

RESEARCH

Open Access

Counteracting geometrical attacks on robust image watermarking by constructing a deformable pyramid transform

Chuntao Wang^{1,2*}, Jiangqun Ni² and Dong Zhang²

Abstract

Counteracting geometrical attacks remains one of the most challenging problems in robust watermarking. In this paper, we resist rotation, scaling, and translation (RST) by constructing a kind of deformable pyramid transform (DPT) that is shift-invariant, steerable, and scalable. The DPT is extended from a closed-form polar-separable steerable pyramid transform (SPT). The radial component of the SPT's basis filters is taken as the kernel of the scalable basis filters, and the angular component is used for the steerable basis filters. The shift-invariance is inherited from the SPT by retaining undecimated high-pass and band-pass subbands. Based on the designed DPT, we theoretically derive interpolation functions for steerability and scalability and synchronization mechanisms for translation, rotation, and scaling. By exploiting the preferable characteristics of DPT, we develop a new template-based robust image watermarking scheme that is resilient to RST. Translation invariance is achieved by taking the Fourier magnitude of the cover image as the DPT's input. The resilience to rotation and scaling is obtained using the synchronization mechanisms for rotation and scaling, for which an efficient template-matching algorithm has been devised. Extensive simulations show that the proposed scheme is highly robust to geometrical attacks, such as RST, cropping, and row/column line removal, as well as common signal processing attacks such as JPEG compression, additive white Gaussian noise, and median filtering.

Keywords: Robust watermarking; RST invariant; Deformable pyramid transform; Template; Robustness

1. Introduction

Counteracting geometrical attacks such as rotation, scaling, and translation (RST) remains one of the most challenging problems for robust watermarking. This is because geometrical attacks easily desynchronize the watermark, degrading its robustness dramatically. To address such problems, a number of RST-invariant blind watermarking schemes have been developed over the past two decades. These schemes can be roughly categorized into five paradigms, namely exhaustive search, invariant domain, auto-correlation, feature-based implicit synchronization, and geometrical correction [1-4]. These are briefly described below.

The exhaustive search method [5-7] iteratively corrects each geometrical distortion in the search space and then evaluates the watermark extracted from the geometrically

corrected carrier accordingly. This method generally leads to high computational complexity and a large probability of false positives. The invariant domain approach [8-14] eliminates the need to identify geometrical distortions by embedding the watermark in a domain that is invariant to such distortions. However, this method may encounter the issue of interpolation approximation during the geometrically invariant transform. Auto-correlation-based techniques periodically insert the watermark in the cover and use a cross-correlation function to locate the periodic autocorrelation peaks, which indicate the geometrical transform that has been performed. The schemes discussed in [15-17] are typical examples of this category. The fourth category exploits salient features to achieve geometrical synchronization, as presented in the schemes of [3,10], and [18-20]. Under this approach, the embedder binds the watermark with the geometrically invariant salient features. The watermark is recovered inversely by the receiver, who seeks the salient features that still exist, even after severe geometrical distortion. In general, this category is

* Correspondence: wct2006@gmail.com

¹College of Information, South China Agricultural University, Guangdong 510642, People's Republic of China

²School of Information Science and Technology, Sun Yat-sen University, Guangdong 510275, People's Republic of China

somewhat robust against geometrical distortions, but it may degrade greatly if the salient feature detection fails. The final category estimates the geometrical distortion parameters, thus permitting geometrical correction and watermark extraction. The template is generally constructed and embedded in the cover, and the geometrical parameters are sought via a particular technique. Several examples are shown in the schemes described in [21-24]. In addition to the template, support vector machines (SVMs) have also been incorporated to obtain geometrical parameters. For example, a number of recently developed schemes [25-28] generate patterns, such as the inserted template and Zernike moments, from geometrically attacked watermarked images. These patterns are then input to the SVM to train the classification model, and finally, the trained model is used to predict the geometrical parameters of the to-be-checked image.

