applied sciences

MDPI

Article

A Fragile Watermark Scheme for Image Recovery Based on Singular Value Decomposition, Edge Detection and Median Filter

Xuan Xie, Chengyou Wang * and Meiling Li

School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai 264209, China
* Correspondence: wangchengyou@sdu.edu.cn; Tel.: +86-631-568-8338

Received: 13 May 2019; Accepted: 22 July 2019; Published: 26 July 2019

Abstract: Many fragile watermark methods have been proposed for image recovery and their performance has been greatly improved. However, jagged edges and confusion still exist in the restored areas and these problems need to be solved to achieve a better visual effect. In this paper, a method for improving recovery quality is proposed that adopts singular value decomposition (SVD) and edge detection for tamper detection and then uses a median filter for image recovery. Variable watermark information can be generated that corresponds to block classifications. With mapping and neighborhood adjustment, the area that has been tampered can be correctly detected. Subsequently, we adopt a filtering operation for the restored image obtained after the inverse watermark embedding process. During the filtering operation, a median filter is used to smooth and remove noise, followed by minimum, maximum and threshold operations to balance the image intensity. Finally, the corresponding pixels of the restored image are replaced with the filtered results. The experimental results of six different tampering attacks conducted on eight test images show that tamper detection method with the edge detection can identify the tampered region correctly but has a higher false alarm rate than other methods. In addition, compared with the other three similar methods previously, using a median filter during image recovery not only improves the visual effect of the restored image but also enhances its quality objectively under most tampering attack conditions.

Keywords: fragile watermark; image recovery; singular value decomposition (SVD); edge detection; median filter

1. Introduction

Modern widespread use of electronic devices and the popularity of social media allows people to easily obtain and share digital images in the manner they prefer. However, at the same time, the improvements in digital image processing software allows anyone to freely alter digital images. After an attacker has modified an image, the authenticity and integrity of the information in that image may be compromised. When an image is maliciously falsified, the top priority in the image processing field is to be able to restore the original image to the greatest extent. To solve this problem, digital image watermarking technology was invented. Digital watermarking capitalizes on the redundancy in digital images to embed hidden information, namely, a watermark, into the image. There are three kinds of watermarks: fragile watermark, semi-fragile watermark and robustness watermark. Among them, fragile watermark is often used in image tampering detection due to its sensitivity [1]. Thus far, scholars have carried out numerous studies in this field and achieved fruitful results, some of which are listed below.

Singh D. and Singh S.K. [1] proposed a fragile watermarking scheme in which the watermark is generated from the five most significant bits (MSBs) of each pixel and embedded into the three least significant bits (LSBs) corresponding to the mapped block. Shivani et al. [2] introduced a self-recovering fragile watermark scheme that embeds 10-bit of recovery data and 2-bit of authentication data into the LSBs of the corresponding mapped block. This method has a high capability for tampering detection and can recover an image effectively. However, it is not suitable for images containing random noise or that have undergone JPEG compression. Roy and Pal [3] proposed a multiple watermarking method that had an improved peak signal-to-noise ratio (PSNR) but that value was achieved at the cost of higher computational complexity. EI'arbi and Amar [4] suggested an image authentication algorithm with recovery capabilities based on neural networks. The methods in References [1–4] are all based on discrete cosine transforms (DCT). Javier et al. [5] introduced a watermark scheme for authentication and self-recovery. Inverse halftoning techniques and median filtering were also used to improve the quality of the recovered image. However, to prevent image tampering attacks, both [4] and [5] take only JPEG compression into account.

Al-Otum [6] proposed a semi-fragile watermarking technique based on a modified discrete wavelet transform quantization-based algorithm. The proposed method is suitable for grayscale image authentication and for tamper detection but was not tested for its ability to handle geometric attacks. Wang et al. [7] developed a three-level strategy to improve tamper detection accuracy and introduced a new block classification scheme based on singular value decomposition (SVD). The length of the generated recovery watermark differs within different blocks. Compared to References [8,9], this method achieved substantial improvements in tamper detection accuracy and recovery ability.

Using self-recovery blocks, Dhole and Patil [10] proposed a modified fragile watermarking scheme in which the detected tampered blocks were used to localize the erroneous regions of the tampered image, while the error-free blocks were used to recover the tampered blocks by applying block chaining. However, room still exists to improve image recovery. Dadkhah et al. [11] presented an SVD-based algorithm that used active watermarking that achieved both satisfactory self-recovery capability and was efficient at tamper detection. However, further research needs to be carried out for situations in which the recovery information is damaged by using the block-neighboring characteristic. Based on SVD, Zhang et al. [12] introduced a pixel-based fragile watermarking algorithm for image authentication. To guarantee the security of the proposed method, they used the Arnold transform twice during the watermark embedding process.

To distinguish the image block type, Hsu and Tu [13] introduced the concept of smoothness; based on the smoothness value, different watermark embedding, tamper detection and recovery strategies were applied. However, in terms of recovery capability, this method is highly applicable only to images with low variation. Sarreshtedari et al. [14] introduced a self-embedding method for digital images in the JPEG domain that applies source coding to recover the lost content using hierarchical trees, by which set partitioning is performed, and applies low-density parity check coding algorithms. Later, Fan and Wang [15] noted that the scheme in Reference [14] is ineffective when the tampering involves the channel parity bits; in such cases the method will use the wrong channel parity bits to perform channel decoding. They also indicated that the scheme had no ability to detect tampering. Lu and Liao [16] introduced a novel multipurpose watermarking scheme that simultaneously embeds both robust and fragile watermarks. However, their approach requires further exploration to eliminate the need to store and retrieve the mapping file and the hidden watermarks. Lee and Lin Reference [17] proposed an effective dual watermark scheme that detects tampered regions hierarchically. A secret key and a public mixing algorithm are used for tamper recovery. However, this method does not perform as well when the tampered area is trivial as when it is large. Qian et al. [18] proposed a novel fragile watermark scheme that uses a dual domain watermark for authentication, tamper detection and image recovery. Later, in Reference [19], Chetan and Shivananda introduced a method that improves the visual quality of the recovered image and can detect the tampered area accurately. In addition, according to the objective index PSNR, its recovery capability outperforms the method in Reference [18].

To enhance the recovery capability of restored images, this paper presents an improved method. Based on the good performance of SVD and edge detection, variable watermark information can be generated corresponding to block classifications. During image recovery, a median filter is applied to smooth and remove noise, followed by three operations on the original restored image that produce a satisfactory effect. Therefore, in the proposed method, SVD and edge detection are used for tamper detection; then, median filtering is used during image recovery. Finally, the corresponding pixels of the original restored image are replaced with the best results among the three operations.

The remainder of this paper is organized as follows. Section 2 provides a brief introduction to the method in Reference [7]. In Section 3, the proposed method is discussed in detail. Section 4 reports the experimental results and provides analyses. Finally, the paper is concluded in Section 5.

2. Related Work

Because SVD achieves good performance when extracting the texture features of an image, many methods based on SVD have been presented in recent years. Among these, the recently published method proposed by Wang et al. [7] demonstrates high tamper location and recovery performances based on DCT and neighborhood adjustment. Building on this method, we develop the proposed scheme to further improve the visual effect of the restored image. In this section, the method of [7] is briefly introduced. The general workflow is illustrated in Figure 1, showing its three main parts: watermark embedding, tamper detection and image recovery.

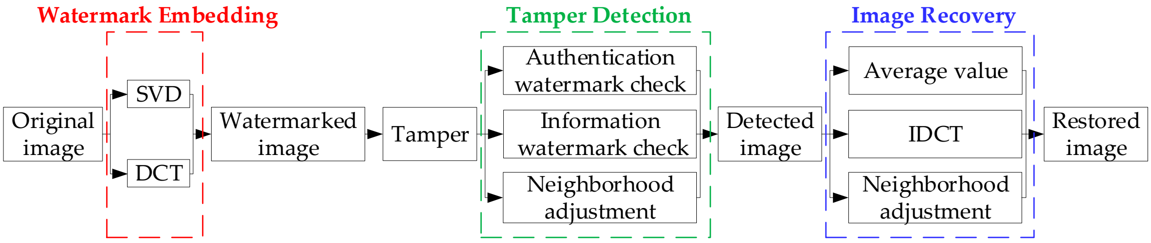

Figure 1. A brief overview of the method in Reference [7].

