

## Research Article

# An Efficient Universal Noise Removal Algorithm Combining Spatial Gradient and Impulse Statistic

**Shuhan Chen, Weiren Shi, and Wenjie Zhang**

*College of Automation, Chongqing University, Chongqing 400030, China*

Correspondence should be addressed to Weiren Shi; wrs@cqu.edu.cn

Received 20 March 2013; Revised 22 May 2013; Accepted 11 June 2013

Academic Editor: Marco Perez-Cisneros

Copyright © 2013 Shuhan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a novel universal noise removal algorithm by combining spatial gradient and a new impulse statistic into the trilateral filter. By introducing a reference image, an impulse statistic is proposed, which is called directional absolute relative differences (DARD) statistic. Operation was carried out in two stages: getting reference image and image denoising. For denoising, we introduce the spatial gradient into the Gaussian filtering framework for Gaussian noise removal and integrate our DARD statistic for impulse noise removal, and finally we combine them together to create a new trilateral filter for mixed noise removal. Simulation results show that our noise detector has a high classification rate, especially for salt-and-pepper noise. And the proposed approach achieves great results both in terms of quantitative measures of signal restoration and qualitative judgments of image quality. In addition, the computational complexity of the proposed method is less than that of many other mixed noise filters.

## 1. Introduction

Noise can be easily introduced into digital images due to analog-to-digital conversion errors and malfunctioning pixel elements in the camera sensors [1, 2]. Noise can significantly degrade the image quality and increase the difficulty in subsequent processing, such as image segmentation, object recognition, and edge detection. Therefore, noise removal becomes a necessary and fundamental step in image processing. However, noise removal is a difficult task because images may be corrupted by different types of noise. Fortunately, most noise added to images can be modeled by Gaussian noise and impulse noise [2].

Gaussian noise is always introduced during acquiring images and can be characterized by adding a zero-mean Gaussian distribution value into each image pixel [2, 3]. Based on this distribution property, it can be removed by locally averaging operation in general [3]. Classical linear filters are the common choice, such as the Gaussian filter which is a widely used method to remove Gaussian noise, however, it blurs edges and details significantly. In order to preserve edges and details in images while removing noise; Tomasi and Manducci proposed a bilateral filter that uses weights based

on spatial and radiometric similarity [4–6]. The bilateral filter has proven to be very useful; however, it is slow. To solve this problem, Paris and Durand proposed a fast approximation of the bilateral filter based on a signal processing interpretation [7]. It downsamples convolution computation significantly without impacting the result accuracy.

Impulse noise can occur in image transmission and characterized by replacing some pixels with noise while retaining the rest [8]. The Gaussian noise removal methods mentioned above cannot adequately remove impulse noise. Therefore, nonlinear filters have been developed for removing impulse noise such as the traditional median filter [8, 9]. The median filter has been widely applied in impulse noise reduction because of its simplicity and high computational efficiency [9]. Since an entire image is replaced by median values, median filter also modifies uncorrupted pixels. To improve performance, extensions of the median filter [10–16] and switching scheme methods are proposed. The switching scheme detects impulse noise pixels first and then replaces them with estimated values while keeping the remaining pixels unchanged [6, 10, 13, 17]. The main drawback of these filters is that they just use median values or their variations to estimate the noisy pixels, which can cause some image details

to be distorted. To overcome this, fuzzy techniques are introduced for noise removal [18–20]. With suitable fuzzy system model, they can preserve image details during noise removal. The performances of these filters also depended on the accuracy of noise detectors. Garnett et al. proposed a rank-ordered absolute differences (ROAD) statistic to identify the impulse noisy pixels and incorporated it into a trilateral filter to remove impulse noise [2]. It has proven to be a good impulse noise detector even with high noise level. Based on ROAD, Dong et al. proposed a rank-ordered logarithmic differences (ROLD) statistic to improve the accuracy of noise detection [21]. Although it obtained better performance, its running time is significantly increased comparing with the previous mentioned filters due to the logarithmic computation. In [22], Yu et al. presented a rank-ordered relative differences (RORD) statistic through introducing a reference image and combining with a simple weighted mean filter. It can not only remove impulse noise but also preserve image details.

Mixed noise could appear during transmitting an already noise corrupted image over faulty communication lines [2]. In such situation, most of the filters mentioned above will be useless. The median-based signal-dependent rank-ordered mean (SD-ROM) filter proposed by Abreu et al. can be used for mixed impulse and Gaussian noise removal [23]. But it often produces visually disappointing output when applied to images with Gaussian or mixed noise [2]. The trilateral filter with the ROAD noise detector [2] can remove both Gaussian and impulse noise effectively; however, it takes a long processing time due to the calculation of radiometric weighting function. In [6], Lin et al. proposed a switching bilateral filter with a texture/noise detector capable of removing mixed noise effectively, but for impulse noise it is not as good as SD-ROM, and for Gaussian noise it is not as good as bilateral filter. More recently, there are also some novel and encouraging approaches proposed [24, 25].

In this paper, we first propose a directional absolute relative differences (DARD) statistic for impulse noise detection by using a reference image which is similar to RORD. Different from RORD detector, our impulse detector does not need sort operation, which can reduce computation complexity. Then, we propose an improved trilateral filter by combining spatial gradient and the DARD statistic. Instead of applying the “detect and replace” methodology of most impulse noise removal techniques, we integrate our two statistics into a filter designed to remove impulse noise, Gaussian noise, and mixed noise. Finally, a two-step iterative algorithm mentioned in [22] is adopted, which includes getting reference image and image denoising.

The remainder of the paper is arranged as follows. In Section 2, we first briefly review ROAD, ROLD, and RORD statistics and then introduce our DARD statistic. Section 3 describes how to incorporate our statistics into the bilateral filter to create two new bilateral filters for Gaussian and impulse noise, respectively, and a new trilateral filter for mixed noise. In Section 4, we provide the simulations on noise detection and noise removal with visual examples and numerical results. Finally, conclusions are drawn in Section 5.

## 2. DARD Statistic for Detecting Impulse Noise

*2.1. Review on ROAD, ROLD, and RORD.* Before introducing our DARD statistic, we first briefly review three statistics: ROAD [2], ROLD [21], and RORD [22].

First, let  $d_{s,t}$  denote the absolute difference between  $I_{i,j}$  and its neighbor  $I_{i+s,j+t}$  in a  $(2N+1) \times (2N+1)$  window as

$$d_{s,t}(i, j) = |I_{i,j} - I_{i+s,j+t}|, \quad s, t \in [-N, 0) \cup (0, N]. \quad (1)$$

Then we define

$$\text{ROAD}_{i,j}^m = \sum_{n=1}^m r_{i,j}^n, \quad (2)$$

where  $r_{i,j}^n$  is the  $n$ th smallest one among  $d_{s,t}$ .

For noisy pixels, their intensities vary greatly from their neighbors and yield large ROAD values, while noise-free pixels should have similar intensities with their neighbors and produce small ROAD values. Thus, we can identify noise by the ROAD value. However, the ROAD value of a pixel may not be large enough for it to be distinguished from noise-free pixel when the noise value is close to its neighbors [21]. Then, ROLD was proposed to solve it by introducing a logarithmic function on the absolute difference  $\dot{d}_{s,t}$  defined as

$$\dot{d}_{s,t}(i, j) = 1 + \frac{\max\{\log_2 |I_{i,j} - I_{i+s,j+t}|, -5\}}{5}. \quad (3)$$

Then, define ROLD as

$$\text{ROLD}_{i,j}^m = \sum_{n=1}^m \dot{r}_{i,j}^n, \quad (4)$$

where  $\dot{r}_{i,j}^n$  is the  $n$ th smallest one among  $\dot{d}_{s,t}$ .