In this paper, we develop a new geometrical correction-based robust image watermarking scheme by constructing a deformable pyramid transform (DPT) that is shift-invariant, steerable, and scalable. This is motivated by the scheme of [24], in which a steerable pyramid transform (SPT) with shift-invariance and steerability [29] is exploited to estimate, with an auxiliary inserted template, the rotation angle. This allows for rotation correction and watermark extraction. Although the scheme in [24] is highly robust against rotation, it cannot resist scaling attacks because of the lack of scalability in SPT. To counteract both the rotation and scaling, a kind of pyramid transform (PT) with shift-invariance, steerability, and scalability is needed. However, such a PT has not, to the best of our knowledge, been reported in the literature. Inspired by this situation, we design a shift-invariant, steerable, and scalable DPT. This is extended from SPT as follows.

We start by introducing the SPT. In essence, SPT is a variant of the wavelet transform (WT). As illustrated in [29], the conventional orthogonal or bi-orthogonal WT is sensitive to translation because of its critical sampling. That is, once the input signal has been translated slightly, its wavelet coefficients are not the translated versions of the original wavelet coefficients, and the information represented within a wavelet subband of the translated signal is not the same as that in the original wavelet subband. To address this issue, Freeman and Adelson [30] proposed a kind of steerable filter that can be used to synthesize any filter at an arbitrary orientation via a linear interpolation. This is termed *steerability*. Furthermore, Perona [31,32] developed scalable filters that can be used to interpolate any filter on a scale within a certain range, which is called *scalability*. These steerable and scalable filters were further integrated to give deformable filters.

In [29], Simoncelli et al. analyzed the translation invariance of WT and then generalized it to the concept of

shiftability. In brief, shiftability implies that any filter at an arbitrary position, orientation, and scale can be obtained through a linear interpolation of the designed shiftable, steerable, and scalable basis filters, respectively. The shiftability in orientation and scale is essentially equivalent to the steerability and scalability proposed in [30-32], respectively. In addition, Simoncelli et al. also proposed the concept of joint shiftability, which allows shiftability in the subset of position, orientation, and scale to be achieved simultaneously. As an illustration of these concepts, they also designed a kind of SPT that is shift-invariant and has shiftability in orientation (i.e., steerability).

In [33], Karasaridis and Simoncelli analyzed constraints for SPT and subsequently designed an SPT under these constraints via a numerical approach. Unfortunately, this SPT has non-perfect reconstruction. In contrast, Portilla et al. [34] developed an SPT with perfect reconstruction.

In summary, the filters developed in [30,33,34] are mainly steerable analysis filters or SPTs with steerability but without scalability. Although the filters designed in [30,31] have both steerability and scalability, they do not incorporate synthesis filters for reconstruction and thus cannot be considered as PTs. To the best of our knowledge, no PTs with shift-invariance, steerability, and scalability have been reported in the literature. In the interest of counteracting RST in robust image watermarking, we are motivated to extend the SPT with shift-invariance and steerability to include scalability. This is termed the DPT for convenience. To this end, we adopt the SPT with perfect reconstruction developed in [34]. The steerable filters of this SPT are represented in a polar-separable form, where the angular components are designed so as to achieve the steerability. This implies that scaling the steerable filters would be equivalent to dilating the radial component. Thus, according to the shiftability framework in [29], we can take the radial component of the SPT's steerable filters as the kernel for constructing the scalable filters. Furthermore, combining the scalable and steerable filters derived from the radial and angular components, respectively, gives rise to the scalability and steerability of the DPT. Its shift-invariance is inherited from the SPT by retaining undecimated high-pass and band-pass coefficients. In this way, we construct a DPT with shift-invariance, steerability, and scalability.

In an attempt to apply the DPT in robust watermarking to counteract RST, we first exploit the shift-invariance, steerability, and scalability of DPT to theoretically derive a mechanism for RST synchronization. As will be shown, the DPT coefficients of the translated signal are the translated versions of those of the original signal, which is the essence of shift-invariance. The relationship between the DPT coefficients of the original signal and those of the

rotated and scaled signal is characterized by a linear interpolation function parameterized using the rotation angle and scaling factor.

