IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 1, 2020, accepted November 23, 2020, date of publication November 26, 2020,
date of current version December 23, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3040760

A Two-Stage Algorithm for the Detection and Removal of Random-Valued Impulse Noise Based on Local Similarity

CONG LIN[1,2], YUCHUN LI[1], SILING FENG[1], AND MENGXING HUANG[1], (Member, IEEE)
[1]College of Information and Communication Engineering, Hainan University, Haikou 570228, China
[2]College of Electronics and Information Engineering, Guangdong Ocean University, Zhanjiang 524000, China

Corresponding authors: Siling Feng (fengsiling2020@163.com) and Mengxing Huang (huangmx09@163.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1404400.

ABSTRACT A two-stage denoising algorithm based on local similarity is proposed to process lowly and moderate corrupted images with random-valued impulse noise in this paper. In the noise detection stage, the pixel to be detected is centered and the local similarity between the pixel and each pixel in its neighborhood is calculated, which can be used as the probability that the pixel is noise. By obtaining the local similarity of each pixel in the image and setting an appropriate threshold, the noise pixels and clean pixels in the damaged image can be detected. In the image restoration stage, an improved bilateral filter based on local similarity and geometric distance is designed. The pixel detected as noise in the first stage is filtered and the new intensity value is the weighted average of all pixel intensities in its neighborhood. A large number of experiments have been conducted on different test images and the results show that compared with the mainstream denoising algorithms, the proposed method can detect and filter out the random-value impulse noise in the image more effectively and faster, while better retaining the edges and other details of the image.

INDEX TERMS Image denoising, random-valued impulse noise, local similarity, bilateral filter.

I. INTRODUCTION

The digital images are often destroyed by impulse noise due to sensor equipment in the process of image acquisition and transmission. The random-valued impulse noise (RVIN) is one of the impulse noises whose noise pixel value is randomly located between 0 to 255. In order to perform operations such as contour extraction, region segmentation and target recognition on the image later, it is necessary to restore the noise image. The image removing algorithm is mainly divided into two stages. Firstly the noise detection is performed on the image and then the detected noise is restored, so that the edge information of the image can be better preserved and the output image is prevented from being blurred [1]. Many researchers have done a lot of research work on image noise detection and denoising algorithms [2]–[8]. The impulse noise detectors and filters based on local statistics have been proposed in the early years. Xiong [9] propose the robust outlyingness ratio (ROR) for measuring how impulse like each pixel is, and combined it with non-local mean (NLM) [10] to eliminate general noise. Garnett [11] proposed a sorted absolute difference (ROAD) statistic and a general triangular filter. Dong [5] proposed a ROAD-based rank logarithmic difference (ROLD). These filters perform well in removing RVIN and retaining edges and details, but their filtering effect is highly dependent on an accurate impulse noise detector.

In recently years, the mainstream denoising algorithms can be divided into methods based on block matching, convolutional neural networks and fuzzy rules [12]–[20]. Gao [21] introduced a two-stage denoising method based on the improvement of the HEIND algorithm which uses the shear wave representation to effectively restore the geometry of the original image. It is particularly effective in eliminating jagged edges and other visual artifacts in images at higher noise levels. Tukkmen [22] proposed a four-phase detection method based on the neighbor criterion of similar values to realize the detection of noisy pixels. The pixels are immediately filtered by the median value after detecting the damaged pixels in each stage. Since three different thresholds are used in each detection stage to obtain a more robust filter, the running time of this method is longer than most other methods. Signh [23] uses three levels of adaptive thresholds and an

The associate editor coordinating the review of this manuscript and approving it for publication was Gerardo Di Martino.

auxiliary condition for detecting noise and restoring images which can improve the miss detection rate and false detection rate of existing noise detection algorithms, but it has better detection and denoising effects only in noise images with high density noise levels. Nadeem [24] proposed an image restoration technique based on adjacent pixels in the spatial link direction and fuzzy logic to solve medium and highly damaged grayscale images with random value impulse noise. The method can adaptively determine and set the threshold so that the filter can automatically process different types of images. Azhar uses a switching method that uses local texture statistics in different directions of the sliding window to identify damaged pixels in an iterative manner. In the noise filtering stage, the fuzzy rules are used to obtain noise-free pixels from the suggested three-way pixels to estimate the intensity value of the identified damaged pixel [25]. But the complexity and running time of the algorithm are greatly increased due to the introduction of fuzzy algorithms and iterative methods. Chen [26] proposed a blind CNN model for RVIN denoising with a variable noise ratio predictor (NRP) as an indicator. This method has the ability to deal with unknown noise ratios under the guidance of NRP. Zou [27] proposed an image denoising block matching method based on convolutional neural network. The solution first needs to apply a denoising algorithm on the noisy image to obtain the pilot signal for training CNN. However, the convolutional neural network model is too complex which requires a long training time and high hardware equipment. This will prevent it from becoming a real-time application. Iqbal [28] proposed an adaptive noise detector and a new weighted average filter which uses an edge recognition stage to ensure that edge pixels are not mistakenly detected as noise pixels to further improve the detection accuracy, but the filter effect is not very good. Veerakumar [29] proposed a novel algorithm to identify and correct images affected by impulse noise in which empirical mode decomposition is used to identify pixels affected by impulse noise and an adaptive bilateral filter is used to restore those noisy pixels. The accuracy of the method may be reduced in a few images with random-valued impulse noise. In addition, this method may not produce better results if impulse noise is affected in smoother or blurred areas of the image. In [30] Pok proposed an effective block-based image denoising method. With the scheme, a block similar to a given block can be processed by considering only the block pointed to by the pointer corresponding to the pixel value of the block. Due to the reason that search without comparing all the blocks in the input image, it performs well in terms of filtering effect and running time but no detailed experimental data is given in the noise pixel detection stage.

Judging from the popular denoising algorithms in recent years, a good filtering effect has been achieved due to the introduction of fuzzy rules and convolutional neural networks, but it has also led to increased algorithm complexity, longer running time and high equipment costs. In order to solve the above problems and improve the performance of image noise detectors and filters, we propose a two-stage denoising algorithm to detect random value impulse noise of images and restore the damaged images. In the noise detection stage, a 5×5 detection window is constructed with any pixel x to be detected as the center, and the similarity between the pixel x and each pixel in its detection window is calculated and summed. Then the weighted average operation and normalization are performed to obtain the local similarity (LS) of a given center pixel, which can certainly indicate the probability of a given pixel being noise free. Through obtaining the local similarity value of each pixel in the image and setting an appropriate LS detection threshold, the noise pixels and clean pixels can be filtered out. In the image restoration stage, an improved bilateral filter based on local similarity and geometric distance is proposed, the pixel detected as noise in the first stage is filtered and its new intensity value is the weighted average of all pixel intensities in its neighborhood. Different from the traditional bilateral filter, the two weights used in proposed filter are related to the geometric distance of the pixel and the local similarity information. The parameters of the proposed filter are small, which greatly reduces the amount of calculation and running time, and the filtering effect is better than many existing filters.

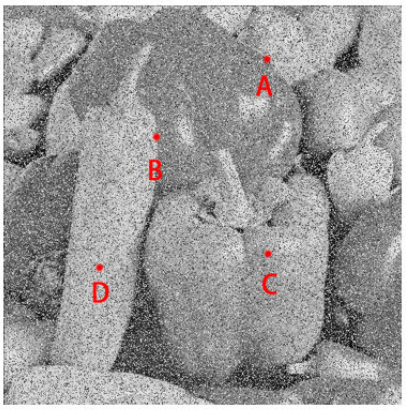
The remainder of this paper is organized as follows: the proposed impulse noise detection scheme is described in Section II, the proposed filter is designed in Section III, simulation results are compared in Section IV and conclusions are presented in Section V.