Although ROLD is more accurate than ROAD to separate noisy pixels from noise-free pixels, it significantly increases computation complexity.

Yu et al. found that ROAD is not accurate at edge pixels because edge details in an image also cause large absolute difference values [22]. Thus, they solved it by introducing a reference image, which is defined as

$$\hat{I}^{\text{ref}} = I - \alpha \tilde{I}^{\text{ref}}. \quad (5)$$

Then,  $\ddot{d}_{s,t}$  and RORD were defined as follows:

$$\ddot{d}_{s,t}(i, j) = |\hat{I}_{i,j}^{\text{ref}} - \hat{I}_{i+s,j+t}^{\text{ref}}|, \quad (6)$$

$$\text{RORD}_{i,j}^m = \sum_{n=1}^m \ddot{r}_{i,j}^n, \quad (7)$$

where  $\ddot{r}_{i,j}^n$  is the  $n$ th smallest one among  $\ddot{d}_{s,t}$ .

Merging (5) into (6),  $\ddot{d}_{s,t}$  can be rewritten as

$$\ddot{d}_{s,t}(i, j) = |(I_{i+s,j+t} - I_{i,j}) - \alpha (\hat{I}_{i+s,j+t}^{\text{ref}} - \hat{I}_{i,j}^{\text{ref}})|. \quad (8)$$

With the help of subtracting  $(\hat{I}_{i+s,j+t}^{\text{ref}} - \hat{I}_{i,j}^{\text{ref}})$ , RORD can distinguish noisy pixels from edge pixels well. However,

<table border="1"> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td><b>255</b></td><td>100</td><td>100</td></tr> <tr><td>200</td><td><b>255</b></td><td>200</td><td><b>255</b></td><td>100</td></tr> <tr><td>200</td><td>200</td><td><b>255</b></td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> </table>	200	200	200	100	100	200	200	<b>255</b>	100	100	200	<b>255</b>	200	<b>255</b>	100	200	200	<b>255</b>	100	100	200	200	200	100	100	<table border="1"> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> </table>	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	<table border="1"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>160</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>110</td><td><b>100</b></td><td>360</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>160</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	160	0	0	0	110	<b>100</b>	360	0	0	0	160	0	0	0	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>110</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>110</td><td><b>0</b></td><td>160</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>110</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	110	0	0	0	110	<b>0</b>	160	0	0	0	110	0	0	0	0	0	0	0
200	200	200	100	100																																																																																																			
200	200	<b>255</b>	100	100																																																																																																			
200	<b>255</b>	200	<b>255</b>	100																																																																																																			
200	200	<b>255</b>	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
0	0	0	0	0																																																																																																			
0	0	160	0	0																																																																																																			
0	110	<b>100</b>	360	0																																																																																																			
0	0	160	0	0																																																																																																			
0	0	0	0	0																																																																																																			
0	0	0	0	0																																																																																																			
0	0	110	0	0																																																																																																			
0	110	<b>0</b>	160	0																																																																																																			
0	0	110	0	0																																																																																																			
0	0	0	0	0																																																																																																			
<i>I</i>	<i>I</i> <sup>ref</sup>	RORD	DARD																																																																																																				

<table border="1"> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td><b>50</b></td><td><b>150</b></td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td><b>230</b></td><td>100</td></tr> <tr><td>200</td><td><b>255</b></td><td><b>50</b></td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> </table>	200	200	200	100	100	200	<b>50</b>	<b>150</b>	100	100	200	200	200	<b>230</b>	100	200	<b>255</b>	<b>50</b>	100	100	200	200	200	100	100	<table border="1"> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> <tr><td>200</td><td>200</td><td>200</td><td>100</td><td>100</td></tr> </table>	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	200	200	200	100	100	<table border="1"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>550</td><td>100</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td><b>150</b></td><td>470</td><td>0</td></tr> <tr><td>0</td><td>220</td><td>500</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	550	100	0	0	0	0	<b>150</b>	470	0	0	220	500	0	0	0	0	0	0	0	<table border="1"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>300</td><td>50</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td><b>0</b></td><td>150</td><td>0</td></tr> <tr><td>0</td><td>110</td><td>150</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	300	50	0	0	0	0	<b>0</b>	150	0	0	110	150	0	0	0	0	0	0	0
200	200	200	100	100																																																																																																			
200	<b>50</b>	<b>150</b>	100	100																																																																																																			
200	200	200	<b>230</b>	100																																																																																																			
200	<b>255</b>	<b>50</b>	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
200	200	200	100	100																																																																																																			
0	0	0	0	0																																																																																																			
0	550	100	0	0																																																																																																			
0	0	<b>150</b>	470	0																																																																																																			
0	220	500	0	0																																																																																																			
0	0	0	0	0																																																																																																			
0	0	0	0	0																																																																																																			
0	300	50	0	0																																																																																																			
0	0	<b>0</b>	150	0																																																																																																			
0	110	150	0	0																																																																																																			
0	0	0	0	0																																																																																																			
<i>I</i>	<i>I</i> <sup>ref</sup>	RORD	DARD																																																																																																				

FIGURE 1: Two failure examples of RORD compared with our DARD. Red denotes noisy pixels.

$I_{i+s,j+t}$  may be noisy especially when the noise level is high. In such case, RORD may be not accurate. Although it may be more robust by accumulating  $m$  smallest ones among  $\ddot{d}_{s,t}$ , it brings errors at the same time. Take Figure 1 for example, in the  $3 \times 3$  window of the center pixel, half or more than half of its neighbors are corrupted with impulse noise, and then RORD will falsely mark the center pixel as a noise. However, the proposed DARD can work well in this case, which will be described in the next subsection.

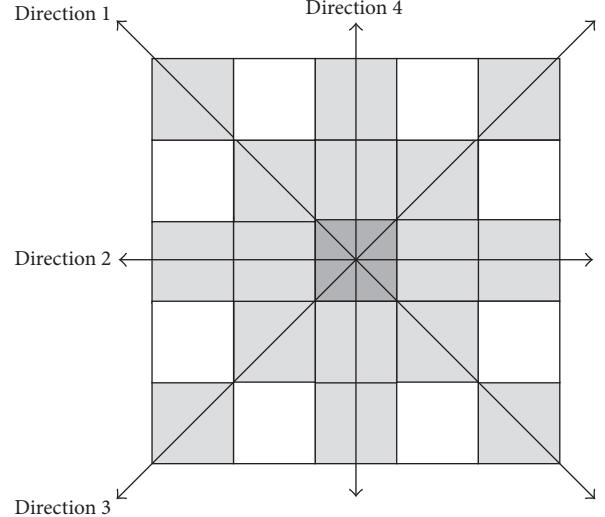
**2.2. Definition of DARD.** Our new impulse detector is based on the following two assumptions. (1) Noise-free images contain locally smoothly varying areas separated by image edges [10]. (2) Noisy pixels take gray scale values substantially smaller or larger than their neighborhoods [26, 27]. Here, we only focus on the edges aligned with four main directions shown in Figure 2. Let  $S_k$  ( $k = 1$  to 4) denote a set of coordinates aligned with the  $k$ th direction centered at  $(0, 0)$  with window size  $(2N + 1) \times (2N + 1)$ , that is,

$$S_1 = \{(-N, -N), \dots, (-1, -1), (0, 0), (1, 1), \dots, (N, N)\} \quad (9a)$$

$$S_2 = \{(0, -N), \dots, (0, -1), (0, 0), (0, 1), \dots, (0, N)\} \quad (9b)$$

$$S_3 = \{(N, -N), \dots, (1, -1), (0, 0), (-1, 1), \dots, (-N, N)\} \quad (9c)$$

$$S_4 = \{(-N, 0), \dots, (-1, 0), (0, 0), (1, 0), \dots, (N, 0)\}. \quad (9d)$$

FIGURE 2: Four directions for noise detection in a  $(2N + 1) \times (2N + 1)$  window ( $N = 2$  for example).

