IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 16, 2020, accepted June 18, 2020, date of publication June 25, 2020, date of current version October 16, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3004841*

# A Novel Zero-Watermarking Scheme Based on Variable Parameter Chaotic Mapping in NSPD-DCT Domain

**RUI WANG[1], HAN SHAOCHENG[2], PENG ZHANG[3], MENG YUE[3], ZHENG CHENG[2], AND YUJIN ZHANG[4], (Member, IEEE)**

[1]College of Science, Civil Aviation University of China, Tianjin 300300, China
[2]Basic Experimental Center, Civil Aviation University of China, Tianjin 300300, China
[3]College of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China
[4]School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China

Corresponding author: Rui Wang (r-wang@cauc.edu.cn)

This work was supported by the Tianjin Education Committee Research Project under Grant 2018KJ246.

**ABSTRACT** In this paper, a novel image zero-watermarking scheme against rotation attacks is proposed based on nonsubsampled pyramid decomposition (NSPD) and discrete cosine transform (DCT). It utilizes the intrinsic characteristics of NSPD and DCT to extract the robust feature of an image as the original zero-watermark. To increase the security of the proposed scheme, a variable parameter chaotic mapping (VPCM) is designed for the processes of watermark encryption and robust feature extraction. Firstly, the host gray-scale image is decomposed by NSPD, and the low-frequency sub-band image is divided into non-overlapping blocks. After the blocks are transformed by DCT, the signs of the first AC coefficients from all the blocks are used to construct a binary feature image. Then an exclusive-or operation is performed between the binary feature image and the encrypted watermark image to obtain the verification zero-watermark image. Furthermore, a method against arbitrary rotation attacks is employed to improve the robustness of the scheme against geometric attacks. The experimental results demonstrate that the proposed scheme is highly robust against various image processing attacks such as filtering, JPEG compression, scaling, translation, rotation and Checkmark attacks.

**INDEX TERMS** Discrete cosine transform, nonsubsampled pyramid decomposition, rotation attacks, variable parameter chaotic mapping, zero-watermarking.

## I. INTRODUCTION

With the rapid development of computer technology and the Internet, digital media such as image, audio and video, can be more easily tampered and distributed than ever before. Therefore, the protection of multimedia content is currently a serious issue. Digital watermarking has been proposed as a potential solution to address this problem. It embeds some special secret information into host data and the information can be retrieved to identify the ownership of these data when necessary [1]–[4]. Digital watermarking has been widely used in many applications, including copyright protection, authentication, transaction tracking, and broadcast monitoring [5], [6]. The characteristics of an effective watermarking scheme include imperceptibility, robustness, security, and embedding capacity [7]–[10]. Imperceptibility means that the original host data and watermarked data can not be differentiated by the human visual system [8]. Robustness refers to the ability of the watermarking scheme to extract the embedded watermark under various attacks [9]. Security ensures that the watermarking scheme is difficult to be cracked, and it depends mainly on the secret keys used in the process of watermark extraction [10]. Embedding capacity denotes the amount of watermark bits.

Depending on the various properties, the watermarking algorithms can be categorized in various ways [8]. According to the required information of original data in the process of watermark extraction, watermarking schemes are categorized into blind, non-blind and semi-blind algorithms [11]. Bind algorithms do not require the original data when extracting the watermark, while non-bind watermarking algorithms require the complete information of original data. On the

The associate editor coordinating the review of this manuscript and approving it for publication was SK Hafizul Islam.

other hand, semi-blind watermarking algorithms only require the partial information of the original data to detect the watermark. According to their applications and purposes, watermarking algorithms can be categorized into fragile and robust watermarking algorithms [12]. Fragile algorithms [13] are used for image authentication and tamper detection, while robust watermarking algorithms are used for copyright protection. According to the domain in which the watermark will be embedded, watermarking algorithms can be categorized into spatial domain algorithms and transform domain algorithms [11]. In spatial domain algorithms [14]–[16], watermarks are usually embedded into the host images by modifying their pixels directly. In transform domain watermarking algorithms, one or more transforms are applied to the host images, then the watermarks are embedded by modifying the transformed coefficients in the frequency domain. Spatial domain algorithms usually have lower computational complexity than the transform domain watermarking algorithms. However, due to the limitations of watermarking in the spatial domain involving visualization and robustness [17], most image watermarking algorithms use the transforms such as Cosine transform [18]–[22], Fourier transform [23], [24], Wavelet transform [25], [26], Contourlet transform [27], [28], and Shearlet transform [29]–[31].

In the above-mentioned traditional watermarking algorithms, whether the original watermarks are embedded in the spatial domain or transform domain, the host images are distorted to some extent. Therefore, there is inevitably a contradiction between imperceptibility and robustness of the watermark. The traditional watermarking methods are usually limited for medical images [32]–[34] and remote-sensing images [35] which require low distortion and high resolution. In this case, the zero-watermarking was firstly proposed by Wen *et al.* [36], which has attracted wide attention of researchers in recent years. Zero-watermarking extracts the intrinsic features of original data without modifying the data. Hence it can address the contradiction between imperceptibility and robustness of the traditional watermarking methods.

Several zero-watermarking algorithms have been proposed in the past 15 years. In [36], Wen *et al.* constructed a zero-watermark by computing high-order cumulants. Their method is robust to common image processing attacks and slight scaling rotation, but it is fragile to large-scale rotations and requires a lot of time to compute the high-order cumulants. Ye [37] proposed a zero-watermarking algorithm based on singular value decomposition (SVD) and discrete cosine transform (DCT). In their algorithm, the host image is firstly divided into non-overlapping blocks. Then, the zero-watermark sequence is derived by comparing the numerical relationship between the direct current (DC) coefficients of two adjacent blocks after every block is performed with SVD and DCT. The algorithm gave out a new idea of zero-watermarking by combining matrix decomposition and traditional transforms, but the attacks used in the experiments were very slight. Based on [36] and [37], Zhang *et al.* [38] presented three improved zero-watermarking schemes respectively named by DC-RE, CU-SVD and CU-SVD-RE. These schemes are more robust than the algorithms in [36] and [37], but the security of these schemes needs to be improved. Rani and Raman [39] also extracted the robust features of an image as the zero-watermark based on DCT and SVD. The scheme divided the host image into overlapping blocks of size $8 \times 8$ and performed DCT and SVD to every block. The zero-watermark is obtained by comparing the relationship between the two largest singular values, which are selected out randomly using four pseudo-random number sequences. The robustness and security of the algorithm are acceptable for the protection of images to some extent. However, dividing the host image into overlapping blocks generates a large number of image blocks, which increased the computational complexity of the algorithm. Ta Minhn Thanh *et al.* [40] extracted the robust feature as the master share from the host image by combining the QR decomposition and one-dimensional (1D) DCT. In their scheme, the host images are RGB images, and two zero-watermarking construction schemes were proposed utilizing the luminance Y-components of the host images. The Y-component is also divided into non-overlapping blocks, and the master share is generated by comparing the DC coefficients of two adjacent blocks after all the blocks have undergone QR decomposition and 1D-DCT.

Lin *et al.* [41] proposed a spatial domain zero-watermarking scheme based on generalized Arnold transform (GAT) and spread spectrum and despreading (SSD) techniques. In their method, the GAT is used to scramble the original watermark for security. Then, a binary feature matrix is obtained from the original host image in quantitative embedding rules. Finally, the zero-watermark is generated by performing an exclusive-or operation between the scrambled watermark and the feature matrix. Considering that better robustness of watermarking can be achieved in transform domain, some multi-scale transforms such as Contourlet [42] and Shearlet [43], [44], regarded as extension of the wavelet transform, have been applied to the zero-watermarkings. In [42], a robust zero-watermarking algorithm was proposed based on contourlet transform (CT) and DCT for medical images, Logistic Map is used to encrypt the original watermark to ensure the security of it. The experimental results show that the method is effective to against common and slightly geometric attacks. Nonsubsampled shearlet transform (NSST) is a new and very important multi-scale and multi-direction analysis tool that can provide nearly optimal approximation properties for image representation [45]. Han *et al.* [43] proposed a zero-watermarking method based on NSST and LU decomposition. In their method, after the host image is decomposed by NSST, a random sub-image of the low-frequency sub-band image is divided into non-overlapping blocks. Then, every block undergoes LU decomposition to obtain the matrix U. The final zero-watermark is generated by comparing the numerical relationship between the sum of the first row elements from every matrix U and the mean of all the sums. In [44], a robust zero-watermarking

scheme was proposed that employed multi-resolution and multi-scale representation characteristics of NSST to analyze the direction features of the host image. In the algorithm, a sub-band image is firstly selected as the embedding position by calculating the direction feature information intensity. Then, the sub-band image is divided into non-overlapping blocks. The zero-watermark is constructed by comparing the 2-norms of every block with a threshold.

Although the above-mentioned zero-watermarking algorithms can resist conventional image processing operations, they are almost fragile to the geometrical distortions such as rotation and translation attacks. Gao and Jiang [46] developed a zero-watermarking algorithm against geometric attacks based on the Bessel-Fourier moment. In their algorithm, the image normalization is applied to the host image. Then, the magnitudes of the Bessel-Fourier moments of the normalized image are computed to construct a binary feature image. The final verification image is generated by performing an exclusive-or operation between the binary feature image and the original watermark image. Wang *et al.* [47] introduced a zero-watermarking algorithm against geometric attacks based on polar complex exponential transform (PCET) and logistic mapping (LM). Their algorithm computes the PCET of the original gray-scale image and randomly selects PCET coefficients based on LM. Then, the magnitudes of the PCET are computed to construct the binary feature image. These above-mentioned zero-watermarking methods are summarized by Table 1.

**TABLE 1. The performance of the related zero-watermarking methods.**

| Methods | Related techiques | Embedding domain | Robustness | Rotation attacks | Complexity |
|---|---|---|---|---|---|
| [36] | Cumulants | — | × | × | × |
| [37] | DCT | Single | × | × | √ |
| [38] | DCT, SVD | Single | × | × | × |
| [39] | DCT, SVD | Single | √ | × | × |
| [40] | QR, DCT | Single | × | × | √ |
| [41] | GAT, SSD | — | × | × | √ |
| [42] | CT, DCT | Hybrid | √ | × | √ |
| [43] | LU, NSST | Single | √ | × | √ |
| [44] | NSST, 2-norm | Single | √ | × | √ |
| [46] | Moments | — | √ | √ | × |
| [47] | PCET, LM | Single | √ | √ | × |

In this paper, we propose a robust zero-watermarking scheme based on NSPD-DCT and VPCM. Our experimental results demonstrate that the proposed algorithm can effectively resist the conventional image processing operations and several geometrical distortions. The contributions of this paper are the following:

(1) A novel invariant feature of an image is found by combining the shift-invariant characteristic of NSPD and the robustness of the signs of some DCT coefficients against various attacks.
(2) In order to improve the security of the proposed scheme, a VPCM system is designed for two processes: watermark encryption and robust feature extraction.
(3) To get better robustness of watermarking against geometrical attacks, a simple method for estimating the rotation angles of images is utilized by computing the normalized cross-correlation (NC) between the scaled down host image and the scaled down image to be detected.

The rest of this paper is organized as follows. Section II presents the preliminaries of the Arnold transform, DCT, and NSPD. This section also briefly describes VPCM. Section III explains the proposed zero-watermarking scheme. In this section, the zero-watermark generation and extraction are illustrated in different subsections. Section IV provides the results and discussions. Finally, Section V presents the conclusions drawn from this work.

## II. PRELIMINARIES

### A. ARNOLD TRANSFORM

Arnold transform is an image scrambling method also regarded as a two-dimensional (2D) chaotic mapping [19]. In many watermarking schemes, Arnold transform is applied to the watermark image to expand the robustness of the proposed algorithm and provide extra security for the embedded watermark [8]. The positions of different pixels in an image can be changed after the image undergoes the Arnold transform. The 2D Arnold transform is defined as:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \left[ \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \right] \mod P, \tag{1}$$

where $(x, y)$ are the location coordinates of the original image pixels, $x, y \in \{0, 1, 2, \ldots, P-1\}$, $(x', y')$ are the location coordinates of the scrambled image pixels, and $P$ is the size of the watermark. When all of the pixels of the image are transformed, the scrambled image is obtained. Arnold scrambling is periodic. Considering that $T$ is the transform period, the image can return to the original state after $T$ iterations [48]. The number of iterations for image scrambling and $T$ can be regarded as the private keys. Without the keys, the original image cannot be restored. In our method, the Arnold transform is also used to scramble the original watermark image.

### B. DISCRETE COSINE TRANSFORM

Discrete cosine transform has been used in the field of various images and signal processing, including in image coding and watermarking. The DCT transforms an image from the spatial domain into the frequency domain [21], [49]. DCT has an excellent energy compactness property, which is why it is used in the JPEG compression technique to separate and remove the insignificant high frequency components in images [50]. The forward 2D-DCT formula is defined as:

$$F(u, v) = c(u)c(v)\frac{2}{N} \cdot \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x, y) \cos\left(\frac{2x+1}{2N}u\pi\right) \cos\left(\frac{2y+1}{2N}v\pi\right) \tag{2}$$

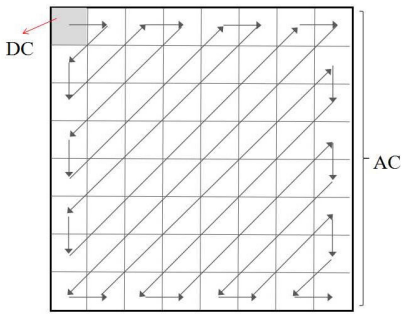



**FIGURE 1.** Zigzag scanning.

where $x, y, u, v = 0, 1, 2, \ldots, N-1$, and

$$c(u) = c(v) = \begin{cases} \frac{1}{\sqrt{2}} & u = 0, v = 0 \\ 1 & otherwise. \end{cases} \quad (3)$$

The inverse 2D-DCT formula is described as

$$f(x,y) = \frac{2}{N} \cdot \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} c(u)c(v)F(u,v) \cos\left(\frac{2x+1}{2N}u\pi\right) \cos\left(\frac{2y+1}{2N}v\pi\right), \quad (4)$$

where $f(x, y)$ is the gray value of a pixel, and $F(u, v)$ is the DCT coefficient. After this transformation, an image consists of one DC coefficient and multiple alternate current (AC) coefficients. The DC and AC coefficients are arranged in the Zigzag scanning order, as shown in Fig. 1. The DC coefficient locating at the top-left corner of the coefficient matrix is a large positive number. It represents most of the energy of the image. Except for the DC coefficient, all the other coefficients are the AC coefficients. The AC coefficients represent the characteristics of different frequency bands, including low, middle and high frequency bands. Moreover, the AC coefficients are either positive or negative numbers, and their amplitudes are less than the DC coefficient [22], [51].