2.1. Brief Introduction to Wang et al.'s Method

The process of watermark embedding is marked with a red dotted box in Figure 1. To generate watermark information, the first step is to divide the original image into 2×2 nonoverlapping blocks and classify them. This is achieved by SVD, as shown in Equation (1):

$$\boldsymbol{A} = \left(a_{ij}\right)_{2\times 2} = \boldsymbol{U}\boldsymbol{\lambda}\boldsymbol{V}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} = \lambda_1 \boldsymbol{u}_1 \boldsymbol{v}_1^{\mathrm{T}} + \lambda_2 \boldsymbol{u}_2 \boldsymbol{v}_2^{\mathrm{T}}, \tag{1}$$

where $\boldsymbol{A}$ is a 2×2 matrix of pixels in a block. The block is divided into three matrices $\boldsymbol{U}$, $\boldsymbol{\lambda}$, $\boldsymbol{V}$, where $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ are the column vectors of $\boldsymbol{U}$ and $\boldsymbol{V}$, respectively. Thus $\boldsymbol{u}_i\boldsymbol{v}_i^{\mathrm{T}}$ $(i = 1, 2)$ are sub-images of the input image. The first sub-image, $\boldsymbol{u}_1\boldsymbol{v}_1^{\mathrm{T}}$, contains most of the information and for smooth blocks, because the intensities are similar, all the element values in the sub-image matrix are concentrated around 0.5. The authentication watermark is generated by the weight matrix $\boldsymbol{\lambda}$ and the information watermark is either the average pixel value in the smooth blocks or the coefficient after DCT [20] in the texture blocks. The final watermark consists of both authentication and information watermarks.

The second step is to insert the watermark into the 3 LSBs of each pixel. Different embedding schemes are implemented according to the classification results. Finally, a map is used to scramble the embedding position of the watermark to enhance security, as presented in Equation (2):

$$P' = (K \times P) \bmod N + 1, \tag{2}$$

where P and P' indicate the position of a block before and after mapping, respectively. N is the number of blocks and K is a custom prime ranging from 1 to $N - 1$.

During the tamper detection process, Wang et al. [7] presented a three-level detection mechanism to achieve higher location accuracy, marked with a green dotted box in Figure 1. This mechanism repeats the same operations used for the watermark generation process on the detected image and then compares the newly generated watermark with the extracted original watermark from two aspects: the authentication watermark and the information watermark. The last step is to adjust the marking status according to the neighborhood conditions, effectively reducing the false alarm rate.

Finally, the tampered region is recovered using the methods shown in the blue dotted box in Figure 1. During the inverse watermark embedding process, the information watermark is fully used to reconstruct the pixel values in the tampered areas. Later, any blocks that were not successfully recovered can be determined through neighborhood adjustment. This method produces final results that perform better than other methods under most tampering attack conditions.

2.2. *Shortcomings of Wang et al.'s Method*

However, after conducting many simulation experiments, some problems arise in the method of Wang et al. [7]. On the one hand, during watermark generation, the portions of the image with sharp intensity transitions include not only the texture information but also the edge regions, both of which can be regarded as important. However, SVD cannot completely detect the edge regions, as shown in Figure 2. The obvious differences, particularly in Figure 2d, show that SVD can detect the texture information of the image well but fails to extract the edge information. In this experiment, the range for determining a block as a smooth block is from 0.48 to 0.52, the same as in Reference [7]. On the other hand, the recovery results show jagged edges and confusion in some regions, especially under large area attack.

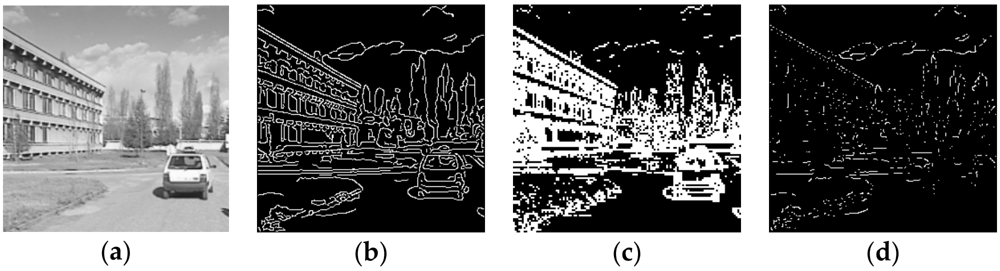

Figure 2. The differences in detection results by singular value decomposition (SVD) and edge detection: (**a**) the test image, Car1; (**b**) a binary image of the edge regions; (**c**) a binary image of the image's SVD result; (**d**) the binary image of the image's edge regions that were not detected by SVD.

3. The Proposed Method

In contrast to the method in Reference [7], the proposed method adopts edge detection during block classification and a filtering operation during the recovery process to solve these problems, as explained in the following subsections. Figure 3 shows an overview of the proposed method's tamper detection and recovery process: the differences between our method and that of [7] are highlighted in blue boxes.

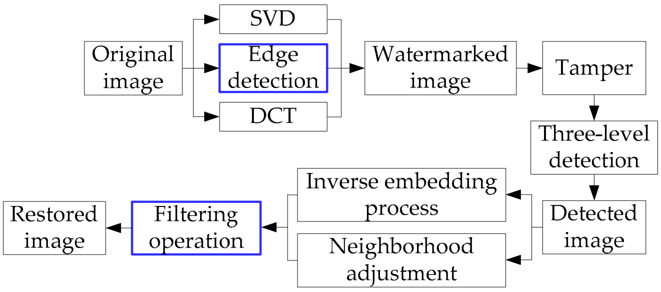

Figure 3. An overview of the proposed method's tamper detection and recovery process.

3.1. Edge Detection

To extract sufficient important information from an image, the edge and texture information needs to be detected. SVD is good at classifying texture information. Edge detection involves extracting object edges, a task at which the Canny operator [21] achieves the best effect. Therefore, we apply these two methods to generate watermarks. The calculation process can be summarized into four steps.

- Step 1. Convolve a Gaussian function $G(x, y)$ and the original image $f(x, y)$ to obtain a smoothed image $f_s(x, y)$, as shown in Equations (3) and (4), respectively:

$$G(x, y) = \mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{3}$$

$$f_s(x, y) = G(x, y) * f(x, y), \tag{4}$$

  where the default value of σ^2 is 2, (x, y) represents the pixel position and the $*$ operator represents a convolution operation.
- Step 2. Compute the gradient magnitude image $M(x, y)$ and angle images $\theta(x, y)$, shown in Equations (5) and (6), respectively:

$$M(x, y) = \sqrt{g_x^2 + g_y^2}, \tag{5}$$

$$\theta(x, y) = \arctan[g_y/g_x], \tag{6}$$

  where g_x and g_y represent the partial derivations of image $f_s(x, y)$ to x and y, respectively.
- Step 3. Apply non-maximum suppression to the gradient magnitude image.
- Step 4. Use double threshold and connectivity analysis to detect and link the edges.

3.2. Filtering Operation

As discussed in Section 2.2, the jagged edges in the restored area are mainly derived from the image blocks and the confused portion, which presents as blurring in the large area attack, stems from the undesirable pixel values introduced by unsuccessful recovery. To eliminate the jagged edges, a pixel-based method is needed instead of a block-based method to preserve the important information and reduce image noise. It is feasible to adjust the confused areas using their neighborhood or background pixels. When an image enhancement method is adopted for the restored image, it restores some obvious details but which method to apply depends primarily subjective judgment. In addition, such an enhancement cannot reduce the confusion. After abundant experiments and a comprehensive literature review, we found that the median filter has a strong effect for edge smoothing and eliminating confusion [5]; however, it causes a contrast reduction in the image and removes some detail in single pixel or small pixel areas. To avoid these defects, some other operations need to be introduced. The method should not only preserve the benefits of the median filter, which is applied to the restored image and the filtered image but increase the contrast by combining low pixel values with high ones using a threshold. The threshold mentioned above is a manually specified pixel value. It is adjustable depending on the image and its restored areas; consequently, it can lead to different recovery results. Based on the above analysis, we introduce a filtering operation to the proposed method. The specific filtering process is illustrated in Figure 4 and explained in the following steps.