II. IMPULSE NOISE DETECTION

The intensity value of the damaged pixel is randomly between 0 and 255 when the image is damaged by random value impulse noise. Figure 1(a) is obtained by applying 40% random impulse noise to the original Peppers image, then do the gray-scale difference operation between it and the original image to know which pixels are noise in the corrupted image. Then the four regions A, B, C and D with a 5×5 patch size are selected from the Figure 1(a), and their pixel intensity are show in Figure 1(b). We use gray and white background colors to present the noise pixels and clean pixels respectively, and red for the center pixel itself. It can be seen that there should be a certain number of pixels with similar intensity in its area if a given center pixel is free of noise. For example, the center pixel of the flat region C has 18 similar pixels in its neighborhood. For an impulse pixel, usually few pixels with similar intensity can be found near it regardless of whether it is located in a complex region B or a flat region D. Therefore, we can measure whether it is noise or not by calculating the similarity between the center pixel and all pixels in its neighborhood.

A. DEFINITION OF LOCAL SIMILARITY

It can be seen from the bilateral filtering algorithm [31] and some other denoising algorithms [25], [32] that in the noise detection and removal stage, not only the influence of the difference in pixel values in the neighborhood of the center pixel,



(a)

109	12	210	109	181
156	23	105	192	212
250	162	139	119	4
131	144	216	150	136
223	104	116	143	160

Intensities of Region A

179	186	184	124	209
137	183	161	171	66
180	187	210	99	70
184	181	183	90	69
66	14	165	84	81

Intensities of Region B

140	134	139	212	141
247	207	139	144	153
113	146	142	151	143
142	146	147	144	121
140	140	144	233	141

Intensities of Region C

176	160	173	174	170
59	181	60	173	177
183	146	204	170	175
149	228	175	242	173
251	1	209	174	169

Intensities of Region D

(b)

FIGURE 1. **(a) Image of Pepper with 40% random-valued impulse noise, marked with four special regions whose intensities within are listed on the right (b), all use gray and white background color to indicate the impulse and clean pixels.**

but also the distance factor must be considered. If the size of detection window is 3×3, the influence of the distance factor on the detection of the center pixel is ignored. For images with fine details, a processing window of 3×3 may not be able to distinguish between noise and details [33]. If the window is too large, it will greatly increase the amount of calculation. Based on the above analysis, a 5×5 neighborhood Ω_x^0 can be constructed to be a window size with any given pixel x as the center and then the similarity between the pixel x and any pixel y in Ω_x^0 can be expressed as [34]:

$$D(x, y) = \exp\left(-\frac{\|\overline{(m, n)} - \overline{(s, t)}\|^2}{2\sigma_D^2}\right), \quad y \in \Omega_x^o \tag{1}$$

$$I(x, y) = \exp\left(-\frac{\|G_x - G_y\|^2}{2\sigma_I^2}\right) \tag{2}$$

$$S(x, y) = D(x, y) \cdot I(x, y) \tag{3}$$

where $S(x, y)$ represents the similarity between the center pixel x and y. (s, t) represents the coordinate of the center pixel x and (m, n) represents the coordinate of any pixel y in the x neighborhood Ω_x^0, G_x and G_y represent the gray value of the pixel x and pixel y respectively. $D(x, y)$ and $I(x, y)$ are Gaussian functions of the geometric distance and intensity difference between pixels x and y respectively. Obviously, $D(x, y)$ and $I(x, y)$ decrease when the distance and the gray level difference between the two pixels get bigger, which also means that if the gray difference between two pixels is large or the distance is far, then the similarity between x and y is very small and even the Euclidean distance can be omitted. The parameters σ_D and σ_I are the standard deviation of the Gaussian function $D(x, y)$ and $I(x, y)$, respectively. They control the sensitivity of the geometric distance and absolute intensity difference of $D(x, y)$ and $I(x, y)$ respectively. Their influence for $D(x, y)$ and $I(x, y)$ can be change by adjusting the values of these two parameters generally. Through a lot of experiments, it is found that the denoising algorithm has achieved good performance in all aspects when $\sigma_D = 10$ and $\sigma_I = 6.7$.

Then the sum of the similarity between pixel x and all pixels in the neighborhood can be obtained by the following formula:

$$\zeta x = \sum_{y \in \Omega_x^0} S(x, y) = \sum_{y \in \Omega_x^0} [D(x, y) \cdot I(x, y)] \tag{4}$$

Since the edges and complex areas of the image refer to the parts where the brightness of the local area of the image changes significantly, the gray profile of this area can generally be regarded as a step, which means that a pixel changes sharply in a small buffer area to another pixel with a large grayscale difference. Therefore, when the central pixel x is in the edge area, the gray scale difference between the pixel x and the pixels in its neighborhood is generally larger. Hence, $I(x, y)$ is smaller and ζx decreases accordingly according to Formulas (2)-(3). Similarly, because the intensity of the pixel in the flat area changes slowly, the grayscale difference between x and any pixel in its neighborhood is relatively small when the central pixel x is in the flat area. Hence, $I(x, y)$ is larger and ζx increases accordingly, so the ζx of the pixels in the flat area is larger than that of the pixels in the edge and texture complex area since the intensity of pixels in the flat area changes more slowly. In order to improve the robustness of ζx, an averaging operation can be performed on the ζx of the pixel x so that the ζx of the noise pixel or the clean pixel almost reaches the same value, and the influence of the complexity of the region on ζx is reduced. The new statistics are defined as follows:

$$\overline{\zeta x} = \zeta x / \overline{\left(\sum_{y \in \Omega_x^0} \zeta y\right)} \tag{5}$$

where $\overline{\left(\sum_{y \in \Omega_x^0} \zeta y\right)}$ denotes the averaging operation for $\sum_{y \in \Omega_x^0} \zeta y$.

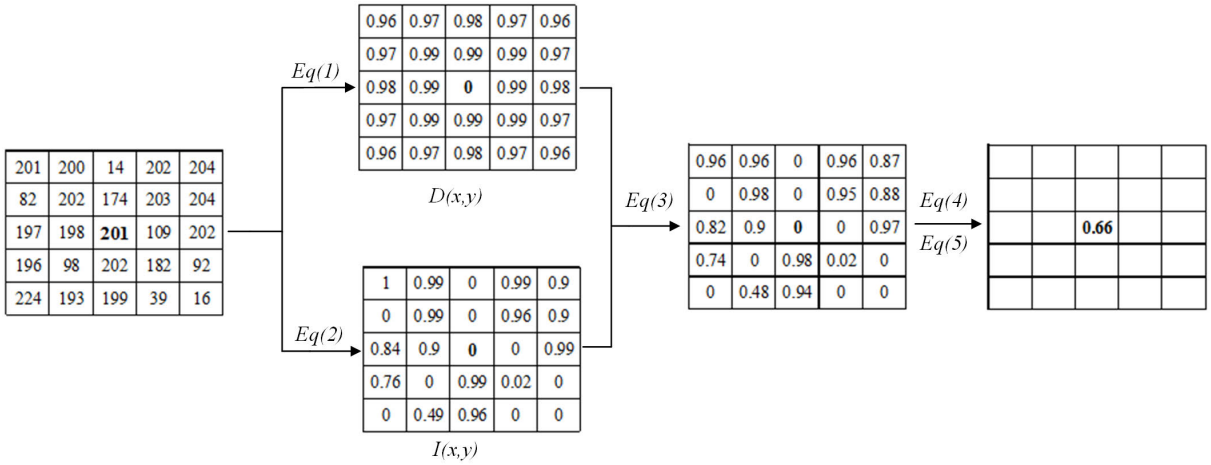


FIGURE 2. Flowchart of the computation process of LS_x for a given center pixel x.