At first, we introduce a reference image ( $I^{\text{ref}}$ ) that contains the original image edge information from the noisy image. In the filter window centered at  $(i, j)$ , for each direction, define  $d_{i,j}^k$  as the sum of all absolute differences of gray-level values between centered pixel in  $I$  and its neighbors in  $I^{\text{ref}}$ . Then, we have

$$d_{i,j}^k = \sum_{(s,t) \in S_k} |I_{i+s,j+t}^{\text{ref}} - I_{i,j}|, \quad k = 1 \sim 4. \quad (10)$$

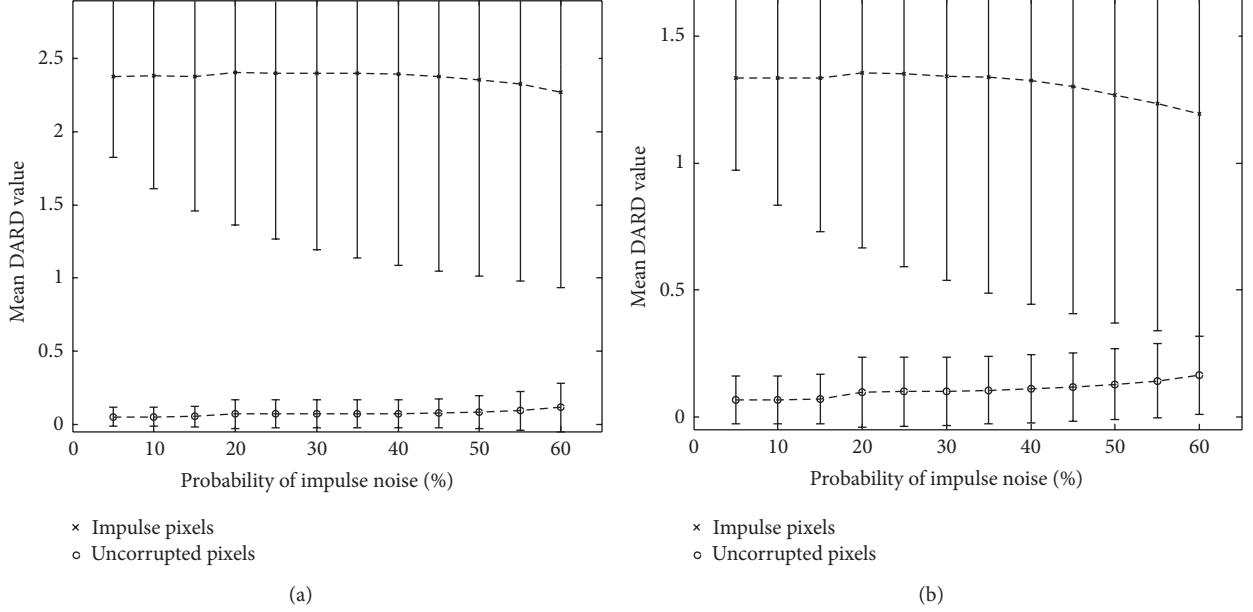


FIGURE 3: Error-bar charts for our DARD statistic on (a) “Lena” with salt-and-pepper impulse noise and (b) “Boats” with random-valued impulse noise.

Similarly, we define  $g_{i,j}^k$  as the sum of all differences of each direction in the reference image as

$$g_{i,j}^k = \sum_{(s,t) \in S_k} |I_{i+s,j+t}^{\text{ref}} - I_{i,j}^{\text{ref}}|, \quad k = 1 \sim 4. \quad (11)$$

Finely textured or detailed regions in an image cause some kind of naturally large absolute difference values. Therefore, it is difficult to tell the difference between a texture and an impulse noise. In order to solve this case and improve the accuracy of impulse noise detection, we define an absolute relative difference statistic for each direction as follows:

$$\text{DARD}_{i,j}^k = |d_{i,j}^k - \alpha g_{i,j}^k|, \quad k = 1 \sim 4, 0 \leq \alpha \leq 1. \quad (12)$$

If  $\alpha$  is set to 1, and the reference image is equal to the original image, then, all intensity variation caused by edges can be eliminated. However, the reference image is just generated from the noisy image by median filter (detailed generation process is shown in Section 3.2), in which only rough information of the original image is contained. In other words, the reference image can still be noisy, not only the edge details, which can also cause large intensity variation. To reduce it, we set  $\alpha = 0.5$  for simplicity. Finally, the directional absolute relative difference statistic is defined as

$$\text{DARD} = \min (\text{DARD}^1, \text{DARD}^2, \text{DARD}^3, \text{DARD}^4). \quad (13)$$

Then, the decision making mechanism can be realized by employing a threshold and the impulse noise detection algorithm is shown as

$$\begin{aligned} & \text{If } \text{DARD}_{i,j} > T, \quad I_{i,j} \text{ is impulse noise} \\ & \text{Else } I_{i,j} \text{ is noise-free.} \end{aligned} \quad (14)$$

In general, salt-and-pepper noisy pixels have larger DARD values while random-valued impulse and Gaussian ones have smaller DARD values, which can be seen in Figure 3.

### 3. The Proposed Method

**3.1. New Weighting Functions.** Bilateral filter presented by Tomasi and Manduchi [4] has been proven to be very useful in removing Gaussian noise and simultaneously preserving edge details. Its main idea is combining grey levels based on both the photometric similarity and geometric closeness. It shows great results but takes a long processing time. Although some improved methods have been proposed, they are not efficient enough [7, 28]. Furthermore, the performance of the bilateral filter is degraded with high noise level as mentioned above. In this work, we present two new weighting functions for designing filters, as discussed below.

Spatial gradient statistic is first introduced into the bilateral filtering framework through replacing the radiometric weighting function. The new created bilateral filter, named SG-BF, is capable of removing Gaussian noise while keeping edge details. The new weighting function is defined as

$$w_G(i, j) = e^{-G(i, j)^2 / 2\sigma_G^2}. \quad (15)$$

Here,  $G(i, j)$  is gradient which can be generated by using Sobel operator. Let  $I_{i,j}$  be the current pixel, and let  $I_{i+s,j+t}$  be the pixels in a  $(2N+1) \times (2N+1)$  window that surrounds  $I_{i,j}$ ;  $(i, j)$  and  $(i+s, j+t)$  are the locations of  $I_{i,j}$  and  $I_{i+s,j+t}$ . Then the new output of SG-BF is defined as

$$\tilde{I}_{i,j} = \frac{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_G(s, t) I_{i+s,j+t}}{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_G(s, t)}, \quad (16)$$

where

$$w_S(i, j) = e^{-(i-s)^2 + (j-t)^2 / 2\sigma_s^2}. \quad (17)$$

By replacing radiometric weighting function with  $w_G$ , our running time can be significantly reduced comparing with the original bilateral filter. In order to let bilateral filter be capable of removing impulse noise, we further incorporate the DARD statistic into the bilateral filter to create a new bilateral filter, and we name it DARD-BF. The new weighting function is defined as

$$w_I(i, j) = e^{-\text{DARD}(i, j)^2 / 2\sigma_I^2}. \quad (18)$$

Then, the new output of DARD-BF bilateral filter is defined as

$$\tilde{I}_{i,j} = \frac{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_I(s, t) I_{i+s, j+t}}{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_I(s, t)}. \quad (19)$$

Finally, we combine the spatial gradient with the DARD statistic to create a new trilateral filter, which can remove both Gaussian and impulse noise, and we name it SG-DARD-TRIF. The new output of SG-DARD-TRIF is defined as

$$\tilde{I}_{i,j} = \frac{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_G(s, t) w_I(s, t) I_{i+s, j+t}}{\sum_{s=-N}^N \sum_{t=-N}^N w_S(s, t) w_G(s, t) w_I(s, t)}. \quad (20)$$

In brief, our new trilateral filter can not only preserve the bilateral filter's ability to remove Gaussian noise but also work well for impulse noise. For images with no impulse noise, the value of impulsive component is nearly to one except for few points with high DARD values, and thus the impulsive component will be "shut off" and only the spatial and gradient weights are used. Essentially, the trilateral filter reverts to the bilateral filter when processing images with Gaussian noise only. For images with impulse noise only, the gradient component will also help to enhance the performance of impulse noise removal.