- Step 1. The restored image R is obtained by performing the inverse embedding process. The binary image B highlights the tampered area detected by the proposed method. To extract the actual restored area in image R, a minimum operation is performed on image R and image B, generating image R_e, which has a black background except for the restored area. Then, median filtering is applied to image R_e and we obtain the filtered image R_ef, which is highly smoothed. However, at the boundaries (i.e., where the black background and the restored areas meet) the original restored pixels are replaced by those with values closer to those of the background, which introduces noise.

- Step 2. The minimum and maximum operations are performed on image R_{e} and image R_{ef}, generating the images G_1 and image G_2, respectively. Image G_1 contains lower pixel values, including the replaced boundary, while image G_2 contains higher pixel values. After removing the boundary, image G_1 is combined with image G_2 to generate image G_3 using a manually selected threshold as presented in detail in Section 4.3.
- Step 3. After the previous two steps, image G_3 not only smooths the noisy areas but also retains the texture and edge information. The final result, R', is obtained by replacing the restored pixels in image R with the corresponding pixels in image G_3.

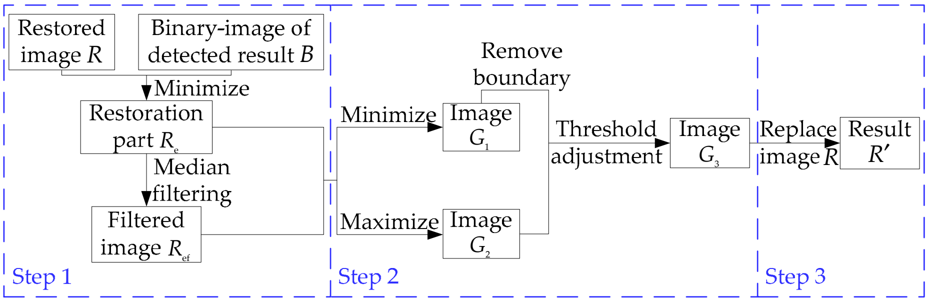

Figure 4. A flowchart of the filtering operation.

3.3. Main Process of Tamper Detection and Recovery

To strike a great balance between using fewer watermark bits and retaining sufficient significant information, we use the 3 LSBs of each pixel to store the watermark. Compared to Reference [7], we apply SVD and edge detection during block classification. After SVD, the sub-images and their weights are generated. The authentication watermark contains two bits, one bit is the result of the exclusive-or operation on each bit of the trace or weight matrix and the other bit is the result of applying the same operation to each even bit of that trace. The first condition for smooth blocks is that the block values of the first sub-image must lie in a range from 0.48 to 0.52. The larger this range is, the greater the number of blocks classified as smooth. After edge detection, a binary image containing 0 s and 1 s is obtained, where 1 represents an edge pixel. The second condition for the smooth blocks is that no block should contain more than one pixel with a value of 1.

After block classification, different types of information are used for watermarks. Because smooth blocks contain less information, we adopt the average pixel value directly, which forms a watermark with 5 bits of information. However, for texture blocks, the pixel values in the spatial domain are converted to the DCT coefficients in the frequency domain. To satisfy the requirements for watermark bits, we first need to scale the blocks. At the end of this process, the watermark information consists of four parts: the sign of the DC coefficient d_{11}, the sign of the first AC coefficient d_{12}, a 4-bit value for $|d_{11}|$ and a 3-bit value for $|d_{12}|$. The authentication watermark is inserted into its original pixels but the embedding position of the information watermark is scrambled through a mapping process, as shown in Equation (2). A schematic map of the embedding bits of the watermark is shown in Figure 5, where the authentication watermark is embedded in the orange bits. w_i $(i = 1, 2, 3, \ldots, 10)$ represents each bit of the information watermark and the embedding positions for these values are marked in green. $falg\_ts$ denotes the block classification, where a 0 represents a smooth block.

During the tamper detection process, a three-level detection mechanism similar to Reference [7] is used to locate tampered regions accurately and reduce the false alarm rate [7]. The image recovery process initially creates an image containing the inverse process of the watermark embedding methods, which is followed by neighborhood adjustment and the filtering operation to further improve the quality of the restored image, as discussed in Section 3.2.

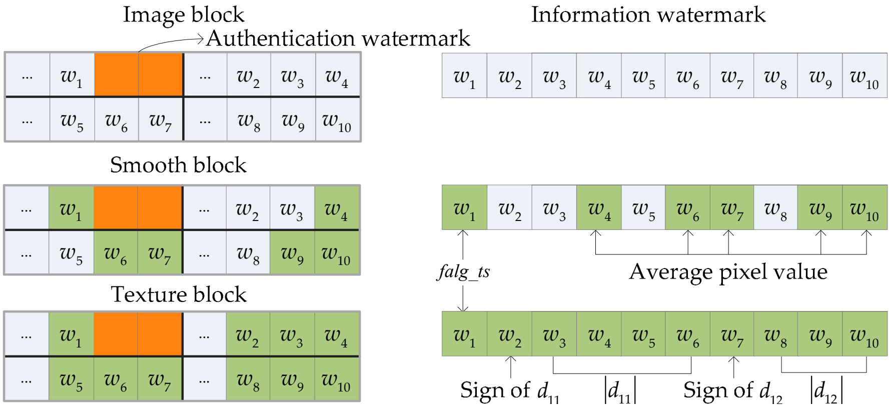

Figure 5. Embedding bits and the corresponding image watermark information.

4. Experimental Results and Analysis

To avoid occasionality, we put the proposed method into practice to test its performance under six different tampering conditions. In this experiment, we adopted the methods proposed by Tong et al. [8], Chen et al. [9] and Wang et al. [7] for comparison purposes. Eight test images from the CVG-UGR image database [22] are displayed in Figure 6: all of which have a size of 256×256. We executed our experiments on MATLAB R2016a.

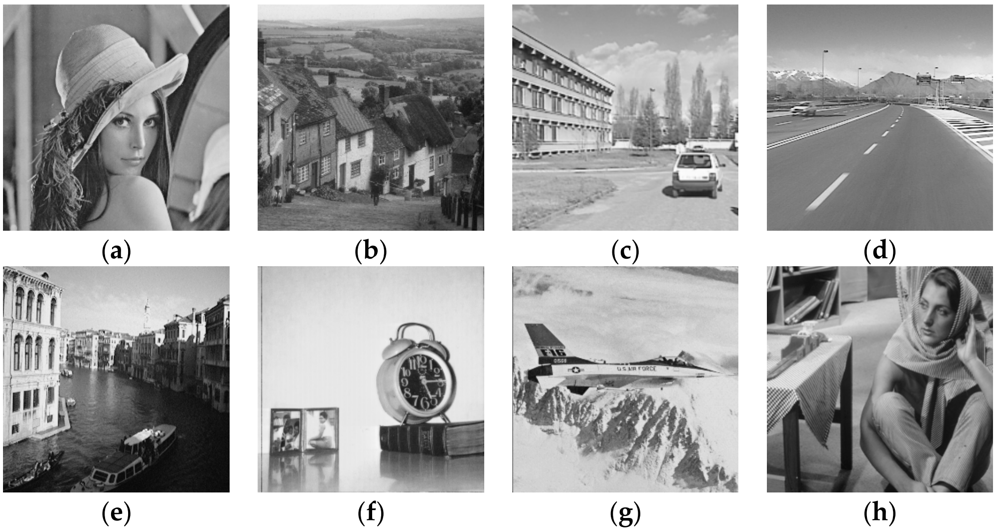
(**a**) (**b**) (**c**) (**d**)
(**e**) (**f**) (**g**) (**h**)

Figure 6. Test images: (**a**) Lena; (**b**) Goldhill; (**c**) Car1; (**d**) Car2; (**e**) Venice; (**f**) Clock; (**g**) Airplane; (**h**) Barbara.