Through observation, it can find that $\overline{\zeta x}$ of each pixel in the noise image is basically scattered in [0, 2.5]. In order to process the data more conveniently and quickly, the following formula can be used to normalize the $\overline{\zeta x}$ of any pixel to be [0, 1] interval:

$$LS_x = \begin{cases} 1, & \overline{\zeta x} > 2.5 \\ \overline{\zeta x}/2.5, & \overline{\zeta x} \le 2.5 \end{cases} \tag{6}$$

where LS_x represents the local similarity between a given central pixel x and the pixels in its neighborhood, which can indicate the probability of whether the pixel x is noise. If the LS_x value is smaller, the similarity between the pixel x and the pixels in its neighborhood is smaller. In other words the pixel x is more likely to be noise. The process of calculating the LS_x value of a given pixel x is shown in Figure 2 for better understanding.

Obviously, the LS_x threshold can be set to filter out noise pixels and clean pixels in the image. Properly setting the threshold of LS_x helps to improve the accuracy of noise detection. As we can know in Section II-A, the ζ x of the noise pixel is smaller than the ζ x of the clean pixel, and the neighborhood of the center pixel contains more noise pixels instead of clean pixels when the noise level of the image increases. Based on the above two premises and Formula (5), it can be known that when calculating the $\overline{\zeta x}$ of the center pixel x, the value will increase as the image noise level increases, which will result in a larger LS_x value of the center pixel. In addition, since the ζ x of the pixels in the flat area is larger than that of the pixels in the edge and texture complex area, the LS_x value of a pixel in a flat area will be larger than that of a pixel in a complex area according to Formula (3). Based on the above analysis, it can be known that the LS detection threshold is related to the noise level of the image, and different LS thresholds should be used for pixels in different regions to determine whether it is noise. Hence, we must first estimate the noise level of the image and determine what region the center pixel is locate before setting the LS detection threshold. The two problems are discussed in the next sections.

B. PIXEL REGION DETECTION

For a noise-free image, the variance is small because the pixel intensity in the flat area changes gently. In other words, the pixel x can be considered to be in a flat area if the intensity variance of all pixels in its neighborhood is small [35]. In order to estimate the variance of a local area more accurately in a noisy image, the intensity of the clean pixels in the area should be used as much as possible to calculate the variance instead of the intensity of the noise pixel. Since LS can indicate the probability of whether a pixel is noise, the variance of the region can be estimated by performing a weighted average operation on all pixels in the pixel neighborhood where the weight of each pixel is related to its corresponding LS value. When the variance of the area where the pixel x located is not greater than the preset threshold, the pixel is regarded as being in a flat area, otherwise it is regarded as being in a complex area. The method can be expressed as:

$$a = \sum_{y \in \Omega_x^0} \left(LS_y\right)^{W_1} \tag{7}$$

$$\mu_x = \sum_{y \in \Omega_x^0} \frac{\left(LS_y\right)^{W_1} \cdot G_y}{a} \tag{8}$$

$$b = \sum_{y \in \Omega_x^0} \left(LS_y\right)^{W_2} \tag{9}$$

$$\sigma_x^2 = \sum_{y \in \Omega_x^0} \frac{\left(LS_y\right)^{W_2} \cdot \left(G_y - \mu_x\right)^2}{b} \tag{10}$$

$$x \in \begin{cases} \text{flat region}, & \sigma_x^2 \le T_\sigma \\ \text{complex region}, & \sigma_x^2 > T_\sigma \end{cases} \tag{11}$$

where σ_x^2 represent the estimated variance deviation of the intensity of all pixels in Ω_x^0. LS_y is the similarity of any pixel y in the neighborhood of the center pixel x, which can be obtained by Formulas (1)-(6). W_1 and W_2 are the weights of LS_y, which are used to adjust the proportion of the influence

of clean pixels and noise pixels on the calculation of local variance. μ_x represents the estimated mean value of all pixels in the neighborhood Ω_x^0 of the center pixel x, G_y is the gray value of any pixel y in Ω_x^0. a and b are the sum of the weights used as a normalizer. T_σ is a preset threshold to distinguish whether the pixel is in a complex area or a flat area. W_1, W_2 and T_σ are adjustable parameters. Through a lot of experiments it is found that the detection accuracy of the area where the pixel located has achieved good performance when $W_1 = 2$, $W_2 = 4$ and $T_\sigma = 0.12$.

C. NOISE LEVEL ESTIMATION

As mentioned earlier, the intensity value of the pixel in the flat area changes smoothly and the larger the LS of a pixel, the more likely it is to be a clean pixel. Based on these two premises, the given pixel x and the pixel y with the largest LS value in Ω_x^0 can be subjected to the intensity difference calculation. The pixel x can be seen as a clean pixel if the difference is not greater than the preset threshold, otherwise it is a noise pixel. The method can be expressed as:

$$x \text{ is } = \begin{cases} \text{an impulse noise}, & I_x - I_y > \theta \\ \text{a clean pixel}, & I_x - I_y \le \theta \end{cases} \tag{12}$$

where I_y is the gray of the pixel with the largest LS value in Ω_x^0. θ is an empirical threshold and we set θ to 5 in this paper. It should be noted that here we only use this formula for noise pixels in flat regions because they are easier to detect [36].

Then select some flat areas with a patch size M in the image, filter out the noise pixels and clean pixels in each area by formula (12) and then use formula (13) to estimate the noise level of each area. Finally the overall noise level of the image can be estimated by performing a average operation on the noise level of these areas [37] as shown in Formula (14):

$$\sigma_i = \frac{Q_n}{Q_c + Q_n}, \quad i = 1, 2, 3, \ldots, d \tag{13}$$

$$\sigma = \frac{1}{d}\sum_{i=1}^{d} \sigma_i \tag{14}$$

where Q_n and Q_c are the quantity of noise pixels and clean pixels given by (12). The parameters d represents the number of selected flat regions and $d = 10$, $M = 11$ in this paper.

D. SELECTION OF LS THRESHOLD

In order to obtain the optimal LS detection threshold, a large number of parameter tuning experiments have been carried out on some common test images and it is found that even if the test images are different but the noise level is similar, the optimal threshold of LS is close. As the noise level increases the optimal threshold becomes larger. In addition, the LS threshold of the pixel in the complex area will be lower than that of the flat area because the pixel intensity of the complex area changes significantly. The above experiment results is in line with our previous theoretical analysis in the last paragraph of the second Chapter II-A. In order to correctly estimate the LS detection threshold of pixels in the flat area and the detailed area under different noise levels, we performed polynomial fitting on a large amount of data obtained from experiments to obtain the calculation formula of the LS threshold with respect to the image noise level, as shown in formulas (15) and (16):

$$\theta_f = -0.12\sigma^3 + 0.07\sigma^2 + 0.75\sigma + 0.19 \tag{15}$$

$$\theta_c = 0.31\sigma^3 + 0.63\sigma^2 + 0.52\sigma + 0.03 \tag{16}$$

where θ_f and θ_c are the LS detection thresholds of pixels in the flat area and the complex area, respectively. σ is the noise level of the corrupted image, which was estimated from Formula (12)-(14). Then the impulse noise detector based on LS can be designed as follows:

(1) If the pixel x is in a complex area, then the formula (17) can be used to detect whether it is noise.

$$x \in \begin{cases} \text{impulse noise}, & LS_x \le \theta_c \\ \text{clean pixel}, & LS_x > \theta_c \end{cases} \tag{17}$$

(2) If the pixel x is in a flat area, the formula (18) can be used.

$$x \in \begin{cases} \text{impulse noise}, & LS_x \le \theta_f \\ \text{clean pixel}, & LS_x > \theta_f \end{cases} \tag{18}$$