In this paper, based on the derived RST synchronization mechanism, we develop a new robust image watermarking that is resilient to RST. According to the aforementioned essence of DPT's shift-invariance, the translation of the input signal should affect the synchronization of rotation and scaling. To uncouple the translation from rotation and scaling, we take the Fourier magnitude of the cover image as the input to the DPT. This achieves true translation invariance. Rotation and scaling attacks are counteracted by first deploying the rotation and scaling synchronization mechanism to estimate their parameters. The estimated parameters are then used to correct the rotation and scaling that has been performed. In blind watermarking, the original signal cannot be used by the receiver, so we resort to the template to estimate the parameters of rotation and scaling attacks. Specifically, we insert the template and watermark at level 1 and levels 2 to 3, respectively, of the DPT pyramid in the embedding process. During the detection process, we exploit the rotation and scaling synchronization mechanism to identify the rotation angle and scaling factor, and further use these estimated parameters to correct the rotation and scaling distortions and recover the watermark from the geometrically corrected signal. Extensive experimental results demonstrate that the proposed algorithm is highly robust against geometrical attacks such as RST and exhibits favorable performance against common signal processing, including JPEG compression, median filtering, Gaussian noise, and low-pass filtering. In addition, we observe comparable or higher robustness with respect to other algorithms in the simulation.

The rest of this paper is organized as follows. Section 2 reviews the SPT with shift-invariance and steerability, and the construction of the DPT with shift-invariance, steerability, and scalability is detailed in Section 3. In Section 4, we theoretically derive the RST synchronization mechanism. We describe the proposed robust image watermarking scheme in Section 5, and present our experimental results in Section 6. Finally, the conclusions are discussed in Section 7.

2. Steerable pyramid transform with shift-invariance and steerability

In this section, we review the SPT with shift-invariance and steerability. We first describe the constraints for SPT given in [33] and then introduce one closed-form SPT presented in [34].

In [33], Karasaridis and Simoncelli evaluated the constraints for the recursive multi-scale SPT. Figure 1 illustrates a single-stage SPT, and the multi-stage version can be formed by recursively inserting the block

enclosed in the dashed box into the filled circle. To reach the perfect construction, the SPT should meet the following constraints [33]:

$$|H_0(\boldsymbol{\omega})|^2 + |L_0(\boldsymbol{\omega})|^2 \left(|L_1(\boldsymbol{\omega})|^2 + \sum_{k=0}^{K-1} |B_k(\boldsymbol{\omega})|^2 \right) = 1 \quad (1-1)$$

$$\left| L_1\left(\frac{\boldsymbol{\omega}}{2}\right) \right|^2 \cdot \left(|L_1(\boldsymbol{\omega})|^2 + \sum_{k=0}^{K-1} |B_k(\boldsymbol{\omega})|^2 \right) = \left| L_1\left(\frac{\boldsymbol{\omega}}{2}\right) \right|^2 \quad (1-2)$$

$$|L_1(\boldsymbol{\omega})| = 0, |\boldsymbol{\omega}| > \frac{\pi}{2}, \quad (1-3)$$

where $\boldsymbol{\omega} = (\omega_x, \omega_y)$ is the frequency vector in the Fourier domain, $H_0(\boldsymbol{\omega})$ and $L_0(\boldsymbol{\omega})$ denote the non-oriented high-pass and low-pass filters, respectively, and $L_1(\boldsymbol{\omega})$ and $B_k(\boldsymbol{\omega})$ ($k = 0, \dots, K-1$) represent the narrow-band low-pass filter and the oriented band-pass filter, respectively. Eqs. (1-1), (1-2), and (1-3) describe the unit system response amplitude, recursion relationship, and aliasing cancellation, respectively. Furthermore, the following constraint must hold to achieve steerability:

$$B_k(\boldsymbol{\omega}) = B(\boldsymbol{\omega})[-j\cos(\theta - \theta_k)]^{K-1}, \quad (2)$$