**3.2. Denoising Algorithm.** As can be seen, the reference image  $I^{\text{ref}}$  plays an important role in our algorithm. In order to get a satisfactory denoising result, we adopt the two-step iterative algorithm mentioned in [22]. In the first step, the initial reference image is generated by using the standard median filter (SMF). If the noise ratio is high, two or more iterations are needed. In the second step, the previously generated restoration result is used as the final reference image. Then the more accurate impulse noise detection result is obtained by using the more satisfactory reference image. Finally, the final restored image will be obtained by using our new bilateral filter (DARD-BF) or trilateral filter (SG-DARD-TRIF). Different from [22], iterative operation is only applied in the first step. Our new denoising algorithm, DARD-BF and SG-DARD-TRIF, is summarized as follows.

#### (1) Getting Reference Image.

- (a) Set  $k = 1$ ,  $I^{\text{ref}} = u^k$ , and  $u^k = \text{SMF}(I)$ .

(b) Restore all pixels by DARD-BF or SG-DARD-TRIF and get the new reference image  $u^{k+1}$ ,  $k = k + 1$ .

(c) If  $k \leq k_{\max}$ , set  $I^{\text{ref}} = u^k$  and then go to step 1(b); Otherwise, stop iteration and get the final reference image.

#### (2) Image Denoising.

If  $\text{DARD}(i, j) > T$ , restore all these pixels by DARD-BF or SG-DARD-TRIF; otherwise, take  $\tilde{I}_{i,j} = I_{i,j}$ .

In general, we use  $3 \times 3$  window median filtering in step 1, and use  $5 \times 5$  window for calculating DARD values in step 2. If the noise ratio is higher than 25%, two or three iterations ( $k_{\max} = 2$  or  $3$ ) and  $5 \times 5$  window median filtering are needed in step 1. By tries and errors, threshold  $T$  is set as  $[0.01, 0.10]$ , with a higher value for salt-and-pepper noise but a lower one for random-valued impulse noise and Gaussian noise. It is worth noting that we do not employ threshold  $T$  in step 1, which is different from many other methods [6, 17, 21, 22]. Although some suggestions of threshold selection are given in their literatures, it still needs trial and error. Therefore, we do not apply "detect and replace" scheme. In this work, threshold  $T$  is not as important as the other methods, which means that the proposed method can still work well when threshold  $T$  is set as 0.

## 4. Simulations

The performance of the proposed filters have been evaluated and compared with those of several existing filters for image restoration. The proposed method produced results superior to other methods in both visual image quality and quantitative measures. Simulations were made on several  $512 \times 512$  gray scale standard test images corrupted with Gaussian noise, salt-and-pepper noise, random-valued impulse noise, and mixed noise. For illustrations, the results for images "Lena", "Boats", "Bridge", "Baboon", and "Barbara" are presented here.

**4.1. Selection of Parameters.** There are three parameters in our algorithm:  $\sigma_S$  (controls spatial weight),  $\sigma_G$  (controls gradient weight), and  $\sigma_I$  (controls DARD weight). From simulations on a large variety of images, we found that the better performance was obtained by the following settings:  $\sigma_G$  should be in the interval  $[0.3, 0.8]$  for all kinds of noise. In general, the selection of  $\sigma_I \in [0.4, 0.6]$  yields satisfactory results for salt-and-pepper noise, while the setting of  $\sigma_I \in [0.2, 0.4]$  consistently performs well for random-valued impulse noise. For Gaussian noise,  $\sigma_S$  should be in the range of  $[0.8, 2.0]$ , while  $[0.4, 1.0]$  for others. In addition, higher values of  $\sigma_S$  and  $\sigma_I$  work better with high noise level and higher  $\sigma_G$  performs well in images with more textured details.

**4.2. Noise Detection.** To demonstrate the effectiveness of our DARD statistic, we make a test on a  $512 \times 512$  Lena image

TABLE 1: Comparison of noise detectors in CR.

Noise ratio	ROAD	ROLD	RORD	DARD
Salt-and-pepper noise				
10%	99.48%	99.37%	99.77%	99.87%
20%	98.75%	98.63%	99.44%	99.73%
30%	97.15%	96.74%	98.53%	99.58%
40%	93.21%	92.65%	95.86%	99.16%
50%	83.58%	83.75%	86.01%	98.89%
Random-valued impulse noise				
10%	97.25%	97.38%	97.71%	97.56%
20%	94.59%	94.83%	95.68%	95.57%
30%	92.54%	92.88%	93.71%	93.54%
40%	90.62%	91.04%	91.75%	91.53%
50%	88.77%	89.31%	89.37%	88.87%
Mixed impulse noise				
10%	97.26%	97.40%	97.87%	97.57%
20%	94.51%	94.77%	95.65%	95.50%
30%	92.06%	92.47%	93.48%	93.36%
40%	89.51%	90.11%	90.81%	90.77%
50%	86.53%	87.52%	87.21%	87.28%

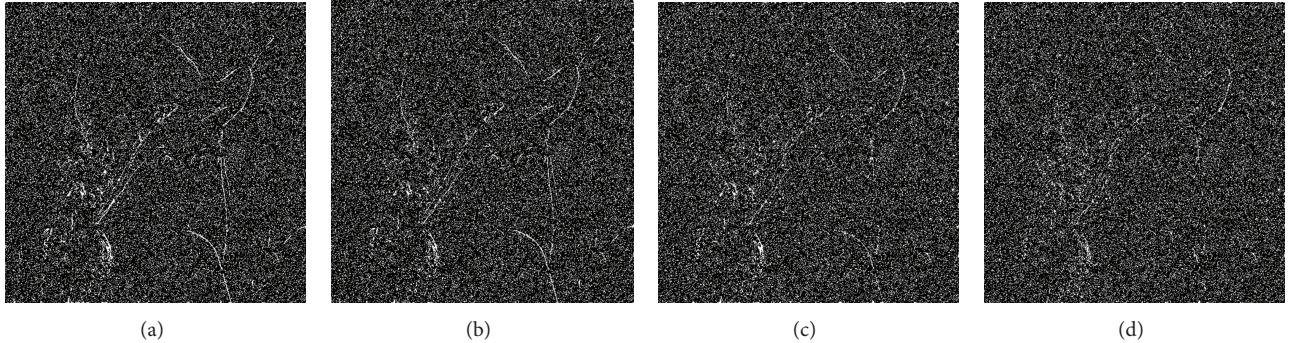


FIGURE 4: Comparison of noise detectors for 20% random-valued impulse noise corrupted image of Lena: (a) ROAD; (b) ROLD; (c) RORD; (d) DARD.

compared to ROAD, ROLD, and RORD. Here, we suppose the locations of all noisy pixels are known in advance, and then all pixels can be grouped into two sets: the noisy pixel set and the noise-free pixel set.