E. IMAGE PREPROCESSING

When a clean pixel is on the edge or contour of the image, the intensity difference between it and nearby pixels is more obvious, which can easily lead to the edge and contour pixels as noise in the noise detection process [28]. In order to make the noise detection scheme more accurate and robust, a limited condition is added to the LS threshold detection method to avoid false detection of edge pixels as noise pixels. This step is performed only when the pixel x is identified as a noisy pixel by the LS detection method. Firstly the original noise image is preprocessed with median filter and Gaussian filter, then the processed image and the original noise image are subjected to intensity difference calculation. The pixel is considered as a noise pixel when the absolute difference between the pixels of the two images at the same coordinate is greater than the threshold. The method is shown in formula (19):

$$x \in \begin{cases} \text{clean pixel}, & \left|I_x - I'_x\right| \le \theta_p \\ \text{impulse noise}, & \left|I_x - I'_x\right| > \theta_p \end{cases} \tag{19}$$

where I_x is the intensity value of pixel x in the original noise image, I'_x is the intensity value of the corresponding pixel x after preprocessing. θ_p is the threshold and $\theta_p = 15$ in this paper. In summary the whole framework of the impulse noise detection algorithm proposed in this paper is shown in Figure 3. The main steps are summarized as follows:

Step 1: Calculate the LS value of each pixel in the noise image according to formulas (1)-(6).

Step 2: Judge whether each pixel is in a flat area or a complex area through formulas (7) - (11).

Step 3: Estimate the overall noise level of the original noisy image through (12)-(14) and then obtain the best LS detection

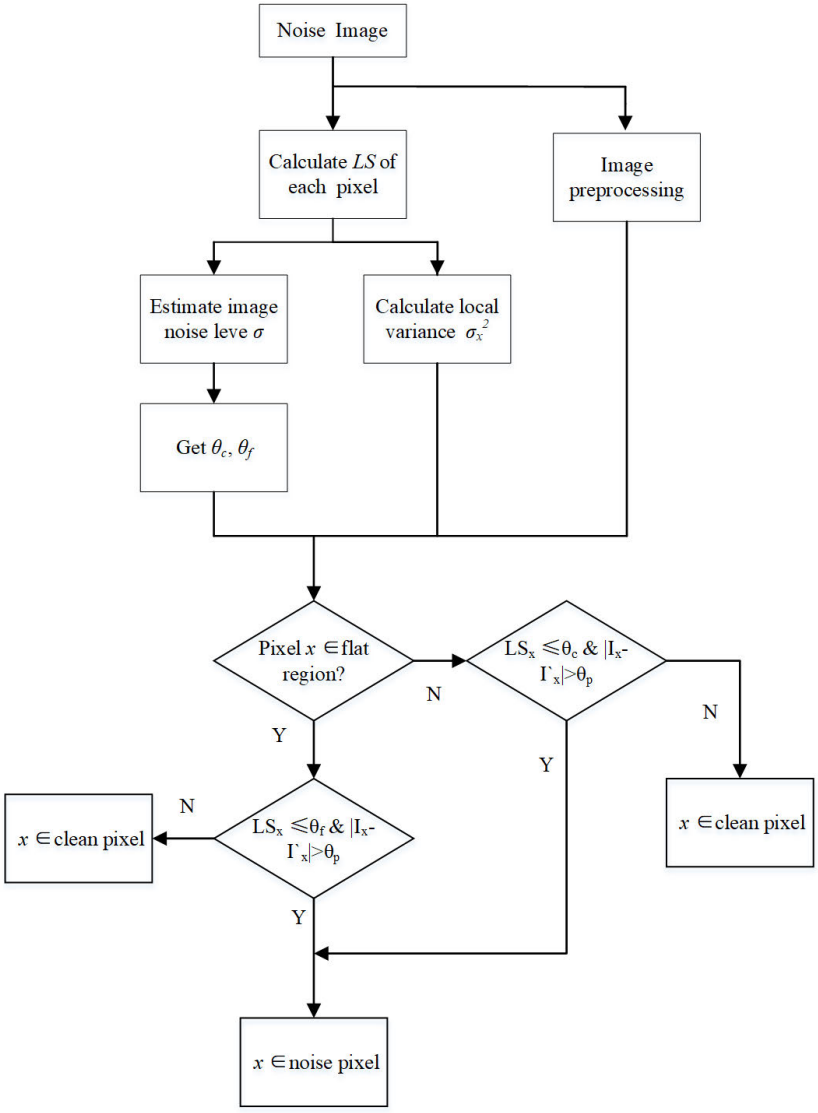


FIGURE 3. Flowchart of the computation process of LS_x for a given center pixel x.

thresholds for pixels in the flat area and the complex area through formulas (15) and (16).

Step 4: When the pixel x of the original noise image meets the conditions of the noise pixel in formulas (17)-(19), it is marked as a noise pixel, otherwise it is a clean pixel.

III. THE IMPULSE NOISE FILTERING METHOD

A good filter is needed to replace the noisy pixels in the second stage after the noise detection. The bilateral filtering proposed by Tomasi is a non-linear filtering method that combines the spatial proximity of the image and the similarity of the pixel value, and considers the spatial information and gray-scale similarity to achieve the purpose of edge preservation and denoising. The basic idea is that the filtered gray value of the noise pixel is determined by the pixel in its neighborhood, and the weight of the pixel in the neighborhood to this value depends on the distance and gray difference between the two pixels [31]. The filter method can be described as follows:

$$D(x,y) = \exp\left(-\frac{\|\overline{(m,n)} - \overline{(s,t)}\|^2}{2\sigma_D^2}\right), \quad y \in \Omega_x^\circ \tag{20}$$

$$I(x,y) = \exp\left(-\frac{\|G_x - G_y\|^2}{2\sigma_I^2}\right) \tag{21}$$

$$G_{xf} = \frac{\sum_{y\in\Omega_x^\circ}\left[D(x,y)\cdot I(x,y)\cdot G_y\right]}{\sum_{y\in\Omega_x^o}[D(x,y)\cdot I(x,y)]} \tag{22}$$

where G_{xf} is the intensity value of the original noise pixel x after filtering, and other parameters are similar to the Formula (1)-(3). Zhang [32] improved the bilateral filter to eliminate impulse noise but the proposed ABF tends not to be good at processing images corrupted by strong block wise noises. This is because ABF is a pixel-based filtering scheme and is not qualified for addressing noises exhibiting regional characteristics.

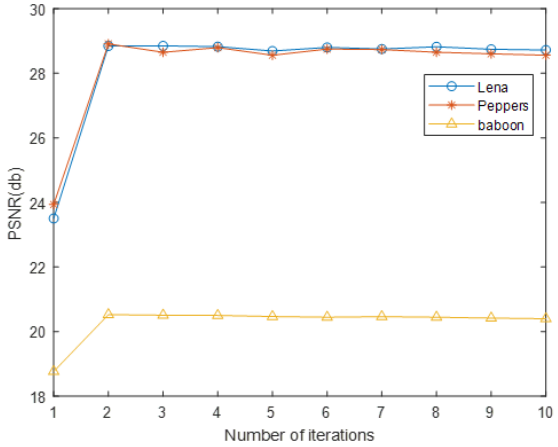


FIGURE 4. The PSNR corresponding to different iteration times of the images.