### C. NONSUBSAMPLED PYRAMID DECOMPOSITION

Nonsubsampled pyramid decomposition was proposed by Cunha *et al.* when they designed the nonsubsampled contourlet transform (NSCT) [52]. The nonsubsampled pyramid (NSP) is achieved by replacing the Laplacian pyramid with two-channel nonsubsampled 2D filter banks as shown in Fig.2 (a). The decomposition filters $H_0(z)$, $H_1(z)$ and the composition filters $G_0(z)$, $G_1(z)$ satisfy the Bezout identical equation $H_0(z)G_0(z)+H_1(z)G_1(z) = 1$. The equation ensures the NSP filter banks satisfying the perfect reconstruction condition, where $H_0(z)$ is a low-pass filter and $H_1(z)$ is a high-pass filter [53].

The frequency expansion of NSPD is conceptually similar to the one-dimensional nonsubsampled wavelet transform (NSWT) computed with the àtrous algorithm. It has $J+1$ redundancy, where $J$ denotes the number of decomposition stages [52]. The sketch map of

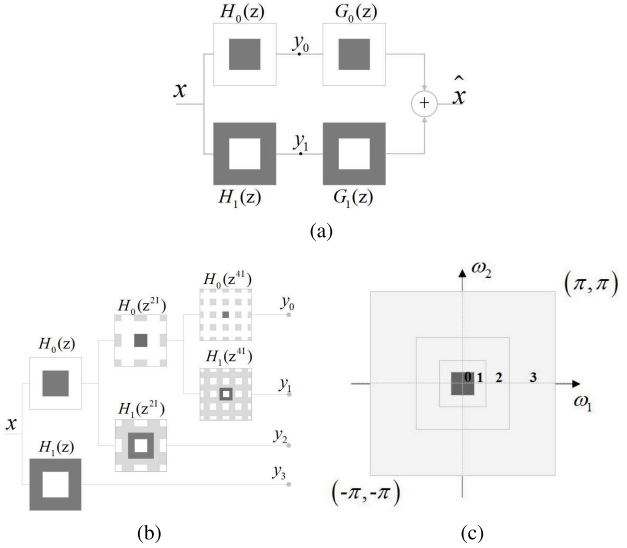



**FIGURE 2.** NSPD: (a) The two-channel NSP filter banks, (b) A 3-stages NSPD, (c) The sub-bands on the 2D frequency plane.

NSPD and the relevant frequency division with $J = 3$ stages are shown in Fig. 2 (b) and (c) respectively. The ideal pass-band support of the low-pass filter at the $j$-th stage is the region $[(-\pi/2^j), (\pi/2^j)]^2$. Accordingly, the ideal support of the equivalent high-pass filter which is the complement of the low-pass filter is the region $[(-\pi/2^{j-1}), (\pi/2^{j-1})]^2 \backslash [(-\pi/2^j), (\pi/2^j)]^2$ [52]. By contrast, the 2D àtrous algorithm has $3J+1$ redundancy. Besides having nice multi-resolution characteristics, NSPD can overcome the weakness of traditional DWT without shift-invariant characteristic and has less redundancy then NSWT [54]. Fig.3 shows an example for NSPD of Man image at $J = 4$. In Fig. 3, (a) is an original image, (b) is a low-pass image, and (c)-(e) are the band-pass images from coarse to fine stages.

### D. VARIABLE PARAMETER CHAOTIC MAPPING

Chaotic mappings are widely used in most watermarking schemes to encrypt the original watermark. In this section, we propose a variable parameter chaotic mapping denoted by VPCM based on two typical chaotic systems: logistic mapping and piecewise linear chaotic mapping (PWLCM). The LM widely used in many watermarking algorithms [10], [47], [55] is defined as:

$$x_{n+1} = \mu x_n(1 - x_n), \quad (5)$$

where $\mu \in [0, 4]$ is the control parameter, and $x_n$ is the chaos sequence of the map. Furthermore, because PWLCM has the properties, including uniform distribution, good ergodicity, confusion, and diffusion, it is also utilized for encrypting the watermarks [56]–[58]. PWLCM can be described as:

$$x_{n+1} = F(x_n, p) = \begin{cases} x_n/p, & x_n \in [0, p] \\ (x_n - p)/(0.5 - p), & x_n \in [p, 0.5] \\ F(1 - x_n, p), & x_n \in [0.5, 1], \end{cases} \quad (6)$$

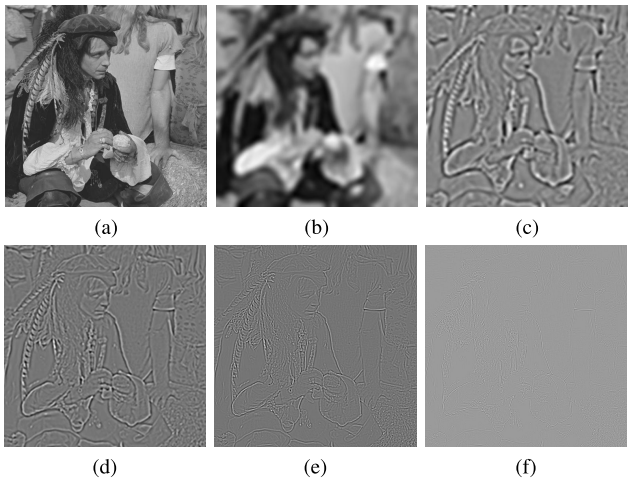



**FIGURE 3.** Original Man image and its NSPD: (a) Original image, (b) Low-pass image, (c) Band-pass image at the 1-th stage, (d) Band-pass image at the 2-th stage, (e) Band-pass image at the 3-th stage, (f) Band-pass image at the 4-th stage.

where $x_n \in (0, 1)$ and the control parameter $p \in (0, 0.5)$.

To enhance the robustness and security of the proposed scheme, we replace the control parameter $p$ of the above PWLCM system with a variable parameter $p_n$. $p_n$ depends on the random sequence $x_n$ generated by the LM. To satisfy $p_n \in (0, 0.5)$, we make $p_n$ to be a third of $x_n$. The proposed VPCM system can be presented as follows:

$$\begin{cases} x_{n+1} = \mu x_n(1 - x_n), p_n = x_n/3 \\ y_{n+1} = \begin{cases} y_n/p_n, & y_n \in [0, p_n] \\ (y_n - p_n)/(0.5 - p_n), & y_n \in [p_n, 0.5] \\ F(1 - y_n, p_n), & y_n \in [0.5, 1]. \end{cases} \end{cases} \tag{7}$$

The parameter $\mu$ and the initial state values of VPCM are regarded as private keys. In our watermarking scheme, we utilize the VPCM system to generate two random sequences to be used for watermark image encryption and robust feature extraction.

## III. PROPOSED ZERO-WATERMARKING SCHEME BASED ON NSPD-DCT AND VPCM

In this section, we introduce a new robust feature extraction method for images. we performed experiments to show that the extracted feature of an image is highly robust to some image processing attacks. Finally, we explain the proposed zero-watermarking scheme based on NSPD-DCT and VPCM in detail. The zero-watermarking scheme mainly includes two procedures: zero-watermark generation and zero-watermark extraction [47]. Zero-watermark is usually generated from the important feature of an image [46]. In our scheme, after the host image undergoes NSPD and DCT, the signs of the first AC coefficients from the DCT blocks are used to construct the zero-watermark. Since the zero-watermarking does not need inverse transformation, the process of zero-watermark extraction is similar to that of zero-watermark generation. Furthermore, in order to improve the performance of the proposed scheme against rotation attacks, we introduce a method of image rotation correction which is applied before the zero-watermark extraction.

### A. ROBUST FEATURE EXTRACTION IN NSPD-DCT DOMAIN

Li *et al.* proposed a zero-watermarking algorithm for protecting medical images in the DCT domain [59]. In their algorithm, they pointed out that the signs of some DCT coefficients remain unchanged against some strong geometric attacks. Their method applies DCT to the host image, then the sign sequence of the low-frequency coefficients is extracted as the feature vector to construct the zero-watermark. The method has been extended to the DWT-DCT hybrid domain in [60]. However, the embedding capacities of watermarks in [59], [60] were limited.

Drawing inspiration from the methods in [59], [60], we here extract robust feature vector based on NSPD and DCT. To describe our method in detail, we performed several experiments as follows. Suppose the original test Man image shown in Fig. 3 (a) is denoted by $H$. We firstly performed the NSPD with $J = 5$ stages on $H$ to obtain the low-frequency subband image $H_L$. Then $H_L$ was divided into non-overlapping blocks of size $8 \times 8$, and every block was transformed by DCT. Finally we selected out eight coefficient blocks denoted by $D_i (i = 1, 2, \ldots, 8)$ for test. The positions of them are shown in Fig. 4.

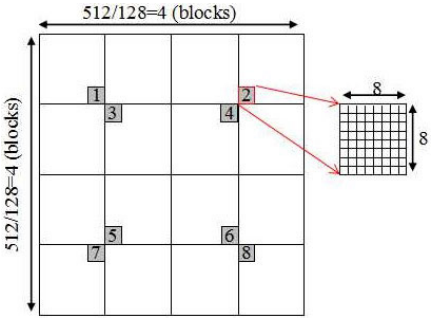



**FIGURE 4.** The positions of the selected blocks for test.

In these experiments, we tested the robustness of eight AC coefficients from each selected block. According to the Zigzag scanning order, the positions of the eight AC coefficients are $(1, 2)$, $(2, 1)$, $(3, 1)$, $(2, 2)$, $(1, 3)$, $(1, 4)$, $(2, 3)$ and $(3, 2)$. The host Man image was distorted via seven image processing attacks, such as (a) Gaussian noise (0.1), (b) Speckle noise (0.2), (c) Median filtering ($7 \times 7$), (d) JPEG compression (5%), (e) Scaling (1/8), (f)Translation (turn down by 5 pixels), (g) Rotation (2°). Table 2 shows the DCT coefficients of $D_1$ for the Man image under the above-mentioned different attacks. The coefficients shown in bold in Table 2 denote that their signs changed after the Man image was subjected to some attacks. The Peak Signal-to-Noise Ratio (PSNR) values of the Man image under different attacks are also shown in the Table 2. As seen from Table 2, all the attacks distort the Man image to have low PSNR values,

**TABLE 2.** The DCT coefficients of $D_1$ for the Man image under different attacks.

| Attacks | PSNR | AC coefficients of matrix $D_1$ for Man image | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $D_1(1,2)$ | $D_1(2,1)$ | $D_1(3,1)$ | $D_1(2,2)$ | $D_1(1,3)$ | $D_1(1,4)$ | $D_1(2,3)$ | $D_1(3,2)$ |
| No | Inf | 25.8756 | −6.9381 | 1.2278 | 0.0225 | −1.2684 | 2.5842 | 0.0915 | 0.0463 |
| (a) | 11.3686 | 19.7750 | −8.0307 | 1.0601 | 0.0785 | −1.1178 | 1.9533 | 0.0863 | 0.0961 |
| (b) | 13.7787 | 26.4954 | −8.6092 | 1.3148 | 0.0551 | −1.3379 | 2.6470 | 0.0996 | 0.0233 |
| (c) | 26.9180 | 25.9576 | −8.7022 | 1.5331 | 0.1864 | −1.1778 | 2.6016 | 0.1177 | 0.0912 |
| (d) | 25.7431 | 28.8509 | −5.7183 | 1.3863 | −0.4789 | −1.4461 | 2.7829 | 0.0657 | 0.0274 |
| (e) | 23.8608 | 25.3355 | −6.3649 | 1.3903 | −0.1494 | −1.2811 | 2.5380 | 0.0848 | 0.0484 |
| (f) | 17.7057 | 28.0652 | −0.3070 | 2.0964 | −0.0548 | 0.0642 | 2.8152 | 0.0918 | 0.0291 |
| (g) | 18.6144 | 25.8909 | 1.7006 | 2.0119 | −0.0545 | −1.1336 | 2.5676 | 0.0275 | −0.0743 |

**TABLE 3.** The constructed sign sequences for the Man image under JPEG compression attacks.

| AC coefficients | No attacks | Quality factors of JPEG compression | | | | | Numbers of changed signs |
|---|---|---|---|---|---|---|---|
| | | 1% | 2% | 3% | 4% | 5% | |
| $D_i(1,2)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,1)$ | 00001111 | 00001111 | 00001111 | 00001111 | 00001111 | 00001111 | 0 |
| $D_i(3,1)$ | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 0 |
| $D_i(2,2)$ | 11010011 | 11010001 | 11010001 | 11010001 | 01010011 | 01010011 | 5 |
| $D_i(1,3)$ | 01110001 | 01110001 | 01110001 | 01110001 | 01110101 | 01110101 | 2 |
| $D_i(1,4)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,3)$ | 11000000 | 11000000 | 11000000 | 11000000 | 11000000 | 10000000 | 1 |
| $D_i(3,2)$ | 10110000 | 10110000 | 10110000 | 10110000 | 10110000 | 10110000 | 0 |

**TABLE 4.** The constructed sign sequences for the Man image under scaling attacks.

| AC coefficients | No attacks | Scaling factors | | | | | Numbers of changed signs |
|---|---|---|---|---|---|---|---|
| | | 1/32 | 1/16 | 1/8 | 1/4 | 1/2 | |
| $D_i(1,2)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,1)$ | 00001111 | 00001011 | 00001111 | 00001111 | 00001111 | 00001111 | 1 |
| $D_i(3,1)$ | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 0 |
| $D_i(2,2)$ | 11010011 | 01010000 | 0101011 | 01010011 | 01010011 | 11010011 | 6 |
| $D_i(1,3)$ | 01110001 | 00110001 | 01110101 | 01110001 | 01110101 | 01110101 | 2 |
| $D_i(1,4)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,3)$ | 11000000 | 11110000 | 11100000 | 10000000 | 11000000 | 10000000 | 4 |
| $D_i(3,2)$ | 10110000 | 00010000 | 10110000 | 10110000 | 10110000 | 10110000 | 2 |

and the DCT coefficients of $D_1$ are changed greatly, but their signs almost did not change.

Moreover, we extracted the signs of the AC coefficients at the same positions for the eight selected blocks to construct a sign sequence of 8 bits, such as $'10101111'$, where $'1'$ represents a positive or zero coefficient, and $'0'$ represents a negative coefficient. Eight sign sequences can be obtained according to the eight different positions of the selected AC coefficients in every block. Tables 3 to 6 show the sign sequences of the Man image and the number of changed signs after different attacks. The elements of the sign sequences shown in bold also denote that the signs of the coefficients have changed after the test image was subjected to some attacks. As seen in Tables 3 to 6, the signs of the selected AC coefficients are robust to the above attacks. We can utilize the polarities of the signs to construct the binary feature vector as the original zero-watermark. In our watermarking scheme, we used the sign of the first AC coefficient at $(1, 2)$ of every block to construct the original zero-watermark because its robustness is the best to various attacks for all the selected coefficients in our experiments. Furthermore, if every block of size $8 \times 8$ can generate at least one bit of zero-watermark, the embedding capacity of the watermark will be increased compared with the methods in [59], [60].

