

## Research Article

# Image Speckle Denoising for Securing Internet of Smart Sensors

Wei Ma,<sup>1,2</sup> Zhihui Xin ,<sup>1,2</sup> Licun Sun,<sup>1,2</sup> and Jun Zhang <sup>1,2</sup>

<sup>1</sup>Yunnan Normal University, School of Physics and Electronic Information, Kunming/650500, China

<sup>2</sup>Yunnan Normal University, Yunnan Key Lab of Opto-Electronic Information Technology, Kunming/650500, China

Correspondence should be addressed to Zhihui Xin; [xinzhihui.luncky@163.com](mailto:xinzhihui.luncky@163.com) and Jun Zhang; [junzhang@ynnu.edu.cn](mailto:junzhang@ynnu.edu.cn)

Received 19 September 2021; Accepted 28 October 2021; Published 18 November 2021

Academic Editor: Weizhi Meng

Copyright © 2021 Wei Ma 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.

How to improve utility performance when securing sensitive data is an important research problem in Internet of smart sensors. In this paper, we study secured image speckle denoising for networked synthetic aperture radar (SAR). Speckle noise of SAR affects image quality and has a great influence on target detection and recognition. MSTAR dataset is often used in image target recognition. In this paper, a subregion-based method is proposed in order to improve the accuracy of target recognition and better retain target information while filtering and denoising the image. The new method applies advanced encryption techniques to protect sensitive data against malicious attack. Firstly, the image is divided into marked areas and unmarked areas through edge extraction and hole filling. Secondly, we use different size windows and filtering methods to filter the image in different areas. The experimental results show that the proposed algorithm has obvious advantages over MR-NLM, SSIM-NLM, Frost, and BM3D filtering in terms of equivalent view number and preserving edge and structure.

## 1. Introduction

Synthetic aperture radar (SAR) is a system of continuous tracking and monitoring imaging that can transmit and receive electromagnetic waves. As a smart sensor, SAR can provide full-time and full-weather observation for targets and is often used in military and civilian applications. Radar networking is an effective method to improve the radar system in the modern war and civil scenario. In contrast to the single radar system, the radar networking system can trace objects more precisely, is more capable to anti-jamming, and has larger scale in space domain, time domain, and frequency domain. When the radar networking is designed, the security of received data and the stability of the system need to be considered. Cyber security is very important in different netting systems. Therefore, detecting and preventing cyber insider threats are necessary [1–4]. The stealing attack based on machine learning is reviewed in perspectives of three categories of targeted controlled information [5]. When the radar networking system is utilized, the data from single radar and different processing flows should be encrypted and kept secure. Malware and software vulnerability detections based on machine learning and deep

neural models are studied to ensure that the Internet system is secure [6–8]. Due to the limitation of the imaging mechanism of radar, the complexity of ground environment usually makes the reflection of the surrounding environment to electromagnetic waves weaken the reflection of the target to electromagnetic waves. Therefore, speckle suppression of SAR image is extremely important for feature extraction of image and target recognition.

At present, MSTAR datasets supported by Defense Advanced Research Projects Agency are widely used for target recognition of the SAR image. Image denoising is needed to improve the efficiency of image recognition. There are two main purposes for image denoising in the MSTAR dataset: one is to smooth out the noise-polluted background area and the other is to retain more texture and detail information in the target area [9]. Commonly used SAR image denoising algorithms are divided into spatial domain filtering [10, 11], transform domain filtering [12–15], and partial differential anisotropic diffusion filtering [16, 17]. Spatial filtering is mainly based on LEE filtering [10] and Frost filtering [11]; in this method, the corresponding pixel values are processed by sliding a fixed-size window and predefined filter coefficients. However, the filtering effect is

limited by the window size and has the disadvantage of insufficient filtering. In recent years, transform domain filtering such as wavelet transform [12] or BM3D and its various improved methods have also achieved good results [13–15], but these methods are easy to introduce interference artificially. Anisotropic diffusion filtering [16, 17] takes an image as a heat field and determines whether to diffuse to the surrounding pixels according to the relationship between the surrounding pixels and the current pixel. However, this algorithm requires multiple iterations and requires a large amount of computation.

Based on bilateral filtering method, the non-local means (NLM) denoising algorithm was developed [18]. This algorithm uses image block similarity instead of traditional single-pixel similarity to construct weights. Making full use of the redundant information in the image, details of the image can be maintained to the greatest extent while denoising [19]. In recent years, scholars have carried out a large number of studies based on NLM filtering [19–25]. Through the SAR image experiment in MSTAR, it can be found that this method can achieve good denoise performance in the target area. However, the denoising performance in the background area and the edge area of the target is poor. NLM filtering effect is also related to the size of the window. Meanwhile, the methods of NLM filtering are impossible to balance the details of preserving details and smoothing noise. Due to the extremely serious noise pollution of the MSTAR data image, the variance-based segmentation algorithm proposed in paper [20] and the cluster-based segmentation algorithm proposed in paper [21] are not suitable for the MSTAR data image.

Each algorithm has its own merits and demerits. This paper proposes a subregion NLM filter method. The image is divided into marked region and unmarked region by extracting target information. Then, we comprehensively consider the noise difference between different regions. An improved NLM filter method is used for the unmarked region. The two regions need to be encrypted through encryption technology. Then, two encryption features are transferred into the next flow to decrypt and denoise with different filter weight, respectively. After this processing, different weighted functions and different sizes of filtering windows are used for NLM filtering in different regions. Experimental results show that the algorithm in this paper can greatly suppress noise while preserving the target details.

