Local Geometric Distortions Resilient Watermarking Scheme Based on Symmetry

Zehua Ma, Weiming Zhang, Han Fang, Xiaoyi Dong, Linfeng Geng, and Nenghai Yu, *Member, IEEE*

***Abstract*— As an efficient watermark attack method, geometric distortions destroy the synchronization between the watermark encoder and decoder. Local geometric distortion is a considerable challenge in the watermarking field. Although many geometric distortion resilient watermarking schemes have been proposed, few perform well against local geometric distortions, such as random bending attacks (RBAs). To address this problem, this paper proposes a novel watermark synchronization process and a corresponding watermarking scheme. In our scheme, the watermark bits are represented by random patterns. The message is encoded to obtain a watermark unit, and the watermark unit is flipped to generate a symmetrical watermark. Then, the symmetrical watermark is additively embedded into the spatial domain of the host image. In watermark extraction, we first obtain the theoretical mean-square error minimized estimation of the watermark. Then, an autoconvolution function is applied to this estimation to detect the symmetry and obtain a watermark unit map. According to this map, the watermark can be accurately synchronized, and then extraction can be performed. Experimental results demonstrate the excellent robustness of the proposed watermarking scheme to local geometric distortions, global geometric distortions, common image processing operations, and some kinds of combined attacks.**

***Index Terms*— Digital watermark, watermark synchronization, local geometric distortions, random bending attack, autoconvolution function.**

I. Introduction

DIGITAL watermarks are widely used for ownership protection, authentication, annotation, etc. [1], [2]. An efficient watermarking scheme should be robust to a variety of distortions. In addition to the robustness toward common image processing, the robustness toward geometric distortions is equally significant. Common image processing, for example, filtering or compression, weakens the watermark by changing the value of pixels. However, geometric distortions destroy the synchronization between the watermark encoder and decoder. Under such distortions, the decoder can no longer decode the watermark without resynchronization even though the watermark still exists.

The geometric distortions can be divided into two categories. One is global geometric distortion, such as rotation, scaling, translation, and cropping. The other is local geometric distortion, such as random bending attacks (RBA) of Stirmark [1], [3]. The local geometric distortions are more complicated and harder to recover. It has been an open problem for years that designing a watermarking scheme performs well under local geometric distortions. In addition, many crossmedia watermarking schemes have been proposed recently, for example, print-scanning watermarking schemes [4], [5], print-camera watermarking schemes [6]–[8], and screen-camera watermarking schemes [9]–[12]. Considering that the crossmedia process introduces unnoticeable local geometric distortions [3], a watermark synchronization process resilient to local geometric distortions may improve the performance of such schemes.

Many geometric distortion resilient watermarking schemes have been proposed. They can be divided into the following five categories. The first category is the watermarking scheme using image normalization to resist geometric transforms [13]–[17]. By normalizing the image according to a set of predefined moment criteria, the watermarked image achieves invariance under affine transformation. However, image normalization-based methods have weak performance under cropping distortions. Cropping changes the moments calculated from the whole image. However, most image normalization-based methods have only one-bit capacity and high-computational costs.

The second category is based on a template [18]–[20]. A template is embedded in the Fourier domain. Before watermark extraction, the affine transformation can be estimated by comparing the detected template with the original template. Then, the inverse affine transformation is implemented, and the watermark can be extracted with a traditional method. However, the performance of template-based watermarks highly depends on the detection accuracy of the template. In addition, such watermarks are vulnerable to template removal attacks [21].

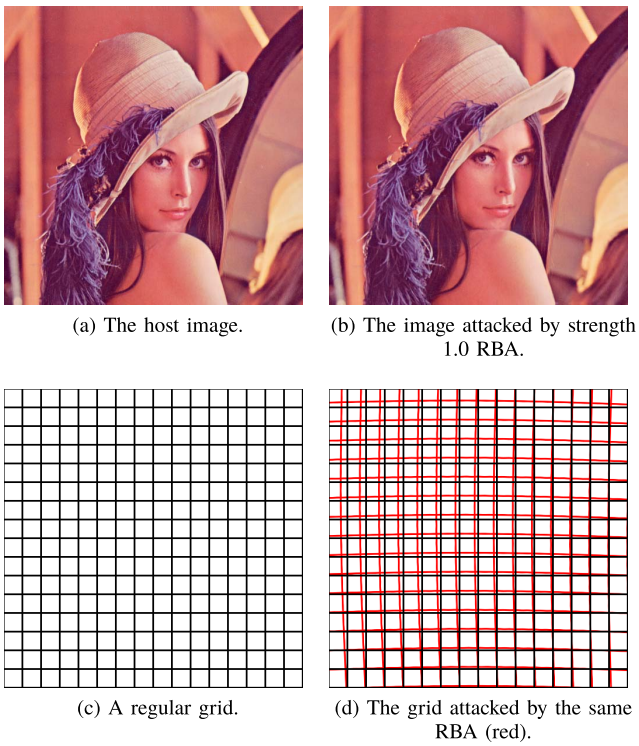
The third category is the watermarking scheme embedding the watermark in the geometric invariant domain [4], [22]–[25]. A rotation, scaling and translation (RST) invariant domain is obtained by applying the Fourier-Mellin transform (discrete Fourier transform and log-polar mapping). The RST operations in the spatial domain can be converted into

Manuscript received March 12, 2020; revised September 15, 2020; accepted January 19, 2021. Date of publication January 28, 2021; date of current version December 6, 2021. This work was supported in part by the Natural Science Foundation of China under Grant 62072421 and Grant 62002334, in part by the Anhui Science Foundation of China under Grant 2008085QF296, and in part by the Fundamental Research Funds for the Central Universities under Grant WK5290000001. This article was recommended by Associate Editor P. Bestagini. *(Corresponding author: Weiming Zhang.)*

The authors are with the CAS Key Laboratory of Electromagnetic Space Information, University of Science and Technology of China, Hefei 230026, China (e-mail: mzh045@mail.ustc.edu.cn; zhangwm@ustc.edu.cn; ynh@ustc.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3055255.

Digital Object Identifier 10.1109/TCSVT.2021.3055255



(a) The host image. (b) The image attacked by strength 1.0 RBA. (c) A regular grid. (d) The grid attacked by the same RBA (red).

Fig. 1. Two images attacked with the same RBA.

parameter translation in the Fourier-Mellin domain. By embedding and extracting the watermark in the Fourier-Mellin domain, there is no need to estimate and invert the geometric distortions. The translation can be easily recovered by using tracking sequences embedded along with the informative watermark. Kang *et al.* [4] proposed a uniform log-polar mapping (ULPM)-based watermarking scheme that eliminates the interpolation distortion and interference distortion introduced in the embedding process. Urvoy *et al.* proposed a perceptual discrete Fourier transform (DFT) watermarking scheme [26], whose embedding strategy can help these DFT-based watermarking schemes obtain perceptually optimal visibility.

However, local geometric distortion is still a challenge to the watermarking schemes belonging to the aforementioned three categories. Comparing Fig. 1(b) to Fig. 1(a), the distortion caused by RBA is almost unnoticeable. However, the distortion is severe, as shown in Fig. 1(d). Frequency values and moments that are calculated from the whole image also change severely after RBA, which severely limits the performance of these watermarking schemes.

The feature-based watermarking schemes belong to the fourth category [16], [27]–[32]. By binding the watermark with the global or local geometrically invariant features, the watermarking scheme obtains corresponding robustness. Most of them are one-bit watermarking schemes. In [32], a multibit watermarking scheme was proposed based on the local daisy feature transform (LDFT). It is resilient to many kinds of local geometric distortions. However, its robustness to common image processing operations and global geometric distortions is weak. A possible reason is that local geometrically invariant features are not robust enough to other distortions. In addition, compared to other watermarking schemes, feature-based methods usually have higher computational costs.

The last category is the watermarking scheme using a periodic watermark as the calibration signal [33]–[37]. A watermark unit is periodically tiled to generate a periodic watermark. In watermark extraction, the period can be detected by an autocorrelation function (ACF) or magnitude spectrum (MS), and such a period can act as a reference to recover global geometric distortions. Considering that local geometric distortions can be regarded as a set of affine transformations in the local region, Voloshynovskiy *et al.* [37] proposed using local ACF to detect and restore local geometric distortions. Moreover, as stated in the paper proposing template removal attack [21], methods in this category are resilient to this attack.

The contributions of this paper can be summarized as follows:

- We propose using symmetry as a calibration signal to synchronize the watermark. Compared to the synchronization process based on periodicity, the advantages and disadvantages of the proposed synchronization process are discussed.
- To the discussed disadvantages of the synchronization process based on symmetry, the corresponding solutions are proposed:
  - We derive the relationship between symmetry and the autoconvolution function and propose a fast method to calculate the symmetry.
  - We propose using hypothesis testing to determine the extra watermark states introduced by flipping.
- Based on the proposed synchronization process, the corresponding watermarking scheme is proposed. In addition to RBA, the proposed watermarking method performs well under global geometric distortions, common image processing operations, and some combined attacks.

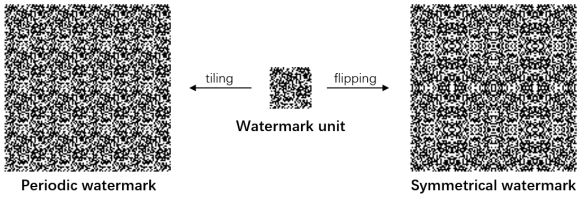
The rest of this paper is organized as follows. In Section II, a universal watermark synchronization process based on periodicity is illustrated first. Then, the difference between it and the proposed synchronization process based on symmetry is discussed. In Section III and IV, we illustrate the watermark embedding process and the extraction process of the proposed watermarking scheme. The experimental results are shown and discussed in Section V. Section VI concludes.

II. Watermark Synchronization Process Analysis

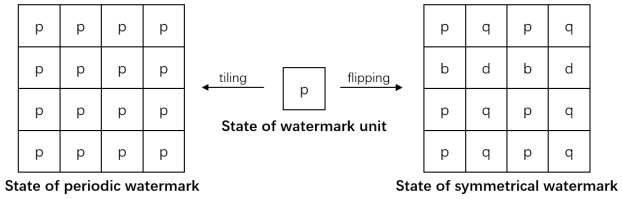
In this section, we discuss and compare periodicity and symmetry and their application to watermark synchronization to illustrate the advantages of symmetry as the watermark calibration signal.

A. Universal Periodic Watermark Synchronization Process

The periodic watermarking schemes, as mentioned in Section I, have been widely used. Among them, the scheme proposed by Voloshynovskiy *et al.* [37] is one of the most representative schemes. That watermarking scheme is robust to local geometric distortions, including random bending attacks. We first illustrate its watermark synchronization process to support later discussion.



(a) An example of watermark generation.



(b) The corresponding state change of (a).