### B. ZERO-WATERMARK GENERATION

Let $I$ be the original host image of size $N \times N$ and $W$ be the original binary watermark image of size $M \times M$, where $N/M = 2^l$. The block diagram of the zero-watermark generation is illustrated in Fig. 5 and described as follows.

*Step 1:* Original watermark image scrambling

The original watermark $W$ is scrambled by the Arnold transform to obtain $W_1$ with the private keys $k_1$ and $T$,

$$W_1 = \text{Arnold}(W). \tag{8}$$

**TABLE 5.** The constructed sign sequences for the Man image under rotation attacks.

| AC coefficients | No attacks | Angles of rotation | | | | | Numbers of changed signs |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | |
| $D_i(1,2)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 101011101 | 1 |
| $D_i(2,1)$ | 00001111 | 00001111 | 00001111 | 10000111 | 10000111 | 10000101 | 7 |
| $D_i(3,1)$ | 11100000 | 11100000 | 11100000 | 10100000 | 10100000 | 10100000 | 3 |
| $D_i(2,2)$ | 11010011 | 11010001 | 0101000 | 11010000 | 11010000 | 11000000 | 11 |
| $D_i(1,3)$ | 01110001 | 01110001 | 11110001 | 10110101 | 10100001 | 10100001 | 10 |
| $D_i(1,4)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10100101 | 2 |
| $D_i(2,3)$ | 11000000 | 11000000 | 11010000 | 01010000 | 01010000 | 0111010 | 9 |
| $D_i(3,2)$ | 10110000 | 11110000 | 11110000 | 11110100 | 11110101 | 11110101 | 10 |

**TABLE 6.** The constructed sign sequences for the Man image under translation attacks.

| AC coefficients | No attacks | Distances of translation | | | | | Numbers of changed signs |
|---|---|---|---|---|---|---|---|
| | | 2 | 4 | 6 | 8 | 10 | |
| $D_i(1,2)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,1)$ | 00001111 | 00001111 | 00001111 | 00001111 | 00001011 | 00001011 | 2 |
| $D_i(3,1)$ | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 11100000 | 0 |
| $D_i(2,2)$ | 11010011 | 11010001 | 1101000 | 11010000 | 11010000 | 11010000 | 9 |
| $D_i(1,3)$ | 01110001 | 01110001 | 01010001 | 01010001 | 01010001 | 01010001 | 4 |
| $D_i(1,4)$ | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 10101111 | 0 |
| $D_i(2,3)$ | 11000000 | 10100000 | 10100000 | 10100000 | 10100000 | 0010100 | 12 |
| $D_i(3,2)$ | 10110000 | 10110000 | 10110000 | 10110000 | 00110100 | 00100100 | 5 |

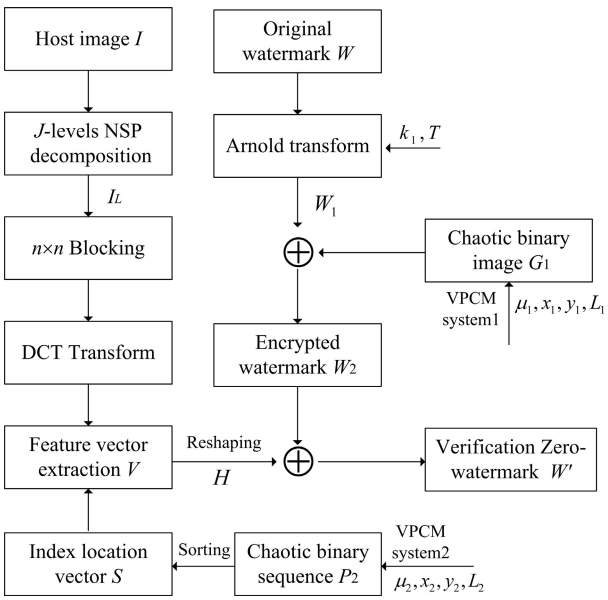



**FIGURE 5.** Block diagram of the zero-watermark generation process.

*Step 2:* Watermark encryption

A random sequence $Y_1 = \{y_n | n = 1, 2, \ldots, M^2 + L_1\}$ is generated by the VPCM system. Then, a new sequence $P_1 = \{y_n | n = L_1 + 1, L_1 + 2, \ldots, M^2 + L_1\}$ is defined. $P_1$ is converted into a binary image denoted by $G_1$. Then an exclusive-or operation is performed between $G_1$ and $W_1$ to obtain the encrypted watermark $W_2$,

$$W_2 = \mathrm{XOR}(W_1, G_1). \tag{9}$$

*Step 3:* Robust feature extraction

NSPD is performed on the original host image $I$. Then, the low-frequency sub-band image $I_L$ is divided into non-overlapping blocks of size $n_1 \times n_1$. Every block is decomposed by 2D-DCT, and the first AC coefficients in the order of Zigzag scanning for all blocks are selected out to construct the 1D feature vector $U(k)$.

*Step 4:* Zero-watermark vector construction

As Step 2., the random sequences $Y_2 = \{z_n | n = 1, 2, \ldots, M^2 + L_2\}$ and $P_2 = \{z_n | n = L_2 + 1, L_2 + 2, \ldots, M^2 + L_2\}$ are generated. Then, the elements of $P_2$ are arranged in ascending order using the equation $[P_3, S] = \mathrm{sort}(P_2)$ to obtain a new sequence $P_3$ and the index location vector $S$. The zero-watermark vector $V(k)$ is constructed by judging whether every element of $U(k)$ is larger than 0 or not as the order of $S$,

$$V(k) = \begin{cases} 1 & \text{if } U(S(k)) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad k = 1, 2, \ldots, M^2. \tag{10}$$

*Step 5:* Generation of the verification zero-watermark image

$V(k)$ is reshaped into a binary image denoted by $Q$. Then, an exclusive-or operation is performed between the encrypted watermark $W_2$ and $Q$ to generate the verification zero-watermark image $W'$,

$$W' = \mathrm{XOR}(W_2, Q). \tag{11}$$

### C. ZERO-WATERMARK EXTRACTION

The procedure of zero-watermark extraction consists of two processes: image rotation correction and final

zero-watermark extraction. After doing a abundant of experiments, we found that the significant differences can exist among the NC values, which are used to assess the similarities between an image and its rotated images with different angles. In addition, when the original image and its rotated images are scaled down to the smaller size simultaneously, the differences among above NC values are not cut down. So this experimental conclusion can be utilized to estimate the angle of the rotated image. Based on the conclusion, the method of image rotation correction proposed in our previous work [61] is used again in this scheme. The block diagram of the zero-watermark extraction process is shown in Fig. 6. Suppose $I_1$ is the image to be verified, which may have undergone some attacks, the detailed steps of zero-watermark extraction are described as follows.

*Step 1:* Image rotation detection and correction

(a) To decide whether $I_1$ undergoes rotation attacks or not, the means of four blocks of size $b_1 \times b_1$, which are located in the four corners of $I_1$, are calculated. If $I_1$ has undergone rotation attacks, the processes from (b) to (e) are performed. Otherwise, Step 2 is performed immediately.

(b) In order to reduce the computational cost, $I$ and $I_1$ of size $N \times N$ are scaled down into two new images $A$ and $A_1$ of size $b_2 \times b_2$ respectively.

(c) $A$ is rotated by $10m$ degrees sequentially, where $m$ is a positive integer from 1 to 36. The NC value (described in Section 4) is calculated between every rotated $A$ and $A_1$ for each $m$. The number $m_1$ is corresponds to the largest NC value.

(d) $A$ is rotated by $10m_1 + e$ degrees sequentially, where $e$ is a positive integer from 1 to 10. The NC value is calculated between every rotated $A$ and $A_1$ in turns again for each $e$.

(e) $I_1$ is rotated by the angle corresponding to the largest NC value in (d) in the opposite direction to obtain the corrected image $I_2$.

*Step 2:* Robust vector extraction

NSPD is performed on the image $I_2$. Note that $I_2$ is the same with $I_1$ if $I_1$ does not undergo rotation attacks. The low-frequency sub-band image $I_{2L}$ is divided into non-overlapping blocks of size $n_1 \times n_1$, and the 1D feature vector $U'(k)$ is constructed by the process of zero-watermark generation.

*Step 3:* Zero-watermark vector construction

The random sequence $P_3$ and its index location vector $S$ are generated by the same secret keys. Then, the zero-watermark vector $V'(k)$ is obtained as follows:

$$V'(k) = \begin{cases} 1 & \text{if } U'(S(k)) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad k = 1, 2, \ldots, M^2. \quad (12)$$

*Step 4:* Final watermark image extraction

$V'(k)$ is reshaped into a binary image $Q'$, and the chaotic binary image $G_1$ is also generated by the saved private keys. Then, an exclusive-or operations and the inverse Arnold transform are performed to extract the final watermark $W^*$ by

$$W^* = \text{Arnold}^{-1}(\text{XOR}(\text{XOR}(Q', W'), G_1)). \quad (13)$$

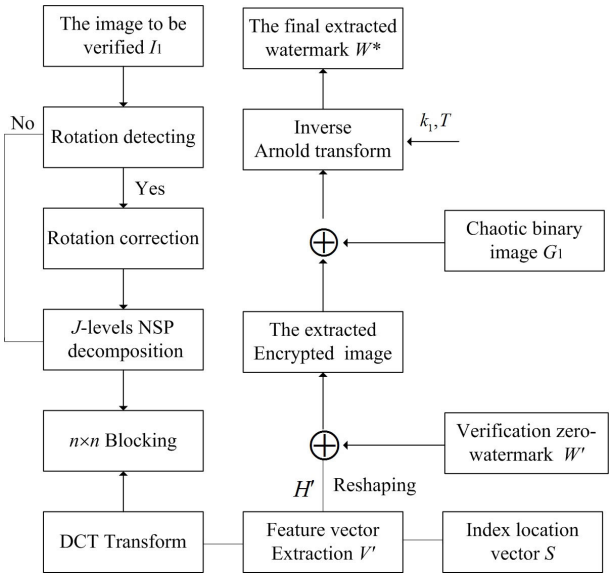



**FIGURE 6.** Block diagram of the zero-watermark extraction process.

## IV. EXPERIMENTAL RESULTS

To verify the validity and feasibility of the proposed scheme, we performed a series of experiments in this section. Eight well-known images of size $512 \times 512$ from the image database of USC-SIPI [62] were taken as the host images, i.e., Man, Tiffany, Elain, Lena, Goldhill, Boat, Bridge and Peppers, as shown in Fig. 7. Four binary watermark images with different sizes were used as the original watermark images, as shown in Fig. 8. The host images were decomposed by NSPD with $J = 5$ stages and 'maxflat' filter. The low-frequency sub-band image of size $512 \times 512$ was divided into non-overlapping blocks according to the size of watermark. For the Arnold transform, the scrambling time $k_1$ was 12 and $T = 24$. The initial state values and control parameters of the VPCM system for the two random sequences were respectively assigned as $\mu_1 = 3.89999$, $x_1 = 0.65555$, $y_1 = 0.10000$, $\mu_2 = 3.98880$, $x_2 = 0.45550$, $y_2 = 0.22220$, and $L_1 = L_2 = 500$. The parameters $b_1$ and $b_2$ in the process of image rotation correction were 2 and 20, respectively.

The objective criteria PSNR, NC and Bit error rate(BER) were used in our experiments. The PSNR was used to evaluate the quality of the host images under different attacks, defined as [63]:

$$\text{PSNR} = 10 \log_{10} \frac{255^2 \times N^2}{\sum_{x=1}^{N} \sum_{y=1}^{N} \left(I(x, y) - I'(x, y)\right)^2}. \quad (14)$$

where $I$ and $I'$ represent the host and the attacked images of size $N \times N$, respectively.

**TABLE 7.** The similarity test of original zero-watermarks for different host images.

| Images | Man | Tiffany | Elain | Lena | Goldhill | Boat | Bridge | Peppers |
|---|---|---|---|---|---|---|---|---|
| Man | 1 | 0.4627 | 0.5602 | 0.4832 | 0.5305 | 0.4459 | 0.5107 | 0.4858 |
| Tiffany | 0.4627 | 1 | 0.4636 | 0.4355 | 0.5004 | 0.4726 | 0.4822 | 0.3834 |
| Elain | 0.5602 | 0.4636 | 1 | 0.5339 | 0.5627 | 0.5080 | 0.5701 | 0.5016 |
| Lena | 0.4832 | 0.4355 | 0.5339 | 1 | 0.4993 | 0.5182 | 0.5257 | 0.4344 |
| Goldhill | 0.5305 | 0.5004 | 0.5627 | 0.4993 | 1 | 0.5417 | 0.5336 | 0.4526 |
| Boat | 0.4459 | 0.4726 | 0.5080 | 0.5182 | 0.5417 | 1 | 0.5830 | 0.4312 |
| Bridge | 0.5107 | 0.4822 | 0.5701 | 0.5257 | 0.5336 | 0.5830 | 1 | 0.4639 |
| Peppers | 0.3834 | 0.3834 | 0.5016 | 0.4344 | 0.4526 | 0.4312 | 0.4639 | 1 |

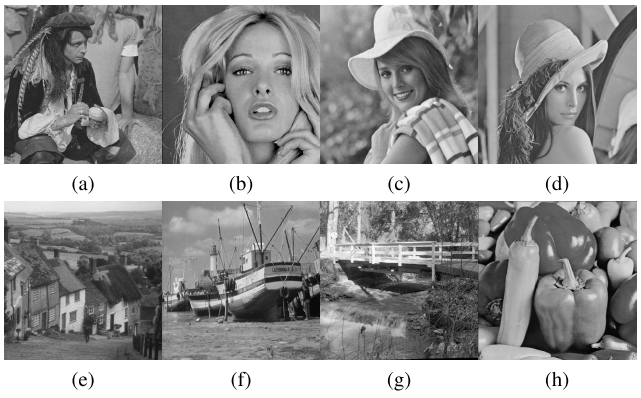



**FIGURE 7.** The host images used in the experiments: (a) Man, (b) Tiffany, (c) Elain, (d) Lena, (e) Goldhill, (f) Boat, (g) Bridge, (h) Peppers.

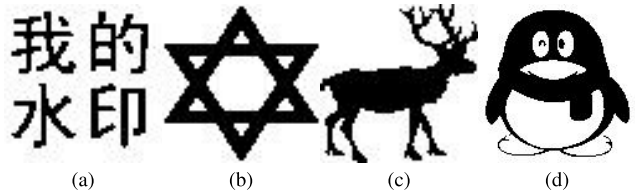



**FIGURE 8.** The original watermarks used in the experiments: (a) Watermark 1 of size 64 × 64, (b) Watermark 2 of size 64 × 64, (c) Watermark 3 of size 64 × 64, (d) Watermark 4 of size 128 × 128.