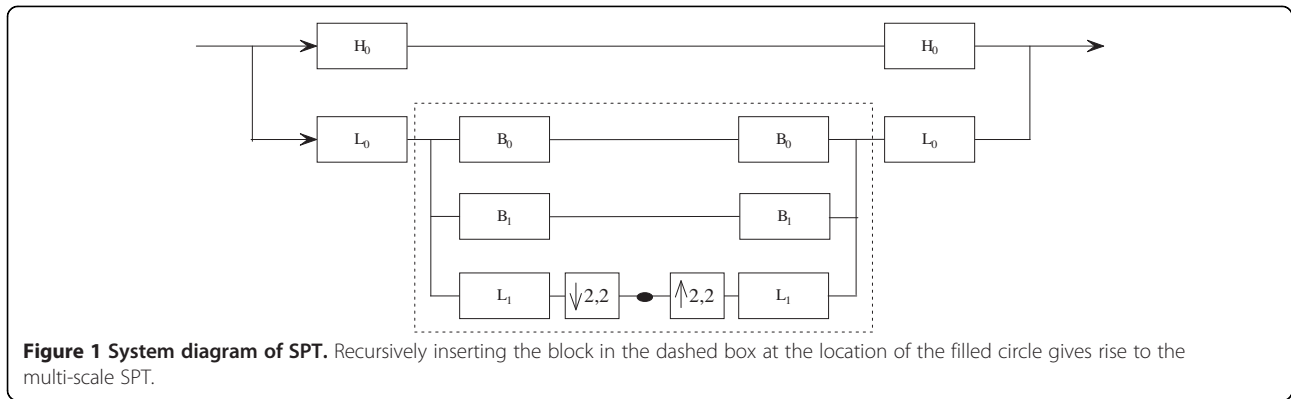
where $\theta = \arg(\boldsymbol{\omega})$, $\theta_k = \pi k/K$, and $B(\boldsymbol{\omega}) = \sqrt{\sum_{k=0}^{K-1} |B_k(\boldsymbol{\omega})|^2}$.

Under the constraints in Eqs. (1) and (2), Karasaridis and Simoncelli employed a numerical technique to design the SPT [33], but unfortunately, this resulted in an SPT with non-perfect reconstruction. In contrast, Portilla et al. [34] devised an SPT with perfect reconstruction. This satisfies the constraints described above and can be represented in a closed form. Because perfect reconstruction is a natural requisite for watermarking, we only introduce the closed-form SPT in [34]. This SPT is represented in the Fourier domain, with polar-separable filters written as:

$$L_1(r, \theta) = \begin{cases} \cos\left(\frac{\pi}{2} \log_2\left(\frac{4r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 1, & r \leq \frac{\pi}{4} \\ 0, & r \geq \frac{\pi}{2} \end{cases} \quad (3)$$

$$B_k(r, \theta) = H(r)G_k(\theta) (k = 0, \dots, K-1), \quad (4)$$

where $r = |\boldsymbol{\omega}| = \sqrt{\omega_x^2 + \omega_y^2}$, $\theta = \arg(\boldsymbol{\omega})$, and $H(r)$ and $G_k(\theta)$ are defined as:



$$H(r) = \begin{cases} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 1, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases} \quad (5)$$

$$G_k(\theta) = \frac{(K-1)!}{\sqrt{K[2(K-1)]!}} \left[2\cos\left(\theta - \frac{\pi k}{K}\right) \right]^{K-1}. \quad (6)$$

The filters $L_0(r, \theta)$ and $H_0(r, \theta)$ are thus constructed as:

$$L_0(r, \theta) = L_1\left(\frac{r}{2}, \theta\right) \quad (7)$$

$$H_0(r, \theta) = H\left(\frac{r}{2}\right). \quad (8)$$

Note that the high-pass filter $H_0(r, \theta)$ in [34] is also split into a number of oriented subbands, i.e., $H_0(r, \theta) = H(r/2)G_k(\theta)$. Because the oriented high-pass subbands will not be used in our scheme, they have been equivalently simplified as Eq. (8).