It can be known from the principle of the bilateral filter that the noise pixels and clean pixels in the filter window have the same influence on the center pixel during filtering, which is obviously unreasonable. Because the gray value of the noise pixel has been damaged to a certain extent, it does not have too high reference value. Therefore, more consideration should be given to the impact of the gray value of the clean pixel on the center pixel when filtering the center pixel. As mentioned in Chapter II-A, the pixel is more likely to be a clean pixel if its *LS* value is larger. Hence, the new filter method based on local statistical (LSBF) is designed by replacing the above Formula (21) with the function of *LS*, as shown in follows:

$$D(x,y) = \exp\left(-\frac{\|\overline{(m,n)} - \overline{(s,t)}\|^2}{2\sigma_D^2}\right), \quad y \in \Omega_x^\circ \tag{23}$$

$$L(x,y) = LS_y^2 \tag{24}$$

$$G_{xf} = \frac{\sum_{y\in\Omega_x^\circ}\left[D(x,y)\cdot L(x,y)\cdot G_y\right]}{\sum_{y\in\Omega_x^\circ}[D(x,y)\cdot L(x,y)]} \tag{25}$$

where LS_y is the local similarity of any pixel y in the neighborhood Ω_x^0, which can be obtained by formulas (1)-(6). Fig.4 shows the three images with a noise level of 60% filtered ten times and their corresponding PSNR values. It can be seen that the proposed filter can obtain good results after the first filtering and the PSNR reaches the highest after the second or third iteration, but the subsequent iterations did not achieve better results because of the reason that most of the noise pixels in the image have been detected and filtered out in the previous iterations. In addition, too much iteration can easily misjudge some normal pixels as noise pixels in the noise detection stage which will cause the PSNR value to decline. Based on above reasons the LSBF filtering steps proposed in this article are as follows:

Step 1: Use formula (25) to recover each noise pixel detected in the first stage.

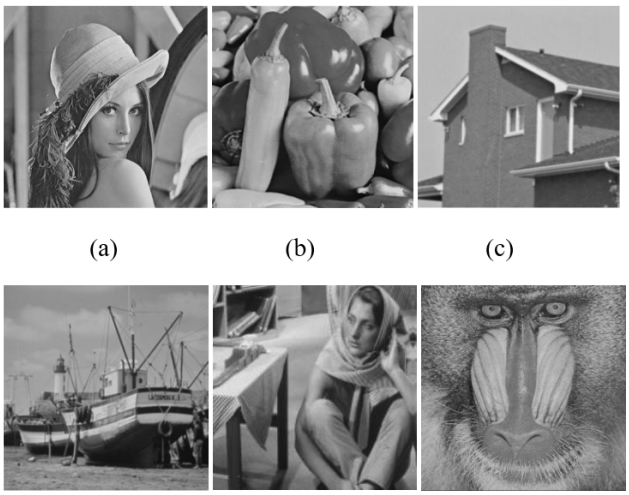


FIGURE 5. Test images: (a) Lena (b) Peppers (c) House (d) Boat (e) Barbara (f) Baboon.

Step 2: Find the *LS* value of each pixel again on the restored image and then detect noise and filter.

Step 3: Repeat step 2 and then stop the iteration when the PSNR value of the restored image starts to decrease.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

In this chapter, a large amount of experimental data is analyzed to illustrate the performance of the proposed impulse noise detection and noise reduction algorithm, and compare it with several latest algorithms. And the size of the test image used in this article is 512 × 512 as shown in Figure 5.

A. PERFORMANCE OF PROPOSED NOISE DETECTOR

Since the intensity value of random impulse noise is between 0 to 255, it is easy to consider the clean pixels as noise or treat noise pixels as clean in the process of detecting noise when the pixel intensity value of the damaged image is not much different from the original value. A good noise detector should have the characteristics of low false detection (FD) and missed detection (MD) rates. Through using the proposed noise detector and several mainstream noise detection algorithms to detect *Lena* images with 40% to 60% RVIN, the results are shown in Table 1. It can be seen that ACWM has obvious advantages in low false detections, but the number of missed detections is very high and DWM has a similar opposite situation. Although the proposed method has not reached the lowest number of missed detections or false detections, it has achieved good results in both aspects and the total number of false detections has reached the lowest under different noise levels. In order to further reflect the performance of the noise detector proposed in this article, the RVIN of 20% to 60% was applied to the test images and the tested results are shown in Table 2. It can be seen that the proposed noise detector shows a very low number of missed detections and false detections in the first three

TABLE 1. Different methods' comparison of detection results for Lena image with 40% to 60% RVIN.

Methods	40%			50%			60%		
	MD	FD	Total	MD	FD	Total	MD	FD	Total
DWM [2]	**9512**	7761	17273	9514	11373	20887	**12676**	12351	25027
ACWM [3]	16052	**1759**	17811	23683	**3623**	27306	32712	7644	40356
NWM [8]	10149	5212	15361	**9116**	11299	20415	15448	7449	22897
AEPWM [28]	10908	7973	18881	11668	9613	21281	13571	9760	23331
ROR-NLM [9]	12890	3328	16218	15297	3487	19084	21827	7808	29635
ROLD-EPR [5]	14373	7158	21531	16682	7619	24301	19245	8235	27480
SDOOD [38]	13269	10324	25393	11742	15574	27316	16989	5923	22912
SRM [39]	21063	2063	23126	24903	3195	28098	32722	**5047**	37769
Proposed	11039	4213	**15252**	13562	5285	**18847**	15071	7521	**22592**

TABLE 2. Detection results of the proposed method for different images with 20% to 60% RVIN.

Images	20%			30%			40%			50%			60%		
	MD	FD	Total	MD	FD	Total	MD	FD	Total	MD	FD	Total	MD	FD	Total
Lena	8167	1214	9381	11704	1738	13442	11039	4213	15252	13562	5285	18847	15071	7521	22592
Peppers	7678	1701	9379	11244	2096	13340	11891	4313	16204	12538	7030	19568	15581	7560	23141
House	6299	52	6351	9304	172	9476	10004	653	10657	12202	1035	13237	12882	2476	15358
Boat	9796	3494	13290	14060	4509	18569	14482	10236	24718	17524	10587	28111	19013	14523	33536
Barbara	10783	2619	13402	15442	3528	18970	16029	9617	25646	19134	10395	29529	20645	14007	34652
Baboon	16465	9667	26132	24192	11506	35698	23592	23816	47408	28371	23356	51727	29396	28328	57724

images with less texture, especially in the *House* image. The number of missed detections and false detections is relatively high since the textures of the last three images are more rich and complex, especially in the *Baboon* image. However, it also can be seen that the noise detector shows a very stable detection effect and good performance with the increase of the image noise level.

B. PERFORMANCE OF PROPOSED FILTER

The proposed filter is used on the test images after the first stage of noise detection. In order to evaluate the filtering effect of the proposed filter, Peak Signal-to-Noise-Ratio (PSNR) [40] and Structure Similarity Index Measure (SSIM) [41] are used as evaluation indicators, which definitions are shown in formulas (26) and (27) respectively.

$$PSNR = 10\log_{10}\frac{255^2}{\frac{1}{M_1M_2}\sum_{i=1}^{M_1}\sum_{j=1}^{M_2}\left(o_{i,j}-r_{i,j}\right)^2} \tag{26}$$

$$SSIM = \frac{\left(2\mu_x\mu_y + C_1\right)\left(2\delta_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\delta_x^2 + \delta_y^2 + C_2\right)} \tag{27}$$

where $M_1 \times M_2$ is the dimension of the image, $o_{i,j}$ and $r_{i,j}$ are the corresponding intensities of pixels in the clean and restored images. μ_x and μ_y are the means of the noisy and restored image respectively. δ_x^2 and δ_y^2 are the variances of the noise and restored image respectively and δ_{xy} is the covariance of the noise and restored image.

Table 3 and Table 4 list the PSNR and SSIM values of *Lena* and *Boat* images contaminated by RVIN with a noise density ranging from 40% to 60%. It can be seen from Table 3 and Table 4 that the proposed filter shows good performance whether it is in *Lena* image with a simple texture or in *Boat* image with a more complex texture. Although DWM, ACWM and AEPWM perform better at lower noise density, the performance decays quickly at higher noise density because these methods mainly replace noise pixels by considering all pixels in the window. The proposed filter is more inclined to replace the intensity value of the central noise pixel with clean pixels through LS weighted average. The performance of the proposed filter decays slowly even with the increase of the noise density level, which indicates that the filter has better stability and robustness.