## 2. Proposed Method

**2.1. Non-Local Mean Algorithm.** The non-local mean algorithm can define the weight by measuring the similarity of the two similar blocks in the large search window when filtering the additive noise image and calculates the pixel value of the target point by weighted average. Its algorithm can be denoted as

$$U(i) = \sum_{j \in \Omega(i)} W(i, j)V(j), \quad (1)$$

where  $\Omega(i)$  is the large-scale search window centering on pixel  $i$ ,  $U(i)$  is the pixel after filtering, and  $V(j)$  is the pixel at any point in the search window.  $W(i, j)$  is the weighted coefficient of pixel  $j$  to pixel  $i$  in the search window. Weight is defined as

$$W(i, j) = \frac{1}{Z(i)} \exp\left(-\frac{d(i, j)}{h^2}\right), \quad (2)$$

where  $h$  controls the rate of decay of the exponential function, and it is usually equal to 10.  $Z(i)$  is the normalized factor, which can be expressed as

$$Z(i) = \sum_j \exp\left(-\frac{d(i, j)}{h^2}\right), \quad (3)$$

where  $d(i, j)$  is the similarity of two pixels in formulas (2) and (3). When calculating  $d(i, j)$ , the first step is to form similar areas of fixed sizes  $N(i)$  and  $N(j)$  with the pixels  $i$  and  $j$  as the center. The similarity is determined by the gray value vectors  $v(N_i)$  and  $v(N_j)$ , which can be measured by Gaussian weighted Euclidean distance which can be expressed as

$$d(i, j) = \left\|v(N_i) - v(N_j)\right\|_{2, \sigma}^2, \quad (4)$$

where  $\| \cdot \|_{2, \sigma}$  denotes the L2 norm.

However, due to the special imaging mechanism of SAR image, its noise is a typical multiplicative noise. Assuming that the observed value of a SAR image at position  $i$  is  $Y(i)$ , its real signal is  $X(i)$ , and the speckle noise is  $N(i)$ , and then its noise model is

$$Y(i) = N(i) * X(i). \quad (5)$$

Therefore, the non-local mean algorithm is not directly applicable to SAR image denoising. On the contrary, it is suitable only when the multiplicative noise is changed into additive noise after logarithmic processing.

**2.2. Subregion NLM Filtering Algorithm.** Classical NLM algorithms cannot maintain the texture of the target while smoothing the noise. A new subregion filtering process and weighting function are proposed in this paper. The proposed algorithm flowchart is shown in Figure 1. The original image is divided into two areas, and different areas are processed using different filtering weights. The edge contour forms a closed area after edge extraction and hole filling. The edge contour is used to mark the original image. The original image is divided into marked region and unmarked region. For the selection of the size of search window and similar window, small window should be used to maintain details and large window should be used to smooth noise. So, the small window is used in the marked area and the large

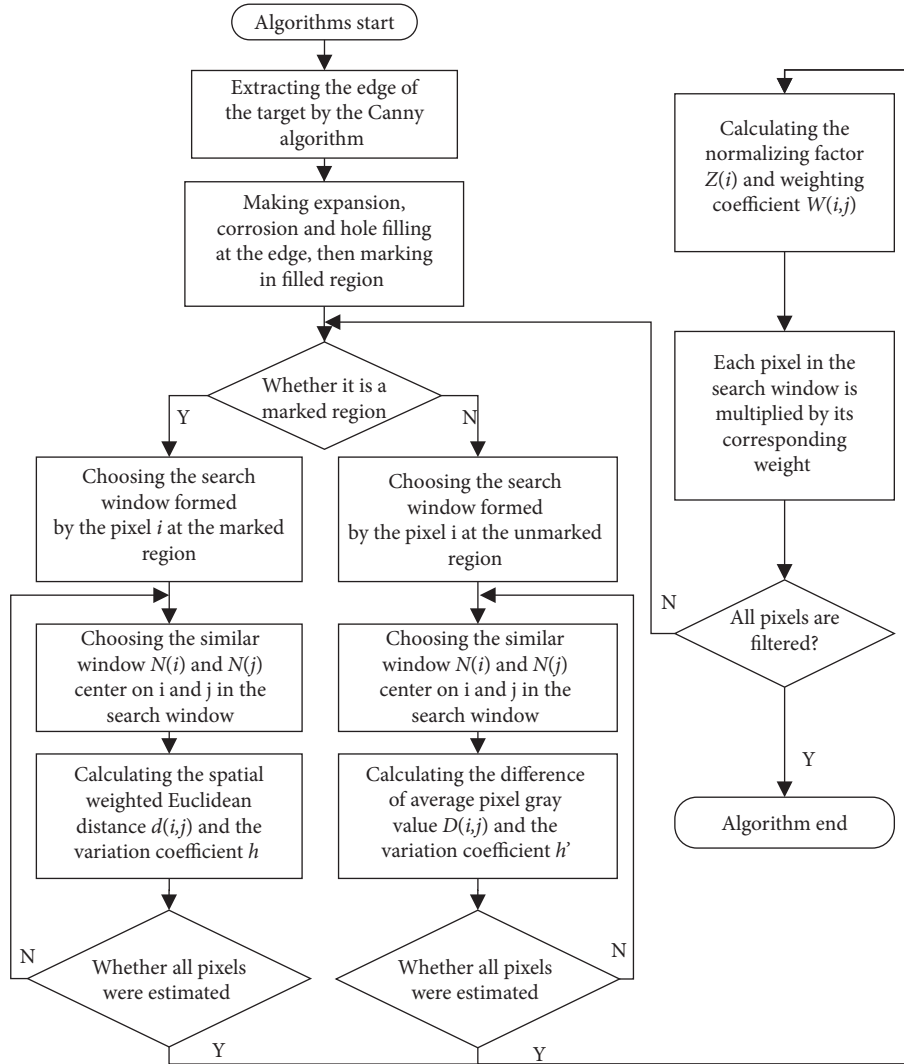


FIGURE 1: Algorithm flowchart.

window is used in the unmarked area. The marked area has a large amount of texture information. So, the similarity of windows is measured through Euclidean distance in NLM algorithm for the marked area. At the same time, the similarity of the window is measured by the difference of the average pixel for the unmarked area. When the image is filtered, the variation coefficient is chosen as the filtering parameter. The main steps of the proposed framework are summarized in Algorithm 1.