Fig. 2. (a) is an example of generating a 4×4 periodic watermark by tiling a watermark unit and generating a 4×4 symmetrical watermark by flipping the same watermark unit. Using symbol 'p' to represent the state of the watermark unit, (b) is the corresponding state change in (a).

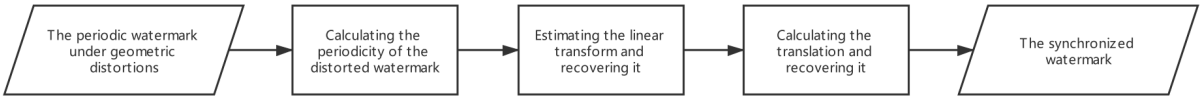


Fig. 3. The flowchart of the universal watermark synchronization process based on periodicity.

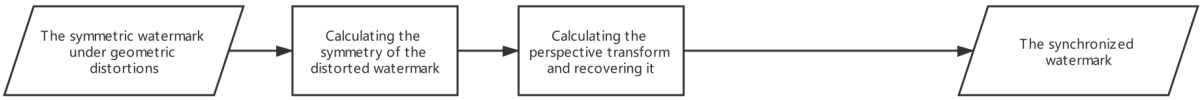


Fig. 4. The flowchart of the proposed watermark synchronization process based on symmetry.

In [37], the watermark unit, named the macroblock, consists of an informative watermark and a reference watermark. As shown in Fig. 2(a), the watermark unit is tiled to generate a periodic watermark. In the case of a rectangular macroblock, the watermark $\boldsymbol{W}$ has the following property:

$$W(x, y) = W(x + m\ T_{len}, y + n\ T_{wid}) \tag{1}$$

where m and n are integers, and T_{len} and T_{wid} are the size of the macroblock. The periodicity $\boldsymbol{P}$ of $\boldsymbol{W}$ is commonly defined as the autocorrelation of $\boldsymbol{W}$:

$$P(i, j) = \sum_x \sum_y W(x, y) W(x + i, y + j) \tag{2}$$

in which $P(i, j)$ represents the correlation between $\boldsymbol{W}$ and the $\boldsymbol{W}$ translated by vector $\boldsymbol{(i, j)}$. $\boldsymbol{P}$ can also be computed by the frequency form of ACF:

$$\boldsymbol{P} = IFFT[FFT(\boldsymbol{W}) FFT(\boldsymbol{W})^*] \tag{3}$$

where FFT represents the fast Fourier transform, $IFFT$ is the corresponding inverse transform, and the $*$ operator denotes complex conjugation. The periodicity changes along with the global geometric distortions.

The case of global geometric distortion is considered first. Most global geometric distortions can be uniquely described by an affine transformation, and the affine transformation can be represented by a linear component matrix $\boldsymbol{A}$, plus a translation component $\boldsymbol{v}$. Considering that the translation can be separately recovered, Voloshynovskiy *et al.* [37] proposed using an approach based on penalized maximum likelihood (ML) estimation to estimate the linear component $\boldsymbol{A}$:

$$\hat{\boldsymbol{A}} = \arg\min_{\boldsymbol{A} \in \Phi} \left\{ \rho \left(\begin{pmatrix} x' \\ y' \end{pmatrix} - \boldsymbol{A} \begin{pmatrix} x \\ y \end{pmatrix} \right) + \mu \Omega(\boldsymbol{A}) \right\} \tag{4}$$

where $\hat{\boldsymbol{A}}$ is the optimal estimation within the set of possible solutions $\boldsymbol{A} \in \Phi$, ρ denotes the cost function, and (x, y) and (x', y') represent the Cartesian coordinates of ACF peaks before and after affine transformation. The last term $\mu\Omega(\boldsymbol{A})$ is weighted prior knowledge, restricting the variations of parameters in $\boldsymbol{A}$. After recovering the linear transformation of the affine transformed watermark, the translation $\boldsymbol{v}$ can be easily recovered, for example, based on an intercorrelation between the extracted watermark and the reference watermark mentioned above.

Considering that the local geometric distortions, such as RBA and perspective transformation, can be approximated as affine transformations in the local region, local geometric distortions can be approximated by a similar process. We can further summarize the watermark synchronization process proposed in [37] as the flowchart shown in Fig. 3.

It should be noted that even though flipping appeared in the scheme proposed by Voloshynovskiy *et al.* [37] for some purposes, the synchronization process of the scheme is still based on periodicity and follows the above discussion.

B. The Proposed Symmetrical Watermark Synchronization Process

However, we believe that the watermark synchronization process based on symmetry may have some advantages over that based on periodicity. These advantages come from the difference between symmetry and periodicity and contribute to a simpler watermark synchronization process, as shown in Fig. 4. Similarly, we define the symmetrical watermark and its symmetry first. In our scheme, as shown in Fig. 2(a), the symmetrical watermark $\boldsymbol{W}$ is generated by flipping the

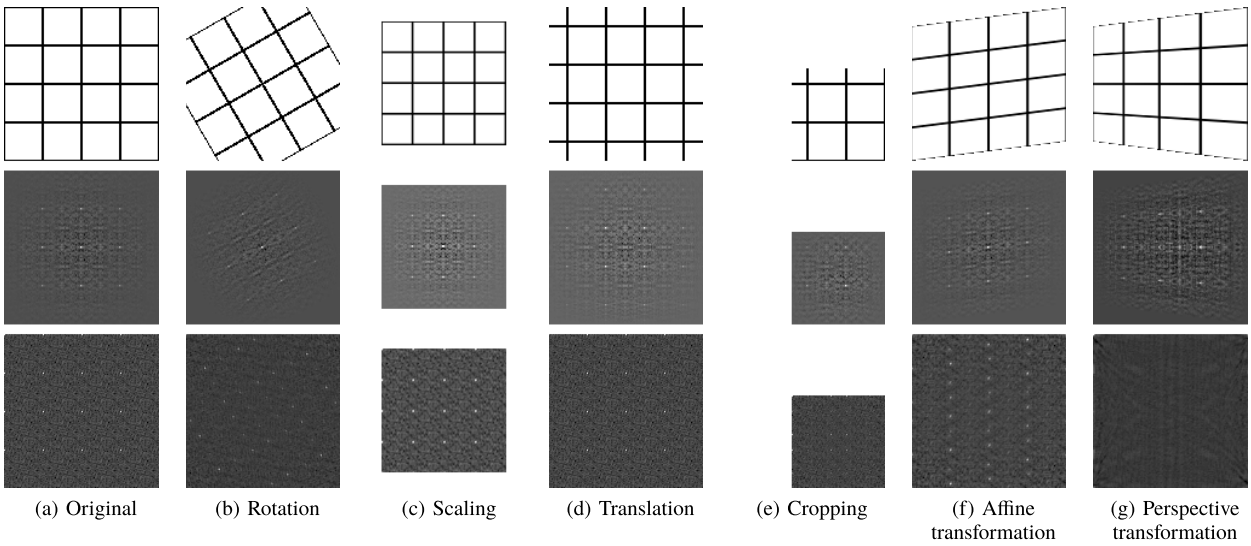


Fig. 5. *Top row:* The grids under the same geometric distortions with watermarks, representing the states of the watermark units; *Middle row:* The symmetry of the symmetrical watermark in Fig. 2(a) under geometric distortions; *Bottom row:* The periodicity of the periodic watermark in Fig. 2(a) under geometric distortions.

rectangular watermark unit, and it satisfies:

$$W(x_s - x, y_s - y) = W(x_s + x, y_s + y) \tag{5}$$

where (x_s, y_s) are the coordinates of one symmetrical center of $\boldsymbol{W}$. In the symmetrical watermark $\boldsymbol{W}$, the symmetrical centers are the corners of the watermark unit. Similar to periodicity, the symmetry $\boldsymbol{S}$ of the watermark about a point (i, j) can be defined by:

$$S(i, j) = \sum_x \sum_y W(i - x, j - y) W(i + x, j + y). \tag{6}$$

$S(i, j)$ is actually the correlation between $\boldsymbol{W}$ and the $\boldsymbol{W}$ flipped around point (i, j).

We believe the most important difference between symmetry and periodicity is that there is a clearer mapping between symmetrical peaks and watermark position. Periodicity is a translation characteristic, and symmetry is a position characteristic. Specifically, in Eq. (2), the coordinates of $\boldsymbol{P}$ represent a translation or a vector, and the corresponding value in $\boldsymbol{P}$ denotes the possibility that the period of $\boldsymbol{W}$ is that vector. The coordinates of $\boldsymbol{P}$ and $\boldsymbol{W}$ have different meanings. However, in Eq. (6), the coordinates of $\boldsymbol{S}$ and $\boldsymbol{W}$ have the same meaning, indicating the position. Given a point (i, j), the value of $S(i, j)$ represents the possibility that point (i, j) is one of the symmetrical centers of $\boldsymbol{W}$. In the ideal case, the positions of symmetrical peaks in $\boldsymbol{S}$ are the positions of symmetrical centers in $\boldsymbol{W}$.

Considering that symmetrical centers are corners of watermark units, knowing the position of symmetrical peaks is equivalent to knowing the position of watermark units and corresponding geometric distortions. Fig. 5 shows the periodicity of the periodic watermark and the symmetry of the symmetrical watermark under various geometric distortions, in which the watermarks in Fig. 2(a) are used as samples and their periodicity and symmetry are calculated by Eq. (2) and Eq. (6), respectively. Observing these peaks in Fig. 5, we find that symmetrical peaks show the position of the current watermark units and the experienced geometric distortions more clearly. Moreover, if the four corner points of a watermark unit are known, the position of the unit can be determined, and most of its geometric distortions can be recovered, even including perspective transformation, as shown in Fig. 5(g). Therefore, the watermark synchronization process based on symmetry shown in Fig. 4 becomes simpler.

As a summary of the above discussion, the advantages of the watermark synchronization process based on symmetry can be listed as follows:

1) The process does not require estimating translation. Comparing Fig. 5(a) and Fig. 5(d), the symmetrical peaks show the position of the watermark unit after translation, while periodic peaks remain the same. Reducing the estimation steps contributes to a simpler synchronization process, which means lower computational costs and higher accuracy. Additionally, the redundancy for estimating translation, such as the reference watermark, is no longer needed in the proposed watermark unit.
2) The process performs better under local geometric distortions. In the proposed synchronization process, the geometric distortion on a watermark unit is regarded as a perspective transformation and recovered based on the four detected corners of the watermark unit. Additionally, the synchronization process based on periodicity uses affine transformations to approximate local geometric distortions, including perspective transformation and RBA. Compared to affine transformation, we think that perspective transformation has more parameters and fits better when approximating local geometric distortions.

In addition, the proposed watermark synchronization process based on symmetry has the potential to become an improved version of the process based on periodicity.