In order to evaluate the effect of the proposed filter intuitively, the *Lena* and *Baboon* images with 50%-60% RVIN are filtered and the results are shown in Figure 6. It can be seen that the proposed filter can restore the edges and detail regions of the image very well even in highly damaged Lena image. As for the Baboon image with complex texture, there are still some noises in the filtered image due to the number

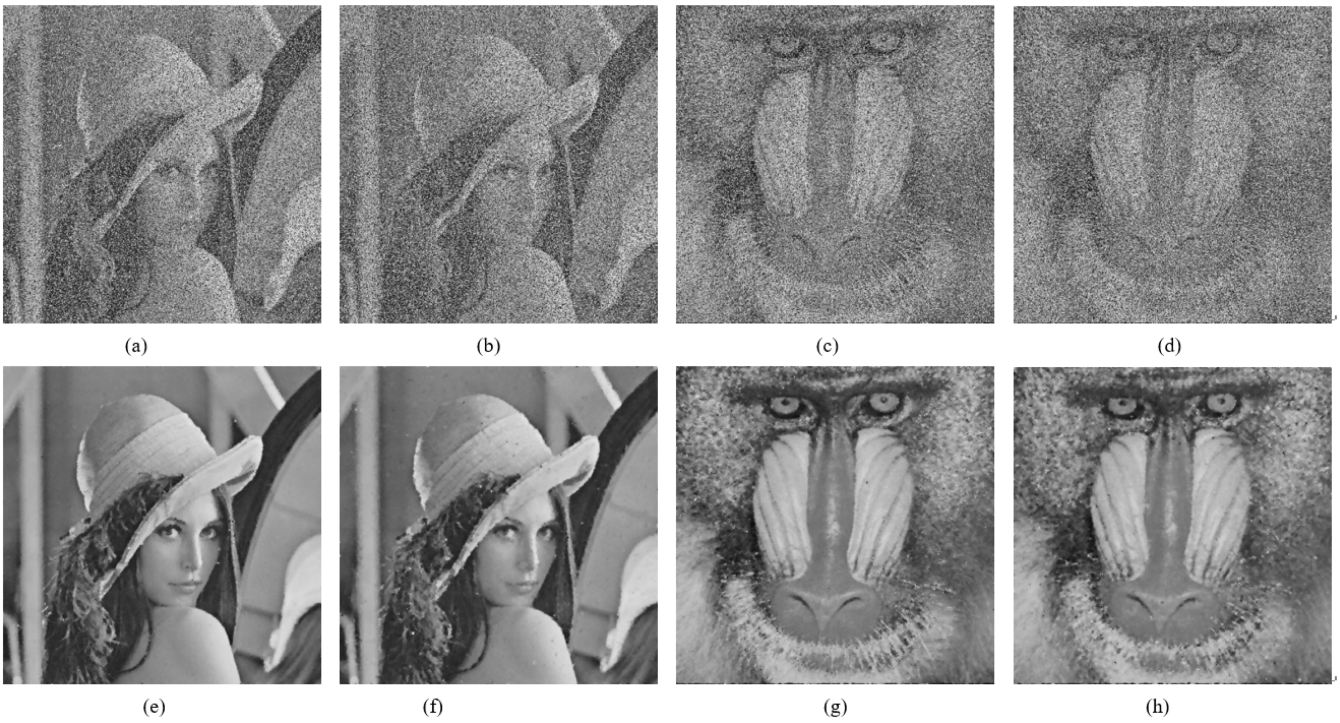


FIGURE 6. The filtering effects of proposed method on Lena and Baboon noisy images: (a)-(b) Lena image with 50%-60% RVIN; (c)-(d) Baboon image with 50%-60% RVIN; (e)-(f) Restored image of Lena with 50%-60% RVIN; (g)-(h) Restored image of Baboon with 50%-60% RVIN.

TABLE 3. Comparison of restoration results in PSNR (dB) on the Lena and boat images corrupted by 40% to 60% RVIN.

Methods	Lena			Boat		
	40%	50%	60%	40%	50%	60%
DWM [2]	**33.12**	**32.98**	**29.76**	27.25	25.97	24.52
ACWM [3]	32.87	31.24	28.8	27.13	25.49	23.76
NWM [8]	27.66	26.34	25.18	27.66	26.34	25.18
AEPWM [28]	31.77	30.01	28.03	**27.85**	26.61	24.87
ROR-NLM [9]	31.42	29.21	25.61	27.23	25.43	24.21
SDOOD [38]	32.06	30.24	27.42	26.78	25.79	24.44
SRM [39]	30.1	29.3	25.8	27.19	25.1	23.6
SBF [33]	30.78	28.16	26.62	27.14	26.01	24.62
TF [11]	31.36	29.44	27.09	27.72	**26.79**	24.91
Proposed	31.14	30.01	28.96	27.25	26.32	**25.34**

TABLE 4. Comparison of restoration results in SSIM on the Lena and Boat images corrupted by 40% to 60% RVIN.

Methods	Lena			Boat		
	40%	50%	60%	40%	50%	60%
DWM [2]	0.901	0.866	0.775	0.816	0.737	0.665
ACWM [3]	0.877	0.779	0.589	0.819	0.715	0.759
SRM [39]	0.886	0.826	0.742	0.865	**0.812**	0.71
SDOOD [38]	0.898	**0.869**	0.797	0.756	0.728	0.649
DnCNN [26]	0.891	0.844	0.793	**0.878**	0.806	**0.760**
SBF [33]	0.889	0.836	0.787	0.764	0.719	0.656
TF [11]	0.866	0.847	0.747	0.731	0.672	0.639
Luo's [42]	0.881	0.796	0.631	0.802	0.719	0.568
Proposed	**0.904**	0.868	**0.817**	0.805	0.761	0.693

of miss and false detected pixels left in the noise detection stage. But the good recovery effects have been obtained for those detected noises and non-detail regions.

Figure 7 shows the filtering results of different filtering methods on the Peppers image of 60% RVIN. It can be seen that the proposed method has fewer noise pixels and higher PSNR values in the filtered image. We can also find that the proposed method can restore the edges and textures of the image better than other methods by observing the detailed regions of the image. In summary, the method proposed in this paper has achieved significant results in both the noise detection stage and the noise removal stage.

C. ALGORITHM EXECUTION COST COMPARISON

The execution time and equipment investment of the algorithm are also used as an important index to measure the performance of the image denoising algorithm. Therefore, we compared the proposed method and two methods with significant filtering effects on the Lena image with 40% to 60% RVIN and the results of running time are listed in Table 5.