A good noise detector should be able to identify most of the noisy pixels and noise-free pixels, and yet its classification rate should be as high as possible. The classification rate (CR) is defined as

$$\begin{aligned} \text{CR} = & \frac{\text{number of correctly detected noise pixels}}{\text{total number of pixels}} \\ & + \frac{\text{number of correctly detected noise-free pixels}}{\text{total number of pixels}}. \end{aligned} \quad (21)$$

The results are shown in Table 1. From the experiment results, it can be seen that our DARD detector achieves

significant improvement over other detectors for salt-and-pepper noise, especially when the noise level is higher than 40%. For random-valued and mixed impulse noise, our DARD detector also shows better results than ROAD and ROLD detectors. Although the classification rate is slightly lower than RORD, our DARD detector has fewer edge pixels falsely detected as noise pixels. The reason is due to the fact that more pixels in subwindow are considered in RORD. Figure 4 shows the results in detecting Lena image, which is corrupted with 20% random-valued impulse noise, where the white denotes detected noise pixels and the black denotes noise-free pixels. It is clear that the DARD detector has the fewest edge pixels falsely detected as noise pixels.

**4.3. Image Quality.** To ensure that our approach provides a visually pleasing output, we make three simulations as follows. One is the Lena image contaminated by mixed salt-and-pepper and Gaussian noise with  $p = 20\%$  and  $\sigma = 10$ .

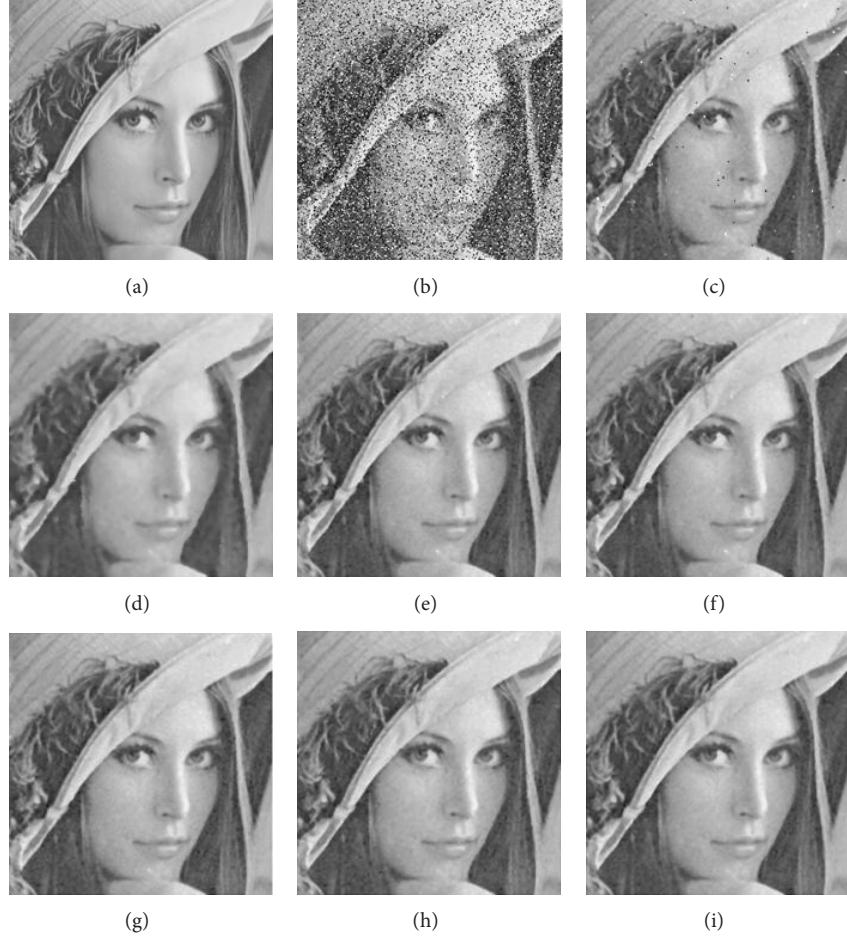


FIGURE 5: (a) Part of the original Lena image. (b) Image corrupted with mixed noise (salt-and-pepper and Gaussian noise;  $p = 20\%$ ,  $\sigma = 10$ ). (c)  $3 \times 3$  median filter. (d)  $5 \times 5$  median filter. (e) ROAD-TRIF. (f) ROLD-EPR. (g) RORD-WMF. (h) DWM filter. (i) SG-DARD-TRIF.

The other one is the Boats image corrupted by mixed random-valued impulse noise and Gaussian noise with  $p = 20\%$  and  $\sigma = 10$ . The last is the Bridge image corrupted by mixed salt-and-pepper and random-valued impulse noise with  $p = 30\%$ . The results are shown in Figures 5, 6, and 7. It is clear to see that the SG-DARD-TRIF can remove noise while preserving the edge details by comparing with the original image.

**4.4. Signal Restoration.** The objective quantitative measures used for comparison are the mean absolute error (MAE) [1] and the peak signal-to-noise ratio (PSNR) [1] between the original and restored images, defined by

$$\text{MAE} = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |I'_{i,j} - I_{i,j}|,$$

$$\text{PSNR}(I'_{i,j}) = 10 \log \frac{255^2}{(1/(M \times N)) \sum_{i=1}^M \sum_{j=1}^N (I'_{i,j} - I_{i,j})^2}, \quad (22)$$

where  $I_{i,j}$  and  $I'_{i,j}$  denote the pixel values of the original image

and the restored image, respectively, and the image size is  $M \times N$ . Larger PSNR value signifies better image restoration while lower for MAE.

Our first goal is to ensure that the proposed filters can effectively restore the pixels corrupted by impulse noise. This can be justified by comparing the performance of our approach with other well-known filters for impulse noise reduction. The group of these filters consists of the standard median filter, the adaptive center-weighted median filter (ACWMF) [29], the SDROM filter, the switching bilateral filter (SBF), ROAD-TRIF, ROLD-EPR, ROLD-WMF, and DWM filter. The results of PSNR and MAE values on salt-and-pepper noise are shown in Table 2. The proposed method shows significant better PSNR and MAE values than other filters both for Lena and Boats images. Table 3 shows the PSNR and MAE values for random-valued impulse noise, and the proposed filters also show better results than other filters except the RORD-WMF. It is noted that the PSNR and MAE values of our SG-DARD-TRIF are slightly better than our DARD-BF, which denotes that the use of the spatial gradient can enhance the filter's performance.

We also compared the performance of the proposed SG-BF with the performance of the previously tested filters on