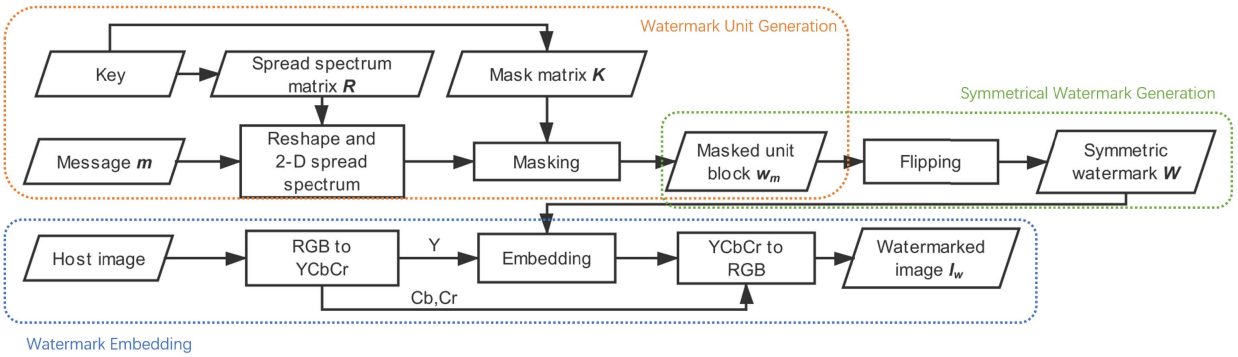


Fig. 6. Watermark embedding process.

Generating the watermark by flipping rather than tiling and detecting the watermark by symmetrical peaks rather than periodic peaks, most periodic watermarking schemes following the synchronization process in Fig. 3 can be modified to the corresponding symmetrical watermarking schemes. If the modification does not introduce new disadvantages, these modified watermarking schemes will perform better.

These disadvantages exist and may limit the performance of the synchronization process based on symmetry. These disadvantages can be listed as follows:

1) The current process lacks a fast method for calculating symmetry. Calculating symmetry in the space domain as in Eq. (6) is inefficient, especially when the host image is large. This problem also exists in the document watermarking scheme proposed by Fang *et al.* [6].
2) The flipping introduces additional states of the watermark unit, as shown in Fig. 2(b), which need to be determined before decoding. A common solution is introducing predefined redundancy in the watermark as a reference. However, such solutions create redundancy, which is released from estimating translation.

In Section III and IV, the proposed watermarking scheme based on symmetry is illustrated, which overcomes the disadvantages mentioned above. Specifically, we discover the relationship between symmetry and the autoconvolution function, similar to periodicity and the autocorrelation function, and obtain a lower computational cost version of Eq. (6). We also propose using hypothesis testing to determine the state of the watermark rather than introducing redundancy. The advantages of symmetry are retained in the proposed scheme to obtain better performance.

III. Watermark Embedding

Fig. 6 shows the framework of the proposed watermark embedding process. It can be divided into three main modules: watermark unit generation, symmetrical watermark generation, and watermark embedding. The implementation details of these modules are illustrated as follows.

A. Watermark Unit Generation

By applying a key, we generate a 2-D bipolar random matrix $\boldsymbol{r}$ of size L_r, $\boldsymbol{r}_{i,j} \in \{-1,1\}$, $i, j = 1, 2 \ldots, L_r$. Then, $\boldsymbol{r}$ is doubly upsampled to obtain $\boldsymbol{R}$. Compared to $\boldsymbol{r}$, the components

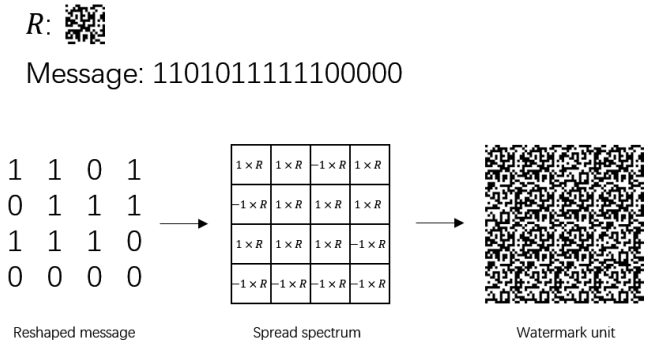


Fig. 7. An example of using $\boldsymbol{R}$ to spread spectrum.

of $\boldsymbol{R}$ are mainly distributed in the middle frequency and more robust to common image processing operations. The size of $\boldsymbol{R}$ is L_R, $L_R = 2L_r$. The watermark sequence is first encoded using an error correction code (ECC) to obtain the message $\boldsymbol{m}$ of length L_m. Then, we reshape $\boldsymbol{m}$ to a 2-D matrix of size $p \times q$, where $p \times q \geq L_m$. Then, $\boldsymbol{m}$ is spread-spectrum encoded using $\boldsymbol{R}$ to obtain a watermark unit $\boldsymbol{w}$. As Fig. 7 shows, in $\boldsymbol{w}$, bit '1' is presented as $+1 \times \boldsymbol{R}$ and bit '0' is presented as $-1 \times \boldsymbol{R}$. Therefore, the size of $\boldsymbol{w}$ is L_R times the size of $\boldsymbol{m}$. Specifically, the length and width of $\boldsymbol{w}$ are defined as m and n, where $m = p \times L_R$, $n = q \times L_R$. Finally, $\boldsymbol{w}$ is masked with a mask matrix $\boldsymbol{K}$, which is also generated from a key and doubly upsampled to the same size of $\boldsymbol{w}$. The masked watermark unit $\boldsymbol{w}_m$ is calculated by

$$w_m(i, j) = w(i, j)K(i, j) \tag{7}$$

where i and j denote the index of the matrix. The masking operation is effective and necessary because it can

1) Offer basic informative security. Without $\boldsymbol{K}$, even if the watermarking scheme is a white box, the adversary cannot accurately extract the message.
2) Eliminate the impact of weak messages. Similar to a weak key in cryptography, a weak message refers to a kind of message that makes the watermark synchronization process behave in some undesirable ways. For example, a symmetrical 2-D message is a kind of weak message. The symmetry of the message and the symmetry generated by flipping are both detected and confuse watermark synchronization. To eliminate the impact of

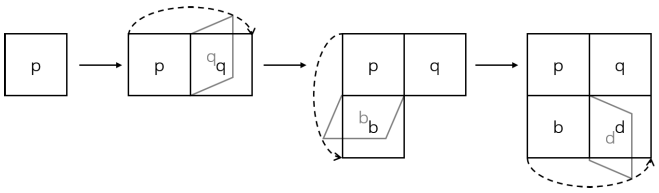


Fig. 8. An example of generating symmetry by flipping the watermark unit.

weak messages, watermark unit $\boldsymbol{w}$ is masked with a mask matrix $\boldsymbol{K}$. Because of the spread-spectrum matrix, the information rate of $\boldsymbol{K}$ is L_R^2 times that of $\boldsymbol{w}$. The difference in the information rate makes the properties of the masked watermark unit $\boldsymbol{w}_m$ more dependent on $\boldsymbol{R}$ than on $\boldsymbol{w}$. Therefore, after masking, most properties of the message are eliminated, and $\boldsymbol{w}$ will be close to a random matrix. As a result, weak messages will no longer impact the synchronization process, and the proposed watermarking scheme can reach its theoretical information rate.

3) Help judge the state of the watermark unit. This part is illustrated in Section IV-C.

B. Symmetrical Watermark Generation

We flip the masked watermark unit $\boldsymbol{w}_m$ to create the complete watermark $\boldsymbol{W}$. In this paper, flipping vertically is defined as flipping along the central horizontal axis of the block, and flipping horizontally is defined as flipping along the central vertical axis of the block. Fig. 8 shows an example of generating symmetry by flipping the watermark unit in which we use the symbol 'p' to represent the watermark unit. The symmetrical watermark is generated by following a *flipping rule*:

- *flipping rule*: The next horizontally adjacent watermark unit is generated by flipping the former unit horizontally, and the next vertically adjacent watermark unit is generated by flipping the former unit vertically.

For a specific image $\boldsymbol{I}$, $\boldsymbol{w}_m$ is flipped repeatedly until the size of $\boldsymbol{W}$ is larger than the size of $\boldsymbol{I}$. Then, we crop $\boldsymbol{W}$ to the size of $\boldsymbol{I}$ to obtain the watermark for embedding.

Note that the flipping process is self-consistent. Different flipping orders will generate the same symmetrical watermark $\boldsymbol{W}$. To this $\boldsymbol{W}$, the adjacent line of two watermark units is the symmetrical axis of $\boldsymbol{W}$, and the adjacent point of four watermark units is the symmetrical center of $\boldsymbol{W}$.

C. Watermark Embedding in Spatial Domain

Watermark is embedded into the luminance of the host image additively. If the input is a color image, we convert it to $YCbCr$ space and take component Y as the luminance, called $\boldsymbol{I}$. To balance robustness and imperceptibility, an adaptive watermark strength strategy is a common solution, for example, the adaptive embedding strategy proposed in [38]. In this paper, we use a simple strategy that embeds watermarks with higher strength on regions with complex textures and with lower strength on regions with simple textures. The complexity of the image could be measured by the local variance of $\boldsymbol{I}$.

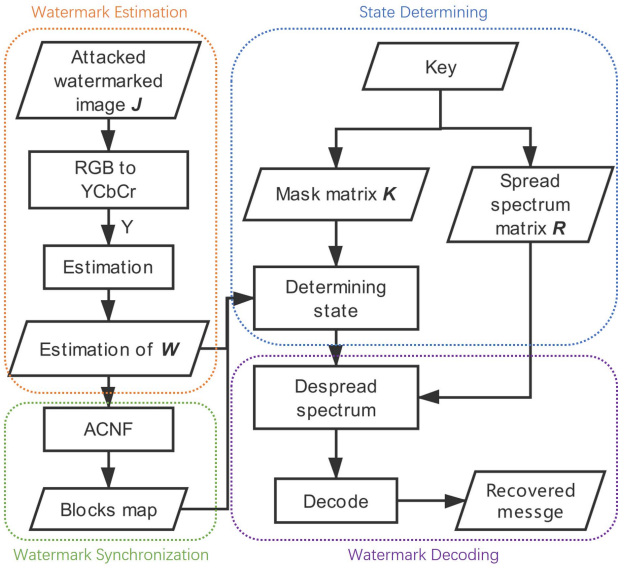


Fig. 9. Watermark extraction process.

Therefore, the adaptive watermark strength $\boldsymbol{s}$ is defined as follows:

$$s(i,j) = \mathcal{F}(\sigma_{\boldsymbol{I}}^2(i,j)) \tag{8}$$

where $\boldsymbol{\sigma}_{\boldsymbol{I}}^2$ is the local variance of $\boldsymbol{I}$ and $\mathcal{F}$ is a nonlinear function. In the proposed scheme, $\mathcal{F}$ is defined as:

$$\mathcal{F}(x) = \begin{cases} \alpha, & if \quad log_2(x) < \alpha \\ log_2(x), & otherwise \end{cases} \tag{9}$$

where α is a global embedding strength and set to 2 in our scheme. Then, the watermarked image $\boldsymbol{I}_w$ is generated by

$$I_w(i,j) = I(i,j) + s(i,j)W(i,j). \tag{10}$$

The symmetrical watermark can be slightly predistorted to resist spatial averaging and removal attacks [37]. This predistortion does not significantly affect the symmetry of the watermark used for the recovery of geometric distortions. In the end, a $YCbCr$ to RGB transform is applied if the host image is a color image.