**2.2.1. Edge Extraction.** The Canny filtering algorithm is used to extract the image edge in the paper. The image is smoothed through Gaussian filtering. A fixed-size convolution template is used to calculate the gradient of the image. The image is divided into two parts by setting a threshold in the maximum suppression method, and then the two parts of the image are connected [26]. The commonly used convolution template includes the Soble operator, Roberts operator, and Prewitt operator. The gradient value and angle value of a certain point of the image are obtained according

to the correlation operator, and the edge of the target is determined according to the method of non-maximum suppression. In this paper, the Soble operator is used to calculate the horizontal and vertical gradients respectively, which can be expressed as

$$\text{Sobel}_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \quad (6)$$

$$\text{Sobel}_y = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}.$$

Because the edge curve obtained by Canny filtering algorithm is not necessarily closed, it needs to be expanded and corroded [19]. The expansion process is shown in Figure 2, in which Figure 2(a) shows the original image, assuming that the pixel value of the white area is 0 and the pixel value of the black area is 1. Set a structural element

**Input:** the SAR image  $A$

- (1) Extract the edge contour of the central target, carry out expansion, corrosion, and hole filling, form a closed area, and generate a binary image  $B$ ;
- (2) Read the image  $A$  and  $B$  size  $[m, n]$ ;
- (3) for  $p = 1:m$
- (4)   for  $q = 1:n$
- (5)     if  $B(i, j) = 255$
- (6)        $A(i, j) \in \varphi_u$ ;  $\varphi_u$  represents the marked area;
- (7)     else
- (8)        $A(i, j) \in \varphi_n$ ;  $\varphi_n$  represents the unmarked area;
- (9)   end; end; end
- (10) Then the marked area  $\varphi_u$  and unmarked area  $\varphi_n$  in the noisy image are obtained;
- (11) Read the image  $A$  size  $[m, n]$ ;
- (12) for  $i = 1:m$
- (13)   for  $j = 1:n$
- (14)     if current position  $(i, j)$  belongs to the marker area  $\varphi_u$
- (15)        $D(i, j) = |\overline{v(N_i)} - \overline{v(N_j)}|^2$  ;
- (16)     else
- (17)        $d(i, j) = \|v(N_i) - v(N_j)\|_{2,\sigma}^2$  ;
- (18)     Use formulas (2) and (11) to calculate the weights;
- (19)   end; end; end

**Output:** the final denoising image

ALGORITHM 1: Secured image speckle denoising for internet of smart sensors.

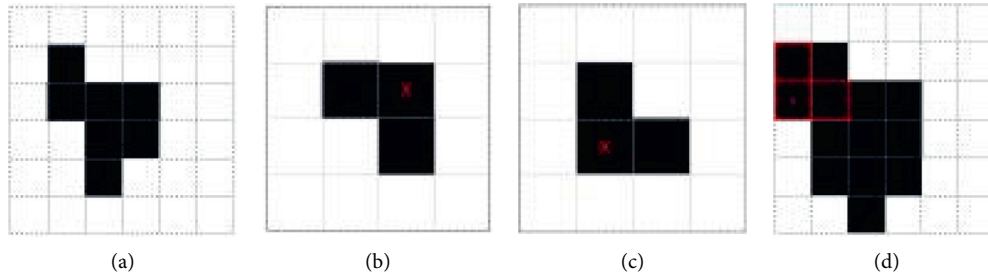


FIGURE 2: Image expansion process. (a) Original image. (b, c) Structural elements. (d) Expansion image.

(Figure 2(b)) and obtain the mapping (Figure 2(c)) of Figure 2(b) about the origin. The process of expansion is to make Figure 2(c) traverse every pixel in the original image. When Figures 2(c) and 2(a) have an intersection, the center pixel of Figure 2(c) is replaced by the maximum value in the structure. Figure 2(d) is the result of expansion (Figure 2(a)). The mathematical expression of expansion is

$$A \oplus B = \{x | (\widehat{B})_x \cap A \neq \emptyset\}, \quad (7)$$

where  $A$  is the original image and  $B$  refers to the structural elements.

The corrosion process is the inverse process of expansion. As shown in Figure 3, the aim of which is to make Figure 3(b) traverse every pixel in the original image (Figure 3(a)), when Figure 3(b) is included in Figure 3(a), the central pixel of Figure 3(b) is retained. Figure 3(d) is a corrosion diagram of Figure 3(a). The mathematical expression of corrosion is

$$A \ominus B = \{x | (B)_x \subseteq A\}. \quad (8)$$

**2.2.2. Hole Filling.** In this algorithm, the edge of the target is extracted by the Canny operator, but the edge is not necessarily continuous. The hole is filled after the expansion and corrosion of the edge. A hole can be understood as a background region surrounded by the boundary connected by foreground pixels. The basis of hole filling is also the expansion of image. After the edge of the target is obtained, the target region is separated from the image through hole filling [27]. As shown in Figure 4, Figure 4(a) is the edge image. First of all, the edge image is inverted to get Figure 4(b) in the process. Then, use an all-white graph with a black spot in the hole (Figure 4(c)) as the initial image and fill it with Figure 4(d). If the expansion result exceeds the size of the hole, use Figure 4(b) to find the intersection of it and limit it to the inside of the hole. Figure 4(e) is the result after filling twice. Assuming that the number of expansion operations is  $k$  and the result is  $X_k$ , when  $X_k$  and  $X_{k-1}$  are the same, the filling image (Figure 4(f)) of the hole is obtained. The filling image (Figure 4(f)) is merged with Figure 4(a) to form the final result (Figure 4(g)).