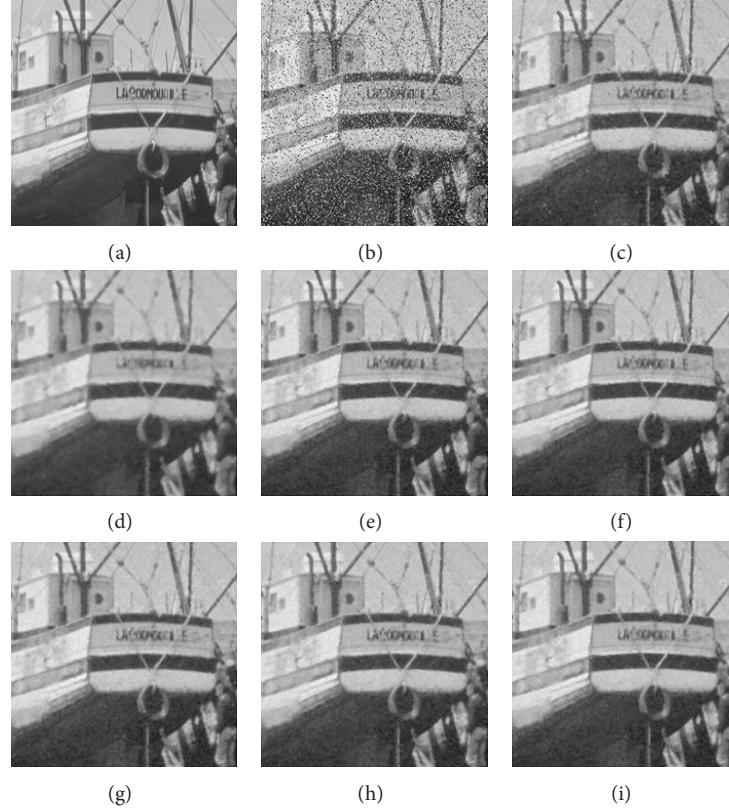


FIGURE 6: (a) Part of the original Boats image. (b) Image corrupted with mixed noise (random-valued impulse and Gaussian noise;  $p = 20\%$ ,  $\sigma = 10$ ). (c)  $3 \times 3$  median filter. (d)  $5 \times 5$  median filter. (e) ROAD-TRIF. (f) ROLD-EPR. (g) RORD-WMF. (h) DWM filter. (i) SG-DARD-TRIF.

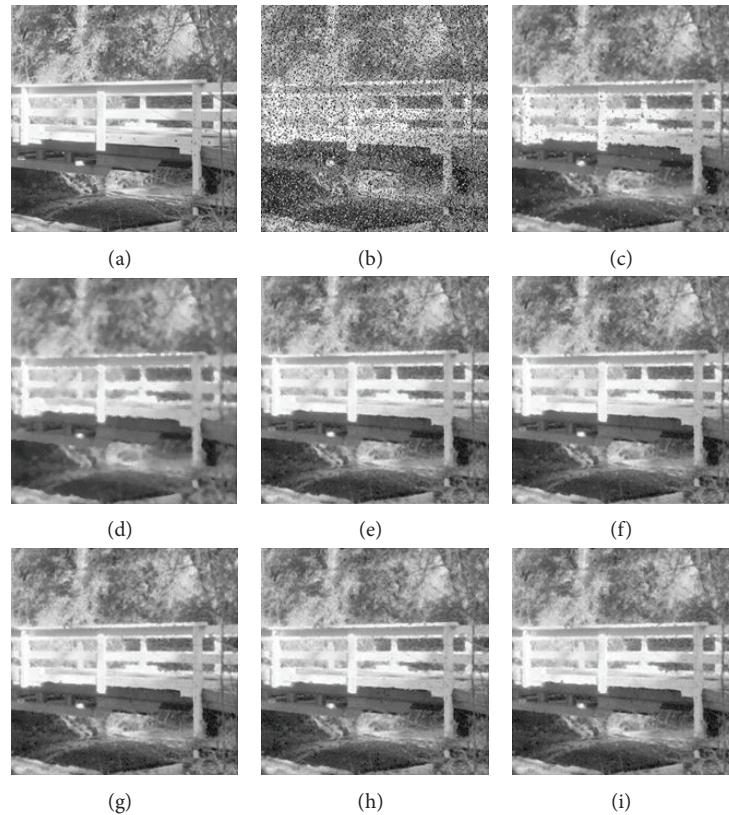


FIGURE 7: (a) Part of the original Bridge image. (b) Image corrupted with mixed noise (salt-and-pepper and random-valued impulse noise;  $p = 30\%$ ). (c)  $3 \times 3$  median filter. (d)  $5 \times 5$  median filter. (e) ROAD-TRIF. (f) ROLD-EPR. (g) RORD-WMF. (h) DWM filter. (i) SG-DARD-TRIF.

TABLE 2: Comparative restoration results in PSNR (dB) and MAE (the second row) for salt-and-pepper noise.

Method	Lena image				Boats image			
	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$
Noisy image	12.42	10.67	9.400	8.460	12.32	10.52	9.300	8.340
	12.42	18.60	24.67	30.95	13.49	20.51	27.27	33.86
$3 \times 3$ median filter	29.57	23.95	19.06	15.39	27.91	23.11	18.90	15.20
	1.663	2.519	4.725	8.473	2.276	3.136	4.834	8.555
$5 \times 5$ median filter	30.22	29.50	28.12	24.41	27.33	26.68	25.70	23.05
	1.963	2.139	2.423	3.052	3.049	3.220	3.390	4.160
ACWMF	31.09	29.81	28.23	24.23	27.41	26.71	25.65	22.89
	1.205	1.886	2.306	3.207	2.977	3.182	3.313	4.404
SDROM	37.74	35.50	32.45	30.81	33.76	30.57	28.86	26.22
	0.511	0.810	1.130	1.546	0.870	1.577	1.991	2.863
ROAD-TRIF	34.81	31.15	27.41	23.10	32.12	27.96	24.18	21.96
	0.847	1.297	2.617	3.612	1.013	2.548	4.177	4.886
DWM	35.64	33.37	30.65	27.02	32.27	29.93	27.95	25.05
	0.789	1.013	1.553	2.652	0.996	1.732	2.780	3.380
DARD-BF	38.44	35.61	33.22	31.34	34.39	31.67	29.39	27.73
	0.382	0.789	1.013	1.290	0.733	1.334	1.672	2.146
SG-DARD-TRIF	38.52	35.75	33.34	31.52	34.45	31.81	29.81	27.94
	0.369	0.757	0.971	1.103	0.716	1.080	1.478	2.091

TABLE 3: Comparative restoration results in PSNR (dB) and MAE (the second row) for random-valued impulse noise.

Method	Lena image				Boats image			
	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$
Noisy image	16.25	14.48	13.19	12.27	15.99	14.25	12.94	11.97
	6.923	10.41	13.96	17.26	8.332	12.37	16.75	20.88
$3 \times 3$ median filter	31.54	28.17	24.68	21.60	29.22	26.55	23.46	20.48
	1.770	2.450	3.692	5.633	2.216	2.846	4.003	5.928
$5 \times 5$ median filter	30.14	29.11	27.77	25.53	27.31	26.64	25.56	23.61
	2.213	2.606	3.214	4.300	3.085	3.349	3.868	5.068
SDROM	35.39	33.44	31.48	29.51	32.40	30.54	28.88	27.43
	0.620	0.972	1.362	1.830	1.195	1.586	2.279	2.864
ROAD-TRIF	34.66	33.05	31.19	29.43	31.85	30.25	28.65	27.41
	1.004	1.171	1.505	1.919	1.477	1.711	2.381	2.869
ROLD-EPR	34.84	33.15	31.28	29.44	32.04	30.32	28.72	27.42
	0.743	1.074	1.411	1.911	1.330	1.607	2.315	2.867
RORD-WMF	35.49	33.52	31.61	29.70	32.45	30.73	29.07	27.37
	0.598	0.960	1.287	1.762	1.187	1.574	2.230	2.884
DWM	34.44	32.51	30.70	28.74	31.55	29.77	28.12	26.40
	0.838	1.546	1.748	2.737	1.510	1.908	2.784	3.233
DARD-BF	35.14	33.08	31.16	29.18	32.21	30.23	28.51	27.20
	0.703	1.120	1.524	2.014	1.286	1.715	2.446	2.997
SG-DARD-TRIF	35.23	33.23	31.37	29.45	32.32	30.34	28.67	27.39
	0.651	1.005	1.389	1.903	1.218	1.604	2.378	2.876

images corrupted with Gaussian noise. From Table 4, we can see that SG-BF produces nearly the same results with the bilateral and trilateral filters when  $\sigma \leq 20$ , but it shows better results than these two filters when  $\sigma = 40$ . Thus, our SG-BF is better than these existing filters when image is highly

corrupted with Gaussian noise. Tables 5 and 6 show PSNR and MAE values corrupted with three kinds of mixed noise: salt-and-pepper and Gaussian with  $p = 20\%$  and  $\sigma = 10$ ; random-valued impulse and Gaussian with  $p = 20\%$  and  $\sigma = 10$ ; salt-and-pepper and random-valued impulse with

TABLE 4: Comparative restoration results in PSNR (dB) and MAE (the second row) for Gaussian noise.