IV. Watermark Extraction

As a blind watermarking scheme, the watermark extractor has no prior knowledge of the host image. However, the key is shared between the encoder and decoder, so the extractor can generate the same spread-spectrum matrix $\boldsymbol{R}$ and the mask matrix $\boldsymbol{K}$. The watermark extraction process is illustrated in Fig. 9 and can be divided into four main modules: watermark estimation, watermark synchronization, watermark state determining, and watermark decoding. These modules are further explained as follows.

A. Watermark Estimation

Part of the discussion in this section refers to [39]. The watermark component is predicted from the attacked watermarked image $\boldsymbol{J}$ received by the extractor. We denote the

distortion and noise as $\boldsymbol{n}$. Then, $\boldsymbol{J}$ can be represented as:

$$J(i,j) = I(i,j) + W(i,j) + n(i,j). \tag{11}$$

$\boldsymbol{n}$ is assumed to be zero-mean white noise for the purpose of tractability. Considering that $\boldsymbol{w}_m$ is almost a random matrix and $\boldsymbol{W}$ is generated by $\boldsymbol{w}_m$, $\boldsymbol{W}$ is similar to additive random noise, similar to $\boldsymbol{n}$. Therefore, $\boldsymbol{J}$ and $\boldsymbol{I}$ have the same local mean $\boldsymbol{\mu}$. Eq. (11) can be further represented as:

$$J'(i,j) = I'(i,j) + W(i,j) + n(i,j) \tag{12}$$

where $\boldsymbol{I}'$ and $\boldsymbol{J}'$ are residual components such that $\boldsymbol{I}' = \boldsymbol{I} - \boldsymbol{\mu}$ and $\boldsymbol{J}' = \boldsymbol{J} - \boldsymbol{\mu}$. We hope to find a kind of $\boldsymbol{h}$ so that $\boldsymbol{W}$ can be estimated as follows:

$$\widehat{\boldsymbol{W}} = \boldsymbol{J}' \otimes \boldsymbol{h} \tag{13}$$

where $\otimes$ represents the convolution operation and $\widehat{\boldsymbol{W}}$ is an estimate of $\boldsymbol{W}$. $\boldsymbol{h}$ is the convolution kernel that can minimize the mean-square error:

$$\boldsymbol{h} = \arg\min\ E\{(\boldsymbol{W} - \widehat{\boldsymbol{W}})^2\} \tag{14}$$

where $E\{\cdot\}$ represents the expectation. Substituting $\widehat{\boldsymbol{W}}$ with Eq. (13), the mean-square error can be rewritten as:

$$E\{(\boldsymbol{W} - \widehat{\boldsymbol{W}})^2\} = E\{(\boldsymbol{W} - \boldsymbol{J}' \otimes \boldsymbol{h}\}. \tag{15}$$

Substituting $\boldsymbol{J}'$ with Eq. (12), Eq. (15) can be rewritten in the frequency domain as:

$$\begin{aligned} E\{(\boldsymbol{W} - \widehat{\boldsymbol{W}})^2\} &= E\{(\overline{\boldsymbol{W}} - \overline{\boldsymbol{J}'}\boldsymbol{H})^2\} \\ &= E\{(\overline{\boldsymbol{W}} - (\overline{\boldsymbol{I}'} + \overline{\boldsymbol{W}} + \boldsymbol{N})\boldsymbol{H})^2\} \\ &= E\{(\overline{\boldsymbol{W}}(1-\boldsymbol{H}) - \overline{\boldsymbol{I}'}\boldsymbol{H} - \boldsymbol{N}\boldsymbol{H})^2\} \\ &= E\{\overline{\boldsymbol{W}}^2(1-\boldsymbol{H})^2 + \overline{\boldsymbol{I}'}^2\boldsymbol{H}^2 + \boldsymbol{N}^2\boldsymbol{H}^2 + \boldsymbol{Z}\} \end{aligned} \tag{16}$$

where $\overline{\boldsymbol{I}'}$, $\overline{\boldsymbol{J}'}$, $\overline{\boldsymbol{W}}$, $\boldsymbol{N}$, $\boldsymbol{H}$ are the frequency forms of $\boldsymbol{I}'$, $\boldsymbol{J}'$, $\boldsymbol{W}$, $\boldsymbol{n}$, $\boldsymbol{h}$, respectively. $\boldsymbol{Z}$ is the summation of the crossproduct. Specifically,

$$\begin{aligned} \boldsymbol{Z} = &-\overline{\boldsymbol{W}}(1-\boldsymbol{H})\overline{\boldsymbol{I}'}^*\boldsymbol{H}^* - \overline{\boldsymbol{W}}^*(1-\boldsymbol{H})^*\overline{\boldsymbol{I}'}\boldsymbol{H} \\ &-\overline{\boldsymbol{W}}(1-\boldsymbol{H})\boldsymbol{N}^*\boldsymbol{H}^* \\ &-\overline{\boldsymbol{W}}^*(1-\boldsymbol{H})^*\boldsymbol{N}\boldsymbol{H} \\ &+\overline{\boldsymbol{I}'}\boldsymbol{H}\boldsymbol{N}^*\boldsymbol{H}^* + \overline{\boldsymbol{I}'}^*\boldsymbol{H}^*\boldsymbol{N}\boldsymbol{H} \\ = &-\overline{\boldsymbol{W}}\overline{\boldsymbol{I}'}^*(1-\boldsymbol{H})\boldsymbol{H}^* \\ &-\overline{\boldsymbol{W}}^*\overline{\boldsymbol{I}'}(1-\boldsymbol{H})^*\boldsymbol{H} - \overline{\boldsymbol{W}}\boldsymbol{N}^*(1-\boldsymbol{H})\boldsymbol{H}^* \\ &-\overline{\boldsymbol{W}}^*\boldsymbol{N}(1-\boldsymbol{H})^*\boldsymbol{H} \\ &+\overline{\boldsymbol{I}'}\boldsymbol{N}^*\boldsymbol{H}\boldsymbol{H}^* + \overline{\boldsymbol{I}'}^*\boldsymbol{N}\boldsymbol{H}^*\boldsymbol{H}. \end{aligned} \tag{17}$$

Note that the cross-correlation of two variables can be calculated by the conjugate product of their frequency form. Because the watermark $\boldsymbol{W}$, residual component $\boldsymbol{I}'$ and noise $\boldsymbol{n}$ are independent, the expectation of their cross-correlation should be zero. Therefore, the expectation of $\boldsymbol{Z}$ is

$$\begin{aligned} E\{\boldsymbol{Z}\} = &-E\{\overline{\boldsymbol{W}}\overline{\boldsymbol{I}'}^*\}(1-\boldsymbol{H})\boldsymbol{H}^* - E\{\overline{\boldsymbol{W}}^*\overline{\boldsymbol{I}'}\}(1-\boldsymbol{H})^*\boldsymbol{H} \\ &-E\{\overline{\boldsymbol{W}}\boldsymbol{N}^*\}(1-\boldsymbol{H})\boldsymbol{H}^* - E\{\overline{\boldsymbol{W}}^*\boldsymbol{N}\}(1-\boldsymbol{H})^*\boldsymbol{H} \\ &+E\{\overline{\boldsymbol{I}'}\boldsymbol{N}^*\}\boldsymbol{H}\boldsymbol{H}^* + E\{\overline{\boldsymbol{I}'}^*\boldsymbol{N}\}\boldsymbol{H}^*\boldsymbol{H} \\ = &\ 0. \end{aligned} \tag{18}$$

Combining Eq. (16) and Eq. (18), we have:

$$\begin{aligned} E\{(\boldsymbol{W} - \widehat{\boldsymbol{W}})^2\} &= E\{\overline{\boldsymbol{W}}^2\}(1-\boldsymbol{H})^2 + E\{\overline{\boldsymbol{I}'}^2\}\boldsymbol{H}^2 \\ &\quad + E\{\boldsymbol{N}^2\}\boldsymbol{H}^2 + E\{\boldsymbol{Z}\} \\ &= P_{\boldsymbol{W}}(1-\boldsymbol{H})^2 + P_{\boldsymbol{I}'}\boldsymbol{H}^2 + P_{\boldsymbol{N}}\boldsymbol{H}^2 + 0 \\ &= P_{\boldsymbol{W}}(1-\boldsymbol{H})^2 + (P_{\boldsymbol{I}'} + P_{\boldsymbol{N}})\boldsymbol{H}^2 \end{aligned} \tag{19}$$

where $P_{\boldsymbol{W}}$, $P_{\boldsymbol{I}'}$, $P_{\boldsymbol{N}}$ are the power spectra of $\boldsymbol{W}$, $\boldsymbol{I}'$, $\boldsymbol{N}$. We find that the mean-square error is a quadratic function of $\boldsymbol{H}$. To find the minimum error value, the derivative of Eq. (19) is calculated and set to zero. Then, we have:

$$\begin{aligned} \boldsymbol{H} &= \frac{P_{\boldsymbol{W}}}{P_{\boldsymbol{W}} + P_{\boldsymbol{I}'} + P_{\boldsymbol{N}}} \\ &= \frac{P_{\boldsymbol{W}}}{P_{\boldsymbol{J}'}} \end{aligned} \tag{20}$$

where $P_{\boldsymbol{J}'}$ is the power spectrum of $\boldsymbol{J}'$. From Eq. (20), we can obtain that $h(i,j)$ is a scaled impulse given by:

$$h(i,j) = \frac{P_W}{P_{J'}}\delta(i,j) \tag{21}$$

where δ is a unit impulse. Substituting the right side of Eq. (13) with Eq. (21), the watermark $\boldsymbol{W}$ within the local region can be expressed as:

$$\begin{aligned} \widehat{\boldsymbol{W}} &= \boldsymbol{J}' \otimes \frac{P_{\boldsymbol{W}}}{P_{\boldsymbol{J}'}}\boldsymbol{\delta} \\ &= (\boldsymbol{J} - \boldsymbol{\mu})\frac{P_{\boldsymbol{W}}}{P_{\boldsymbol{J}'}}. \end{aligned} \tag{22}$$

Considering that $\boldsymbol{J}'$ and W are both zero means in the local region, their power spectra are their local variance. Therefore, the mean-square error minimized estimation of $\boldsymbol{W}$ can be calculated by:

$$\widehat{\boldsymbol{W}} = (\boldsymbol{J} - \boldsymbol{\mu})\frac{\boldsymbol{\sigma}^2_{\boldsymbol{W}}}{\boldsymbol{\sigma}^2_{\boldsymbol{J}'}}. \tag{23}$$

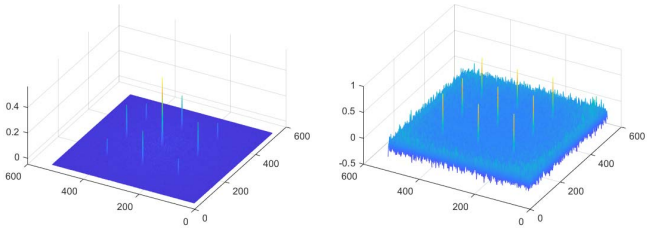
where $\boldsymbol{\sigma}^2_{\boldsymbol{W}}$ and $\boldsymbol{\sigma}^2_{\boldsymbol{J}'}$ are the local variances of $\boldsymbol{W}$ and $\boldsymbol{J}'$, respectively. We can calculate $\boldsymbol{\sigma}^2_{\boldsymbol{J}'}$ from the attacked watermarked image $\boldsymbol{J}$. Note that $\boldsymbol{W}$ is similar to random noise, so its local variance $\boldsymbol{\sigma}^2_{\boldsymbol{W}}$ depends on its embedding strength $\boldsymbol{s}$, which can be estimated from $\boldsymbol{J}$ using Eq. (8).