The NC and BER were used to evaluate the robustness of the proposed scheme. They are respectively defined as [3], [47]:

$$\mathrm{NC}(W, W^*) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M} W(i,j)W^*(i,j)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{M} W^2(i,j)}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{M} W^{*2}(i,j)}}, \quad (15)$$

$$\mathrm{BER} = \frac{B}{M \times M} \times 100. \quad (16)$$

where $W$ and $W^*$ represent the original and the extracted watermarks of size $M \times M$, respectively. $B$ is the sum of inaccurate bits.

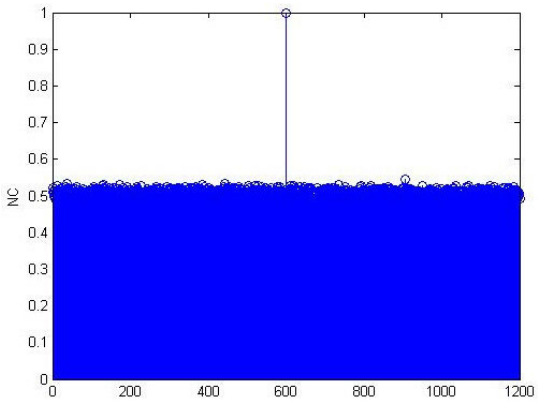



**FIGURE 9.** Similarities between the original zero-watermark and the random binary matrices.

### A. UNIQUENESS VERIFICATION OF ZERO-WATERMARKS

In this subsection, we discuss the uniqueness verification of zero-watermarks similar to the method used in [37]. The zero-watermark constructed from an image should only be relevant to this image. In other words, the zero-watermarks constructed from different images should be different so that the zero-watermark constructed from an image just can uniquely identify this image [43]. The similarity between two different zero-watermark binary images constructed from different images is shown in Table 7. Table 7 shows that the maximum NC value of two different images is 0.5701, which is far less than 1. Therefore, the zero-watermark constructed from an image can be effectively distinguished from the zero-watermarks constructed from other different images.

To further verify the uniqueness of the zero-watermark constructed from an image in the proposed scheme, we produced 1200 random binary sequences from a uniform distribution and reshaped them into 2D images. Then, we calculated their respective NC values with the zero-watermark binary image constructed from the Man image. The results are shown in Fig. 9, where the 600th sequence corresponds to the zero-watermark of the Man image. Fig. 9 shows that all the NC values fluctuate around 0.5 in an extremely small range except the right zero-watermark of the Man image. Hence, it can be concluded that the zero-watermark constructed from an image in our proposed method can not be generated randomly and the image can be identified uniquely.

**TABLE 8.** The PSNR/NC values for different images under Gaussian noise attacks.

| Host image | Variances of Gaussian noise (Mean 0) | | | | |
|---|---|---|---|---|---|
| | 0.01 | 0.05 | 0.1 | 0.15 | 0.2 |
| Man | 20.07/0.9908 | 13.68/0.9817 | 11.38/0.9703 | 10.24/0.9649 | 9.53/0.9635 |
| Tiffany | 20.04/0.9886 | 13.54/0.9776 | 11.26/0.9628 | 10.14/0.9608 | 9.50/0.9516 |
| Elain | 20.07/0.9898 | 13.62/0.9764 | 11.31/0.9696 | 10.19/0.9656 | 9.51/0.9514 |
| Lena | 20.02/0.9901 | 13.56/0.9750 | 11.24/0.9684 | 10.15/0.9591 | 9.49/0.9598 |
| Goldhill | 20.12/0.9741 | 13.68/0.9458 | 11.40/0.9167 | 10.25/0.9082 | 9.55/0.9090 |
| Boat | 20.11/0.9770 | 13.61/0.9413 | 11.29/0.9283 | 10.18/0.9140 | 9.52/0.8972 |
| Bridge | 20.18/0.9873 | 13.79/0.9665 | 11.45/0.9547 | 10.31/0.9468 | 9.58/0.9336 |
| Peppers | 20.16/0.9913 | 13.76/0.9804 | 11.44/0.9744 | 10.27/0.9679 | 9.56/0.9624 |

**TABLE 9.** The PSNR/NC values for different images under Salt and Peppers noise attacks.

| Host image | Noise densities | | | | |
|---|---|---|---|---|---|
| | 0.01 | 0.05 | 0.1 | 0.15 | 0.2 |
| Man | 25.18/0.9946 | 18.37/0.9869 | 15.37/0.9826 | 13.66/0.9759 | 12.40/0.9749 |
| Tiffany | 25.55/0.9944 | 18.56/0.9847 | 15.53/0.9791 | 13.75/0.9743 | 12.53/0.9714 |
| Elain | 25.42/0.9941 | 18.48/0.9887 | 15.47/0.9825 | 13.73/0.9744 | 12.44/0.9726 |
| Lena | 25.63/0.9957 | 18.59/0.9879 | 15.57/0.9815 | 13.84/0.9770 | 12.58/0.9643 |
| Goldhill | 25.28/0.9852 | 18.32/0.9680 | 15.32/0.9584 | 13.62/0.9459 | 12.43/0.9377 |
| Boat | 25.42/0.9885 | 18.45/0.9675 | 15.44/0.9583 | 13.69/0.9413 | 12.48/0.9359 |
| Bridge | 25.24/0.9925 | 18.28/0.9796 | 15.26/0.9726 | 13.48/0.9651 | 12.27/0.9532 |
| Peppers | 25.24/0.9954 | 18.27/0.9908 | 15.33/0.9865 | 13.51/0.9784 | 12.26/0.9755 |

**TABLE 10.** The PSNR/NC values for different images under Median filtering attacks.

| Host image | Sizes of Median filter | | | | |
|---|---|---|---|---|---|
| | $3 \times 3$ | $4 \times 4$ | $5 \times 5$ | $6 \times 6$ | $7 \times 7$ |
| Man | 32.24/0.9978 | 27.50/0.9802 | 28.64/0.9927 | 26.30/0.9793 | 26.92/0.9871 |
| Tiffany | 31.12/0.9956 | 26.96/0.9823 | 28.04/0.9913 | 25.88/0.9789 | 26.56/0.9898 |
| Elain | 32.91/0.9989 | 29.38/0.9897 | 31.72/0.9973 | 28.96/0.9881 | 30.89/0.9957 |
| Lena | 36.36/0.9986 | 29.65/0.9850 | 32.24/0.9964 | 28.59/0.9850 | 30.11/0.9951 |
| Goldhill | 31.71/0.9956 | 28.30/0.9855 | 28.83/0.9893 | 27.08/0.9821 | 27.27/0.9852 |
| Boat | 30.95/0.9940 | 26.51/0.9796 | 27.19/0.9841 | 25.04/0.9705 | 25.19/0.9733 |
| Bridge | 26.73/0.9940 | 23.97/0.9812 | 24.02/0.9855 | 22.76/0.9739 | 22.68/0.9796 |
| Peppers | 35.12/0.9992 | 28.53/0.9847 | 32.35/0.9973 | 27.85/0.9836 | 30.36/0.9933 |

### *B. ROBUSTNESS TESTS OF THE PROPOSED SCHEME*

In this subsection, we present the experimental results for the robustness test of the proposed scheme under various attacks. The watermark image 1 was firstly used in this part. Several conventional image processing operations (adding noise, filtering, and JPEG compression) and the geometrical distortions (scaling, cropping, and rotation) were performed on the original host images. The robustness of the proposed scheme under the above-mentioned attacks was evaluated as follows.

#### 1) ADDITION OF NOISE

Gaussian noise, Salt and Peppers noise are usually used to test the robustness of watermarking algorithms. For the Gaussian noise, we varied the amount of noise with respect to its variance while fixing its mean. For the Salt and Peppers noise, we degraded the quality by changing the noise densities. Table 8 shows the results of adding the Gaussian white noise with a mean of 0 and variances of 0.01 to 0.2 to the host images in terms of PSNR and NC values. Meanwhile, Table 9 shows the results of adding the Salt and Peppers noise with densities of 0.01 to 0.2 to the host images. As can be seen in the Tables, even though the test images are degraded greatly by the two types of noise with various noise variances or densities, the high NC values can be obtained. This means that the proposed scheme is robust against noise attacks.

#### 2) FILTERING ATTACKS

Test images were filtered using Median filters and Wiener filters with size of $3 \times 3$, $4 \times 4$, $5 \times 5$, $6 \times 6$ and $7 \times 7$. Table 10 and 11 show the results in terms of PSNR and NC values for the Median and Wiener filters, respectively. The Tables show that the proposed scheme is highly robust to the Median filtering and Wiener filtering attacks. The NC values are very near to 1 in almost all of the cases, which demonstrates that the proposed scheme has

**TABLE 11.** The PSNR/NC values for different images under Wiener filtering attacks.

| Host image | Sizes of Wiener filter | | | | |
|---|---|---|---|---|---|
| | $3 \times 3$ | $4 \times 4$ | $5 \times 5$ | $6 \times 6$ | $7 \times 7$ |
| Man | 35.14/0.9992 | 32.54/0.9948 | 31.72/0.9967 | 30.53/0.9929 | 29.92/0.9952 |
| Tiffany | 34.76/0.9987 | 32.85/0.9946 | 32.41/0.9978 | 31.42/0.9938 | 31.10/0.9959 |
| Elain | 33.66/0.9998 | 32.63/0.9935 | 32.53/0.9983 | 32.12/0.9935 | 31.96/0.9978 |
| Lena | 38.85/0.9992 | 36.19/0.9946 | 35.74/0.9989 | 34.29/0.9948 | 33.73/0.9983 |
| Goldhill | 34.12/0.9983 | 32.30/0.9933 | 31.35/0.9949 | 30.38/0.9906 | 29.67/0.9929 |
| Boat | 34.10/0.9981 | 32.02/0.9930 | 31.18/0.9965 | 30.01/0.9917 | 29.29/0.9940 |
| Bridge | 29.54/0.9986 | 27.44/0.9922 | 26.51/0.9956 | 25.57/0.9905 | 24.97/0.9924 |
| Peppers | 36.62/0.9995 | 34.85/0.9957 | 34.69/0.9986 | 33.55/0.9948 | 33.18/0.9959 |

**TABLE 12.** The PSNR/NC values for different images under JPEG compression attacks.

| Host image | Quality factors(%) | | | | |
|---|---|---|---|---|---|
| | 1 | 5 | 10 | 20 | 40 |
| Man | 23.16/0.9960 | 25.74/0.9773 | 28.27/0.9929 | 30.55/0.9964 | 32.64/0.9981 |
| Tiffany | 23.49/0.9611 | 25.97/0.9727 | 28.40/0.9890 | 30.35/0.9960 | 32.25/0.9986 |
| Elain | 24.88/0.9592 | 27.50/0.9831 | 29.94/0.9906 | 31.60/0.9970 | 32.73/0.9976 |
| Lena | 24.78/0.9518 | 27.80/0.9744 | 30.99/0.9885 | 33.63/0.9952 | 35.87/0.9975 |
| Goldhill | 23.74/0.9243 | 26.16/0.9458 | 28.65/0.9746 | 30.87/0.9887 | 32.90/0.9897 |
| Boat | 22.99/0.9095 | 25.56/0.9515 | 28.13/0.9747 | 30.49/0.9882 | 32.75/0.9948 |
| Bridge | 21.08/0.9578 | 23.06/0.9780 | 25.13/0.9892 | 27.01/0.9952 | 28.85/0.9981 |
| Peppers | 24.29/0.9699 | 27.18/0.9830 | 30.13/0.9909 | 32.42/0.9964 | 34.20/0.9975 |

**TABLE 13.** The PSNR/NC values for different images under cropping attacks attacks.

| Host image | Types of cropping | | | | |
|---|---|---|---|---|---|
| | Top-left $64 \times 64$ | Top-left $128 \times 128$ | Center $128 \times 128$ | Edge $64 \times 512$ | Edge $128 \times 512$ |
| Man | 24.47/0.9933 | 16.79/0.9770 | 15.45/0.9741 | 12.81/0.9335 | 9.88/0.9077 |
| Tiffany | 21.46/0.9925 | 16.03/0.9749 | 17.99/0.9607 | 12.33/0.9347 | 10.53/0.8951 |
| Elain | 28.08/0.9921 | 20.31/0.9783 | 17.29/0.9686 | 15.65/0.9553 | 12.52/0.9023 |
| Lena | 26.74/0.9978 | 18.47/0.9709 | 18.33/0.9699 | 15.72/0.9813 | 11.56/0.8970 |
| Goldhill | 29.67/0.9901 | 19.25/0.9741 | 15.90/0.9699 | 14.07/0.9560 | 11.03/0.8961 |
| Boat | 24.73/0.9930 | 19.71/0.9792 | 18.34/0.9652 | 15.96/0.9553 | 12.70/0.9069 |
| Bridge | 25.84/0.9976 | 19.84/0.9791 | 15.42/0.9627 | 16.21/0.9683 | 12.83/0.9210 |
| Peppers | 24.39/0.9938 | 16.87/0.9713 | 17.77/0.9774 | 13.01/0.9311 | 10.76/0.8838 |

superior performance against these two types of filtering attacks.

### 3) JPEG COMPRESSION

Any effective watermarking algorithm should be robust against compression attacks. We test our watermarking algorithm against JPEG compression, which is one of the most widely used common compression attacks. JPEG compression attacks with different quality factors of 1%, 5%, 10%, 20% and 40% were applied to the host images. The results in terms of PSNR and NC are summarized in Table 12. Table 12 shows that even though the quality factor is 1%, high NC values can be achieved. Therefore, the proposed scheme is highly robust against the JPEG compression attacks because the low-frequency components of the host images are used to construct the zero-watermarks.

### 4) CROPPING ATTACKS

To test the robustness against cropping attacks, the host images were cropped with different cropping windows. We used cropping window sizes of $64 \times 64$, $128 \times 128$, $64 \times 512$, and $128 \times 512$, and the cropping windows were black. The results in terms of PSNR and NC values are shown in Table 13. As seen in Table 13, even though the cropping window size is $128 \times 512$, all the NC values are larger than 0.88 for the different test images. This verifies that the proposed scheme is robust against cropping attacks.

### 5) SCALING ATTACKS

For scaling attacks, we resized the original images by multiplying by a scaling factor, and then scaled back the images to their original sizes. If the scaling factor is less than 1, the images are scaled down, otherwise they are scaled up.