Method	Lena image			Boats image		
	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$	$\sigma = 10$	$\sigma = 20$	$\sigma = 40$
Noisy image	28.13	22.14	16.37	28.13	22.17	16.40
	4.002	7.961	15.50	3.983	7.914	15.23
$3 \times 3$ median filter	32.13	28.44	23.44	30.25	27.43	23.07
	2.267	3.702	6.744	2.734	4.086	7.059
$5 \times 5$ median filter	30.44	28.94	25.95	27.67	26.72	24.63
	2.280	3.083	4.787	3.213	3.906	5.502
Gaussian filter	33.14	30.12	27.08	32.04	28.83	25.86
	2.216	2.856	4.459	2.885	3.475	5.133
Bilateral filter	33.90	30.67	27.19	32.69	29.22	25.94
	1.975	2.785	4.293	2.322	3.358	5.113
ROAD-TRIF	33.97	30.57	27.11	32.92	29.23	25.93
	1.847	2.811	4.239	2.180	3.359	5.111
SG-BF	33.68	30.68	27.32	32.44	29.19	26.03
	2.004	2.764	4.117	2.322	3.391	5.023

TABLE 5: Comparative restoration results in PSNR (dB) and MAE (the second row) for mixed noise ( $p = 20\%$ sp,  $\sigma = 10$ : salt-and-pepper and Gaussian;  $p = 20\%$ rv,  $\sigma = 10$ : random-valued impulse and Gaussian;  $p = 30\%$ : salt-and-pepper and random-valued impulse noise).

Method	Lena image			Boats image		
	$p = 20\%$ sp, $\sigma = 10$	$p = 20\%$ rv, $\sigma = 10$	$p = 30\%$	$p = 20\%$ sp, $\sigma = 10$	$p = 20\%$ rv, $\sigma = 10$	$p = 30\%$
Noisy image	12.36	16.03	13.87	12.25	15.76	13.27
	15.54	10.12	15.86	16.70	11.56	18.59
$3 \times 3$ median filter	27.98	29.38	27.94	26.91	27.70	25.86
	3.021	3.081	1.746	3.505	3.388	2.020
$5 \times 5$ median filter	29.39	29.22	28.92	26.87	26.78	26.23
	2.616	2.867	1.998	3.554	3.607	2.549
Bilateral filter	23.66	25.00	22.77	22.72	23.64	21.33
	7.302	6.107	4.205	6.959	5.754	4.587
SDROM	30.84	30.77	33.13	29.23	29.34	29.91
	2.256	2.894	0.980	2.847	3.001	1.401
ROAD-TRIF	31.34	31.53	32.74	29.49	29.62	29.94
	2.221	2.301	1.201	2.823	2.938	1.336
ROLD-EPR	30.95	31.53	32.89	28.91	29.64	30.12
	2.233	2.300	1.140	2.898	2.911	1.201
RORD-WMF	31.63	31.54	33.53	29.75	29.70	30.31
	2.181	2.298	0.917	2.716	2.908	1.129
DWM	30.92	30.79	32.18	29.03	28.89	29.10
	2.251	2.836	1.268	3.317	3.247	1.354
SBF	31.57	30.97	32.81	29.64	29.13	29.97
	2.208	2.599	1.155	2.797	3.120	1.313
SG-DARD-TRIF	31.93	31.56	33.06	30.24	29.79	30.02
	2.177	2.291	1.004	2.645	2.898	1.293

$p = 30\%$ . The proposed SG-DARD-TRIF consistently yields the best PSNR and MAE values for each image corrupted with the first two mixed noises. Note that it outperforms the other methods a large margin in images with salt-and-

pepper and Gaussian noise due to its high salt-and-pepper noise detection rate. For mixed impulse noise, the RORD-WMF shows better results in each image and the SG-DARD-TRIF is very close to it.

TABLE 6: Comparative restoration results in PSNR (dB) and MAE (the second row) for mixed noise ( $p = 20\%sp, \sigma = 10$ : salt-and-pepper and Gaussian;  $p = 20\%rv, \sigma = 10$ : random-valued impulse and Gaussian;  $p = 30\%$ : salt-and-pepper and random-valued impulse noise).

Method	Baboon image			Barbara image		
	$p = 20\%sp, \sigma = 10$	$p = 20\%rv, \sigma = 10$	$p = 30\%$	$p = 20\%sp, \sigma = 10$	$p = 20\%rv, \sigma = 10$	$p = 30\%$
Noisy image	11.98	15.25	13.98	12.30	15.92	14.17
	21.04	15.32	16.39	14.35	9.000	13.75
$3 \times 3$ median filter	20.43	21.01	21.76	23.06	23.62	23.17
	8.754	8.396	5.215	5.197	5.433	3.881
$5 \times 5$ median filter	20.59	20.66	20.90	22.81	22.86	22.79
	8.369	8.419	6.368	5.275	5.674	4.422
Bilateral filter	19.81	21.00	19.80	21.70	22.65	21.43
	9.612	8.404	5.958	8.006	7.822	6.120
SDROM	21.23	22.19	24.07	25.29	25.58	25.71
	7.860	7.403	4.256	4.143	4.420	3.329
ROAD-TRIF	21.71	22.32	24.13	25.66	25.94	25.66
	7.403	7.314	4.221	3.800	4.205	3.410
ROLD-EPR	21.38	22.26	24.03	25.48	26.02	25.88
	7.614	7.329	4.261	3.961	4.229	3.230
RORD-WMF	21.87	22.32	24.30	25.79	26.11	25.94
	7.363	7.300	4.062	3.787	4.268	2.978
DWM	21.34	22.15	23.89	25.44	25.65	25.30
	7.681	7.411	4.402	4.005	4.404	3.664
SBF	21.79	22.20	24.06	25.72	25.87	25.75
	7.404	7.387	4.257	3.802	4.339	3.323
SG-DARD-TRIF	22.41	22.37	24.15	26.71	26.15	26.03
	6.983	7.293	4.079	3.625	4.123	2.619

## 5. Conclusion

Many noise removal algorithms, such as the ROAD-TRIF and ROLD-EPR, tend to neglect the image edge information and, hence, end with unsatisfactory results. RORD-WMF introduces a reference image which is obtained by standard median filter to solve this problem; however, the edge direction information is still not considered. Although the DWM filter introduces the edge direction information into the median filter, the performance hardly depends on the accuracy of edge direction calculation. Furthermore, most of the existing filters do not have the ability of removing both impulse noise and Gaussian noise or cannot perform well. Therefore, a new trilateral filter based on DARD statistic and spatial gradient is proposed to handle these problems. The DARD statistic represents how impulse-like a particular pixel is in the sense that the larger the impulse, the greater the DARD value, while the spatial gradient is for Gaussian component weight. We incorporate the DARD statistic and spatial gradient into the Gaussian filtering framework by adding two components to the weighting functions. The weighting functions of the new trilateral filter contain spatial, gradient, and impulsive component. The gradient component combined with the spatial component smoothes away the Gaussian noise, while the impulsive component removes larger impulse noise. To demonstrate the superior performance of the proposed method, extensive experiments have

been conducted on several standard test images to compare our method with many other well-known techniques. Experimental results indicate that the proposed method performs better in removing Gaussian and mixed noise as well as in removing impulse noise than many other existing techniques.