B. Watermark Synchronization Based on Symmetry

To synchronize the masked watermark unit $\boldsymbol{w}_m$, the symmetry of $\widehat{\boldsymbol{W}}$ should be calculated first. In Section. II-B, we defined the symmetry of the symmetrical watermark and proposed Eq. (6) to calculate it. Additionally, in Section. II-B, we believe that a lower computational cost version of Eq. (6) is needed.

First, Eq. (6) can be rewritten as:

$$\begin{aligned} S(i,j) &= \sum_x \sum_y \widehat{W}(i-x, j-y)\widehat{W}(i+x, j+y) \\ &= \sum_x \sum_y \widehat{W}(x,y)\widehat{W}(2i-x, 2j-y) \end{aligned} \tag{24}$$



(a) Original symmetrical peaks. (b) Uniform symmetrical peaks.

Fig. 10. Original and uniform symmetrical peaks generated from the same symmetrical watermark. The symmetrical watermark has 4×4 watermark units.

where x and y run over all values that lead to legal subscripts of $\widehat{\boldsymbol{W}}$. Defining a temporary matrix $\boldsymbol{T}$, $T(2i, 2j) = S(i, j)$, we have:

$$T(2i, 2j) = \sum_x \sum_y \widehat{W}(x, y)\widehat{W}(2i - x, 2j - y)$$
$$T(u, v) = \sum_x \sum_y \widehat{W}_p(x, y)\widehat{W}_p(u - x, v - y) \quad (25)$$

where $\widehat{\boldsymbol{W}}_p$ is $\widehat{\boldsymbol{W}}$ zero-padding to double the original size. We discover that $\boldsymbol{T}$ is the autoconvolution of $\widehat{\boldsymbol{W}}_p$. Similar to the periodicity and autocorrelation function (ACF), symmetry has a relationship with the autoconvolution function. In this paper, the **A**uto-**C**o**N**volution **F**unction is abbreviated as ACNF to distinguish it from ACF. Therefore, the convolution theorem can be used to obtain the frequency form of Eq. (25) as follows:

$$\boldsymbol{T} = IFFT[FFT(\widehat{\boldsymbol{W}}_p)FFT(\widehat{\boldsymbol{W}}_p)] \quad (26)$$

where FFT represents the fast Fourier transform and $IFFT$ is the corresponding inverse transform. According to the definition of $\boldsymbol{T}$, $\boldsymbol{T}$ is actually the double upsampling of $\boldsymbol{S}$, so the symmetry $\boldsymbol{S}$ of $\widehat{\boldsymbol{W}}$ can be calculated by:

$$\boldsymbol{S} = \mathcal{D}(IFFT[FFT(\widehat{\boldsymbol{W}}_p)FFT(\widehat{\boldsymbol{W}}_p)]) \quad (27)$$

where $\mathcal{D}(\cdot)$ is a downsampling function that scales its input matrix to the half size. Using the frequency form of ACNF, from Eq. (24) to Eq. (27), the computational cost of calculating $\boldsymbol{S}$ is greatly reduced.

Another problem is that, in Eq. (24), different i and j will contribute different summation sizes. As a result, $\boldsymbol{S}$ has higher peaks in the middle region, as shown in Fig. 10(a). This phenomenon is also reflected in the brightness of symmetrical peaks in Fig. 5. Actually, in the spatial domain, the uniform $\boldsymbol{S}'$ is easy to obtain by dividing $\boldsymbol{S}$ by its summation size as follows:

$$S'(i, j) = \frac{\sum_x \sum_y \widehat{W}(i + x, j + y)\widehat{W}(i - x, j - y)}{\sum_x \sum_y 1} \quad (28)$$

where $S'(i, j)$ is a uniform symmetry generated by scaling $S(i, j)$ to the unit value. It is difficult to find the frequency form of Eq. (28) directly. Kang *et al.* [4] proposed customized phase correlation to solve a similar problem and obtain an approximate solution. In our scheme, we first calculate the numerator and denominator of Eq. (28) separately and then calculate their quotient. We note that the denominator also has frequency form similar to the numerator. Thus, the uniform symmetry $\boldsymbol{S}'$ can be calculated by:

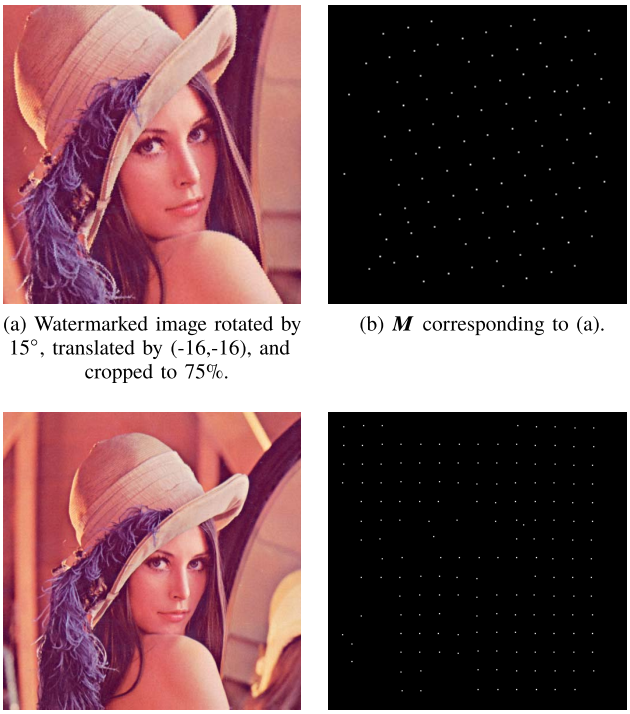
$$\boldsymbol{S}' = \mathcal{D}\left(\frac{IFFT[FFT(\widehat{\boldsymbol{W}}_p)FFT(\widehat{\boldsymbol{W}}_p)]}{IFFT[FFT(\boldsymbol{O}_p)FFT(\boldsymbol{O}_p)]}\right) \quad (29)$$

where $\boldsymbol{O}_p$ is the zero-padding of $\boldsymbol{O}$ and $\boldsymbol{O}$ is a matrix of ones having the same size as $\widehat{\boldsymbol{W}}$. Fig. 10 shows the symmetrical peaks of the original $\boldsymbol{S}$ and uniform $\boldsymbol{S}'$. The peaks of $\boldsymbol{S}'$ have similar heights.

To separate symmetrical peaks from noise peaks and obtain a watermark unit map $\boldsymbol{M}$, an adaptive threshold is applied as follows:

$$\boldsymbol{M} = \begin{cases} 1, & if \quad \boldsymbol{S}' > \boldsymbol{\mu}_{\boldsymbol{S}'} + \beta\boldsymbol{\sigma}^2_{\boldsymbol{S}'} \\ 0, & otherwise \end{cases} \quad (30)$$

where $\boldsymbol{\mu}_{\boldsymbol{S}'}$ and $\boldsymbol{\sigma}^2_{\boldsymbol{S}'}$ denote the local average and the standard deviation of $\boldsymbol{S}'$, respectively. β is an empirical coefficient that is set from 3.0 to 4.3. $\boldsymbol{M}$ is a binary matrix in which element 1's represent the possible corner points of watermark units, and we can use it to resist various geometric distortions. As shown in Fig. 11, $\boldsymbol{M}$ clearly shows the position of the watermark units under geometric distortions.



(a) Watermarked image rotated by 15°, translated by (-16,-16), and cropped to 75%. (b) $\boldsymbol{M}$ corresponding to (a).

(c) Watermarked image attacked by strength 1.0 RBA. (d) $\boldsymbol{M}$ corresponding to (c).

Fig. 11. Watermarked images under geometric distortions and their corresponding watermark units map $\boldsymbol{M}$. The embedded symmetrical watermark has 16×16 watermark units.

C. Watermark State Determining

Before the despread spectrum and decoding the masked watermark unit $\boldsymbol{w}_m$, the state of $\boldsymbol{w}_m$ should be determined. Flipping offers watermark symmetry but introduces additional

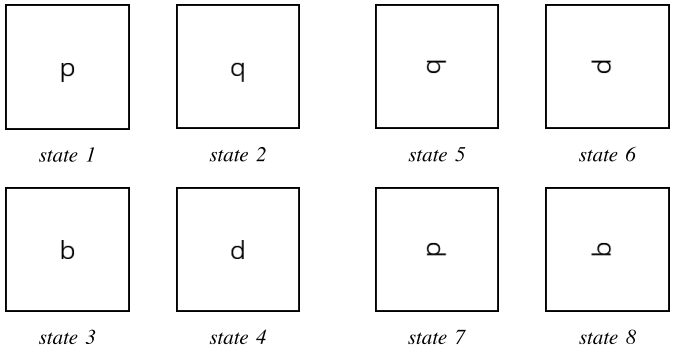


Fig. 12. Four original states and four extra states generated by rotation.

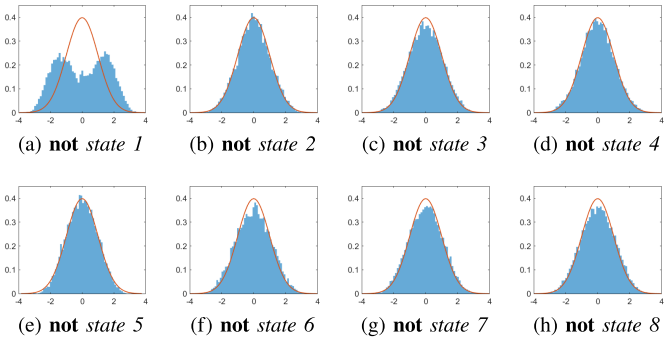


Fig. 13. Different distributions of y with different null hypotheses of $\boldsymbol{w}_s$ when $\boldsymbol{w}_s$ is *state 1*. The red line follows $N(0, 1)$.

states. As Fig. 12 shows, the original watermark unit has four states. If rotation is considered, the states increase to eight. Rather than introducing a reference watermark in the watermark unit, we propose a state determining method by combining the mask matrix $\boldsymbol{K}$ and the statistical characteristics of $\boldsymbol{w}_m$.