**TABLE 14.** The PSNR/NC values for different images under scaling attacks attacks.

| Host image | Scaling factors | | | | |
|---|---|---|---|---|---|
| | 1/16 | 1/8 | 1/4 | 1/2 | 2 |
| Man | 21.48/0.9776 | 23.86/0.9927 | 26.85/0.9987 | 31.04/0.9998 | 41.03/0.9998 |
| Tiffany | 21.46/0.9925 | 16.03/0.9749 | 17.99/0.9607 | 12.33/0.9347 | 10.53/0.8951 |
| Elain | 23.03/0.9809 | 27.01/0.9960 | 30.48/0.9992 | 33.09/1.0000 | 39.68/0.9998 |
| Lena | 23.18/0.9829 | 26.35/0.9952 | 30.01/0.9995 | 35.09/0.9995 | 45.09/0.9994 |
| Goldhill | 22.47/0.9711 | 24.93/0.9892 | 27.65/0.9967 | 31.44/0.9992 | 41.06/0.9994 |
| Boat | 20.81/0.9679 | 22.83/0.9870 | 25.53/0.9971 | 29.94/0.9986 | 39.53/0.9998 |
| Bridge | 18.98/0.9759 | 20.85/0.9909 | 23.06/0.9979 | 26.50/0.9995 | 35.83/0.9998 |
| Peppers | 21.22/0.9749 | 24.58/0.9954 | 27.92/0.9992 | 31.74/0.9995 | 40.98/0.9998 |

**TABLE 15.** The PSNR/NC values for different images under translation attacks.

| Host image | Distances of translation from left to right (pixels) | | | | |
|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 |
| Man | 24.96/0.9783 | 21.47/0.9570 | 19.89/0.9402 | 18.86/0.9232 | 18.10/0.9060 |
| Tiffany | 24.76/0.9827 | 22.01/0.9665 | 20.79/0.9542 | 19.93/0.9389 | 19.21/0.9245 |
| Elain | 26.65/0.9734 | 23.71/0.9580 | 21.93/0.9440 | 20.34/0.9261 | 19.35/0.9113 |
| Lena | 27.14/0.9853 | 23.23/0.9713 | 21.30/0.9598 | 20.03/0.9475 | 19.09/0.9332 |
| Goldhill | 26.08/0.9838 | 23.05/0.9693 | 21.57/0.9548 | 20.38/0.9434 | 19.67/0.9296 |
| Boat | 23.40/0.9828 | 20.50/0.9651 | 19.18/0.9507 | 18.44/0.9329 | 17.92/0.9177 |
| Bridge | 22.17/0.9844 | 19.34/0.9699 | 18.19/0.9554 | 17.45/0.9419 | 16.88/0.9292 |
| Peppers | 26.79/0.9815 | 23.05/0.9624 | 20.81/0.9441 | 19.33/0.9281 | 18.24/0.9122 |

**TABLE 16.** The PSNR/NC values for different images under rotation attacks.

| Host image | Angles of rotation | | | | |
|---|---|---|---|---|---|
| | 5° | 10° | 20° | 40° | 90° |
| Man | 14.86/0.9633 | 12.88/0.9474 | 11.14/0.9307 | 9.89/0.9189 | 11.64/1.0000 |
| Tiffany | 15.56/0.9734 | 13.61/0.9664 | 12.03/0.9351 | 11.19/0.9182 | 13.13/1.0000 |
| Elain | 14.53/0.9574 | 12.87/0.9412 | 11.17/0.9299 | 9.77/0.9149 | 10.90/1.0000 |
| Lena | 15.33/0.9776 | 13.04/0.9646 | 11.66/0.9407 | 10.83/0.9125 | 12.42/1.0000 |
| Goldhill | 15.12/0.9715 | 13.10/0.9623 | 11.43/0.9372 | 10.38/0.9196 | 10.65/1.0000 |
| Boat | 23.40/0.9575 | 20.50/0.9419 | 19.18/0.9156 | 18.44/0.8974 | 17.92/1.0000 |
| Bridge | 13.57/0.9691 | 12.09/0.9520 | 10.70/0.9153 | 9.67/0.9139 | 10.33/1.0000 |
| Peppers | 13.56/0.9680 | 11.47/0.9498 | 9.99/0.9273 | 8.92/0.9176 | 10.48/1.0000 |

Table 14 shows the results in terms of PSNR and NC values for the eight test images under scaling attacks with scaling factors of 1/16 to 2. From the Table 14, it can be demonstrated that the proposed watermarking scheme shows strong resistance to the scaling attacks.

### 6) TRANSLATION ATTACKS

For translation attacks, we translate the host images from left to right by 1, 2, 3, 4 and 5 pixels in turns before performing watermark extraction. The experimental results are shown in Table 15. As seen in Table 15, the NC values of the extracted zero-watermarks for translation attacks are not larger than those for other attacks, such as the addition of noise, filtering, and JPEG compression, but the minimum NC value is larger than 0.91. This means that the proposed scheme can effectively resist to the translation attacks.

### 7) ROTATION ATTACKS

The host images were also tested for geometric attacks of rotation. The images were respectively rotated by 5°, 10°, 20°, 40° and 90° before performing watermark detection. The results in terms of PSNR and NC values are shown in Table 16. Table 16 shows that the proposed scheme can successfully resist rotation attacks. Furthermore, the proposed scheme performs best against rotation attacks when the rotation angle is 90°. In fact, the same result can be obtained if the rotation angle is 180° or 270° because of the proposed method of image rotation correction. Since the proposed scheme can resist the rotation, scaling, and translation (RST) attacks, it can be considered as an RST-invariant watermarking algorithm. From the experimental results shown in Tables 8 to 16, we can conclude that the proposed scheme is highly robust against conventional image processing operations and some geometrical distortions.

**TABLE 17.** The PSNR/NC values for Lena image under different attacks.

| Attacks | PNSR | Watermark 1 | | | Watermark 3 | Watermark 4 |
|---|---|---|---|---|---|---|
| | | $32 \times 32$ | $64 \times 64$ | $128 \times 128$ | $64 \times 64$ | $128 \times 128$ |
| Gaussian noise (0.05) | 13.5306 | 0.9679/0.0459 | 0.9703/0.0452 | 0.9695/0.0464 | 0.9659/0.0454 | 0.9636/0.0457 |
| Gaussian noise (0.15) | 10.1598 | 0.9652/0.0498 | 0.9553/0.0667 | 0.9625/0.0569 | 0.9601/0.0530 | 0.9570/0.0540 |
| Salt and Peppers noise (0.1) | 15.6000 | 0.9864/0.0195 | 0.9831/0.0259 | 0.9827/0.0265 | 0.9786/0.0286 | 0.9725/0.0346 |
| Salt and Peppers noise (0.2) | 12.5448 | 0.9740/0.0371 | 0.9725/0.0420 | 0.9649/0.0466 | 0.9621/0.0505 | 0.9641/0.0451 |
| Median filtering($5 \times 5$) | 32.2413 | 0.9664/0.0479 | 0.9584/0.0630 | 0.9616/0.0583 | 0.9525/0.0630 | 0.9535/0.0583 |
| Wiener filtering($5 \times 5$) | 35.7395 | 0.9986/0.0020 | 0.9989/0.0017 | 0.9983/0.0026 | 0.9987/0.0017 | 0.9980/0.0026 |
| JPEG compression (5%) | 27.7996 | 0.9623/0.0537 | 0.9518/0.0728 | 0.9505/0.0749 | 0.9450/0.0728 | 0.9400/0.0749 |
| JPEG compression (20%) | 33.6275 | 0.9959/0.0059 | 0.9952/0.0073 | 0.9954/0.0071 | 0.9945/0.0073 | 0.9944/0.0071 |
| Scaling (1/8) | 26.3543 | 0.9857/0.0205 | 0.9829/0.0261 | 0.9825/0.0269 | 0.9805/0.0261 | 0.9787/0.0269 |
| Cropping (center $128 \times 128$) | 18.3265 | 0.9678/0.0459 | 0.9722/0.0425 | 0.9699/0.0459 | 0.9682/0.0425 | 0.9634/0.0459 |
| Cropping (Edge $64 \times 512$) | 15.7225 | 0.8947/0.0146 | 0.8906/0.1614 | 0.8902/0.1619 | 0.8748/0.1614 | 0.86800.1619/ |
| Translation (5 pixels) | 19.0941 | 0.8661/0.1855 | 0.8676/0.1936 | 0.8672/0.1943 | 0.8500/0.1936 | 08414/0.1943 |
| Rotation(5°) | 15.3284 | 0.9748/0.0361 | 0.9776/0.0342 | 0.9769/0.0353 | 0.9744/0.0342 | 0.9719/0.0353 |
| Rotation(10°) | 13.0434 | 0.9665/0.0479 | 0.9646/0.0537 | 0.9647/0.0536 | 0.9596/0.0537 | 0.9572/0.0536 |
| Rotation(40°) | 10.8319 | 0.9109/0.1250 | 0.9125/0.1299 | 0.9103/0.1335 | 0.9005/0.1299 | 0.8919/0.1335 |
| Rotation(90°) | 12.4175 | 1.0000/0.0000 | 1.0000/0.0000 | 1.0000/0.0000 | 1.0000/0.0000 | 1.0000/0.0000 |

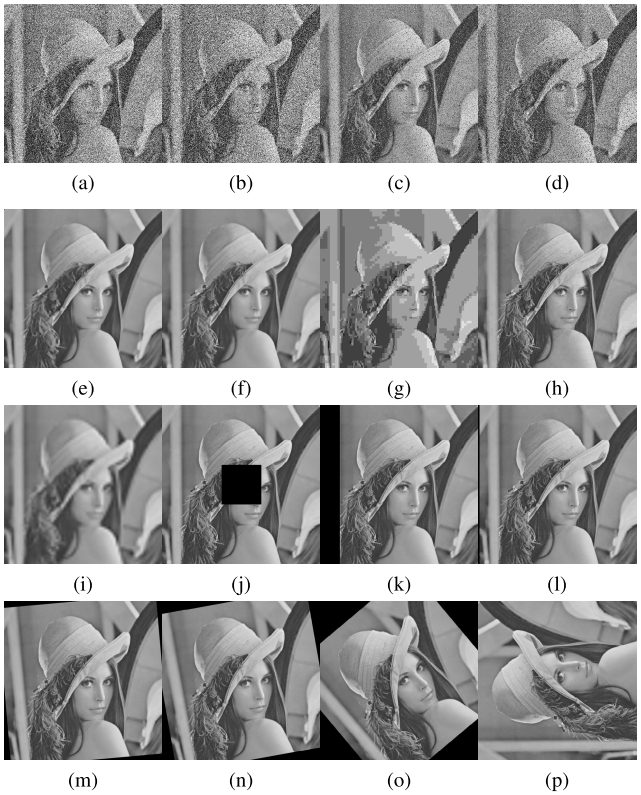



**FIGURE 10.** The attacked Lena image under different attacks (a) Gaussian noise (0.05), (b) Gaussian noise (0.15), (c) Salt and Peppers noise(0.1), (d) Salt and Peppers noise (0.2), (e) Median filtering ($5 \times 5$), (f) Wiener filtering ($5 \times 5$), (g) JPEG compression(5%), (h) JPEG compression (20%), (i) Scaling (1/8), (j) Cropping (center $128 \times 128$), (k) Cropping (Edge $64 \times 512$), (l) Translation (5 pixels), (m) Rotation (5°), (n) Rotation (10°), (o) Rotation(40°), (p) Rotation (90°).

To further demonstrate the robustness of the proposed scheme, more experiments for robustness test were completed. In this part, sixteen attacks were selected shown in Table 17, and Lena was used as host image. After above sixteen attacks, attacked Lena images are shown in Fig. 10.

**TABLE 18.** Robustness test of proposed scheme under Checkmark attacks.

| Checkmark attacks | PSNR | NC | BER |
|---|---|---|---|
| Soft thresh(2) | 33.9399 | 0.9968 | 0.0049 |
| Hard thresh (2) | 34.9442 | 0.9983 | 0.0027 |
| Dither (2) | 6.5338 | 0.9946 | 0.0083 |
| Dprorr (1) | 29.9750 | 0.9964 | 0.0056 |
| Linear (5) | 16.5825 | 0.9323 | 0.1016 |
| Midpoint (2) | 26.9431 | 0.9449 | 0.0078 |
| Nulineremove (2) | 35.3325 | 0.9960 | 0.0061 |
| Projective (2) | 19.9948 | 0.9557 | 0.0671 |
| Ratio (4) | 28.2486 | 0.9956 | 0.0068 |
| Rotation scale(12) | 19.3888 | 0.9122 | 0.1304 |
| Row col (12) | 31.7333 | 0.9927 | 0.0112 |
| Sample down up (4) | 33.2087 | 0.9944 | 0.0085 |
| Scale (3) | 36.0917 | 0.9949 | 0.0078 |
| Sharpening (1) | 23.3342 | 0.9818 | 0.0278 |
| Shearing (4) | 15.6781 | 0.8807 | 0.1758 |
| Stimark (1) | 25.5343 | 0.9801 | 0.0305 |
| Template remove (1) | 38.1953 | 0.9992 | 0.0012 |
| Trimmed mean (2) | 31.3160 | 0.9989 | 0.0071 |
| Warp (1) | 18.2210 | 0.9141 | 0.1279 |
| Wavelet compression (10%) | 30.5847 | 0.9895 | 0.0161 |

The PSNR values of attacked Lena images and the NC/BER values of watermarks extracted from them are also shown in Table 17. The PSNR values indicate that the host images are badly distorted under different attacks. The NC and BER values of the extracted watermarks prove that the extracted watermarks are very closely similar to the original watermark excepting for the cropping attacks with large areas. These data generally illustrate that the proposed scheme is robust to different attacks for multiple watermarks also with different sizes. For watermark image 1 of size $64 \times 64$, watermark image 3 of size $64 \times 64$ and watermark image 4 of size $128 \times 128$, the extracted watermarks from the corresponding attacked Lena images are shown in Fig. 11 to Fig. 13. In any case, it can be observed form the Fig. 10 to Fig. 13 that even though the host images are seriously degraded by different attacks, the corresponding extracted watermarks with different contents and sizes are basically clear enough. This


**FIGURE 11.** The extracted watermark image 1 for Lena image under different attacks: (a) Gaussian noise (0.05), (b) Gaussian noise (0.15), (c) Salt and Peppers noise(0.1), (d) Salt and Peppers noise (0.2), (e) Median filtering (5 × 5), (f) Wiener filtering (5 × 5), (g) JPEG compression(5%), (h) JPEG compression (20%), (i) Scaling (1/8), (j) Cropping (center 128 × 128), (k) Cropping (Edge 64 × 512), (l) Translation (5 pixels), (m) Rotation (5°), (n) Rotation (10°), (o) Rotation(40°), (p) Rotation (90°).