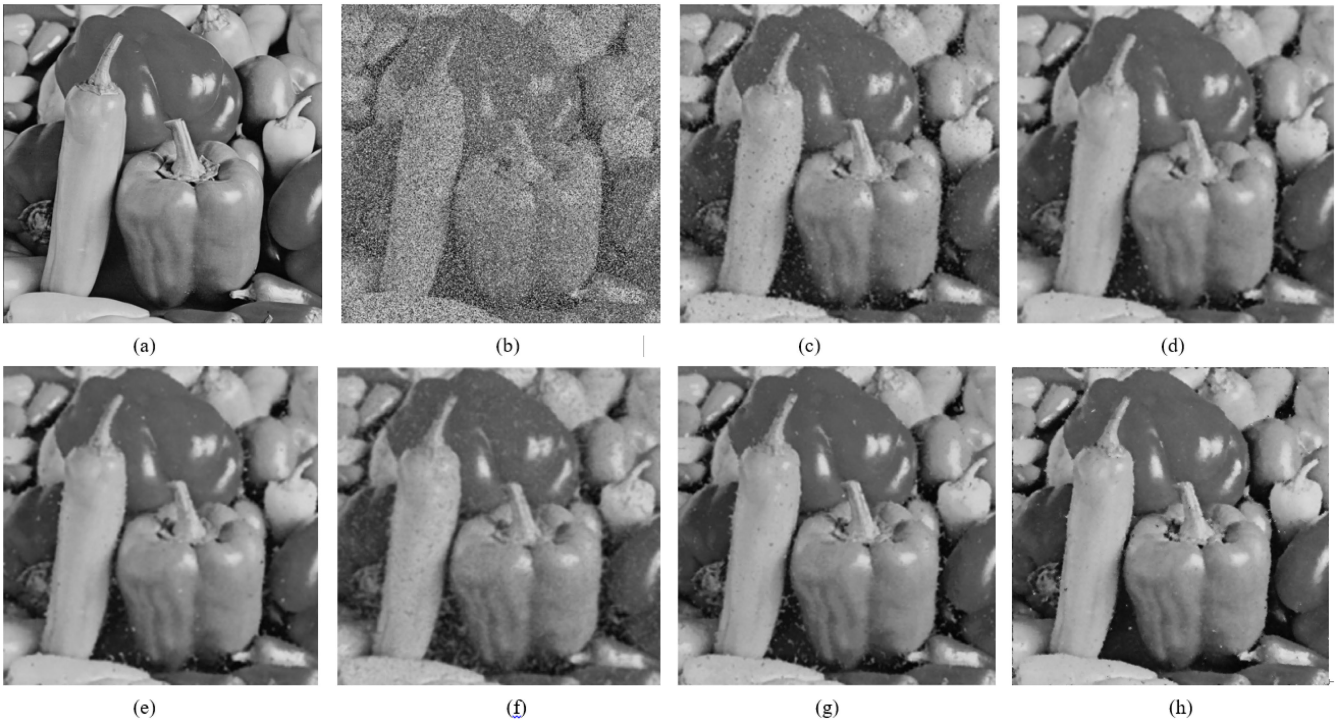


FIGURE 7. The filter results of different methods on Peppers corrupted by 60% RVIN: (a) Original image; (b) Damaged image with 60% RVIN, (c) Luos with PSNR=23.89 dB; (d) DWM with PSNR=25.48 dB; (e) ACWM with PSNR=25.33 dB; (f) SBF with PSNR=24.67 dB; (g) ROR-NLM with PSNR=25.45 dB; (h) Proposed with PSNR=27.01 dB.

TABLE 5. Comparison of running time consumption in Lena with 40% to 60% RVIN (in seconds).

Methods	40%	50%	60%
SAFF [24]	82.62	83.37	84.18
DnCNN [26]	6.34	6.38	6.75
Proposed	2.95	3.14	3.87

Our computing environment is: Processor: Intel Core i3-6100 CPU 3.70 GHz; running memory: 8.0 GB; system type: Windows 7 of 64-bit operating system. It can be seen that although [24] and [26] achieve very good filtering effects, the time cost is huge due to the addition of iteration and fuzzy control in [24]. And [26] uses a convolutional neural network so more expensive hardware equipment is required. However, the proposed method only iterates 2 or 3 times in the filtering stage so it spends a lower cost in hardware equipment and algorithm execution time.

V. CONCLUSION

In this paper we propose a two-stage denoising algorithm based on local similarity to detect and remove random-valued impulse noise in the image. The proposed method first uses the neighborhood information of the pixels to be detected to identify the impulse noise in the image, and then improves the traditional bilateral filter based on the local similarity of the pixels to recover the pixels affected by the impulse noise. Through a large number of experiments and analysis on different test images, it can be found that the proposed method has shown significant effects in both the noise detection stage and the image restoration stage. In the image restoration stage, the method only needs a few iterations to achieve the optimal image restoration effect. Although other denoising algorithms have achieved higher PSNR and SSIM values after multiple iterations, it increases the calculation time and equipment cost. In summary, the proposed method has achieved satisfactory results in terms of image processing performance and operating cost, which will be more helpful for algorithm transplantation in small devices.

Through observation it is found that the miss detected and false detected pixels in the first stage mainly exist in the contour and edge regions of the image, which is caused by the large difference between the intensity values of the pixels on the contour and the surrounding pixels. In a sense, the pixels in the contour region are also regarded as a kind of noise. When there are too many detail regions in the image or too much noise in the detail regions, this will not only increase the difficulty of noise detection, but also affect the filtering effect in the image restoration stage. In the subsequent improvement work, we will continue to discuss and analyze how to obtain the optimal parameters when judging whether the pixel is in a detailed area or a flat area. In order to improve the detection accuracy of the noise detector in the detail region of the image, we will also consider introducing other better edge noise detection methods for improvement in the later stage.

REFERENCES

[1] N. Singh, T. Thilagavathy, R. T. Lakshmipriya, and O. Umamaheswari, "Some studies on detection and filtering algorithms for the removal of random valued impulse noise," *IET Image Process.*, vol. 11, no. 11, pp. 953–963, Nov. 2017.

[2] Y. Dong and S. Xu, "A new directional weighted median filter for removal of random-valued impulse noise," *IEEE Signal Process. Lett.*, vol. 14, no. 3, pp. 193–196, Mar. 2007.

[3] T.-C. Lin, "A new adaptive center weighted median filter for suppressing impulsive noise in images," *Inf. Sci.*, vol. 177, no. 4, pp. 1073–1087, Feb. 2007.

[4] H. Dawood, H. Dawood, and P. Guo, "Removal of random-valued impulse noise by local statistics," *Multimedia Tools Appl.*, vol. 74, no. 24, pp. 11485–11498, Dec. 2015.

[5] Y. Dong, R. H. Chan, and S. Xu, "A detection statistic for random-valued impulse noise," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1112–1120, Apr. 2007.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[7] M. Tiwari and B. Gupta, "Maximum absolute relative differences statistic for removing random-valued impulse noise from given image," *Circuits, Syst., Signal Process.*, vol. 37, no. 5, pp. 2098–2116, May 2018.

[8] L. Liu, C. L. P. Chen, Y. Zhou, and X. You, "A new weighted mean filter with a two-phase detector for removing impulse noise," *Inf. Sci.*, vol. 315, pp. 1–16, Sep. 2015.

[9] B. Xiong and Z. Yin, "A universal denoising framework with a new impulse detector and nonlocal means," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1663–1675, Apr. 2012.

[10] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 60–65.

[11] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1747–1754, Nov. 2005.

[12] H. Dawood, M. Iqbal, M. Azhar, H. Ahmad, H. Dawood, Z. Mehmood, and J. S. Alowibdi, "Texture-preserving denoising method for the removal of random-valued impulse noise in gray-scale images," *Opt. Eng.*, vol. 58, no. 2, pp. 023103.1–023103.14, 2019.

[13] L. Xuegang, L. Junrui, and W. Juan, "Nonconvex low rank approximation with phase congruency regularization for mixed noise removal," *IEEE Access*, vol. 7, pp. 179538–179551, 2019.

[14] S. Xu, G. Zhang, L. Hu, and T. Liu, "Convolutional neural network-based detector for random-valued impulse noise," *J. Electron. Imag.*, vol. 27, no. 5, p. 1, Oct. 2018.

[15] A. Roy, L. Manam, and R. H. Laskar, "Region adaptive fuzzy filter: An approach for removal of random-valued impulse noise," *IEEE Trans. Ind. Electron.*, vol. 65, no. 9, pp. 7268–7278, Sep. 2018.