The hole filling formula can be expressed as

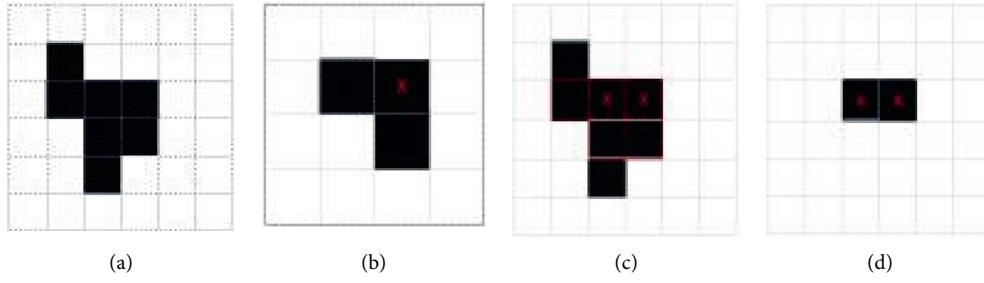


FIGURE 3: Image etching process. (a) Original image. (b) Structural element. (c) Corrosion process. (d) Corrosion image.

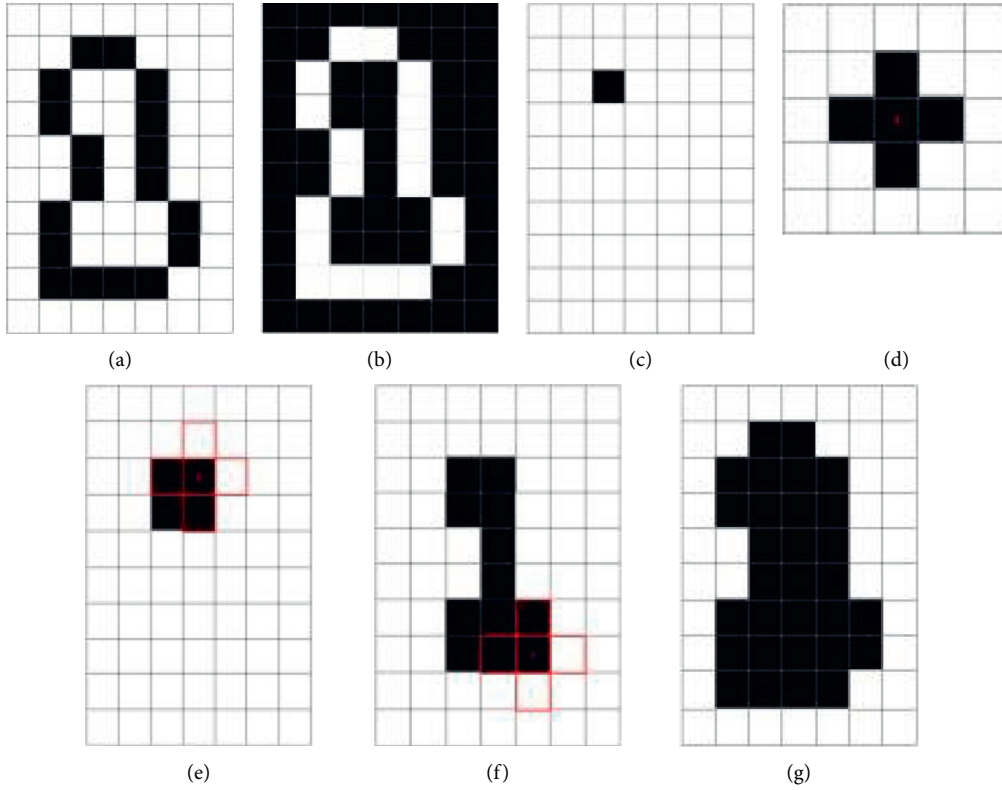


FIGURE 4: Hole filling process. (a) Edge image. (b) Edge image is inverted. (c) The first image of the hole filling. (d) Structural element. (e) The image after two fillings. (f) Filling image of the hole. (g) Final image.

$$\begin{aligned} x_k &= (x_{k-1} \oplus B) \cap A^c, \\ E &= x_k \cup A^c. \end{aligned} \quad (9)$$

**2.2.3. Improved Weight Function.** As shown in Figure 5, (a) is the similarity window of the center pixel of the search window, where the white pixel is noise, and (b) and (c) represent two different sliding windows, respectively.  $d(a, c) > d(a, b)$  can be obtained according to formula (4), and  $w(a, c) < w(a, b)$  can be obtained according to formula (2). But for (a), the white pixel at the center is noise and should be cleared, so the ideal weighting situation should be  $w(a, c) > w(a, b)$ . Borrowing the idea of neighborhood filtering, the difference between the gray values of the average pixels of similar blocks is used as the similarity function here so as to redefine the weighted function. When the difference

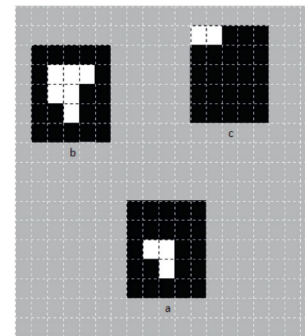


FIGURE 5: Pixel weighted weight analysis.

between the gray values of the two is smaller, a relatively large weight will be obtained. The smoothing effect of the isolated noise points is related to the size of the window. A

large window will make the noise smoother better, but the amount of calculation will increase accordingly.

The new similarity function is defined as

$$D(i, j) = \left| \overline{v(N_i)} - \overline{v(N_j)} \right|^2, \quad (10)$$

where  $\overline{v(N_i)}$  and  $\overline{v(N_j)}$  represent the average gray value of the similar box centering on  $i$  and  $j$ , respectively, and then the weight calculation formula of the gray scale of the pixel in the center of the image block is defined as

$$W'(i, j) = \frac{1}{Z(i)} \sum \left[ \exp\left(-\frac{D(i, j)}{h'^2}\right) \right], \quad (11)$$

where  $Z(i)$  is expressed as a normalized factor,  $D(i, j)$  denotes the grayscale distance of the mean pixels of similar blocks, and  $h'$  is the corresponding filtering parameter.