The state of $\boldsymbol{w}_m$ without any flipping and rotation is denoted as *state 1*, and the rest are denoted as *state 2* to *8*. First, according to the watermark unit map $\boldsymbol{M}$, we select a watermark unit and restore its geometric distortions to obtain $\boldsymbol{w}_s$, whose state is unknown. Before demasking $\boldsymbol{w}_s$ with $\boldsymbol{K}$, $\boldsymbol{w}_s$ should be flipped and rotated from the current state to *state 1*. A null hypothesis is set as follows:

$$H_0\text{: } \boldsymbol{w}_s \text{ is } \textbf{not } \textit{state 1}.$$

Contrary to H_0, $\boldsymbol{w}_s$ is demasked with $\boldsymbol{K}$ directly to obtain $\boldsymbol{w}_r$. According to H_0, $\boldsymbol{w}_s$ is incorrectly demasked, so $\boldsymbol{w}_r$ is almost a random matrix and has no property of spread-spectrum matrix $\boldsymbol{R}$. We divide $\boldsymbol{w}_r$ into nonoverlapping small blocks that have the same size as $\boldsymbol{R}$ and use $\boldsymbol{w}_r^{i,j}$ to represent the small block i-th in row and j-th in column. Then, $\boldsymbol{w}_r^{i,j}$ is normalized to the mean value of 0 and the variance of 1. It should be noted that $\boldsymbol{R}$ is also zero-mean and one-variance. Matrix $\boldsymbol{x}$ is defined by:

$$x(p,q) = w_r^{i,j}(p,q)R(p,q) \tag{31}$$

where p and q are integers ranging from 1 to L_R. According to H_0, $w_r^{i,j}(p,q)$ is independent of $R(p,q)$. Thus, the expectation and the variance in $x(p,q)$ are:

$$\begin{aligned}\mu_x &= E\{x(p,q)\}\\ &= E\{w_r^{i,j}(p,q)\}E\{R(p,q)\}\\ &= 0 \end{aligned}\tag{32}$$

$$\begin{aligned}\sigma_x^2 &= D\{x(p,q)\}\\ &= E\{(w_r^{i,j}(p,q)R(p,q))^2\} - E\{w_r^{i,j}(p,q)R(p,q)\}^2\\ &= D\{w_r^{i,j}(p,q)\}D\{R(p,q)\} - E\{w_r^{i,j}(p,q)\}^2 E\{R(p,q)\}^2\\ &= 1. \end{aligned}\tag{33}$$

Consider the following equation:

$$\begin{aligned} y &= \frac{1}{\sqrt{L_R^2\sigma_x}}\sum_p^{L_R}\sum_q^{L_R}(x(p,q)-\mu_x)\\ &= \frac{1}{L_R}\sum_p^{L_R}\sum_q^{L_R}x(p,q). \end{aligned}\tag{34}$$

If the size of $\boldsymbol{R}$ is sufficiently large, y will be a random variable following a standard normal distribution $N(0, 1)$ by exploiting the central limit theorem. One selected watermark unit $\boldsymbol{w}_s$ can offer a few samples, and we can roughly evaluate whether y follows $N(0, 1)$ according to these samples.

Note that H_0 is actually an hypothesis for all blocks' states in addition to the state of $\boldsymbol{w}_s$ because the watermark unit states are related. For example, according to the *flipping rule* mentioned in Section III-B, a watermark unit $\boldsymbol{w}$ on the right side of $\boldsymbol{w}_s$ is generated by flipping $\boldsymbol{w}_s$ horizontally. Therefore, if $\boldsymbol{w}_s$ is *state 1*, the $\boldsymbol{w}$ on the right side of $\boldsymbol{w}_s$ will be *state 2*. If $\boldsymbol{w}_s$ is not *state 1*, the $\boldsymbol{w}$ on the right side of $\boldsymbol{w}_s$ will not be *state 2*. Similarly, combining H_0 and the flipping rule, it is easy to conclude an equivalent null hypothesis H_0':

$$\begin{gathered} H_0'\text{: } \boldsymbol{w}_s \text{ is } \textbf{not } \textit{state 1};\\ \boldsymbol{w} \text{ on the right side of } \boldsymbol{w}_s \text{ is } \textbf{not } \textit{state 2};\\ \boldsymbol{w} \text{ on the upper side of } \boldsymbol{w}_s \text{ is } \textbf{not } \textit{state 3};\\ \ldots. \end{gathered}$$

H_0' makes hypotheses for all watermark units. For every block, we can obtain a set of samples of y. Therefore, we have enough samples of y from these blocks and can evaluate the distribution of y accurately. Fig. 13(a) shows the distribution of y with H_0' when the selected $\boldsymbol{w}_s$ is *state 1*. It is obvious that y deviates on a large scale from $N(0, 1)$, so H_0' is rejected. H_0 is also rejected since H_0 and H_0' are equivalent. Finally, we can conclude that $\boldsymbol{w}_s$ is *state 1*.

As Fig. 13 shows, if we make a different null hypothesis, y will follow $N(0, 1)$. In practice, we use Kullback-Leibler divergence (KLD) to measure the distance between y and $N(0, 1)$. For eight different null hypotheses, the distance between y and $N(0, 1)$ is calculated. Then, the hypothesis corresponding to the farthest distance is rejected, and the state of $\boldsymbol{w}_s$ is determined.

Fig. 14. *Top row:* Host images; *Bottom row:* Watermarked images generated by the proposed scheme.

D. Watermark Decoding

Now, we obtain the states of all watermark units and restore all available watermark units to *state 1*. All available blocks are accumulated to obtain $\widehat{\boldsymbol{w}}$. Similarly, $\widehat{\boldsymbol{w}}$ is divided into nonoverlapping small blocks that have the same size as the spread-spectrum matrix $\boldsymbol{R}$. These small blocks are denoted as $\widehat{\boldsymbol{w}}^{i,j}$. To despread $\widehat{\boldsymbol{w}}$, the correlation value $\rho_{i,j}$ between $\widehat{\boldsymbol{w}}^{i,j}$ and $\boldsymbol{R}$ is obtained as:

$$\rho_{i,j} = \sum_{p}^{L_R} \sum_{q}^{L_R} \widehat{w}^{i,j}(p,q) R(p,q) \tag{35}$$

Then, the message bit is determined by:

$$\widehat{m}_{i,j} = \begin{cases} 0, & \rho_{i,\ j} < 0 \\ 1, & \rho_{i,\ j} \geq 0 \end{cases} \tag{36}$$

where $\widehat{m}_{i,j}$ is the extracted message bit. The obtained message matrix $\widehat{\boldsymbol{m}}$ is now reshaped decoded according to its error correction code (ECC) to recover the embedded message $\boldsymbol{m}'$.

V. Experimental Results

In this section, the proposed watermarking scheme based on symmetry is compared with some state-of-art watermarking schemes. The experimental parameters, setup of the comparative experiment and test database are described in Section V-A. In Section V-B, V-C, and V-D, the robustness in different aspects is compared between the proposed scheme and other watermarking schemes.

A. Implementation Details

In our scheme, the size of the spread-spectrum matrix $\boldsymbol{R}$ is an important parameter. Obviously, a larger $\boldsymbol{R}$ makes the watermark more robust to common image processing operations. However, a smaller $\boldsymbol{R}$ means more watermark units within the same size area; that is, there are more symmetrical peaks to better approximate the geometric distortions. In our experiment, we choose $L_R = 4$, so the watermark is robust to various geometric distortions, even RBA. For all watermarking schemes in the comparative experiments, their message length L_m is set as 64 bits. To evaluate the performance between the proposed scheme and the state-of-art schemes fairly, the message is embedded directly without ECC encoded, and the bit error quantity (BEQ) is recorded.

TABLE I

Average PSNR of Watermarked Images Generated by Different Schemes

Scheme	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
PSNR(dB)	40.74	40.61	41.03	41.35

As mentioned in Section I, only watermarking schemes belonging to the last two categories have the possibility of being robust to local geometric distortions. Voloshynovskiy *et al.* [37] and Tian *et al.* [32] belong to these two categories and are resilient to local geometric distortions. In the remaining categories, the robustness of Kang *et al.* [4] to global geometric distortions and common image processing operations is one of the best. We compare the proposed scheme with these three schemes. To illustrate the experimental results concisely, in the following part, we use V_ACF [37], T_LDFT [32], and K_ULPM [4] to represent the three watermarking schemes above.

The test host images include eight colorful images selected from the USC-SIPI image database [40] and one hundred gray images randomly selected from the BOSSBase database [41]. Fig. 14 shows part of the host images and their corresponding watermarked images generated by the proposed scheme. The average peak signal-to-noise ratio (PSNR) of 108 watermarked images is 41.35 dB. The robustness of the proposed scheme to distortions is evaluated by the average bit error quantity (BEQ) of watermarked images under the corresponding distortions. For other watermarking schemes in the comparative experiment, the same experimental setup and process are applied. As shown in Table I, the average PSNR values of watermarked images generated by four different watermarking schemes are set to the same level of 41 ± 0.4 dB.

Most distortions in our experiment are realized by Stirmark 4.0 Benchmark [1], [3].

B. Robustness to Common Image Processing Operations

In this subsection, we compare the capability of V_ACF [37], T_LDFT [32], K_ULPM [4], and the proposed scheme to recover the hidden message under common image processing operations, including JPEG compression, Gaussian noise, and average filtering. The experimental results are shown as follows.

1) Robustness to JPEG Compression: The watermarked images are compressed with JPEG compression with a quality

TABLE II
AVERAGE BEQ OF WATERMARKED IMAGES UNDER JPEG COMPRESSION

Quality Factor	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
90	0	6.111	0.037	**0**
80	0	16.352	0.343	**0**
70	0	20.740	0.482	**0**
60	0.046	22.815	0.630	**0**
50	0.213	24.102	1.278	**0.157**
40	0.657	26.963	1.954	**0.481**
30	**0.843**	29.926	1.676	1.741
20	4.704	32.176	**2.389**	4.657
15	7.037	31.278	**4.102**	6.046

TABLE III
AVERAGE BEQ OF WATERMARKED IMAGES UNDER GAUSS NOISE

Variance	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
0.0001	0	2.824	1.380	**0**
0.001	0.185	27.491	1.435	**0.056**
0.01	0.796	31.759	2.380	**0.370**

TABLE IV
AVERAGE BEQ OF WATERMARKED IMAGES WITH AVERAGE FILTERING

Filter Size	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
3×3	0.083	21.648	1.704	**0.028**
5×5	14.750	28.278	**2.676**	13.657
7×7	31.167	31.120	**8.500**	27.954

factor (QF) from 15 to 90. The average BEQ is listed in Table II. The proposed scheme performs well under JPEG compression with all quality factors. Specifically, the performance of the proposed scheme is similar to K_ULPM [4]'s and better than the rest of the schemes'.

