2, \Theta = 20$, 2 iterations; (e) Difference between (a) and (d); (f) Classical median filter; (g) Difference between (a) and (f);

REFERENCES

1. J. Astola and P. Kuosmanen, *Fundamentals of Non-Linear Digital Filtering*, CRC Press, Boca Raton, N.Y., 1997.
2. H.-L. Eng and K.-K. Ma, "Noise adaptive soft-switching median filter," *IEEE Transactions on Image Processing, IP-10* , pp. 242–251, 2001.
3. T. Chen and H. R. Wu, "Adaptive impulse detection using center-weighted median filters," *IEEE Signal Processing Letters* **8**(1), pp. 1–3, Jan. 2001.
4. K.-C. Lee, H.-J. Song, and K.-H. Sohn, "Detection-estimation based approach for impulsive noise removal," *IEEE Electronics Letters* **34**(5), pp. 449–450, Mar. 1998.
5. P. Windyga, "Fast impulsive noise removal," *IEEE Transactions on Image Processing* **10**(1), pp. 173–179, Jan. 2001.
6. Y.-H. Fung and Y.-H. Chan, "An improved algorithm for removing impulse noise based on long-range correlation in an image," *Multimedia and Expo, Proceedings of IEEE International Conference* **1**, pp. 157 –160, 2002.
7. M. Yu, G. Jiang, and B. Yu, "Noise detection based impulse noise removal for color image," *The 2000 IEEE Asia-Pacific Conference on Circuits and Systems* , pp. 453–456, 2000.
8. X. Xu and E. Miller, "Adaptive two-pass median filter to remove impulsive noise," *IEEE ICIP 2002* **1**, pp. 808–811, 2002.
9. I. N. Aizenberg, N. N. Aizenberg, S. Agaian, J. Astola, and K. Egiazarian, "Nonlinear cellular neural filtering for noise reduction and extraction of image details," *SPIE Proceedings* **3646**, pp. 100–111, 1999.
10. L. P. Yaroslavsky and M. Eden, *Fundamentals of Digital Optics: Digital Signal Processing in Optics and Holography*, Springer Verlag, 1996.
11. I. Aizenberg and C. Butakoff, "Image processing using cellular neural networks based on multi-valued and universal binary neurons," *Journal of VLSI signal processing (Kluwer)* **32**, pp. 169–188, 2002.