3. Design of deformable pyramid transform with shift-invariance, steerability, and scalability

In the interest of counteracting RST in robust watermarking, we are motivated to design a DPT with shift-invariance, steerability, and scalability. We take the SPT in [34] with shift-invariance and steerability as the starting point. Based on such an SPT, we further achieve scalability by constructing scalable basis filters from the steerable ones, $B_k(r, \theta)$. According to the theory of shiftability in [29], scalable basis filters are essentially scaled versions of $B_k(r, \theta)$. As scaling $B_k(r, \theta)$ is equivalent to scaling $H(r)$ according to Eq. (4), $H(r)$ can be taken as the kernel for constructing scalable basis filters. That is, the radial component $H(r)$ in Eq. (4) is used to achieve scalability, and the angular components $G_k(\theta)$ are used to satisfy steerability. Together, this results in

the joint steerability and scalability. Furthermore, keeping the high-pass and band-pass subbands undecimated, as in the SPT, yields the property of shift-invariance. Continuing with this line of thought, we can construct the DPT, as shown in Figure 2. This achieves the desired characteristics of shift-invariance, steerability, and scalability. The $C_j(r)$ ($j = 0, 1, \dots, J-1$) in Figure 2 denote the scalable filters designed from the kernel $H(r)$.

It can be observed from Figure 2 that perfect reconstruction requires the following constraint to be satisfied:

$$|H_0(r)|^2 + |L_0(r)|^2 \left(|L_1(r)|^2 + \sum_{j=0}^{J-1} \left(|C_j(r)|^2 \cdot \sum_{k=0}^{K-1} |G_k(\theta)|^2 \right) \right) = 1. \quad (9)$$

By comparing Eq. (9) to Eqs. (1–1) and (4), we have

$$\sum_{j=0}^{J-1} |C_j(r)|^2 = |H(r)|^2. \quad (10)$$

Below, we determine a suitable number of scalable basis filters, J , derive the closed-form $C_j(r)$, and obtain the interpolation functions for steerability and scalability.

3.1. Construction of scalable basis filters

According to the sufficient and necessary condition of shiftability in [29], the number of basis filters is equal to or greater than the number of Fourier frequencies with non-zero magnitude, where the Fourier frequency denotes the kernel's frequency in the form of an imaginary exponent. Because $H(r)$ is a piecewise function in the Fourier domain, we determine the number of scalable basis filters in a piecewise fashion, as follows.

First, consider the case $r \in (\pi/4, \pi/2)$ where $H(r) = \cos((\pi/2) \log_2(2r/\pi))$. Here, $H(r)$ can be treated as a function that has undergone a logarithmic warping operation, i.e., $H(r) = \cos(\rho(2r/\pi))$, where $\rho(2r/\pi) = \pi \log_2(2r/\pi)/2 \in (-\pi/2, 0)$. Because warping operations