The selection of filter parameters plays an important role in the effect of filtering, the size of which directly determines the effect of denoising. The filtering parameters used in the traditional NLM filtering process are all constants, larger values are easy to lose details, and smaller values will retain more noise. Variation coefficient can well evaluate the degree of fluctuation of image pixels, so it is introduced into the filtering parameters. The coefficient of variation is defined as

$$CV(i) = \frac{\sigma(N(i))}{N(i)}, \quad (12)$$

where  $N(i)$  is a similar block with the pixel as the center of  $i$ ,  $\sigma(N(i))$  is the standard deviation of the similar block, and  $N(i)$  represents the mean value.

**2.2.4. NLM Filtering Algorithm Based on Subregion Improved Weights.** We can get the image contour information from the Canny filter algorithm, but because the outline is not closed, expansion, corrosion, and hole filling processing are needed. After edge extraction and hole filling, the original image is divided into marked area and unmarked area by marking the original image. The marked area is the central target with less noise, while the unmarked area is the target background with a large number of isolated noise points. To keep the target information and make effective denoising at the same time, this paper conducts the image in different filtering functions. First, we use the spatially weighted Euclidean distance as the similarity function in the marked area, as shown in formula (4), and appropriate filtering parameters are selected for filtering. Meanwhile, the mean difference of similar blocks is used as the similarity function for weighting in the unmarked region, as shown in equations (10) and (11), and appropriate filtering parameters are selected. According to the characteristics of MSTAR image, the new attenuation factor is defined as

$$\begin{aligned} h &= \beta * CV, \\ h' &= \alpha * CV, \end{aligned} \quad (13)$$

where  $h$  and  $h'$  are the filter parameters in the marked area and the unmarked area, respectively, and  $\beta$  and  $\alpha$  are constants.

### 3. Experiments and Results

**3.1. Experiment Settings.** Experimental data were obtained from the measured SAR ground stationary target data. They were published by the MSTAR project supported by DARPA. The radar operates in the X band, and polarization mode is HH polarization. The resolution is 0.3 m\*0.3 m. The pixel size is 100\*100. There are 7 types of ground targets in 3 categories.

The images of armored vehicles (SAR1) and tanks (SAR2) in the MSTAR dataset are used for verification. We use Frost algorithm [11], SSIM-NLM method [22], MR-NLM method [24], and block-matching 3D (BM3D) algorithm [15] to compare with the proposed algorithm to verify its superiority. The experiment adopted Intel (R) Core (TM) i7-10700F CPU@ 2.90Ghz. During the experiment, the filter window size of MR-NLM was 1 and 2, and the filter parameters were  $1/(1.5 * CV)$ . The size of the filter window in the SSIM-NLM algorithm is 1 and 2, and the filter parameter is  $22\sigma$ , where  $\sigma$  is the standard deviation obtained from paper [28] and the number of sights  $L$  in the BM3D algorithm is set to 3. In the algorithm of this paper, the search window radius in the marked area is 2, the similar window radius is 1, and the constant  $\alpha$  is 0.25. In order to verify that the weighted algorithm proposed in this paper is more beneficial to smooth the noise, two groups of experiments are carried out. In the first group, the radius of the search window in the unmarked region is 5, and the radius of the similarity window is 2. In the second group, the radius of the search window in the unmarked region is 10, and the radius of the similarity window is 3. In the experiments, the constant  $\beta$  is 0.2, and the threshold of Canny edge detection is 0.8.

**3.2. Denoising Quality Evaluation Index.** Subjective denoising quality evaluation uses naked eyes to observe the denoising degree of images, while objective quality evaluation uses equivalent number of looks, edge preserve index, and structure similarity index to evaluate.

**3.2.1. Equivalent Number of Looks (ENL).** The equivalent number is an indicator used to view the smoothing effect of the image, and its definition is as follows:

$$ENL = \frac{\mu^2}{\sigma^2}, \quad (14)$$

where  $\mu$  is the mean value and  $\sigma^2$  is the variance. The larger the equivalent number, the better the smoothing effect of the image. In this paper, the equivalent number is calculated only in the specific area.

**3.2.2. Edge Preservation Index (EPI).** The edge preservation index is used to evaluate the degree of image edge preservation, and its definition is as follows:

$$EPI = \frac{\sum (|U(i, j) - U(i+1, j)| + |U(i, j) - U(i, j+1)|)}{\sum (|V(i, j) - V(i+1, j)| + |V(i, j) - V(i, j+1)|)}, \quad (15)$$

where  $U(i, j)$  is the denoised image and  $V(i, j)$  is the original image. The larger the edge preservation index is, the better

the effect will be. In this paper, the edge preservation index is calculated only for the marked area, and the unmarked area is the noise area, which has no such index.

**3.2.3. Structure Similarity Index (SSIM).** The structural similarity index is used to measure the retention degree of image structure before and after denoising. The definition formula is as follows:

$$\text{SSIM}(i, j) = \frac{(2\mu_i\mu_j + C_1)(2\sigma_{ij} + C_2)}{(\mu_i^2 + \mu_j^2 + C_1)(\sigma_i^2 + \sigma_j^2 + C_2)}, \quad (16)$$

where  $\mu$  is the mean value,  $\sigma$  is standard deviation, and subscripts  $i$  and  $j$  show the similarity window, respectively, with their centers  $i$  and  $j$ . The larger the structure similarity index is, the better the effect will be. In this paper, the structure similarity index is calculated only for the marked area, and the unmarked area is the noise area, which has no such index.

**3.3. Experimental Results.** We apply the Frost algorithm [11], BM3D algorithm [15], SSIM-NLM algorithm [22], MR-NLM algorithm [24], and the algorithm in this paper to the SAR images of armored vehicles and tanks, respectively. Then, images and related parameters of the obtained armored vehicle are shown in Figure 6 and Table 1. The images and related parameters of the obtained tank are shown in Figure 7 and Table 2. Table 3 presents the results of running time of different algorithms in two groups of experiments, and it can be used to analyze the efficiency of the algorithm.