2) Robustness to Gauss Noise: The watermarked images are corrupted by Gaussian noise with a variance from 0.0001 to 0.01. As Table III shows, the 64-bit message can be recovered with few errors by the proposed watermarking scheme. V_ACF [37] also performs well. The proposed watermarking scheme and V_ACF are both spatial watermarks and have a similar framework. Their low BEQ is the result of accumulation operation, which is a common spatial watermark enhancement method. The impact of Gaussian noise can be considerably decreased by accumulating repeated watermark units.

3) Robustness to Average Filtering: Average filters of different sizes are applied to the watermarked images, and the average BEQ is recorded in Table IV. The proposed watermarking scheme can recover the message without error under 3×3 average filtering. When the filter size increases, the performance of the proposed scheme is worse than that of K_ULPM [4] but better than that of the other two schemes.

C. Robustness to Geometric Distortions

The robustness of watermarking schemes to geometric distortions, including global and local geometric distortions, is investigated in this subsection. In addition to rotation, scaling, translation, and cropping, affine transformation, line removal, and aspect ratio change are also set as global geometric distortions in comparative experiments. Additionally, RBA proposed in Stirmark is set as local geometric distortion in the comparative experiment. There is no specific experiment to translation because translation is a basic geometric distortion and usually exists along with other geometric distortions, such as rotation and cropping.

TABLE V
AVERAGE BEQ OF ROTATED WATERMARKED IMAGES

Rotate Angle	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
0.25°	3.343	5.148	3.204	**0.093**
0.5°	2.585	5.620	2.259	**0.019**
1°	1.222	4.481	2.352	**0.046**
5°	0.750	8.593	4.833	**0.046**
30°	0.259	7.556	5.926	**0.139**
45°	0.509	12.759	8.287	**0.157**
90°	0.167	28.574	24.389	**0.065**

TABLE VI
AVERAGE BEQ OF SCALED WATERMARKED IMAGES

Scaling Factor	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
0.5	0.083	24.537	11.787	**0**
0.75	0	11.574	4.935	**0**
0.9	0	6.473	3.139	**0**
1.1	0	8.019	3.056	**0**
1.5	0	20.343	6.315	**0**
2	0	26.167	18.796	**0**

1) Robustness to Rotation: The watermarked images are rotated by some specific angles, including some small, precise angles, to test the robustness of watermarking schemes to rotation. The experimental results are listed in Table V. The proposed scheme almost recovers the message with very few errors from all of the rotation angles. The remaining schemes have a higher average BEQ, especially under some specific rotation angles.

2) Robustness to Scaling: The width and height of watermarked images are scaled proportionally with the scaling factor from 0.5 to 2.0. As shown in Table VI, under all scaling factors, the average BEQ of the proposed scheme is 0. Our scheme and V_ACF [37] have similar performance and are much better than the rest.

3) Robustness to Cropping: The watermarked images are cropped by the ratio from 1% to 75%. Both width and height are cropped. For example, if a watermarked image of size 512×512 is cropped 75%, only the central region of size 128×128 is reserved whose area is only 6.25% of the original image. The experimental results are listed in Table VII. The proposed watermarking scheme can recover the message almost without error when the cropping ratio is from 1% to 50%. When the cropping ratio is 75%, the average BEQ of the proposed scheme is only 5.537, much less than the rest of the schemes.

TABLE VII
AVERAGE BEQ OF CROPPED WATERMARKED IMAGES

Cropping Ratio	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
1%	0.093	0.231	3.917	**0**
5%	0	1.157	4.833	**0**
10%	0.269	1.083	5.546	**0**
25%	0.657	3.000	8.111	**0**
50%	0.694	5.444	17.963	**0**
75%	9.824	12.778	28.009	**5.537**

TABLE VIII
AVERAGE BEQ OF AFFINE TRANSFORMED WATERMARKED IMAGES

Affine Transform Matrix a	b	c	d	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
1	0	0.01	1	0	1.139	5.556	**0**
1	0	0.05	1	0	3.870	28.148	**0**
1	0.01	0	1	0	2.583	6.861	**0**
1	0.05	0	1	0.046	2.269	30.917	**0**
1	0.01	0.01	1	0.185	5.806	11.435	**0**
1	0.05	0.05	1	0.407	8.269	30.870	**0**

TABLE IX
AVERAGE BEQ OF DISPROPORTIONATELY SCALED WATERMARKED IMAGES

Scaling Factors	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
0.9×1.1	0	9.630	32.407	**0**
1.5×0.8	0	21.352	31.083	**0**
0.7×1.8	0	25.519	32.271	**0**

4) Robustness to Affine Transformation: The watermarked images are transformed with different affine transformation matrices as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{37}$$

By adjusting a, b, c, and d, X-shearing, Y-shearing, and XY-shearing of different strengths is applied to watermarked images. According to the experimental results in Table VIII, under all the tested affine transformations, the BEQ of the message recovered by the proposed scheme is 0. Additionally, under a specific affine transformation, the BEQ of V_ACF [37] is 0.407, the BEQ of T_LDFT [32] is 8.269, and the BEQ of K_ULPM [4] is up to 30.870.

5) Robustness to Aspect Ratio Change: The width and height of watermarked images are scaled with different scaling factors. In Table IX, the height of the watermarked images is scaled with the first scaling factor, and the width is scaled with the second scaling factor. As Table IX shows, the proposed watermarking scheme can recover the embedded message without error with all aspect ratio changes. V_ACF [37] performs as well as the proposed scheme. The average BEQ of T_LDFT [32] is over 9, and the average BEQ of K_ULPM [4] is over 31.

6) Robustness to Lines Removal: To investigate the robustness to line removal, 1%, 5% and 10% lines in the row and column of watermarked images are removed. These removed lines are distributed at regular intervals. As Table X shows, under all tested lines removed, the proposed scheme has a BEQ of 0, which is much lower than the BEQ of the rest of the schemes.

TABLE X
AVERAGE BEQ OF WATERMARKED IMAGES UNDER LINES REMOVAL

Lines Removal Ratio	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
1%	0	0.426	5.907	**0**
5%	0.056	1.241	4.954	**0**
10%	0.083	4.185	6.500	**0**

TABLE XI
AVERAGE BEQ OF WATERMARKED IMAGES UNDER RBA

Strength of RBA	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
0.1	0	5.113	1.370	**0**
0.2	0	5.391	2.243	**0**
0.3	0.074	5.406	6.133	**0**
0.4	0.065	5.957	9.943	**0.056**
0.5	0.248	5.576	12.061	**0.124**
0.6	2.764	5.254	13.235	**0.278**
0.7	6.191	5.822	15.017	**0.472**
0.8	5.687	5.843	17.170	**0.659**
0.9	6.185	5.791	20.044	**0.674**
1.0	9.369	5.828	22.650	**0.602**

7) Robustness to Random Bending Attack: The RBA of the Stirmark is one of the most typical local geometric distortions. Considering the randomness of RBA, to a specific attack strength, each watermarked image is attacked 5 times to obtain the average performance. Thus, each average BEQ in Table XI is calculated from the messages recovered from 108×5 watermarked images. The experimental results with varying strengths are shown in Table XI, and the average BEQ of the proposed watermarking scheme is no more than 0.7. The proposed watermarking scheme has excellent performance under RBA, even better than the performance of T_LDFT [32], which is based on local geometrically invariant features.

In addition, the proposed watermarking scheme has a better performance compared with V_ACF [37], especially under the RBA with large strength. As mentioned in Section. II-B, the symmetrical peaks in the proposed scheme are resilient to perspective transformation, and perspective transformation can better approximate local geometric distortions. The parameters of the perspective transformations can be obtained directly from the watermark unit map $\boldsymbol{M}$, as shown in Fig. 11(d).

D. Robustness to Combined Attacks

In this subsection, we select representative single attacks, that is, JPEG compression with QF 70, Gauss noise with variance 0.001, 3×3 average filtering, rotating 30°, scaling with factor 0.75, cropping 25%, and RBA with strength 0.3. One combined attack is generated by combining two of these single attacks. The experimental results are listed in Table XII, where the attack parameters are omitted. The average BEQ of the proposed scheme is less than 1, which is much lower than that of the other schemes. This is not a surprising result because the proposed scheme, unlike others, has better performance under all these single attacks.

TABLE XII
AVERAGE BEQ OF WATERMARKED IMAGES UNDER COMBINED ATTACKS

Combined Attack Types	V_ACF [37]	T_LDFT [32]	K_ULPM [4]	Proposed
Average + Gauss	0.111	29.519	1.824	**0.083**
JPEG + Average	0.083	26.130	1.639	**0.021**
JPEG + Gauss	0	28.287	2.102	**0**
Cropping + Rotation	1.120	5.556	11.963	**0.157**
Cropping + Scaling	0.713	22.167	9.972	**0**
Scaling + Rotation	0.611	18.972	7.963	**0.222**
Cropping + Average	1.157	25.806	8.491	**0.044**
Cropping + Gauss	0.935	24.426	8.148	**0**
JPEG + Cropping	1.111	23.917	8.796	**0**
JPEG + Rotation	0.731	24.657	5.963	**0.231**
JPEG + Scaling	0	26.037	5.852	**0**
Rotation + Average	0.451	24.935	6.685	**0.204**
Rotation + Gauss	0.583	30.759	5.750	**0.185**
Scaling + Average	0.241	28.398	5.657	**0.093**
Scaling + Gauss	0	32.176	5.713	**0**
RBA + Average	0.102	25.037	10.278	**0.102**
RBA + Cropping	0.685	8.046	28.102	**0.074**
RBA + Gauss	0.111	31.704	10.324	**0.009**
RBA + JPEG	0.185	24.954	10.037	**0.102**
RBA + Rotation	0.343	16.444	3.389	**0.157**
RBA + Scaling	0.194	22.796	6.769	**0.019**

TABLE XIII
THE COMPUTATIONAL COST OF CALCULATING SYMMETRY BY EQ. (6) AND EQ. (27)

Scaling Factor	0.5	0.75	1	1.5	2
Time cost(s) of Eq. (6)	0.42	2.10	6.18	27.70	102.37
Time cost(s) of Eq. (27)	**0.02**	**0.05**	**0.07**	**0.12**	**0.20**

TABLE XIV
THE COMPUTATIONAL COST OF THE SYNCHRONIZATION PROCESS IN THE PROPOSED SCHEME AND V_ACF

Scaling Factor	0.5	0.75	1	1.5	2
Time cost(s) of V_ACF	0.142	0.205	0.287	0.638	1.205
Time cost(s) of proposed	**0.135**	**0.183**	**0.266**	**0.588**	**1.101**

E. Computational Cost

As mentioned in Sections I and IV-B, we reduced the computational cost of the proposed synchronization process from two aspects:

1) We calculate the symmetry by the frequency form of the autoconvolution function (Eq. (27)), which is faster than calculating symmetry pixel by pixel in the space domain (Eq. (6)).
2) Compared to V_ACF [37], the proposed synchronization process does not require translation estimation, which saves computational cost.