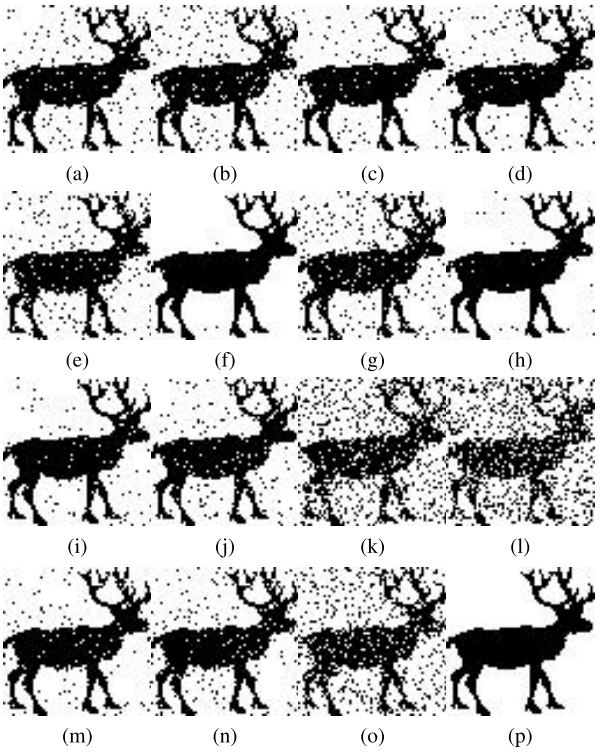


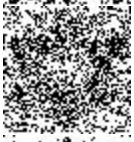

**FIGURE 12.** The extracted watermark image 3 for Lena image under different attacks: (a) Gaussian noise (0.05), (b) Gaussian noise (0.15), (c) Salt and Peppers noise(0.1), (d) Salt and Peppers noise (0.2), (e) Median filtering (5 × 5), (f) Wiener filtering (5 × 5), (g) JPEG compression(5%), (h) JPEG compression (20%), (i) Scaling (1/8), (j) Cropping (center 128 × 128), (k) Cropping (Edge 64 × 512), (l) Translation (5 pixels), (m) Rotation (5°), (n) Rotation (10°), (o) Rotation(40°), (p) Rotation (90°).

demonstrates that the proposed scheme exhibits excellent performance against the attacks mentioned in Table 17. Table 18 shows the robustness test results of the proposed scheme against the standard benchmark software Checkmark [64]. In Table 18, twenty kinds of attacks in Checkmark, which are not almost mentioned in Table 17, are selected out to test the proposed scheme, and the PSNR, NC and BER values for different attacks are shown respectively. For these attacks, the NC and BER values are well accepted except the rotation scale, shearing and warp attacks, but the minimum NC value is 0.8807 while the maximum BER value is 0.1758. As can be seen from Table 18, the proposed scheme is robust to Checkmark attacks to some extent.

### C. COMPARISON WITH OTHER ZERO-WATERMARKING ALGORITHMS

To evaluate the robustness of the proposed scheme in comparison with other methods, we have compared the results of our scheme with the three methods in [38], [40], [42]. We firstly embedded the 64 × 64 watermark image 2, shown in Fig. 8, into the host gray-scale images in all the algorithms. In the experiments, eight common attacks were adopted, including (1) Gaussian noise (0.05), (2) Salt and Peppers noise (0.1), (3) Median filtering (5 × 5), (4) JPEG compression (10%), (5) Scaling (1/8), (6) Cropping (center 128 × 128), (7) Translation (5 pixels) and (8) Rotation (5°). Since several attacks can happen on an image simultaneously, it is also important to investigate the robustness of a watermarking scheme against combined attacks. Hence, we additionally tested four types of combined attacks, including (9) Gaussian noise (0.01)+ JPEG compression (10%), (10) Median filtering (3 × 3)+ Scaling (1/8), (11) Salt and Peppers noise (0.02)+ Median filtering (5 × 5)+ Rotation (1°), and (12) Gaussian noise (0.01)+ Scaling (1/8)+ JPEG compression (5%). The Peppers image shown in Fig. 7 was used as the test image at this time. Table 19 shows the extracted watermarks from attacked Peppers images using the proposed scheme and the three methods in [38], [40], [42], and it can prove that the proposed scheme is more robust than other three methods. Because of the strong attack parameters used in the comparison experiments, some of the extracted watermarks in other three comparative methods are almost not clear enough in Table 19. For the watermark image 2 of size 32 × 32, watermark image 3 of size 64 × 64 and watermark image 4 of size 128 × 128, the same host image and attacks mentioned in Table 19 are used to test the robustness among the different methods. Fig.14 to Fig. 16 show the comparison of the results in terms of average BER values of the extracted different

**TABLE 19.** The extracted watermarks for Peppers image using different methods.

| No. | Attack types and the corresponding PSNR values | Extracted watermarks | | | |
|---|---|---|---|---|---|
| | | DC-RE in [38] | PVMF in [40] | Method in [42] | Proposed scheme |
| 1 | Gaussian noise (0.05) PSNR=13.7545 |  |  |  |  |
| 2 | Salt and Peppers noise (0.1) PSNR=15.3457 |  |  |  |  |
| 3 | Median filtering ($5 \times 5$) PSNR=32.3463 |  |  |  |  |
| 4 | JPEG compression (10%) PSNR=31.1288 |  |  |  |  |
| 5 | Scaling (1/8) PSNR=24.5768 |  |  |  |  |
| 6 | Cropping (center $128 \times 128$) PSNR=17.7666 |  |  |  |  |
| 7 | Translation (5 pixels) PSNR=18.2372 |  |  |  |  |
| 8 | Rotation (5°) PSNR=15.5551 |  |  |  |  |
| 9 | Gaussian noise (0.1) +JPEG compression (10%) PSNR=24.1970 |  |  |  |  |
| 10 | Gaussian noise (0.1) +Scaling (1/8), PSNR=24.5514 |  |  |  |  |
| 11 | Salt and Peppers noise (0.02) +Median filtering ($5 \times 5$) +Rotation (1°), PSNR=20.7589 |  |  |  |  |
| 12 | Salt and Peppers noise (0.01) +JPEG compression (10%) +Scaling (1/8), PSNR=23.3525 |  |  |  |  |

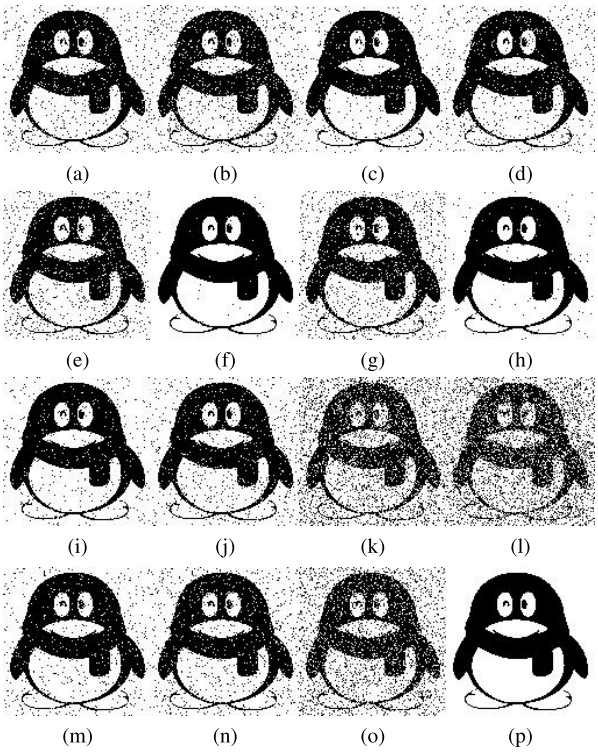



**FIGURE 13.** The extracted watermark image 4 for Lena image under different attacks: (a) Gaussian noise (0.05), (b) Gaussian noise (0.15), (c) Salt and Peppers noise(0.1), (d) Salt and Peppers noise (0.2), (e) Median filtering (5 × 5), (f) Wiener filtering (5 × 5), (g) JPEG compression(5%), (h) JPEG compression (20%), (i) Scaling (1/8), (j) Cropping (center 128 × 128), (k) Cropping (Edge 64 × 512), (l) Translation (5 pixels), (m) Rotation (5°), (n) Rotation (10°), (o) Rotation(40°), (p) Rotation (90°).

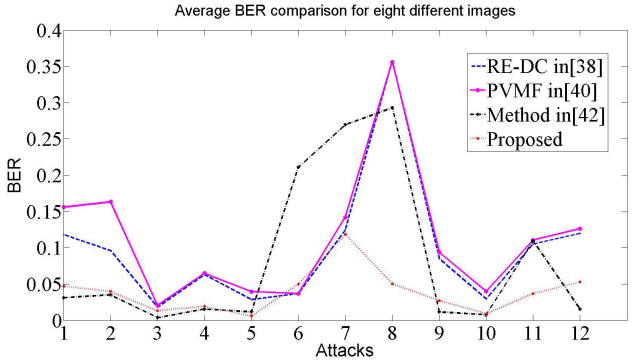



**FIGURE 14.** The average BER values in different methods with watermark image 2 of sizes 32 × 32.

watermarks for eight test images shown in Fig. 7. As can be seen from Fig. 14 to Fig. 16, it is clear that the proposed scheme generally outperforms the other three watermarking methods in [38], [40], [42] for the above attacks. Especially, this advantage is more obvious when the size of watermark is large, for instance, the watermark image 4 used in Fig.16.

### D. STATISTICAL ANALYSIS

In this section, a statistical analysis is performed between our proposed scheme and the each method in [38], [40], [42].

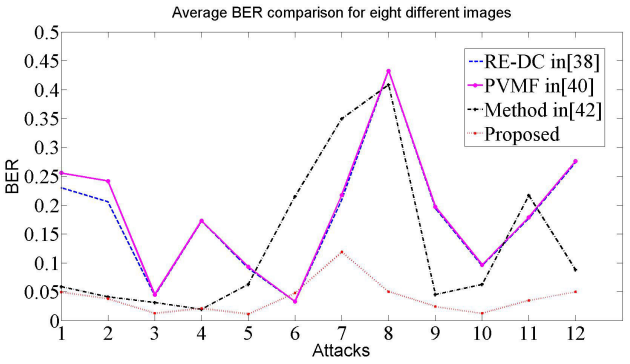



**FIGURE 15.** The average BER values in different methods with watermark image 3 of sizes 64 × 64.

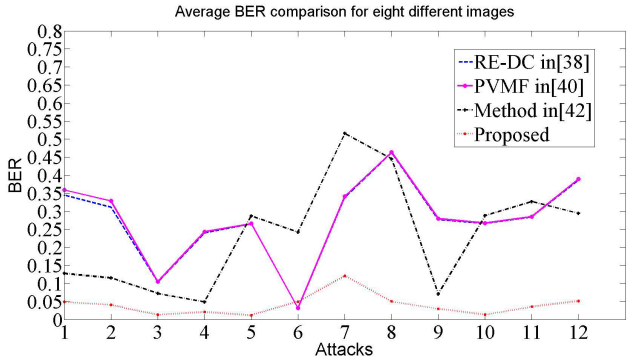



**FIGURE 16.** The average BER values in different methods with watermark image 4 of sizes 128 × 128.

**TABLE 20.** The average PSNR/NC values under Gaussian noise attacks.

| Noise variances | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 0.01 | 20.0975 | 0.9861 | 0.9031 | 0.8978 | 0.9873 |
| 0.02 | 17.2101 | 0.9793 | 0.8776 | 0.8688 | 0.9704 |
| 0.05 | 13.6507 | 0.9686 | 0.8398 | 0.8213 | 0.9691 |
| 0.1 | 11.3515 | 0.9563 | 0.8108 | 0.7873 | 0.9455 |
| 0.15 | 10.2231 | 0.9493 | 0.7890 | 0.7644 | 0.9454 |
| 0.2 | 9.5321 | 0.9415 | 0.7770 | 0.7475 | 0.9310 |
| 0.25 | 9.0570 | 0.9365 | 0.7652 | 0.7395 | 0.9252 |

According to the way mentioned in [65], Wilcoxon signed ranks test with a significance level of 0.05 is same to be used to assess the statistical significance of the difference between the proposed scheme and the each method in [38], [40], [42]. The robustness tests of the these compared algorithms against the attacks are shown in Tables 20 to 28. In the Tables 20 to 28, the average PSNR/NC values under different attacks for all the eight test images, shown in Fig. 7, are calculated. Furthermore, in order to satisfy the requirement for the number of statistical samples, the number of a certain type attacks in Tables 20 to 28 is increased form 5 to 7 comparing with that in the Tables 8 to 16.

**TABLE 21.** The average PSNR/NC values under Salt and Peppers noise attacks.

| Noise densities | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 0.01 | 25.4547 | 0.9925 | 0.9464 | 0.9403 | 0.9911 |
| 0.02 | 22.3985 | 0.9895 | 0.9229 | 0.9119 | 0.9872 |
| 0.05 | 18.4239 | 0.9817 | 0.8924 | 0.8683 | 0.9833 |
| 0.1 | 15.4139 | 0.9732 | 0.8613 | 0.8318 | 0.9637 |
| 0.15 | 13.6794 | 0.9667 | 0.8348 | 0.8087 | 0.9599 |
| 0.2 | 12.4090 | 0.9606 | 0.8169 | 0.7871 | 0.9520 |
| 0.25 | 11.4291 | 0.9527 | 0.8015 | 0.7760 | 0.9373 |

**TABLE 22.** The average PSNR/NC values under Median filtering attacks.

| Sizes of filter | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| $3 \times 3$ | 32.1413 | 0.9967 | 0.9834 | 0.9824 | 0.9872 |
| $4 \times 4$ | 27.5993 | 0.9835 | 0.9618 | 0.9608 | 0.9454 |
| $5 \times 5$ | 29.1302 | 0.9917 | 0.9717 | 0.9708 | 0.9795 |
| $6 \times 6$ | 26.5578 | 0.9802 | 0.9542 | 0.9526 | 0.9321 |
| $7 \times 7$ | 27.4955 | 0.9874 | 0.9603 | 0.9592 | 0.9665 |
| $9 \times 9$ | 26.3120 | 0.8558 | 0.9472 | 0.9461 | 0.9572 |
| $11 \times 11$ | 25.4070 | 0.8549 | 0.9324 | 0.9314 | 0.9375 |

**TABLE 23.** The average PSNR/NC values under Wiener filtering attacks.

| Sizes of filter | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| $3 \times 3$ | 34.5984 | 0.9989 | 0.9894 | 0.9877 | 0.9924 |
| $4 \times 4$ | 32.6026 | 0.9939 | 0.9751 | 0.9741 | 0.9898 |
| $5 \times 5$ | 32.0164 | 0.9972 | 0.9796 | 0.9782 | 0.9834 |
| $6 \times 6$ | 30.9845 | 0.9928 | 0.9678 | 0.9667 | 0.9860 |
| $7 \times 7$ | 30.4787 | 0.9953 | 0.9677 | 0.9665 | 0.9783 |
| $9 \times 9$ | 29.3087 | 0.9933 | 0.9547 | 0.9541 | 0.9704 |
| $11 \times 11$ | 28.3664 | 0.9901 | 0.9400 | 0.9390 | 0.9652 |