Armored vehicles have less edge information, while tanks have more edge information, so the robustness of the algorithm in this paper is verified. In this paper, a fixed area is selected in the image and marked by a white box in the original image, and the equivalent number of each algorithm is calculated in this fixed area. Then, the search window radius of 5 and similar block radius of 2 and the search window radius of 10 and similar block radius of 3 were defined for the unmarked area, respectively, in order to analyze the effectiveness of the new weighting function in smoothing noise.

As shown in Figure 6, the SSIM-NLM algorithm and the proposed algorithm have the best effect on noise smoothing, but the SSIM-NLM algorithm causes the image to be blurry. The contour information and texture information of the center target from Figure 6(c) are all lost. The EPI index of SSIM-NLM algorithm from Table 1 is less than 0.5, while that of the proposed method is 0.8625. EPI of the Frost algorithm from Table 1 is only 0.64, so the edge of the target is not well kept. The noise smoothing effect of Frost algorithm from Figure 6(d) is not obvious, though its ENL index from Table 1 is 6.7322. The detail texture and edge contour information of the BM3D algorithm are better preserved, but the noise suppression effect is the worst. The MR-NLM algorithm has made some progress in noise suppression compared to the BM3D algorithm. According to Table 1, MR-NLM algorithm and BM3D algorithm have good effect on target preservation, but the highest ENL index is only

4.2787. Compared with the proposed algorithm, the proposed algorithm has obvious advantages in noise suppression. From the analysis of the two denoising effect diagrams in this paper, when the window size is set to 5 and 2, respectively, the noise is not smoothed completely, and there are still some white patches. When the window is set to 10 and 3, the white patches are also smoothed out, which proves that the weighting function proposed in this paper has obvious advantages in noise smoothing. Because the same parameters and algorithms are used in the target area, the relevant parameters of the target area have not changed. To sum up, the algorithm in this paper has the optimal effect on noise smoothing, target contour, and detail preservation.

As shown in Figure 7, the central target of the original image includes a square body area and a protruding barrel area, especially an isolated white area directly below the body, which represents the vehicle wheel of the tank. The smoothing effect of the SSIM-NLM algorithm is too serious. While smoothing the noise, it also oversmooths the target. So, the target information is almost completely lost. The images of the vehicle wheel area the barrel area are completely erased from Figure 7(c). The denoising image does not retain any detailed texture information of the target, so its EPI and SSIM are the lowest among various denoising algorithms. The Frost algorithm makes the detailed information of the target lost from Figure 7(d), so EPI of the algorithm is relatively low. The denoising effect of the background of Frost algorithm is not ideal though ENL is 7.2377 in Table 2. The MR-NLM algorithm is excessively smooth in the target area, resulting in a decrease in the SSIM index, and its ENL index is the lowest. Compared with the MR-NLM algorithm, the BM3D algorithm has made certain progress in noise suppression and target retention. However, the BM3D algorithm with the highest EPI index has also reduced by 10% compared with EPI in this paper. The two denoising images in this paper indicate that window will smooth the noise to a greater extent without affecting the center target. The algorithm in this paper can control the ENL index by adjusting the window while maintaining a high level of EPI and SSIM indexes. To sum up, the algorithm in this paper has the optimal effect on noise smoothing, target contour, and detail preservation. The window size can be chosen to obtain an ideal denoising image using the proposed algorithm to recognize the target better.

Table 3 records the running times of the various algorithms in Figures 6 and 7. Since the window size of the algorithm in this paper is set to 2 and 5, the MR-NLM algorithm and the SSIM-NLM algorithm also adopt the same size window as that in this paper. As can be seen from this table, Frost algorithm has the shortest running time. BM3D algorithm is a block method of 3-dimension algorithm based on the similarity between image blocks; although it has shorter running time, the denoising is not good. Among the various improved algorithms of NLM, when MR-NLM adopts a smaller window, the computation time will be shorter, while a larger window will make the computation time longer. When a window of the same size as the one in this paper is used, the running time difference between the algorithm in this paper and the MR-NLM algorithm is less, but the denoising effect in this paper is

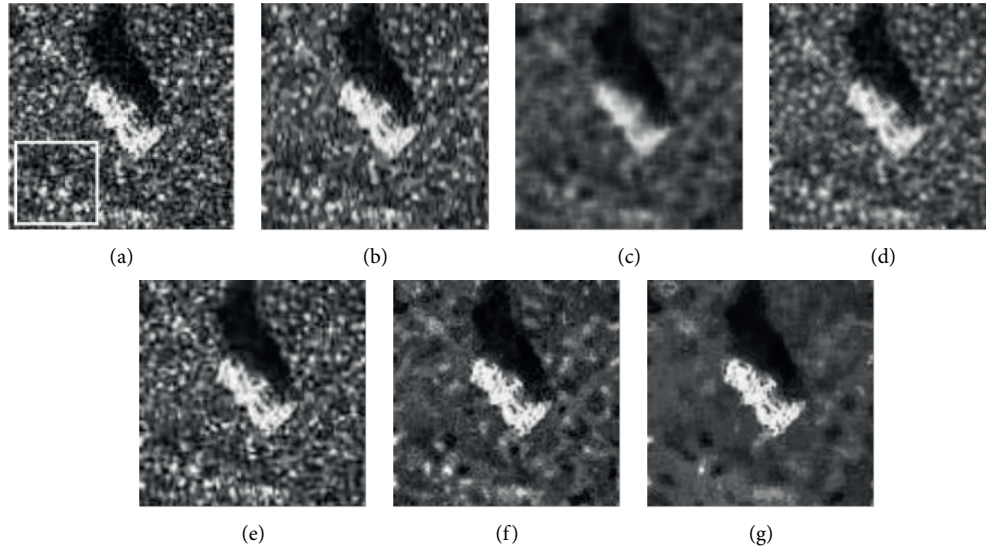


FIGURE 6: SAR images of armored vehicle. (a) Original. (b) MR-NLM. (c) SSIM-NLM. (d) Frost. (e) BM3D. (f) This paper (search window radius and the similarity window radius are set to 5 and 2). (g) This paper (search window radius and the similarity window radius are set to 10 and 3).