We test the running time with an unoptimized MATLAB implementation on a PC with AMD Ryzen5 1600X CPU and 16 GB ram. The scaled watermarking images are used as input. The original image size is 512×512, and the scaling factor varies from 0.5 to 2. The running time of calculating symmetry by Eq. (6) and Eq. (27) is shown in Table XIII. Compared to calculating symmetry pixel by pixel, calculating symmetry by an autoconvolution function greatly reduces the computational cost, especially when the input image size is large.

We also compare the synchronization time of the proposed watermarking scheme and V_ACF. As shown in Table XIV, compared to V_ACF, the proposed scheme has a lower time cost, where the difference is close to the computational cost of the translation estimation process.

VI. CONCLUSION

This paper presents a blind and robust watermarking scheme, including a novel watermark synchronization process. The proposed synchronization process is based on the symmetry of the symmetrical watermark, which has several advantages compared to prior similar schemes. In addition, we propose minimizing the mean-square error to obtain the watermark estimation using the autoconvolution function to quickly calculate the symmetry, and apply hypothesis testing to determine the watermark state. We believe that the proposed watermark synchronization process based on symmetry has the potential to become an improved version of the process based on periodicity and help similar schemes improve their performance. According to the experimental results, the proposed watermarking scheme has excellent performance under various distortions, including RBA, global geometric distortions, common image processing operations, and combined attacks. In future work, we will focus on the application of the proposed watermarking scheme on the print-camera process.

REFERENCES

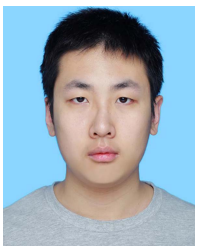
[1] F. A. P. Petitcolas, "Watermarking schemes evaluation," *IEEE Signal Process. Mag.*, vol. 17, no. 5, pp. 58–64, Sep. 2000.
[2] C. I. Podilchuk and E. J. Delp, "Digital watermarking: Algorithms and applications," *IEEE Signal Process. Mag.*, vol. 18, no. 4, pp. 33–46, Jul. 2001.
[3] F. A. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Attacks on copyright marking systems," in *Proc. Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, 1998, pp. 218–238.
[4] X. Kang, J. Huang, and W. Zeng, "Efficient general print-scanning resilient data hiding based on uniform log-polar mapping," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 1, pp. 1–12, Mar. 2010.
[5] J.-M. Guo, S.-C. Pei, and H. Lee, "Paired subimage matching watermarking method on ordered dither images and its high-quality progressive coding," *IEEE Trans. Multimedia*, vol. 10, no. 1, pp. 16–30, Jan. 2008.
[6] H. Fang *et al.*, "A camera shooting resilient watermarking scheme for underpainting documents," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 4075–4089, Nov. 2020.
[7] C. Chen, W. Huang, B. Zhou, C. Liu, and W. H. Mow, "PiCode: A new picture-embedding 2D barcode," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3444–3458, Aug. 2016.
[8] C. Chen, B. Zhou, and W. H. Mow, "RA code: A robust and aesthetic code for resolution-constrained applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 11, pp. 3300–3312, Nov. 2018.
[9] C. Chen, W. Huang, L. Zhang, and W. H. Mow, "Robust and unobtrusive display-to-camera communications via blue channel embedding," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 156–169, Jan. 2019.
[10] H. Fang, W. Zhang, H. Zhou, H. Cui, and N. Yu, "Screen-shooting resilient watermarking," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 6, pp. 1403–1418, Jun. 2019.
[11] M.-J. Lee, K.-S. Kim, and H.-K. Lee, "Digital cinema watermarking for estimating the position of the pirate," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 605–621, Nov. 2010.
[12] H. Cui, H. Bian, W. Zhang, and N. Yu, "Unseencode: Invisible on-screen barcode with image-based extraction," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 1315–1323.
[13] C.-W. Tang and H.-M. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 950–959, Apr. 2003.
[14] D. Zheng, S. Wang, and J. Zhao, "RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes," *IEEE Trans. Image Process.*, vol. 18, no. 5, pp. 1055–1068, May 2009.

[15] J. S. Seo and C. D. Yoo, "Image watermarking based on invariant regions of scale-space representation," *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1537–1549, Apr. 2006.
[16] P. Dong, J. G. Brankov, N. P. Galatsanos, Y. Yang, and F. Davoine, "Digital watermarking robust to geometric distortions," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2140–2150, Dec. 2005.
[17] H. Shin Kim and H.-K. Lee, "Invariant image watermark using Zernike moments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 766–775, Aug. 2003.
[18] A. Herrigel, J. Ó. Ruanaidh, H. Petersen, S. Pereira, and T. Pun, "Secure copyright protection techniques for digital images," in *Proc. Int. Workshop Inf. Hiding*. Berlin, Germany: Springer, 1998, pp. 169–190.
[19] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 1123–1129, Jun. 2000.
[20] X. Kang, J. Huang, Y. Q. Shi, and Y. Lin, "A DWT-DFT composite watermarking scheme robust to both affine transform and JPEG compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 776–786, Aug. 2003.
[21] A. Herrigel, S. V. Voloshynovskiy, and Y. B. Rytsar, "Watermark template attack," in *Proc. 2nd Secur. Watermarking Multimedia Contents*, vol. 4314, Aug. 2001, pp. 394–405.
[22] C.-Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation, scale, and translation resilient watermarking for images," *IEEE Trans. Image Process.*, vol. 10, no. 5, pp. 767–782, May 2001.
[23] D. Zheng, J. Zhao, and A. El Saddik, "RST-invariant digital image watermarking based on log-polar mapping and phase correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 753–765, Aug. 2003.
[24] J. J. K. O'Ruanaidh and T. Pun, "Rotation, scale and translation invariant digital image watermarking," in *Proc. Int. Conf. Image Process.*, vol. 1, 1997, pp. 536–539.
[25] D. He and Q. Sun, "A RST resilient object-based video watermarking scheme," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 2, Oct. 2004, pp. 737–740.
[26] M. Urvoy, D. Goudia, and F. Autrusseau, "Perceptual DFT watermarking with improved detection and robustness to geometrical distortions," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 7, pp. 1108–1119, Jul. 2014.
[27] H. Kim, "Robust image watermarking using local invariant features," *Opt. Eng.*, vol. 45, no. 3, Mar. 2006, Art. no. 037002.
[28] P. Bas, J.-M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 1014–1028, Sep. 2002.
[29] X. Gao, C. Deng, X. Li, and D. Tao, "Geometric distortion insensitive image watermarking in affine covariant regions," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 3, pp. 278–286, May 2010.
[30] S. Xiang, H. J. Kim, and J. Huang, "Invariant image watermarking based on statistical features in the low-frequency domain," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 6, pp. 777–790, Jun. 2008.
[31] J.-L. Dugelay, S. Roche, C. Rey, and G. Doërr, "Still-image watermarking robust to local geometric distortions," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2831–2842, Sep. 2006.
[32] H. Tian, Y. Zhao, R. Ni, L. Qin, and X. Li, "LDFT-based watermarking resilient to local desynchronization attacks," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2190–2201, Dec. 2013.
[33] M. Kutter, "Watermarking resistance to translation, rotation, and scaling," in *Proc. Int. Soc. Opt. Photonics, Multimedia Syst. Appl.*, vol. 3528, 1999, pp. 423–431.
[34] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Content adaptive watermarking based on a stochastic multiresolution image modeling," in *Proc. 10th Eur. Signal Process. Conf.*, 2000, pp. 1–4.
[35] F. Deguillaume, S. V. Voloshynovskiy, and T. Pun, "Method for the estimation and recovering from general affine transforms in digital watermarking applications," in *Proc. 4th Secur. Watermarking Multimedia Contents*, vol. 4675, Apr. 2002, pp. 313–322.
[36] T. Kalker, G. Depovere, J. Haitsma, and M. J. Maes, "Video watermarking system for broadcast monitoring," in *Proc. Secur. Watermarking Multimedia Contents, Int. Soc. Opt. Photon.*, vol. 3657, Apr. 1999, pp. 103–112.
[37] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local nonlinear geometrical distortions," in *Proc. Int. Conf. Image Process.*, vol. 3, 2001, pp. 999–1002.
[38] Y. Huang, B. Niu, H. Guan, and S. Zhang, "Enhancing image watermarking with adaptive embedding parameter and PSNR guarantee," *IEEE Trans. Multimedia*, vol. 21, no. 10, pp. 2447–2460, Oct. 2019.
[39] J. S. Lim, *Two-Dimensional Signal and Image Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1990, p. 548.
[40] *The USC-SIPI Image Database*. Accessed: Jun. 13, 2019. [Online]. Available: http://sipi.usc.edu/database/
[41] *The BOSSBase Database*. Accessed: Mar. 8, 2020. [Online]. Available: http://agents.fel.cvut.cz/boss/

Zehua Ma received the B.S. degree in information security from the University of Science and Technology of China (USTC) in 2018, where he is currently pursuing the M.S. degree in information security. His research interests include image watermarking, information hiding, and image processing.

Weiming Zhang received the M.S. and Ph.D. degrees from the Zhengzhou Information Science and Technology Institute, China, in 2002 and 2005 respectively. He is currently a Professor with the School of Information Science and Technology, University of Science and Technology of China. His research interests include information hiding and multimedia security.

Han Fang received the B.S. degree from the Nanjing University of Aeronautics and Astronautics (NUAA) in 2016. He is currently pursuing the Ph.D. degree in information security with the University of Science and Technology of China (USTC). His research interests include image watermarking, information hiding, and image processing.



Xiaoyi Dong received the B.S. degree in information security from the University of Science and Technology of China (USTC) in 2018, where he is currently pursuing the Ph.D. degree in information security. His research interests include adversarial sample, 3D point cloud recognition, and DeepFake detection.

Linfeng Geng received the B.S. and M.S. degrees from the University of Science and Technology of China (USTC) in 2017 and 2020, respectively. His research interests include information security, image watermarking, and image processing.

Nenghai Yu (Member, IEEE) received the B.S. degree from the Nanjing University of Posts and Telecommunications in 1987, the M.E. degree from Tsinghua University in 1992, and the Ph.D. degree from the University of Science and Technology of China in 2004. He is currently a Professor with the University of Science and Technology of China. His research interests include multimedia security, multimedia information retrieval, video processing, and information hiding.