To ensure fairness and consistency, we conducted objective assessments of the experimental results as measured uniformly by PSNR and structural similarity (SSIM) [23]. The PSNR value reflects the differences in corresponding pixels between two images. The larger the PSNR value is, the better the objective quality of the result is. When the PSNR exceeds 35 dB, humans are unable to discern the difference between two images. However, the PSNR value does not completely reflect with the subjective effect. Therefore, we also introduce the SSIM value, which measures image quality from three aspects: brightness, contrast and structure. Similar to PSNR, the SSIM value is positively correlated with the objective quality.

4.1. Performance in Watermark Embedding

To test the differences between SVD and edge detection, the proposed method and the method with only SVD [7] are compared along four aspects: the number of smooth blocks N_{SB}, the number of texture blocks N_{TB}, the value of PSNR in dB and the value of SSIM. The results are listed in Table 1.

Table 1. Comparison of watermark embedding results along four aspects.

Test Images	N_{SB}		N_{TB}		PSNR (dB)		SSIM	
	Wang et al. [7]	Proposed	Wang et al. [7]	Proposed	Wang et al. [7]	Proposed	Wang et al. [7]	Proposed
Lena	8561	8372	7823	8012	39.82	39.74	0.9636	0.9985
Goldhill	5038	4637	11,346	11,747	38.91	39.08	0.9734	0.9984
Car1	10,516	9998	5868	6386	40.41	40.07	0.9650	0.9978
Car2	13,572	12,872	2812	3512	42.20	40.76	0.9627	0.9981
Venice	6062	5758	10,322	10,626	39.15	39.21	0.9627	0.9993
Clock	12,366	11,842	4018	4542	41.22	40.49	0.9575	0.9991
Airplane	10,946	10,673	5438	5711	40.56	40.40	0.9629	0.9986
Barbara	7321	7163	9063	9221	39.52	39.42	0.9697	0.9987

The proposed method marks more texture blocks with the implementation of edge detection, which also proves that SVD cannot extract all the edge information. As the number of texture blocks increases, more information can be stored in the watermarks; therefore, the PSNR values generally show a downward trend. However, all the values exceed 35 dB, which means that the information is invisible. The SSIM value indicates the similarity between the watermarked image and the original image. Similar to the PSNR, the higher the SSIM value is, the better the objective quality of the results is.

4.2. Analysis of Different Tampering Test Results

In this section, the edge detection and filtering operation performances are tested. For comparison, the differences among the four methods are as follows: the method in Reference [8] uses only chaotic mapping; the method in Reference [9] uses variable watermarks extracted from roughness information; the method in Reference [7] extracts texture information and uses neighborhood adjustment and the proposed method extracts texture and edge information and also adds the final filtering operation. Due to the diversity of images and tampering operations, the tampered regions have different characteristics. Some mainly contain smooth information, while some are full of texture details. To analyze the adaptability and performance of the methods under different conditions, different images are selected and displayed in the following subsections.

4.2.1. Text Addition Attack

A text addition attack introduces some of textual information to a watermarked image—the word "LENA" in this experiment. The images in Figure 7 show the experimental results and Table 2 gives the PSNR and SSIM values of the four different methods.

By comparing the four images listed in Figure 7e–h, it is obvious that tamper detection is more accurate when using watermarks with texture information. However, the addition of edge information causes tamper detection more sensitive to edge changes, resulting in a higher false alarm rate. From the recovery results shown in Figure 7i–l, it can be seen that the final results of [8,9] do not fully recover the tampered region. But with neighborhood adjustment, both the proposed method and the method in Reference [7] restore the detection area without any visible missing information. However, the PSNR and SSIM values in Table 2 show that the quality of the final image is slightly better after the filtering operation.

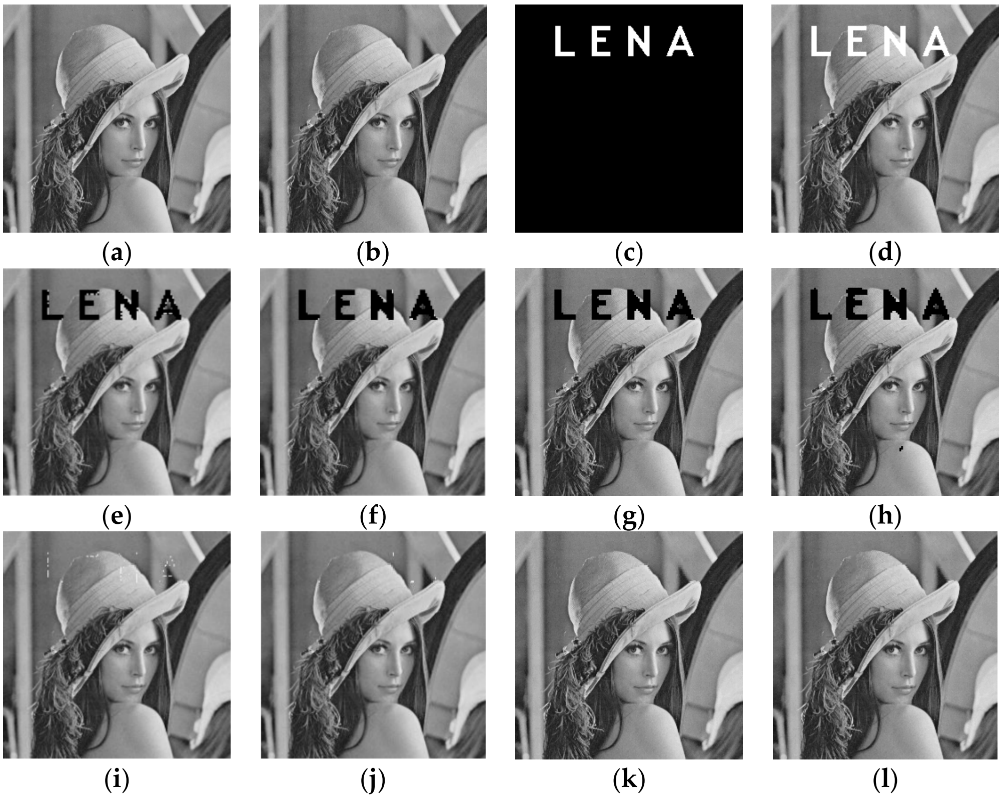

Figure 7. Tamper detection and recovery results of text addition attack on Lena image: (**a**) original image; (**b**) watermarked image; (**c**) tamper reference; (**d**) tampered image; (**e**) tamper detection result of [8]; (**f**) tamper detection result of [9]; (**g**) tamper detection result of [7]; (**h**) tamper detection result of the proposed method; (**i**) recovery result of [8]; (**j**) recovery result of [9]; (**k**) recovery result of [7]; (**l**) recovery result of the proposed method.

Table 2. Comparison of recovery performance for text addition attack on Lena image.

Indexes	**Tong et al. [8]**	**Chen et al. [9]**	**Wang et al. [7]**	**Proposed**
PSNR (dB)	35.01	40.85	45.52	45.68
SSIM	0.8441	0.8559	0.9925	0.9930

4.2.2. Copy-Move Attack

A copy-move attack involves copying part of an original image and moving it into another image. Figure 8 displays the specific process and Table 3 gives the PSNR and SSIM values of the different methods. As shown in Figure 8c, a part of the sky is copied from image Car1 and moved into the top of the watermarked Goldhill image. Based on Figure 8e–h, the tamper detection performance is essentially the same as that of the text addition attack and indicates that texture information and neighborhood adjustment play a significant role in image authentication and tamper location discovery. Although edge information is also an important part of an image, it reduces the detection accuracy and increases the redundancy. Compared with the other three recovery results in Figure 8i–k, the final image of the proposed method visually appears more similar to the watermarked image. Moreover, the values listed in Table 3 objectively demonstrate its quality. The tampered area consists of a large scale of smooth background at a distance, which is suitable for exploiting the advantages of the median filter. In this case, the image after filtering operation exhibits a better quality.