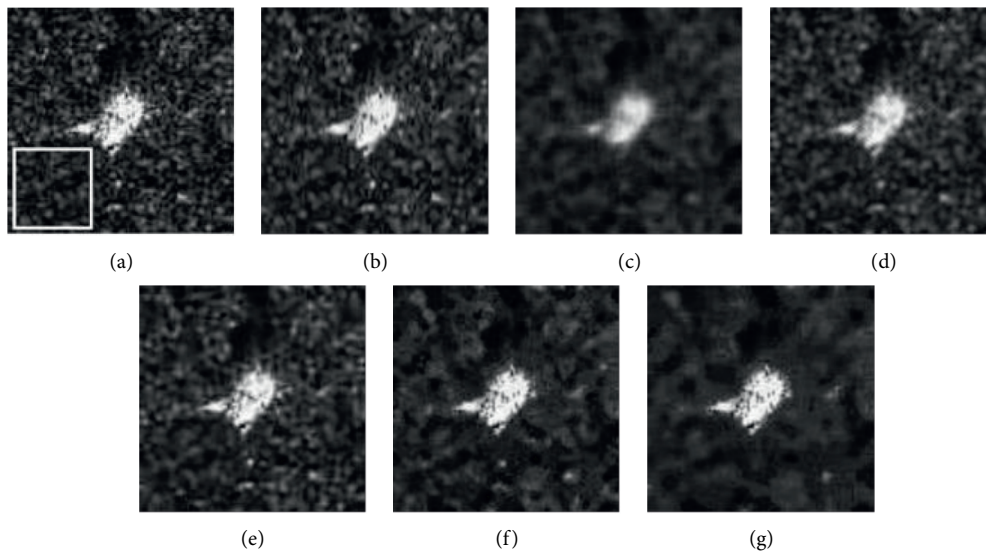


FIGURE 7: SAR images of tank. (a) Original. (b) MR-NLM. (c) SSIM-NLM. (d) Frost. (e) BM3D. (f) This paper (search window radius and the similarity window radius are set to 5 and 2). (g) This paper (search window radius and the similarity window radius are set to 10 and 3).

TABLE 1: Performance comparison of several algorithms in Figure 6.

Algorithms	EPI	SSIM	ENL
MR-NLM	0.7134	0.9799	4.2787
SSIM-NLM	0.4646	0.9546	13.7213
Frost	0.6442	0.9740	6.7322
BM3D	0.7955	0.9995	3.6986
This paper (2/5)	0.8625	0.9908	6.0072
This paper (3/10)	0.8625	0.9908	16.1660

better. SSIM-NLM algorithm introduces the structural similarity index, and the increase of parameters in the operation process leads to the increase of operation time and low

efficiency. Since the NLM algorithm itself has a high amount of computation, how to improve the computing efficiency of the algorithm will be the focus of future research.



TABLE 2: Performance comparison of several algorithms in Figure 7.

Algorithms	EPI	SSIM	ENL
MR-NLM	0.7169	0.9827	5.3808
SSIM-NLM	0.4147	0.9504	14.1114
Frost	0.6153	0.9741	7.2377
BM3D	0.7676	0.9994	5.9258
This paper (2/5)	0.8679	0.9918	7.3157
This paper (3/10)	0.8679	0.9918	13.6220

TABLE 3: Efficiency comparison of several algorithms in Figures 6 and 7.

	Frost (s)	BM3D (s)	MR-NLM		SSIM-NLM		This paper (2/5) (s)
			(1/2) (s)	(2/5) (s)	(1/2) (s)	(2/5) (s)	
Figure 6	0.3	4	74	129	1098	6389	118
Figure 7	0.3	3	72	129	1241	6369	117

This paper adopts different weighting methods in different regions through regional processing. From the analysis of SAR denoising images and relevant indicators of the three targets, the proposed method can achieve good denoising performance compared with the other methods. The texture information of the image is retained well and the noise is thoroughly smoothed. ENL, EPI, and SSIM of the proposed algorithm have great improvement compared with other algorithms.

#### 4. Conclusions

In this paper, a new segmentation filtering algorithm based on edge extraction was proposed, in which edge extraction is applied to SAR image segmentation algorithm. The edge of the target is first determined by edge extraction, and then the image is divided into different areas by expansion corrosion and hole filling. The separated regions are both encrypted before they are transferred to next procedure for security. In the process of filtering, each area is decrypted and filtered separately. Meanwhile, a new NLM filter weighting algorithm is proposed for the background areas with serious noise. The new segmentation filtering algorithm can preserve the characteristics of central target and smooth the background noise well. Performance indexes such as ENL, EPI, and SSIM are obtained to evaluate the algorithm compared with existing methods. Finally, experimental results and simulation data demonstrate the effectiveness of the proposed algorithm.

#### Data Availability

The MSTAR data used to support the findings of this study have been deposited in the following website: [https://download.csdn.net/download/a1367666195/12302537?utm\\_source=iteye\\_new](https://download.csdn.net/download/a1367666195/12302537?utm_source=iteye_new).

#### Conflicts of Interest

The authors declare that they have no conflicts of interest.

#### Acknowledgments

This research was supported by the National Natural Science Foundation of China (61801419) and Natural Science Foundation of Yunnan Province (2019FD114).