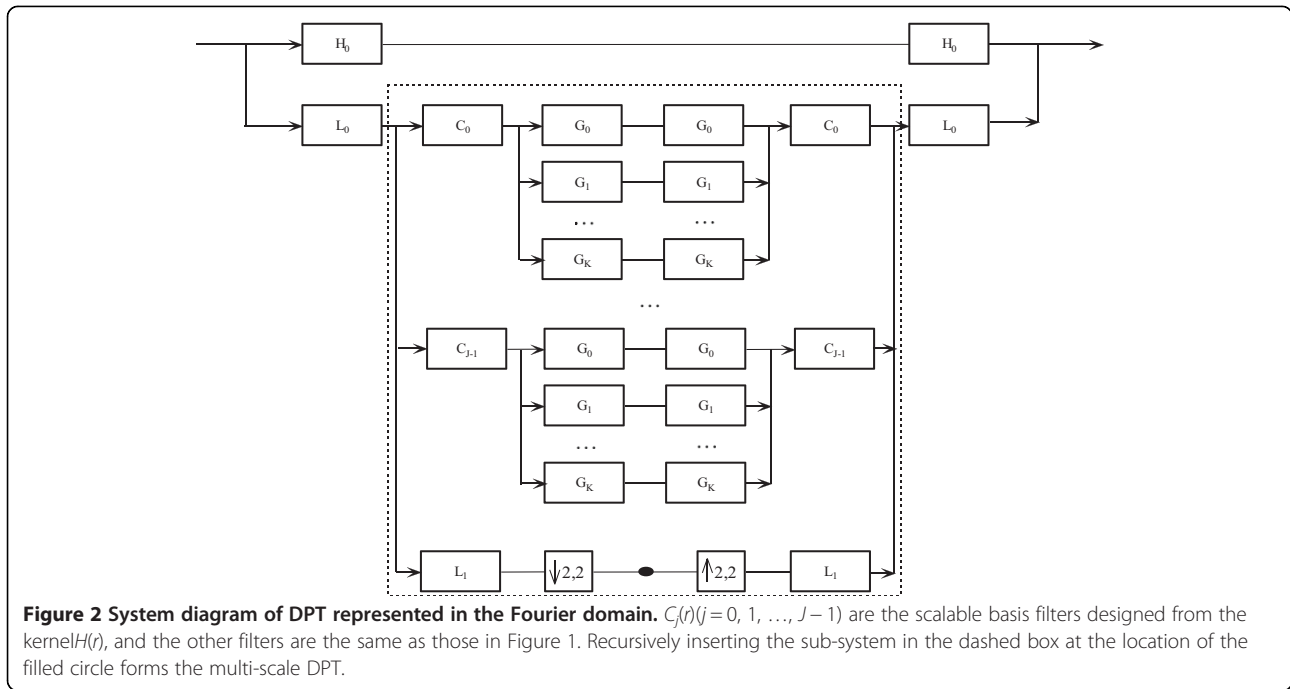


Figure 2 System diagram of DPT represented in the Fourier domain. $C_j(r)(j = 0, 1, \dots, J - 1)$ are the scalable basis filters designed from the kernel $H(r)$, and the other filters are the same as those in Figure 1. Recursively inserting the sub-system in the dashed box at the location of the filled circle forms the multi-scale DPT.

do not, according to [29], affect the property of shift-ability, the number of scalable basis filters for $r \in (\pi/4, \pi/2)$ depends on the non-warping kernel $\tilde{H}(r) = \cos(2r/\pi) = (e^{2r/\pi} + e^{-2r/\pi})/2$. Clearly, there are two Fourier frequencies with non-zero magnitude, and thus, the number of scalable basis filters for $r \in (\pi/4, \pi/2)$ satisfies $J \geq 2$. For simplicity, we choose $J = 2$ and construct, according to [29], the two scalable basis filters $C_j(r)(j = 0, 1)$ as:

$$C_j(r) = a_j \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right) + R_j\right), \quad (11)$$

where $a_j(a_j > 0)$ meets the constraint in Eq. (10), and $R_j \in (-\pi/2, 0)$ is set as:

$$R_j = -\frac{\pi}{2} + \frac{(j+1)\pi}{6} \quad (j = 0, 1), \quad (12)$$

which aims to generate frequency subbands with equal size on a logarithmic axis. To make the scalable basis filter reflection-shiftable [29], we further design $C_j(r)(j = 0, 1)$ as:

$$\begin{aligned} C_j(r) &= a_j \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right) + R_j\right) + a_j \cos\left(-\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right) + R_j\right) \\ &= 2a_j \cos(R_j) \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right). \end{aligned} \quad (13)$$

By substituting Eqs. (13) and (12) into Eq. (10), we have

$$4a_0^2 \cos^2\left(-\frac{\pi}{3}\right) + 4a_1^2 \cos^2\left(-\frac{\pi}{6}\right) = 1. \quad (14)$$

As Eq. (14) is underdetermined, there exist many values of a_j that satisfy Eq. (14). By simply setting $a_0 = a_1$, we have the following solutions:

$$a_0 = a_1 = \frac{1}{2}. \quad (15)$$

Therefore, the two scalable basis filters for the case $r \in (\pi/4, \pi/2)$ are constructed as:

$$\begin{cases} C_0(r) = \cos\left(-\frac{\pi}{3}\right) \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right) = \frac{1}{2} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right) \\ C_1(r) = \cos\left(-\frac{\pi}{6}\right) \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right) = \frac{\sqrt{3}}{2} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right) \end{cases} \quad (16)$$

We proceed to handle the case $r \in (0, \pi/4]$. The kernel $H(r)$ is $H(r) = 0$, and thus, $J \geq 0$ holds. For the case $r \in [\pi/2, \pi]$, $H(r)$ is represented as $H(r) = 1$, and hence, we have $J \geq 1$. For the convenience of construction, we uniformly adopt $J = 2$ scalable basis filters for all three cases. Under the constraint of Eq. (10), the two scalable basis filters for $r \in (0, \pi/4]$ and $r \in [\pi/2, \pi]$ are derived as $C_0(r) = C_1(r) = 0$ and $C_0(r) = C_1(r) = 1/\sqrt{2}$, respectively.