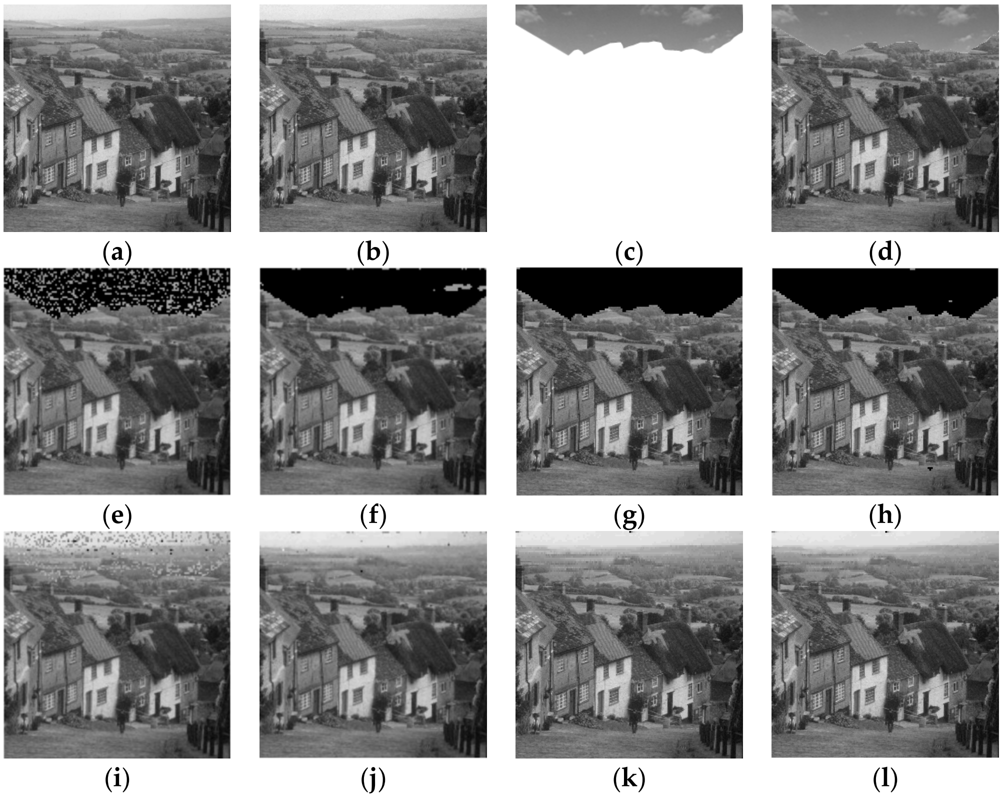

Figure 8. Tamper detection and recovery results of copy-move attack on Goldhill image: (**a**) original image; (**b**) watermarked image; (**c**) tamper reference; (**d**) tampered image; (**e**) tamper detection result of [8]; (**f**) tamper detection result of [9]; (**g**) tamper detection result of [7]; (**h**) tamper detection result of the proposed method; (**i**) recovery result of [8]; (**j**) recovery result of [9]; (**k**) recovery result of [7]; (**l**) recovery result of the proposed method.

Table 3. Objective comparison of the recovery performance for a copy-move attack on the Goldhill image.

Indexes	**Tong et al.** [8]	**Chen et al.** [9]	**Wang et al.** [7]	**Proposed**
PSNR (dB)	25.90	35.05	36.13	36.34
SSIM	0.6861	0.7910	0.9925	0.9930

4.2.3. Splicing Attack

Unlike a copy-move attack, a splicing attack extracts a portion of a watermarked image and then splices it into another watermarked image. Figure 9 displays the tamper location and recovery results for the different methods and Table 4 lists their PSNR and SSIM values. Figure 9c shows that a car from the watermarked image Car1 was inserted into the same position in another watermarked image, Car2. In Figure 9e, the tamper detection result locates only a portion of the boundary of the tampered region but the other three methods detect the tampered region correctly. This results occurs because the method in Reference [8] cannot achieve independent block authentication. Comparing the images in Figure 9j–l and the values listed in Table 4, the proposed method and the methods of [7,9] all achieve high performance; however, the proposed method is the best. The reasons are as follows. In this experiment, the tampered region in the Car2 image contain only smooth information. Therefore, no jagged edges or confused areas affect the recovery quality. In addition, median filtering removes some noises without affecting the edge characteristics, further improving the subjective and objective qualities of the final image.

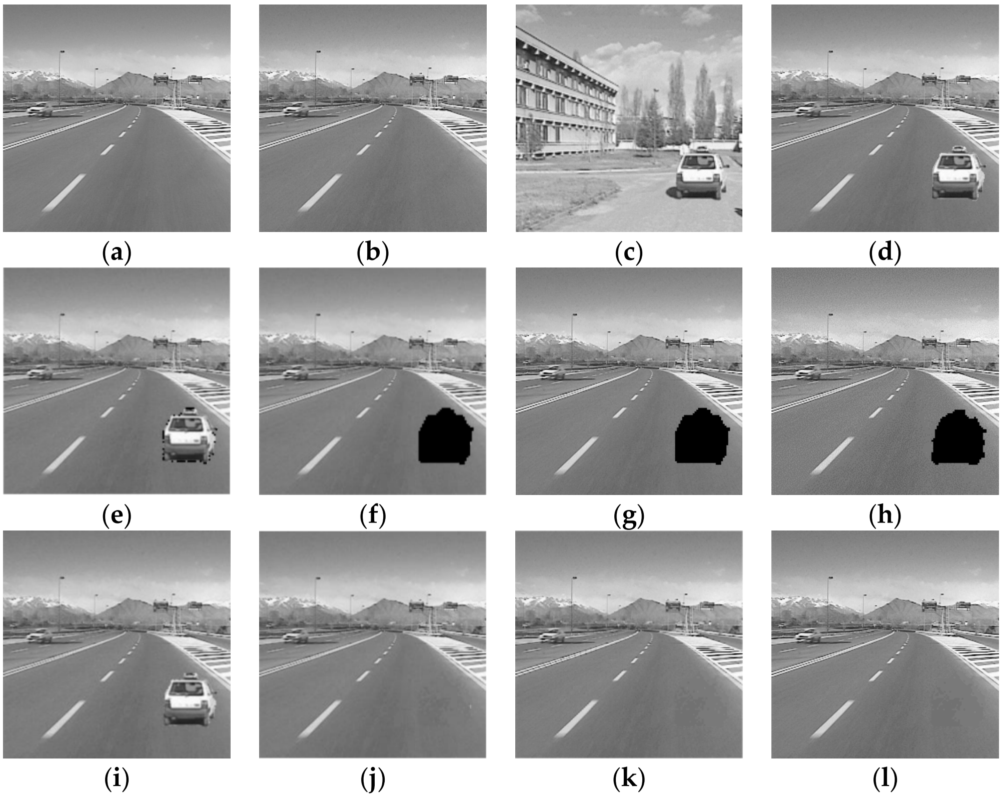

Figure 9. Tamper detection and recovery results of a splicing attack on the Car2 image: (**a**) original image; (**b**) watermarked image; (**c**) tamper reference; (**d**) tampered image; (**e**) tamper detection result of [8]; (**f**) tamper detection result of [9]; (**g**) tamper detection result of [7]; (**h**) tamper detection result of the proposed method; (**i**) recovery result of [8]; (**j**) recovery result of [9]; (**k**) recovery result of [7]; (**l**) recovery result of the proposed method.

Table 4. Objective comparison of the recovery performances for a splicing attack on the Car2 image.