**TABLE 24.** The average PSNR/NC values under JPEG compression attacks.

| Quality factors | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 1% | 23.5509 | 0.9500 | 0.7942 | 0.7922 | 0.9303 |
| 5% | 26.1208 | 0.9707 | 0.8286 | 0.8294 | 0.9678 |
| 10% | 28.7050 | 0.9863 | 0.8828 | 0.8822 | 0.9872 |
| 20% | 30.8657 | 0.9941 | 0.9287 | 0.9281 | 0.9936 |
| 30% | 32.0099 | 0.9955 | 0.9450 | 0.9455 | 0.9924 |
| 40% | 32.7744 | 0.9965 | 0.9571 | 0.9574 | 0.9962 |
| 50% | 33.4156 | 0.9976 | 0.9632 | 0.9654 | 0.9949 |

**TABLE 25.** The average PSNR/NC values under cropping attacks.

| Types of croppping | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| Top-left $32 \times 32$ | 32.7752 | 0.9960 | 0.9985 | 0.9985 | 0.9691 |
| Top-left $64 \times 64$ | 25.6729 | 0.9938 | 0.9951 | 0.9951 | 0.9413 |
| Top-left $64 \times 128$ | 22.1912 | 0.9872 | 0.9906 | 0.9906 | 0.9455 |
| Top-left $128 \times 128$ | 18.4087 | 0.9756 | 0.9813 | 0.9813 | 0.8967 |
| Center $128 \times 128$ | 17.0610 | 0.9686 | 0.9785 | 0.9784 | 0.8508 |
| Edge $64 \times 512$ | 14.4700 | 0.9519 | 0.9605 | 0.9607 | 0.6732 |
| Edge $128 \times 512$ | 8.4867 | 0.7921 | 0.8361 | 0.8364 | 0.7224 |

**TABLE 26.** The average PSNR/NC values under scaling attacks.

| Scaling factors | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 1/32 | 32.7752 | 0.8831 | 0.7343 | 0.7333 | 0.7361 |
| 1/16 | 25.6729 | 0.9765 | 0.8144 | 0.8139 | 0.8317 |
| 1/8 | 22.1912 | 0.9927 | 0.9397 | 0.9382 | 0.9584 |
| 1/4 | 18.4087 | 0.9985 | 0.9780 | 0.9758 | 0.9949 |
| 1/2 | 17.0610 | 0.9994 | 0.9905 | 0.9887 | 0.9975 |
| 2 | 14.4700 | 0.9997 | 0.9959 | 0.9947 | 1.0000 |
| 4 | 8.4867 | 0.9999 | 0.9967 | 0.9954 | 1.0000 |

**TABLE 27.** The average PSNR/NC values under translation attacks.

| Distances of translation | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 1 pixel | 25.2435 | 0.9815 | 0.9659 | 0.9583 | 0.8913 |
| 2 pixels | 22.0444 | 0.9649 | 0.9362 | 0.9289 | 0.8076 |
| 3 pixels | 20.4578 | 0.9504 | 0.9076 | 0.9000 | 0.7789 |
| 4 pixels | 19.3440 | 0.9353 | 0.8803 | 0.8740 | 0.7671 |
| 5 pixels | 18.5579 | 0.9205 | 0.8561 | 0.8500 | 0.7472 |
| 6 pixels | 17.9150 | 0.9058 | 0.8323 | 0.8270 | 0.7264 |
| 7 pixels | 17.3867 | 0.8914 | 0.8095 | 0.8059 | 0.6955 |

**TABLE 28.** The average PSNR/NC values under rotation attacks.

| Angles of rotation | PSNR | NC | | | |
|---|---|---|---|---|---|
| | | Proposed scheme | Method in [38] | PVMF in [40] | Method in [42] |
| 5° | 14.6148 | 0.9672 | 0.6799 | 0.6808 | 0.7014 |
| 10° | 12.7213 | 0.9532 | 0.6488 | 0.6489 | 0.6840 |
| 20° | 11.1588 | 0.9290 | 0.6287 | 0.6276 | 0.6488 |
| 30° | 10.4677 | 0.9204 | 0.9294 | 0.6299 | 0.6360 |
| 40° | 10.0856 | 0.9141 | 0.6240 | 0.6239 | 0.5895 |
| 50° | 9.9846 | 0.9096 | 0.6256 | 0.6239 | 0.6401 |
| 90° | 11.4102 | 1.0000 | 0.6207 | 0.6193 | 0.6306 |

In this part, IBM SPSS Statistics 26 was used to perform the statistical calculations. The null hypothesis $H_0$ indicates that there is no significant difference between the proposed scheme and the each method in [38], [40], [42]. In contrast, the alternative hypothesis $H_1$ indicates that a significant difference exists between the proposed scheme and the each method in [38], [40], [42]. With a 95% confidence level, $H_0$ is rejected if $\alpha \leq 0.05$, and $H_0$ can not be rejected if $\alpha > 0.05$. In the case of rejecting $H_0$ ($\alpha \leq 0.05$), the final decision is made according to the result whether the sum of the negative ranks $w^-$ is smaller than or equal to the critical value for the Wilcoxon signed rank test. If the sum of the negative ranks

**TABLE 29. Statistical comparison using the Wilcoxon signed-rank test.**

| Attacks | Proposed scheme vs method in [38] | Proposed scheme vs PVMF in [40] | Proposed scheme vs method in [42] |
|---|---|---|---|
| Gaussian noise (Table 18) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.063$ $w_\alpha^* = 3, w^- = 3, w^- \leq w_\alpha^*$ |
| Salt and Peppers noise (Table 19) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.043$ $w_\alpha^* = 3, w^- = 2, w^- \leq w_\alpha^*$ |
| Median filtering (Table 20) | Can not reject $H_0$, with $\alpha = 0.866$ $w_\alpha^* = 3, w^- = 13, w^- > w_\alpha^*$ | Can not reject $H_0$, with $\alpha = 0.866$ $w_\alpha^* = 3, w^- = 13, w^- > w_\alpha^*$ | Can not reject $H_0$, with $\alpha = 0.866$ $w_\alpha^* = 3, w^- = 13, w^- > w_\alpha^*$ |
| Wiener filtering (Table 21) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ |
| JPEG compression (Table 22) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 3, w^- \leq w_\alpha^*$ |
| Cropping (Table 23) | Can not reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 28, w^- > w_\alpha^*$ | Can not reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 28, w^- > w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ |
| Scaling (Table 24) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.063$ $w_\alpha^* = 3, w^- = 3, w^- \leq w_\alpha^*$ |
| Translation (Table 25) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ |
| Rotation (Table 26) | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ | Reject $H_0$, with $\alpha = 0.018$ $w_\alpha^* = 3, w^- = 0, w^- \leq w_\alpha^*$ |

value is smaller than or equal to the critical value for the Wilcoxon signed rank test (i.e., $w^- \leq w_\alpha^*$), the proposed scheme is more robust than the each method in [38], [40], [42]. Otherwise, the proposed scheme can not be obviously recognized as the better one [65].

The results of this statistical analysis is shown in Table 29. As can be seen from Table 29, the proposed scheme generally outperforms the each method in [38], [40], [42] except Median filtering and Cropping attacks. For Median filtering, the proposed scheme is better than the each method in [38], [40], [42] when the window sizes are $3 \times 3$, $4 \times 4$, $5 \times 5$, $6 \times 6$ and $7 \times 7$. But the nice average NC values are not obtained when the window sizes are $9 \times 9$, $11 \times 11$ due to the three images, including Elain, Goldhill, and Bridge. Moreover, the each method in [38], [40] is more superior than the proposed scheme against Cropping attacks, but the superiority is not very significant. Finally, it is not difficult to find that the proposed scheme is more robust than the methods in [38], [40], [42] based on the Tables 20 to 29.

### E. SECURITY ANALYSIS

The security of a watermarking scheme mainly depends on the encryption algorithms used in the watermarking scheme. In this paper, the secret keys from Arnold transform and the two random sequences produced by VPCM system together ensure the high security of the proposed scheme. Even though the algorithm of zero-watermark generation is open, the right zero-watermark image will not be extracted in the process of watermark detection if any one of the secret keys is incorrect. Due to the limited length of this paper, we only analyse how the secret key $\mu_1$ affects the security of the proposed scheme in this subsection. In the previous experiments, we know that the correct $\mu_1$ is 3.89999. To test the security of the scheme, we design the experiments as follows. Assuming that all the secret keys in the process of watermark extraction are correct except $\mu_1$, which is variable, we varied $\mu_1$ from

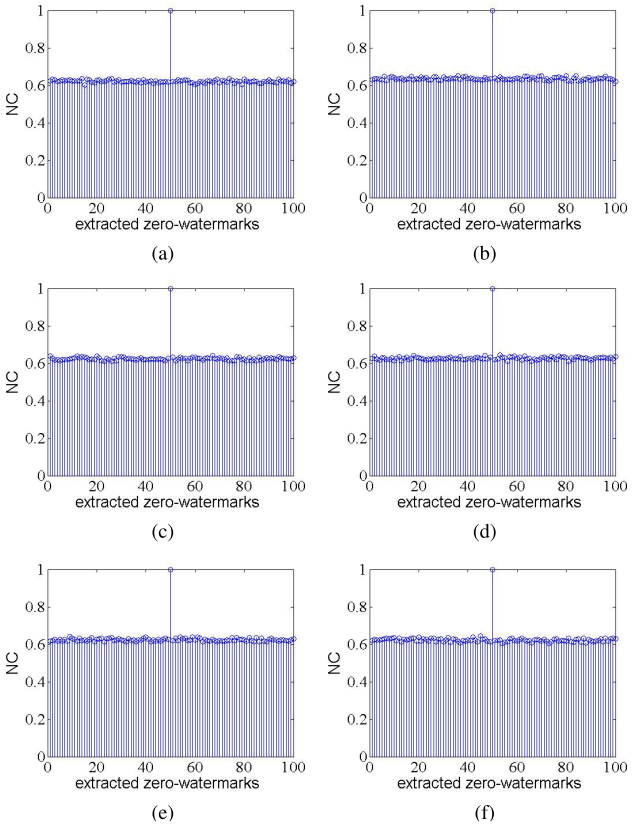



**FIGURE 17. Security test of the proposed scheme for different images: (a) Man, (b) Tiffany, (c) Elain, (d) Lena, (e) Goldhill, (f) Boat.**

3.89950 to 3.90049 with 0.00001 steps to extract 100 zero-watermarks. The 100 NC values were respectively calculated between the 100 extracted zero-watermarks and the original zero-watermark generated with the correct $\mu_1$ for different images. The experimental results are shown in Fig. 17. Fig. 17 shows that the NC value is 1 when $\mu_1$ is correct. However, the NC value is just about 0.5 when $\mu_1$ is incorrect, which

**TABLE 30. Average computation time of the proposed scheme for all the test images.**

| | Zero-watermark generation | Zero-watermark extraction under rotation attacks | | | | |
|---|---|---|---|---|---|---|
| | | 0° | 5° | 10° | 40° | 90° |
| Time (s) | 1.3514 | 1.2500 | 1.3260 | 1.2636 | 1.3026 | 1.2792 |

means that the watermark is extracted unsuccessfully. It is important to note that finding out the correct $\mu_1$ randomly is very difficult. Therefore, all the secret keys perfectly ensure the security of the proposed scheme, and it is impossible to extract the correct zero-watermark without all the correct secret keys.

### F. COMPUTATION TIME

The average computation time of the zero-watermark generation and extraction for the eight test images mentioned in Fig.7 are shown in Table 30. The computer used for our experiments had a 3.20 GHz Processor, 4-GB RAM, and a Microsoft Windows 7 operating system with 32 bits. The experiments were completed in the environment of MATLAB 2014. Table 30 illustrates that the computation time of the zero-watermark generation and extraction in the proposed scheme are acceptable.

## V. CONCLUSION

In the proposed scheme for the copyright protection of digital images, NSPD and DCT were used to extract the robust feature of a host image as the original zero-watermark. Several experiments were performed to demonstrate that the signs of some AC coefficients of an image in the NSPD-DCT domain almost do not change under various attacks. To enhance the security of the proposed scheme, we combined PWLCM with LM to generate the VPCM system used for the watermark encryption and the robust feature vector extraction. Furthermore, an image rotation correction procedure was used before zero-watermark detection to improve the robustness of the scheme against rotation attacks. The experimental results have demonstrated that the NSPD-DCT based zero-watermarking scheme was highly robust to common image processing operations, several geometric attacks and some Checkmark attacks, and it had better performance than some existing algorithms. In the future, we will further improve the robustness of the proposed zero-watermarking scheme against combined attacks and extend this scheme to video watermarking.

## REFERENCES


[1] G. Bhatnagar, Q. M. J. Wu, and P. K. Atrey, "Secure randomized image watermarking based on singular value decomposition," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 10, no. 1, pp. 1–21, Dec. 2013.

[2] S. Liu, B. M. Hennelly, C. Guo, and J. T. Sheridan, "Robustness of double random phase encoding spread-space spread-spectrum watermarking technique," *Signal Process.*, vol. 109, pp. 345–361, Apr. 2015.

[3] Z. Shao, Y. Shang, Y. Zhang, X. Liu, and G. Guo, "Robust watermarking using orthogonal Fourier–Mellin moments and chaotic map for double images," *Signal Process.*, vol. 120, pp. 522–531, Mar. 2016.

[4] H.-Y. Yang, X.-Y. Wang, P.-P. Niu, and A.-L. Wang, "Robust color image watermarking using geometric invariant quaternion polar harmonic transform," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 11, no. 3, pp. 1–26, Feb. 2015.

[5] M. Amirmazlaghani, "Additive watermark detection in the wavelet domain using 2D-GARCH model," *Inf. Sci.*, vols. 370–371, pp. 1–17, Nov. 2016.

[6] S. Singh, V. S. Rathore, R. Singh, and M. K. Singh, "Hybrid semi-blind image watermarking in redundant wavelet domain," *Multimedia Tools Appl.*, vol. 76, no. 18, pp. 19113–19137, Sep. 2017.

[7] H.-T. Hu and L.-Y. Hsu, "A mixed modulation scheme for blind image watermarking," *AEU-Int. J. Electron. Commun.*, vol. 70, no. 2, pp. 172–178, Feb. 2016.

[8] S. Roy and A. K. Pal, "A robust blind hybrid image watermarking scheme in RDWT-DCT domain using Arnold scrambling," *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 3577–3616, 2017.

[9] X.-Y. Wang, Y.-N. Liu, H. Xu, A.-L. Wang, and H.-Y. Yang, "Blind optimum detector for robust image watermarking in nonsubsampled Shearlet domain," *Inf. Sci.*, vol. 372, pp. 634–654, Dec. 2016.