[16] M. Azhar, H. Dawood, and H. Dawood, "Texture-oriented image denoising technique for the removal of random-valued impulse noise," *J. Electron. Imag.*, vol. 27, no. 3, pp. 033028.1–033028.15, 2018.

[17] Q. Xu, Y. Li, Y. Guo, S. Wu, and M. Sbert, "Random-valued impulse noise removal using adaptive ranked-ordered impulse detector," *J. Electron. Imag.*, vol. 27, no. 1, pp. 148–167, 2018.

[18] I. Turkmen, "The ANN based detector to remove random-valued impulse noise in images," *J. Vis. Commun. Image Represent.*, vol. 34, pp. 28–36, Jan. 2016.

[19] K. Ashok, A. Kalaiselvi, and V. R. Vijaykumar, "Adaptive impulse detection based selective window median filter for removal of random-valued impulse noise in digital images," *COMPEL-Int. J. Comput. Math. Electr. Electron. Eng.*, vol. 35, no. 5, pp. 1604–1616, Sep. 2016.

[20] T. Yamaguchi, A. Suzuki, and M. Ikehara, "Detail preserving mixed noise removal by DWM filter and BM3D," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vol. E100.A, no. 11, pp. 2451–2457, 2017.

[21] G. Gao, Y. Liu, and D. Labate, "A two-stage shearlet-based approach for the removal of random-valued impulse noise in images," *J. Vis. Commun. Image Represent.*, vol. 32, pp. 83–94, Oct. 2015.

[22] I. Turkmen, "A new method to remove random-valued impulse noise in images," *AEU-Int. J. Electron. Commun.*, vol. 67, no. 9, pp. 771–779, Sep. 2013.

[23] N. Singh and U. Oorkavalan, "Triple threshold statistical detection filter for removing high density random-valued impulse noise in images," *EURASIP J. Image Video Process.*, vol. 2018, no. 1, p. 22, Dec. 2018.

[24] M. Nadeem, A. Hussain, A. Munir, M. Habib, and M. T. Naseem, "Removal of random valued impulse noise from grayscale images using quadrant based spatially adaptive fuzzy filter," *Signal Process.*, vol. 169, Apr. 2020, Art. no. 107403.

[25] M. Azhar, H. Dawood, H. Dawood, G. I. Choudhary, A. K. Bashir, and S. H. Chauhdary, "Detail-preserving switching algorithm for the removal of random-valued impulse noise," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 10, pp. 3925–3945, Oct. 2019.

[26] J. Chen, G. Zhang, S. Xu, and H. Yu, "A blind CNN denoising model for random-valued impulse noise," *IEEE Access*, vol. 7, pp. 124647–124661, 2019.

[27] B.-J. Zou, Y.-D. Guo, Q. He, P.-B. Ouyang, K. Liu, and Z.-L. Chen, "3D filtering by block matching and convolutional neural network for image denoising," *J. Comput. Sci. Technol.*, vol. 33, no. 4, pp. 838–848, Jul. 2018.

[28] N. Iqbal, S. Ali, I. Khan, and B. Lee, "Adaptive edge preserving weighted mean filter for removing random-valued impulse noise," *Symmetry*, vol. 11, no. 3, p. 395, Mar. 2019.

[29] T. Veerakumar, B. N. Subudhi, and S. Esakkirajan, "Empirical mode decomposition and adaptive bilateral filter approach for impulse noise removal," *Expert Syst. Appl.*, vol. 121, pp. 18–27, May 2019.

[30] G. Pok and K. H. Ryu, "Efficient block matching for removing impulse noise," *IEEE Signal Process. Lett.*, vol. 25, no. 8, pp. 1176–1180, Aug. 2018.

[31] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. 6th Int. Conf. Comput. Vis.*, 2002, pp. 839–846.

[32] Y. Zhang, X. Tian, and P. Ren, "An adaptive bilateral filter based framework for image denoising," *Neurocomputing*, vol. 140, pp. 299–316, Sep. 2014.

[33] C. H. Lin, J. S. Tsai, and C. T. Chiu, "Switching bilateral filter with a texture/noise detector for universal noise removal," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2307–2320, Apr. 2010.

[34] X. Xiao, N. N. Xiong, J. Lai, C.-D. Wang, Z. Sun, and J. Yan, "A local consensus index scheme for random-valued impulse noise detection systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Jul. 23, 2019, doi: 10.1109/TSMC.2019.2925886.

[35] G. Chen, F. Zhu, and P. A. Heng, "An efficient statistical method for image noise level estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 477–485.

[36] Z. Huang, S. Li, L. Fang, H. Li, and J. A. Benediktsson, "Hyperspectral image denoising with group sparse and low-rank tensor decomposition," *IEEE Access*, vol. 6, pp. 1380–1390, 2018.

[37] P. Jiang and J. Z. Zhang, *Fast and Reliable Noise Level Estimation Based on Local Statistic*. Amsterdam, The Netherlands: Elsevier, 2016.

[38] A. S. Awad, "Standard deviation for obtaining the optimal direction in the removal of impulse noise," *IEEE Signal Process. Lett.*, vol. 18, no. 7, pp. 407–410, Jul. 2011.

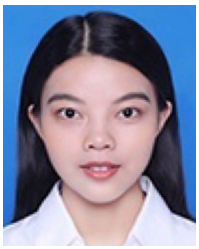
[39] B. Deka, M. Handique, and S. Datta, "Sparse regularization method for the detection and removal of random-valued impulse noise," *Multimedia Tools Appl.*, vol. 76, no. 5, pp. 6355–6388, Mar. 2017.

[40] P. N. T. Wells, "Handbook of image and video processing," *Physiological Meas.*, vol. 22, no. 1, p. 263, 2001.

[41] K. Gu, S. Wang, G. Zhai, W. Lin, X. Yang, and W. Zhang, "Analysis of distortion distribution for pooling in image quality prediction," *IEEE Trans. Broadcast.*, vol. 62, no. 2, pp. 446–456, Jun. 2016.

[42] W. Luo, "A new efficient impulse detection algorithm for the removal of impulse noise," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E88-A, no. 10, pp. 2579–2586, Oct. 2005.

CONG LIN received the B.S. and M.S. degrees from the Guangxi University of Science and Technology, China, in 2011 and 2015, respectively. He is currently pursuing the Ph.D. degree with Hainan University, China. He is currently a Lecturer with the School of Information and Communication Engineering, Guangdong Ocean University. His current research interests include machine learning and image processing.



YUCHUN LI received the B.S. degree from the University of Jinan, Jinan, China, in 2016, and the M.S. degree from the Nanjing University of Science and Technology, Nanjing, China, in 2019. She is currently pursuing the Ph.D. degree with Hainan University, China. Her current research interests include computer-aided diagnostic medical image processing and artificial intelligence.

SILING FENG received the Ph.D. degree from the University of Electronic Science and Technology of China, in 2014. She is currently an Associate Professor and a Ph.D. Supervisor with the School of Information and Communication Engineering, Hainan University. Her research interests include intelligent computing, big data analysis, and intelligent recommendation.

MENGXING HUANG (Member, IEEE) received the Ph.D. degree from Northwestern Polytechnical University, in 2007. He then joined the Research Institute of Information Technology, Tsinghua University, as a Postdoctoral Researcher. In 2009, he joined Hainan University. He is currently a Professor and a Ph.D. Supervisor of computer science and technology, and the Dean of the College of Information Science and Technology. He is also the Executive Vice-President of the Hainan Province Institute of Smart City, and the Leader of the Service Science and Technology Team, Hainan University. He has published more than 60 academic articles as the first or corresponding author. He has reported 12 patents of invention, owns three software copyright, and published two monographs and two translations. His current research interests include signal processing for sensor systems, big data, and intelligent information processing. He was awarded one Second Class and one Third Class Prizes of the Hainan Provincial Scientific and Technological Progress.