Indexes	Tong et al. [8]	Chen et al. [9]	Wang et al. [7]	Proposed
PSNR (dB)	24.76	47.17	47.79	47.97
SSIM	0.8016	0.8390	0.9925	0.9930

4.2.4. Image Deletion Attack

An image deletion attack is used to remove part of an image; this operation can be approximately reconstructed after image recovery. The tamper detection and recovery results are shown in Figure 10 and partially enlarged details are presented in Figure 11. Table 5 lists the PSNR and SSIM values of the different methods. As Figure 10e–l shows, all four methods achieve high tamper location and image recovery performances. Figure 11 shows some of the differences. Figure 11a shows some black noise. Compared with Figure 11b, Figure 11c presents a clearer visual effect and Figure 11d has the smoothest edges among all the results. With the threshold adjustment process, several unsuccessfully restored pixels are recovered. Meanwhile, the filtering operation, effectively smooths the jagged edges without confusing their characteristics. However, as shown by comparison values listed in Table 5, the objective quality of the proposed method does not outperform the best result, obtained by [7]. This can be explained by the drawback of the median filter. As can be seen in the enlarged images, several details exist in single or small pixel regions that can easily be lost if they are not in the middle. However, this problem can be solved by threshold adjustment to a large extent. In general, the proposed method achieves a better visual effect and a fairly good objective quality compared with the other methods.

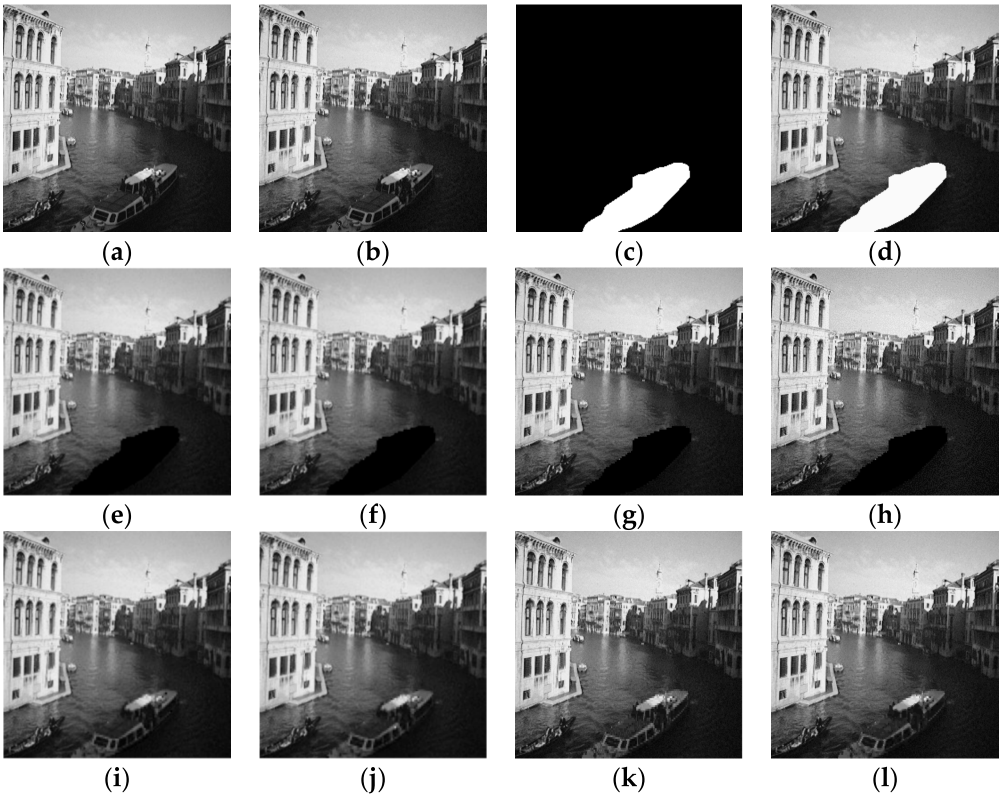

Figure 10. Tamper detection and recovery results of an image deletion attack on the Venice image: (**a**) original image; (**b**) watermarked image; (**c**) tamper reference; (**d**) tampered image; (**e**) tamper detection result of [8]; (**f**) tamper detection result of [9]; (**g**) tamper detection result of [7]; (**h**) tamper detection result of the proposed method; (**i**) recovery result of [8]; (**j**) recovery result of [9]; (**k**) recovery result of [7]; (**l**) recovery result of the proposed method.

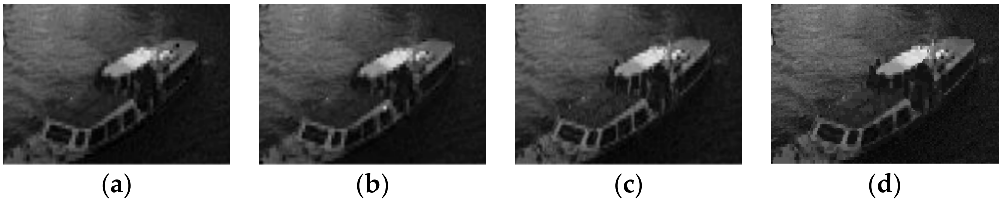

Figure 11. Partially enlarged details of restored images: (**a**) details in Reference [8]; (**b**) details in Reference [9]; (**c**) details in Reference [7]; (**d**) details in the proposed method.

Table 5. Objective comparison of the recovery performance for the image deletion attack on the Venice image.

Indexes	Tong et al. [8]	Chen et al. [9]	Wang et al. [7]	Proposed
PSNR (dB)	35.19	35.21	37.27	37.05
SSIM	0.7194	0.7911	0.9925	0.9930

4.2.5. Content-Only Attack

A content-only attack is an operation that tampers with the image content without changing the embedded watermark information. After tampering, the watermark information in the image is well preserved. However, the new watermark generated by the tampered image will be different from the original embedded watermark. The results of each method are displayed in Figure 12 and Table 6 gives their PSNR and SSIM values.

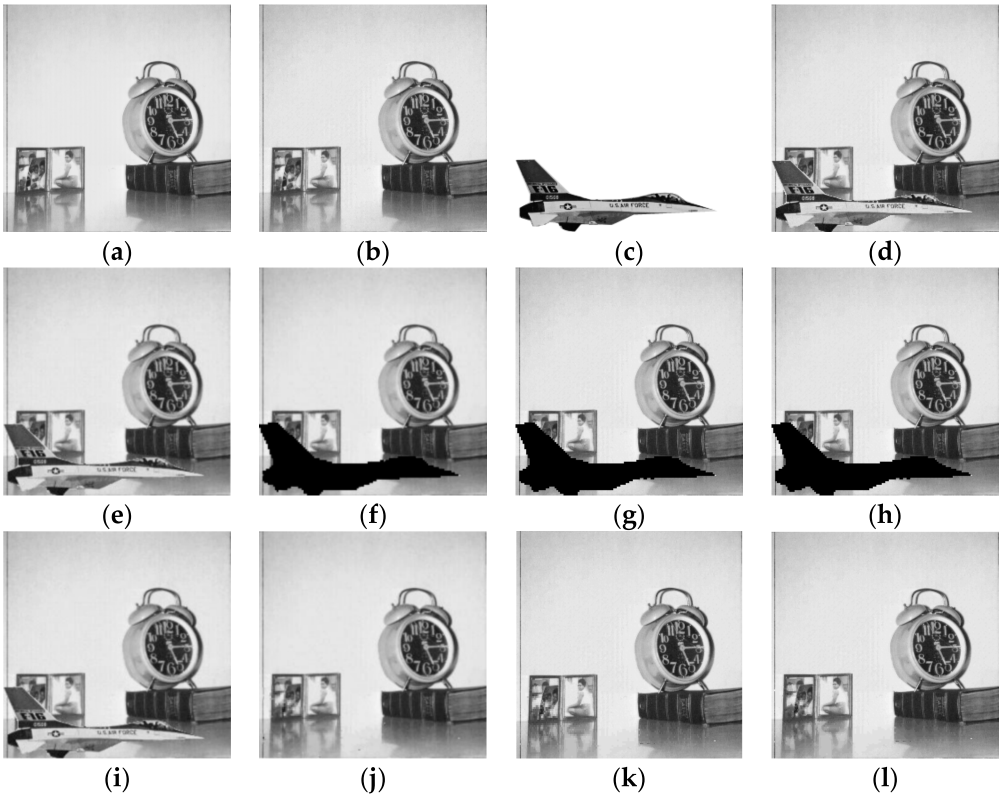

Figure 12. Tamper detection and recovery results of a content-only attack on the Clock image: (**a**) original image; (**b**) watermarked image; (**c**) tamper reference; (**d**) tampered image; (**e**) tamper detection result of [8]; (**f**) tamper detection result of [9]; (**g**) tamper detection result of [7]; (**h**) tamper detection result of the proposed method; (**i**) recovery result of [8]; (**j**) recovery result of [9]; (**k**) recovery result of [7]; (**l**) recovery result of the proposed method.