#### References

- [1] R. Coulter, Q.-L. Han, L. Pan, Z. Jun, and X. Yang, "Data driven cyber security in perspective - intelligent traffic analysis," *IEEE Transactions on Cybernetics*, vol. 50, no. 7, pp. 3081–3093, 2020.
- [2] N. Sun, J. Zhang, P. Rimba, Y. Z. Leo, G. Shang, and X. Yang, "Data-driven cybersecurity incident prediction: a survey," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1744–1772, 2019.
- [3] L. Liu, O. D. Vel, Q. L. Han, Z. Jun, and X. Yang, "Detecting and preventing cyber insider threats: a survey," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 2, pp. 1397–1417, 2018.
- [4] M. H. Wang, T. Q. Zhu, T. Zhang, Z. Wanlei, S. Yu, and Z. Jun, "Security and privacy in 6G networks: new areas and new challenges," *Digital Communications and Networks*, vol. 6, no. 3, pp. 281–291, 2020.
- [5] Y. T. Miao, C. Chen, L. Pan, Q. H. Long, Z. Jun, and X. Yang, "Machine learning based cyber attacks targeting on controlled information: a survey," *ACM Computing Surveys*, vol. 54, no. 7, 2021.
- [6] J. Y. Qiu, J. Zhang, L. Pan, L. Wei, N. Surya, and X. Yang, "A survey of android malware detection with deep neural models," *ACM Computing Surveys*, vol. 53, no. 6, pp. 1–36, 2021.
- [7] X. Chen, C. Li, D. Wang et al., "Android HIV: a study of repackaging malware for evading machine-learning detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 1, pp. 987–1001, 2020.
- [8] G. J. Lin, S. Wen, Q. L. Han, J. Zhang, and Y. Xiang, "Software vulnerability detection using deep neural networks: a survey," *Proceedings of the IEEE*, vol. 108, no. 10, pp. 1825–1848, 2020.
- [9] C. Liu, R. Szeliski, S. B. Kang, C. L. Zitnick, and W. T. Freeman, "Automatic estimation and removal of noise

- from a single image,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 299–314, 2008.
- [10] J. S. Lee, “Digital image enhancement and noise filtering by using local statistics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 2, pp. 165–168, 1980.
- [11] Y. Pan, Y. Meng, and L. Zhu, “SAR image despeckling method based on improved Frost filtering,” *Signal Image and Video Processing*, vol. 15, pp. 1–8, 2020.
- [12] X. Q. Yang, Z. H. Jia, J. Yang, and K. Nikola, “Change detection of optical remote sensing image disturbed by thin cloud using wavelet coefficient substitution algorithm,” *Sensors*, vol. 19, no. 9, pp. 1972–1986, 2019.
- [13] M. Lebrun, “An analysis and implementation of the BM3D image denoising method,” *Image Processing On Line*, vol. 2, no. 25, pp. 175–213, 2012.
- [14] M. M. Hasan, *Adaptive Edge-Guided Block-Matching and 3D Filtering (BM3D) Image Denoising Algorithm*, University of Western Ontario, London, United Kingdom, 2014.
- [15] Y. Mäkinen, L. Azzari, and A. Foi, “Collaborative filtering of correlated noise: exact transform-domain variance for improved shrinkage and patch matching,” *IEEE Transactions on Image Processing*, vol. 29, pp. 8339–8354, 2020.
- [16] Y. J. Yu and S. T. Acton, “Speckle reducing anisotropic diffusion,” *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 11, no. 11, pp. 1260–1270, 2002.
- [17] E. Cuevas, H. Becerra, and A. Luque, “Anisotropic diffusion filtering through multi-objective optimization,” *Mathematics and Computers in Simulation*, vol. 183, no. 2, pp. 1410–1429, 2021.
- [18] A. Buades, B. Coll, and J. M. Morel, “A non-local algorithm for image denoising,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 60–65, San Diego, CA, USA, June 2005.
- [19] S. Y. Chen and X. J. Li, “SAR image despeckling based on adaptive non-local means,” *Systems Engineering and Electronics*, vol. 39, no. 12, pp. 2683–2690, 2017.
- [20] S. J. Liu, G. Q. Wu, X. Z. Zhang, T. Yang, and Y. M. Li, “SAR image despeckling via the classification-based non-local clustering,” *Systems Engineering and Electronics*, vol. 38, no. 3, pp. 551–556, 2016.
- [21] J. M. Li, L. Zhu, B. Zhang, and Y. Pan, “A non-local means speckle suppression algorithm with restrained search window,” *Journal of Xi’an Jiaotong University*, vol. 54, no. 10, pp. 54–62, 2020.
- [22] Y. J. Wang, *The Study of Image Denoising Methods Based on the Non-local Means*, Harbin University of Science and Technology, Harbin, China, 2019.
- [23] B. G. Kim, S. H. Kang, R. P. Chan, H. W. Jeong, and Y. Lee, “Noise level and similarity analysis for computed tomographic thoracic image with fast non-local means denoising algorithm,” *Applied Sciences*, vol. 10, no. 21, 2020.
- [24] L. Zhu, F. F. Cai, Y. N. Wang, and L. Y. Guo, “A non-local means filtering algorithm for despeckling of SAR images,” *Journal of Xi’an Jiaotong University*, vol. 52, no. 4, pp. 98–104, 2018.
- [25] D. Devapal, S. S. Kumar, and C. Jojy, “A novel approach of despeckling SAR images using nonlocal means filtering,” *Journal of the Indian Society of Remote Sensing*, vol. 45, no. 3, pp. 443–450, 2016.
- [26] C. Huang, W. Jin, Q. Xu, Z. Q. Liu, and Z. L. Xu, “Sub-pixel edge detection algorithm based on canny–zernike moment method,” *Journal of Circuits, Systems, and Computers*, vol. 29, no. 15, pp. 1–18, 2020.
- [27] X. Liu and X. Ji, “Weld pool image processing and feature extraction based on the vision of the co2 welding,” vol. 355, pp. 625–633, in *Proceedings of the 4th International Conference on Computer Engineering and Networks*, vol. 355, Springer International Publishing, Harbin, China, December 2015.
- [28] A. F. Santiago, V. S. F. Gonzalo, M. F. Marcos, and A. L. Carlos, “Automatic noise estimation in images using local statistics. Additive and multiplicative cases,” *Image and Vision Computing*, vol. 27, no. 6, pp. 756–770, 2009.