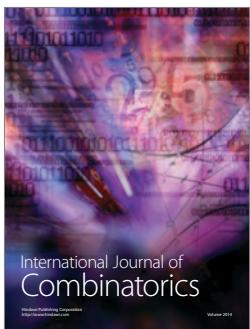
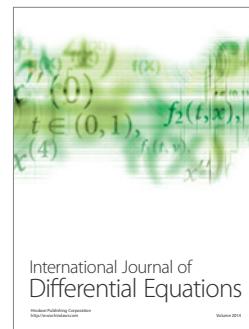
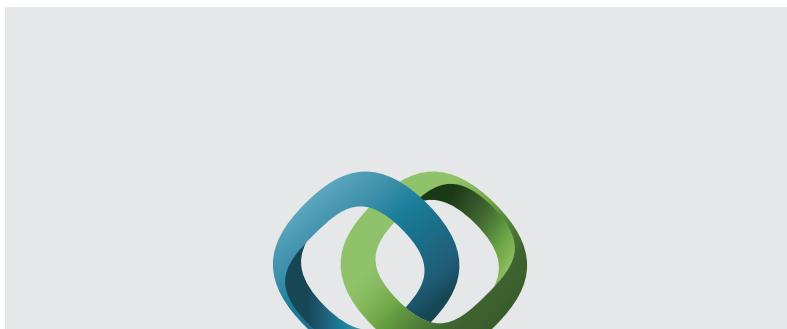
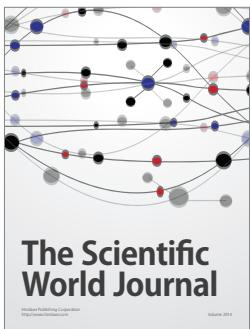
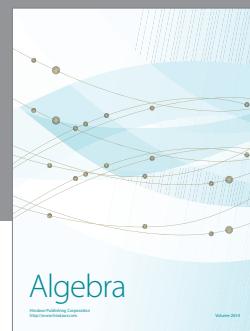
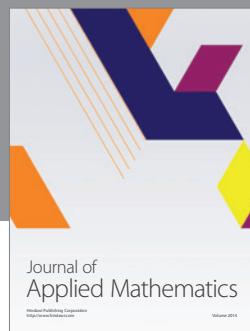
## Acknowledgments

The authors thank the anonymous reviewers for the help in improving this paper. This work is partly supported by the National Science Foundation of China (90820017), National Key Technology R&D Program (2011BAK07B03), and National Science and Technology Major Project (2009ZX07528-003-09).

## References

- [1] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2002.
- [2] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1747–1754, 2005.
- [3] W. K. Pratt, *Digital Image Processing*, Wiley, New York, NY, USA, 1978.
- [4] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 1998 IEEE 6th International Conference on Computer Vision*, pp. 839–846, January 1998.

- [5] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1141–1151, 2002.
- [6] C. H. Lin, J. S. Tsai, and C. T. Chiu, "Switching bilateral filter with a texture/noise detector for universal noise removal," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2307–2320, 2010.
- [7] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," in *Proceedings of the 9th European Conference on Computer Vision*, vol. 3954 of *Lecture Notes in Computer Science*, pp. 568–580, May 2006.
- [8] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Kluwer, Norwell, Mass, USA, 1990.
- [9] W. K. Pratt, "Median filtering," Tech. Rep., Signal and Image Processing Institute, University of Southern California, Los Angeles, Calif, USA, 1975.
- [10] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing," *Pattern Recognition Letters*, vol. 15, no. 4, pp. 341–347, 1994.
- [11] Z. Wang and D. Zhang, "Progressive switching median filter for the removal of impulse noise from highly corrupted images," *IEEE Transactions on Circuits and Systems II*, vol. 46, no. 1, pp. 78–80, 1999.
- [12] T. Chen and H. R. Wu, "Space variant median filters for the restoration of impulse noise corrupted images," *IEEE Transactions on Circuits and Systems II*, vol. 48, no. 8, pp. 784–789, 2001.
- [13] C. T. Lu and T. C. Chou, "Denoising of salt-and-pepper noise corrupted image using modified directional-weighted-median filter," *Pattern Recognition Letters*, vol. 33, no. 10, pp. 1287–1295, 2012.
- [14] G. Pok, J. C. Liu, and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Transactions on Image Processing*, vol. 12, no. 1, pp. 85–92, 2003.
- [15] A. Nasimudeen, M. S. Nair, and R. Tatavarti, "Directional switching median filter using boundary discriminative noise detection by elimination," *Signal, Image and Video Processing*, vol. 6, no. 4, pp. 613–624, 2012.
- [16] K. S. Srinivasan and D. Ebenezer, "A new fast and efficient decision-based algorithm for removal of high-density impulse noises," *IEEE Signal Processing Letters*, vol. 14, no. 3, pp. 189–192, 2007.
- [17] Y. Dong and S. Xu, "A new directional weighted median filter for removal of random-valued impulse noise," *IEEE Signal Processing Letters*, vol. 14, no. 3, pp. 193–196, 2007.
- [18] H. Xu, G. Zhu, H. Peng, and D. Wang, "Adaptive fuzzy switching filter for images corrupted by impulse noise," *Pattern Recognition Letters*, vol. 25, no. 15, pp. 1657–1663, 2004.
- [19] S. Schulte, V. De Witte, M. Nachtegael, D. Van der Weken, and E. E. Kerre, "Fuzzy random impulse noise reduction method," *Fuzzy Sets and Systems*, vol. 158, no. 3, pp. 270–283, 2007.
- [20] C. C. Kang and W. J. Wang, "Fuzzy reasoning-based directional median filter design," *Signal Processing*, vol. 89, no. 3, pp. 344–351, 2009.
- [21] Y. Dong, R. H. Chan, and S. Xu, "A detection statistic for random-valued impulse noise," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1112–1120, 2007.
- [22] H. Yu, L. Zhao, and H. Wang, "An efficient procedure for removing random-valued impulse noise in images," *IEEE Signal Processing Letters*, vol. 15, pp. 922–925, 2008.
- [23] 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 Transactions on Image Processing*, vol. 5, no. 6, pp. 1012–1025, 1996.
- [24] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng, "Restoration of images corrupted by mixed Gaussian-impulse noise via  $\ell_1\ell_0$  minimization," *Pattern Recognition*, vol. 44, no. 8, pp. 1708–1720, 2011.
- [25] Y. Zhou, Z. Ye, and Y. Xiao, "A restoration algorithm for images contaminated by mixed Gaussian plus random-valued impulse noise," *Journal of Visual Communication and Image Representation*, vol. 24, pp. 283–294, 2013.
- [26] D. Zhang and Z. Wang, "Impulse noise removal using polynomial approximation," *Optical Engineering*, vol. 37, no. 4, pp. 1275–1282, 1998.
- [27] S. Zhang and M. A. Karim, "A new impulse detector for switching median filters," *IEEE Signal Processing Letters*, vol. 9, no. 11, pp. 360–363, 2002.
- [28] B. Zhang and J. P. Allebach, "Adaptive bilateral filter for sharpness enhancement and noise removal," *IEEE Transactions on Image Processing*, vol. 17, no. 5, pp. 664–678, 2008.
- [29] T. Chen and H. R. Wu, "Adaptive impulse detection using center-weighted median filters," *IEEE Signal Processing Letters*, vol. 8, no. 1, pp. 1–3, 2001.



Submit your manuscripts at  
<http://www.hindawi.com>