Table 6. Objective comparison of the recovery performances for the content-only attack on the Clock image.

Indexes	Tong et al. [8]	Chen et al. [9]	Wang et al. [7]	Proposed
PSNR (dB)	19.48	36.36	39.53	39.35
SSIM	0.7968	0.9010	0.9925	0.9930

In Figure 12c, the airplane stems from the original image Airplane and the 5 MSBs of its information are inserted into the watermarked image Clock. If the authentication watermark is completely irrelevant to the image content, then this type of attack cannot be detected and located, as shown by the detection results in Figure 12e. Therefore, the three methods shown in Figure 12f–h all generate watermarks from the image itself. In Figure 12k–l, Figure 12k eliminates several of the black noises in Figure 12i through its neighborhood adjustment, while Figure 12l has a gap in the dark vertical line on the left due to the filtering operation. The background line in the tampered area interferes with the median filtering operation, making it unable to apply its full advantages. Combining the results of the deletion attack and the content-only attack, we can conclude that while the proposed method improves the subjective quality of the restored image, it is unsuitable for tampered regions containing details within a single pixel or small area.

4.2.6. Large Area Attack

This tamper type is used primarily to test the performance of the filtering operation. The location and recovery results are shown in Figure 13.

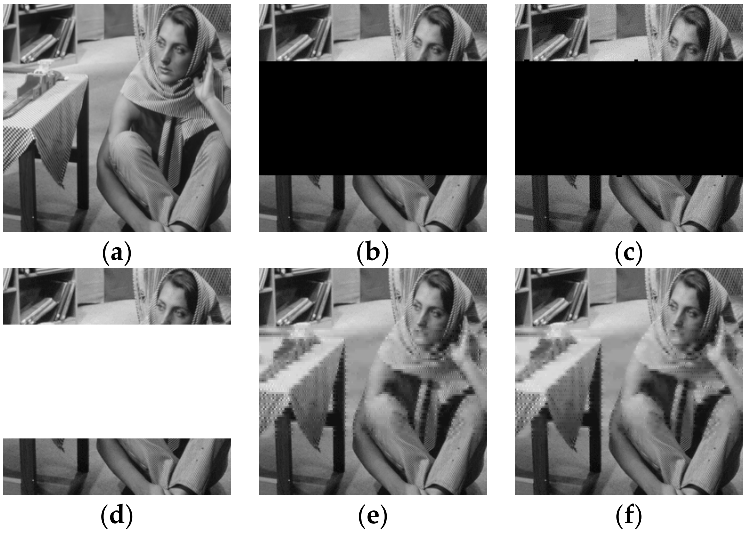

Figure 13. Tamper detection and recovery results of a large area attack on the Barbara image: (**a**) watermarked image; (**b**) tampered result of [7]; (**c**) tampered result of the proposed method; (**d**) tampered image; (**e**) recovery result of [7] (Peak signal-to-noise ratio (PSNR) = 27.34 dB); (**f**) recovery result of the proposed method (PSNR = 27.40 dB).

Due to the implementation of chaotic mapping, even though the original image is largely tampered, most of the affected pixels can still be detected and recovered by the watermark information extracted from the corresponding blocks. Moreover, the neighborhood adjustment is used to further recover the tampered area from the neighboring pixel values. This approach generates a fully restored image but with more distortion. As shown in Figure 13, the proposed method with edge detection can identify all tampered areas. However, just as in the previous attacks, many misjudgments are present. Figure 13e shows the obvious confusions in the restored region, which is exactly what we want to improve. After the filtering operation, the final result of the proposed method possesses better quality as reflected by the PSNR and SSIM values displayed in parentheses in the corresponding figure captions.

As the enlarged images in Figure 14 show, due to the median filtering, the woman's chin, hand and right leg are clearer and contain less noise. Meanwhile, the texture information on the scarf demonstrates that the threshold adjustment can offset the detail loss caused by filtering.

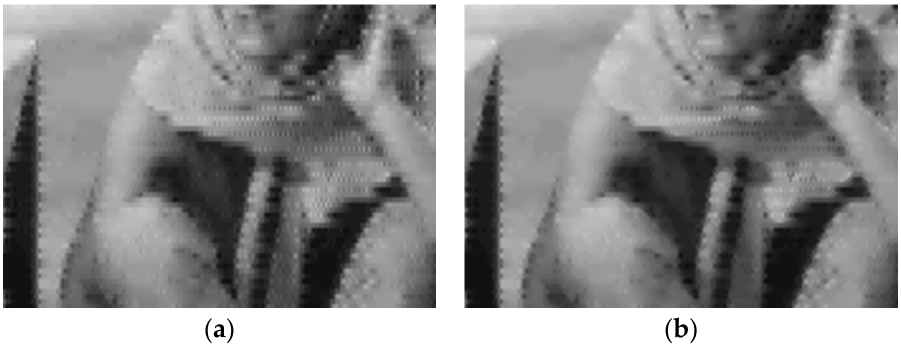

Figure 14. Enlarged recovery details of a large area attack on the Barbara image: (**a**) the method of [7]; (**b**) the proposed method.

4.3. Selection of Threshold in Filtering Operation

The filtering operation of the proposed method uses an adjustable threshold set to a pixel value. Different values lead to different qualities in the final results. After many tests with different pixels, the optimal threshold value was found to be the range from 110 to 150. Figure 15 clarifies the relationships of the pixel threshold value to the recovery quality in different tampering experiments, where each image name reflects its relative tampering type. In Figure 15, the horizontal ordinate represents the pixel threshold, which ranges from 110 to 150 at a step size of 5. On the vertical axis, we label the highest PSNR values both among the methods [7–9] and under different thresholds. As shown in Figure 15, the optimal pixel threshold varies among different images; thus, it should be selected according to different tests. In addition, the PSNR values after the threshold adjustment are close to or even better than those of the previous methods. This result demonstrates that the proposed method can achieve a great visual effect without losing the objective quality of an image.

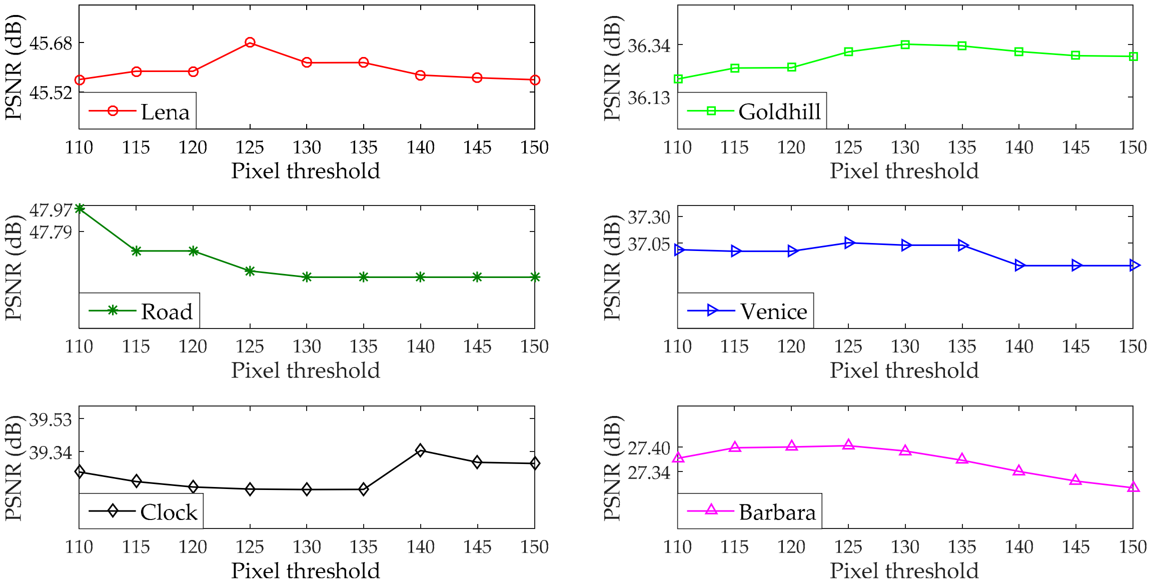

Figure 15. Relationships of pixel threshold to the PSNR value in different experiments.