[10] N. Liu, H. Li, H. Dai, D. Guo, and D. Chen, "Robust blind image watermarking based on chaotic mixtures," *Nonlinear Dyn.*, vol. 80, no. 3, pp. 1329–1355, May 2015.

[11] H. Agarwal, P. K. Atrey, and B. Raman, "Image watermarking in real oriented wavelet transform domain," *Multimedia Tools Appl.*, vol. 74, no. 23, pp. 10883–10921, Dec. 2015.

[12] N. M. Makbol and B. E. Khoo, "Robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition," *AEU-Int. J. Electron. Commun.*, vol. 67, no. 2, pp. 102–112, Feb. 2013.

[13] M. Botta, D. Cavagnino, and V. Pomponiu, "A modular framework for color image watermarking," *Signal Process.*, vol. 119, pp. 102–114, Feb. 2016.

[14] Y. Naderahmadian and S. Hosseini-Khayat, "Fast and robust watermarking in still images based on QR decomposition," *Multimedia Tools Appl.*, vol. 72, no. 3, pp. 2597–2618, Oct. 2014.

[15] I. Nasir, Y. Weng, and J. Jiang, "Novel multiple spatial watermarking technique in color images," in *Proc. 5th Inf. Int. Conf. Inf. Technol., New Gener.*, vol. 4, Apr. 2008, pp. 777–782.

[16] Q. Su and B. Chen, "An improved color image watermarking scheme based on Schur decomposition," *Multimedia Tools Appl.*, vol. 76, no. 22, pp. 24221–24249, 2017.

[17] T. Huynh-The, C.-H. Hua, N. A. Tu, T. Hur, J. Bang, D. Kim, M. B. Amin, B. H. Kang, H. Seung, and S. Lee, "Selective bit embedding scheme for robust blind color image watermarking," *Inf. Sci.*, vol. 426, pp. 1–18, Feb. 2018.

[18] H.-T. Hu and J.-R. Chang, "Dual image watermarking by exploiting the properties of selected DCT coefficients with JND modeling," *Multimedia Tools Appl.*, vol. 77, no. 20, pp. 26965–26990, Oct. 2018.

[19] M. Khalili, "DCT-Arnold chaotic based watermarking using JPEG-YCbCr," *Optik-Int. J. Light Electron Opt.*, vol. 126, no. 23, pp. 4367–4371, Dec. 2015.

[20] Y.-K. Lin, C.-H. Yang, and J.-T. Tsai, "More secure lossless visible watermarking by DCT," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 8579–8601, Apr. 2018.

[21] S. Liu, Z. Pan, and H. Song, "Digital image watermarking method based on DCT and fractal encoding," *IET Image Process.*, vol. 11, no. 10, pp. 815–821, Oct. 2017.

[22] S. Roy and A. K. Pal, "A blind DCT based color watermarking algorithm for embedding multiple watermarks," *AEU-Int. J. Electron. Commun.*, vol. 72, pp. 149–161, Feb. 2017.

[23] A. M. Abdelhakim, M. H. Saad, M. Sayed, and H. I. Saleh, "Optimized SVD-based robust watermarking in the fractional Fourier domain," *Multimedia Tools Appl.*, vol. 77, no. 21, pp. 27895–27917, Nov. 2018.

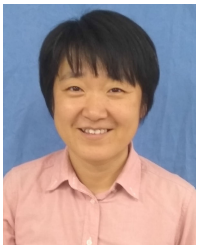
[24] T. K. Tsui, X.-P. Zhang, and D. Androutsos, "Color image watermarking using multidimensional Fourier transforms," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 16–28, Mar. 2008.

[25] C. Wang, J. Ni, and J. Huang, "An informed watermarking scheme using hidden Markov model in the wavelet domain," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 853–867, Jun. 2012.

[26] C.-C. Lai and C.-C. Tsai, "Digital image watermarking using discrete wavelet transform and singular value decomposition," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 11, pp. 3060–3063, Nov. 2010.

[27] M. A. Akhaee, S. M. E. Sahraeian, and F. Marvasti, "Contourlet-based image watermarking using optimum detector in a noisy environment," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 967–980, Apr. 2010.

[28] S. Etemad and M. Amirmazlaghani, "A new multiplicative watermark detector in the contourlet domain using t location-scale distribution," *Pattern Recognit.*, vol. 77, pp. 99–112, May 2018.
[29] B. Ahmaderaghi, F. Kurugollu, J. M. D. Rincon, and A. Bouridane, "Blind image watermark detection algorithm based on discrete Shearlet transform using statistical decision theory," *IEEE Trans. Comput. Imag.*, vol. 4, no. 1, pp. 46–59, Mar. 2018.
[30] M. Mardanpour and M. A. Z. Chahooki, "Robust transparent image watermarking with Shearlet transform and bidiagonal singular value decomposition," *AEU-Int. J. Electron. Commun.*, vol. 70, no. 6, pp. 790–798, Jun. 2016.
[31] Z. Jian, L. Shan, J. Jian, Z. Wanru, and Z. Shunli, "Extended Shearlet-based image watermarking algorithm using selective coefficients in horizontal cone," *Arabian J. Sci. Eng.*, vol. 42, no. 8, pp. 3597–3607, 2017.
[32] M. Ghadi, L. Laouamer, L. Nana, and A. Pascu, "A novel zero-watermarking approach of medical images based on Jacobian matrix model," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 5203–5218, Dec. 2016.
[33] M. Shakeri and M. Jamzad, "A robust zero-watermarking scheme using Canny edge detector," *Int. J. Electron. Secur. Digit. Forensics*, vol. 5, no. 1, pp. 25–44, 2013.
[34] S. Vellaisamy and V. Ramesh, "Inversion attack resilient zero-watermarking scheme for medical image authentication," *IET Image Process.*, vol. 8, no. 12, pp. 718–727, Dec. 2014.
[35] P. Zhu, F. Jia, and J. Zhang, "A copyright protection watermarking algorithm for remote sensing image based on binary image watermark," *Optik-Int. J. Light Electron Opt.*, vol. 124, no. 20, pp. 4177–4181, Oct. 2013.
[36] Q. Wen, T. Sun, and S. Wang, "Concept and application of zero-watermark," (in Chinese), *Acta Electron. Sinica*, vol. 31, pp. 214–216, Feb. 2003.
[37] T. Ye, "A robust zero-watermark algorithm based on singular value decomposition and discreet cosine transform," in *Parallel and Distributed Computing and Networks*, vol. 137. 2011, pp. 1–8.
[38] Y. Zhang, C. Jia, X. Wang, K. Wang, and W. Pei, "Robust zero-watermark algorithms based on numerical relationship between adjacent blocks," *J. Electron. (China)*, vol. 29, no. 5, pp. 392–399, Sep. 2012.
[39] A. Rani and B. Raman, "An image copyright protection scheme by encrypting secret data with the host image," *Multimedia Tools Appl.*, vol. 75, no. 2, pp. 1027–1042, Jan. 2016.
[40] T. M. Thanh and K. Tanaka, "An image zero-watermarking algorithm based on the encryption of visual map feature with watermark information," *Multimedia Tools Appl.*, vol. 76, no. 11, pp. 13455–13471, Jun. 2017.
[41] S. Lin, X. Jiucheng, Z. Xingxing, D. Wan, and T. Yun, "A novel generalized Arnold transform-based zero-watermarking scheme," *Appl. Math. Inf. Sci.*, vol. 9, no. 4, pp. 2023–2035, 2015.
[42] X. Wu, J. Li, R. Tu, J. Cheng, U. A. Bhatti, and J. Ma, "Contourlet-DCT based multiple robust watermarkings for medical images," *Multimedia Tools Appl.*, vol. 78, no. 7, pp. 8463–8480, Apr. 2019.
[43] S.-C. Han and Z.-N. Zhang, "A novel zero-watermark algorithm based on LU decomposition in NSST domain," in *Proc. IEEE 11th Int. Conf. Signal Process.*, vol. 3, Oct. 2012, pp. 1592–1596.
[44] J. Zhao, W. Xu, S. Zhang, S. Fan, and W. Zhang, "A strong robust zero-watermarking scheme based on Shearlets' high ability for capturing directional features," *Math. Problems Eng.*, vol. 2016, pp. 1–11, Oct. 2016.
[45] W.-Q. Lin, "The discrete Shearlet transform: A new directional transform and compactly supported Shearlet frames," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1166–1180, May 2010.
[46] G. Gao and G. Jiang, "Bessel-Fourier moment-based robust image zero-watermarking," *Multimedia Tools Appl.*, vol. 74, no. 3, pp. 841–858, Feb. 2015.
[47] C.-P. Wang, X.-Y. Wang, X.-J. Chen, and C. Zhang, "Robust zero-watermarking algorithm based on polar complex exponential transform and logistic mapping," *Multimedia Tools Appl.*, vol. 76, no. 24, pp. 26355–26376, Dec. 2017.
[48] R. Keshavarzian and A. Aghagolzadeh, "ROI based robust and secure image watermarking using DWT and Arnold map," *AEU-Int. J. Electron. Commun.*, vol. 70, no. 3, pp. 278–288, Mar. 2016.
[49] L.-Y. Hsu and H.-T. Hu, "Blind image watermarking via exploitation of inter-block prediction and visibility threshold in DCT domain," *J. Vis. Commun. Image Represent.*, vol. 32, pp. 130–143, Oct. 2015.
[50] D. Singh and S. K. Singh, "DCT based efficient fragile watermarking scheme for image authentication and restoration," *Multimedia Tools Appl.*, vol. 76, no. 1, pp. 1–25, 2015.
[51] M. Moosazadeh and G. Ekbatanifard, "An improved robust image watermarking method using DCT and YCoCg-R color space," *Optik-Int. J. Light Electron Opt.*, vol. 140, pp. 975–988, Jul. 2017.
[52] A. L. Da Cunha, J. Zhou, and M. N. Do, "The nonsubsampled contourlet transform: Theory, design, and applications," *IEEE Trans. Image Process.*, vol. 15, no. 10, pp. 3089–3101, Oct. 2006.
[53] Z. Hui-Xin, R. Sheng-Hui, Q. Han-Lin, L. Rui, and Z. Jun, "Anomaly detection algorithm based on nonsubsampled pyramid decomposition and kernel unsharp masking for hyperspectral image," in *Proc. Int. Conf. Ind. Control Electron. Eng.*, Aug. 2012, pp. 1320–1323.
[54] X. Zhang, W. Wang, D. Wang, and Y. Zhang, "A fusion algorithm for remote sensing images based on nonsub-sampled pyramids and bidimensional empirical decomposition," *Sci. China Technol. Sci.*, vol. 53, no. 1, pp. 196–204, May 2010.
[55] S. S. Jamal, T. Shah, and I. Hussain, "An efficient scheme for digital watermarking using chaotic map," *Nonlinear Dyn.*, vol. 73, no. 3, pp. 1469–1474, Aug. 2013.
[56] D. Caragata, S. E. Assad, and M. Luduena, "An improved fragile watermarking algorithm for JPEG images," *AEU-Int. J. Electron. Commun.*, vol. 69, no. 12, pp. 1783–1794, Dec. 2015.
[57] Y. Hu, C. Zhu, and Z. Wang, "An improved piecewise linear chaotic map based image encryption algorithm," *Sci. World J.*, vol. 2014, pp. 1–7, Jan. 2014.
[58] B. Lei, D. Ni, S. Chen, T. Wang, and F. Zhou, "Optimal image watermarking scheme based on chaotic map and quaternion wavelet transform," *Nonlinear Dyn.*, vol. 78, no. 4, pp. 2897–2907, Dec. 2014.
[59] J. Li, C. Dong, M. Huang, H. Zhang, and Y. Chen, "A novel robust watermarking for medical image," *Adv. Inf. Sci. Service Sci.*, vol. 4, no. 11, pp. 28–36, 2012.
[60] J. Dong and J. Li, "A robust zero-watermarking algorithm for encrypted medical images in the DWT-DCT encrypted domain," *Int. J. Simul.—Syst., Sci. Technol.*, vol. 17, no. 43, pp. 34.1–34.7, Dec. 2016.
[61] S.-C. Han, J.-F. Yang, R. Wang, and G.-M. Jia, "A robust color image watermarking algorithm against rotation attacks," *Optoelectron. Lett.*, vol. 14, no. 1, pp. 61–66, Jan. 2018.
[62] *The USC-SIPI Image Database*. [Online]. Available: http://sipi.usc.edu/database/
[63] M. Ali and C. W. Ahn, "An optimized watermarking technique based on self-adaptive DE in DWT–SVD transform domain," *Signal Process.*, vol. 94, pp. 545–556, Jan. 2014.
[64] S. Pereira, S. Voloshynovskiy, S. Marchand-Maillet, T. Pun, and M. Madueno, "Second generation benchmarking and application oriented evaluation," in *Proc. 4th Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, 2001.
[65] N. M. Makbol and B. E. Khoo, "A new robust and secure digital image watermarking scheme based on the integer wavelet transform and singular value decomposition," *Digit. Signal Process.*, vol. 33, pp. 134–147, Oct. 2014.



**RUI WANG** received the B.S. degree in applied mathematics from Tianjin University, in 2006, and the D.S. degree in applied mathematics from Nankai University, in 2011. She is currently a Lecturer with the College of Science, Civil Aviation University of China, Tianjin, China. Her research interests include image processing, information security, and coding theory.

**HAN SHAOCHENG** received the B.S. degree in information and communication engineering from the North University of China, in 2004, and the M.S. degree in signal and information processing from the Civil Aviation University of China, Tianjin, China, in 2009. He is currently an Associate Professor with the Basic Experimental Center, Civil Aviation University of China. His research interests include digital watermarking and information hiding.

**PENG ZHANG** received the B.S. degree in electronic information science and technology from the Zhengzhou University of Light Industry, Henan, China, in 2018. He is currently pursuing the master's degree with the College of Electronic Information and Automation, Civil Aviation University of China, Tianjin. His research interests include image processing and information hiding.

**MENG YUE** received the Ph.D. degree in information and communication engineering from Tianjin University, China, in 2017. He is currently a Lecturer with the School of Electronics, Information Engineering and Automation, Civil Aviation University of China. His current research interests include information security and cloud computing.

**ZHENG CHENG** received the B.S. degree in electronic and communication engineering from the Civil Aviation University of China, Tianjin, China, in 2014. He is currently an Assistant Experimentalist with the Basic Experimental Center, Civil Aviation University of China. His research interests include SAR image processing and machine learning.

**YUJIN ZHANG** (Member, IEEE) received the Ph.D. degree in communication and information system from Shanghai Jiao Tong University, Shanghai, China, in 2014. Since 2015, he has been with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science. His research interests include multimedia forensics, signal processing, artificial intelligence, and pattern recognition.