5. Conclusions

In this paper, to achieve a better subjective quality of a restored image, we proposed an improved method that uses edge detection and SVD for tamper detection and median filtering during image recovery. With the SVD and edge detection characteristics applied, the watermark block is classified to ensure it contains sufficient texture and edge information. After a three-level tampering detection process, the tampered region can be located accurately. Using the inverse process of the watermark embedding method and the filtering operation, the recovery result is finally obtained. Under the effect of the median filter and proper threshold selection, the proposed method objectively achieves high quality. The performance of the proposed method is tested on eight classic images under six different types of tampering attacks and its results compared with the results of the methods in References [7–9]. The experimental results show that the tampered region can be accurately located but has a slightly higher false alarm rate than do other methods due to the sensitivity of edge detection. Using the adopted filtering operation, the restored result has a better visual effect without losing its objective quality. However, these operations are not suitable for images where the tampered regions contain details represented in a single pixel or by small pixel areas. In this case, the better subjective quality is obtained only at the cost of eliminating such details. In future research, we will further explore the effects of different filters on image recovery quality and discuss a more automated method for selecting the threshold.

Author Contributions: X.X., C.W. and M.L. conceived the algorithm and designed the experiments; X.X. and M.L. performed the experiments; X.X., C.W. and M.L. analyzed the results; X.X. drew the block diagrams; X.X. and M.L. drafted the manuscript; X.X., C.W. and M.L. revised the manuscript. C.W. supervised the project. All authors read and approved the final manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 61702303); the Shandong Provincial Natural Science Foundation, China (No. ZR2017MF020); and the 13th student research training program (SRTP) at Shandong University (Weihai), China.

Acknowledgments: The authors thank Xiaoyan Yu, Mingtong Zhang, Zhi Zhang and Qianwen Li for their help in revising this paper. The authors also gratefully acknowledge the technical assistance of DL850E ScopeCorder administrated by Xiaoli Wang.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Singh, D.; Singh, S.K. DCT based efficient fragile watermarking scheme for image authentication and restoration. *Multimed. Tools Appl.* **2017**, *76*, 953–977. [CrossRef]
2. Shivani, S.; Singh, D.; Agarwal, S. DCT based approach for tampered image detection and recovery using block wise fragile watermarking scheme. In Proceedings of the 6th Iberian Conference on Pattern Recognition and Image Analysis, Funchal, Portugal, 5–7 June 2013; Lecture Notes in Computer Science. Springer: Berlin, Germany, 2013; Volume 7887, pp. 640–647.
3. Roy, S.; Pal, A.K. A blind DCT based color watermarking algorithm for embedding multiple watermarks. *AEU Int. J. Electron. Commun.* **2017**, *72*, 149–161. [CrossRef]
4. El'arbi, M.; Amar, C.B. Image authentication algorithm with recovery capabilities based on neural networks in the DCT domain. *IET Image Process.* **2014**, *8*, 619–626. [CrossRef]
5. Molina-Garcia, J.; Reyes-Reyes, R.; Ponomaryov, V.; Cruz-Ramos, C. Watermarking algorithm for authentication and self-recovery of tampered images using DWT. In Proceedings of the 9th International Kharkiv Symposium on Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves, Kharkiv, Ukraine, 20–24 June 2016; pp. 1–4.
6. Al-Otum, H.M. Semi-fragile watermarking for grayscale image authentication and tamper detection based on an adjusted expanded-bit multiscale quantization-based technique. *J. Vis. Commun. Image Represent.* **2014**, *25*, 1064–1081. [CrossRef]
7. Wang, C.Y.; Zhang, H.; Zhou, X. A self-recovery fragile image watermarking with variable watermark capacity. *Appl. Sci.* **2018**, *8*, 548. [CrossRef]
8. Tong, X.J.; Liu, Y.; Zhang, M.; Chen, Y. A novel chaos-based fragile watermarking for image tampering detection and self-recovery. *Signal Process. Image Commun.* **2013**, *28*, 301–308. [CrossRef]
9. Chen, F.; He, H.J.; Tai, H.M.; Wang, H.X. Chaos-based self-embedding fragile watermarking with flexible watermark payload. *Multimed. Tools Appl.* **2014**, *72*, 41–56. [CrossRef]
10. Dhole, V.S.; Patil, N.N. Self embedding fragile watermarking for image tampering detection and image recovery using self recovery blocks. In Proceedings of the International Conference on Computing Communication Control and Automation, Pune, India, 26–27 February 2015; pp. 752–757.
11. Dadkhah, S.; Abd Manaf, A.; Hori, Y.; Hassanien, A.E.; Sadeghi, S. An effective SVD-based image tampering detection and self-recovery using active watermarking. *Signal Process. Image Commun.* **2014**, *29*, 1197–1210. [CrossRef]
12. Zhang, H.; Wang, C.Y.; Zhou, X. Fragile watermarking for image authentication using the characteristic of SVD. *Algorithms* **2017**, *10*, 27. [CrossRef]
13. Hsu, C.S.; Tu, S.F. Image tamper detection and recovery using adaptive embedding rules. *Measurement* **2016**, *88*, 287–296. [CrossRef]
14. Sarreshtedari, S.; Akhaee, M.A.; Abbasfar, A. Source–channel coding-based watermarking for self-embedding of JPEG images. *Signal Process. Image Commun.* **2018**, *62*, 106–116. [CrossRef]
15. Fan, M.Q.; Wang, H.X. An enhanced fragile watermarking scheme to digital image protection and self-recovery. *Signal Process. Image Commun.* **2018**, *66*, 19–29. [CrossRef]
16. Lu, C.S.; Liao, H.Y.M. Multipurpose watermarking for image authentication and protection. *IEEE Trans. Image Process.* **2001**, *10*, 1579–1592. [PubMed]
17. Lee, T.Y.; Lin, S.D. Dual watermark for image tamper detection and recovery. *Pattern Recognit.* **2008**, *41*, 3497–3506. [CrossRef]

18. Qian, Z.X.; Feng, G.R.; Ren, Y.L. Fragile watermarking for color image recovery based on color filter array interpolation. In Proceedings of the 11th International Conference on Web-Age Information Management, Jiuzhaigou, China, 15–17 July 2010; Lecture Notes in Computer Science. Springer: Berlin, Germany, 2010; Volume 6184, pp. 537–543.
19. Chetan, K.R.; Shivananda, N. A new fragile watermarking approach for tamper detection and recovery of document images. In Proceedings of the 3rd International Conference on Advances in Computing, Communications and Informatics, New Delhi, India, 24–27 September 2014; pp. 1494–1498.
20. Yu, X.Y.; Wang, C.Y.; Zhou, X. Review on semi-fragile watermarking algorithms for content authentication of digital images. *Future Internet* **2017**, *9*, 56.
21. Canny, J. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *8*, 679–698. [CrossRef] [PubMed]
22. CVG-UGR Image Database. Available online: http://decsai.ugr.es/cvg/dbimagenes/ (accessed on 26 July 2019).
23. Shi, H.; Wang, X.H.; Li, M.C. Secure variable-capacity self-recovery watermarking scheme. *Multimed. Tools Appl.* **2017**, *76*, 6941–6972. [CrossRef]