In summary, the two scalable basis filters are constructed as follows:

$$C_0(r) = \begin{cases} \frac{1}{2} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ \frac{1}{\sqrt{2}}, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases} \quad (17)$$

$$C_1(r) = \begin{cases} \frac{\sqrt{3}}{2} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ \frac{1}{\sqrt{2}}, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases}$$

3.2. Derivation of interpolation function

Under the shiftability framework [29], the interpolation function is parameterized by translation distance, rotation angle, or scaling factor, and will be used to interpolate the filter (response) at an arbitrary spatial position, orientation, or scale. Because the designed DPT is shift-invariant, we mainly derive interpolation functions for steerability and scalability.

We start with the derivation of the interpolation function for steerability. In the interest of reducing the computational complexity of geometrical synchronization, we adopt $K = 2$ steerable basis filters, i.e., $G_0(\theta) = \cos(\theta)$ and $G_1(\theta) = \cos(\theta - \pi/2)$ according to Eq. (6). From the sufficient and necessary condition of shiftability [29], the steerable interpolation function $b_k(\phi)$ satisfies the following equation:

$$e^{j\phi} = \begin{pmatrix} e^{j0} & e^{j\frac{\pi}{2}} \end{pmatrix} \begin{pmatrix} b_0(\phi) \\ b_1(\phi) \end{pmatrix}, \quad (18)$$

where ϕ denotes an arbitrary rotation angle. By requiring that both the real and imaginary parts of Eq. (18) agree, we obtain the following interpolation function for steerability:

$$\begin{cases} b_0(\phi) = \cos(\phi) \\ b_1(\phi) = \sin(\phi) \end{cases}. \quad (19)$$

We proceed to derive the interpolation function for scalability. As mentioned previously, both $H(r)$ and $C_j(r)$ ($j = 0, 1$) are piecewise. Thus, the scalable interpolation functions, say $s_j(\sigma)$, should also be piecewise, where $\sigma(\sigma > 0)$ is an arbitrary scaling factor. We first handle the case $r \in (\pi/4, \pi/2)$. As analyzed in Section 3.1, the Fourier frequency with non-zero amplitude merely depends on that before the un-warping operation. Therefore, the Fourier frequency in this case is

equal to $k = 2/\pi$. According to [29], $s_j(\sigma)$ satisfies the following equation:

$$e^{j\frac{2}{\pi}\sigma} = \begin{pmatrix} e^{j\frac{2}{\pi}R_0} & e^{j\frac{2}{\pi}R_1} \end{pmatrix} \begin{pmatrix} s_0(\sigma) \\ s_1(\sigma) \end{pmatrix}, \quad (20)$$

where R_j ($j = 0, 1$) is defined in Eq. (12). Given that both the real and imaginary parts of Eq. (20) agree, we obtain

$$\begin{cases} s_0(\sigma) = \sin\left[\frac{2}{\pi}(\sigma - R_1)\right] / \sin\left[\frac{2}{\pi}(R_0 - R_1)\right] \\ s_1(\sigma) = \sin\left[\frac{2}{\pi}(\sigma - R_0)\right] / \sin\left[\frac{2}{\pi}(R_1 - R_0)\right] \end{cases}. \quad (21)$$

For the case $r \in (0, \pi/4]$, no Fourier frequency has non-zero amplitude, and hence, $s_j(\sigma)$ can be any value. In our scheme, we simply set $s_j(\sigma) = 0$ for $r \in (0, \pi/4]$. For the case $r \in [\pi/2, \pi]$, the Fourier frequency with non-zero amplitude is $k = 0$. As a result, we have

$$1 = \begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} s_0(\sigma) \\ s_1(\sigma) \end{pmatrix}. \quad (22)$$

Because $C_0(r) = C_1(r)$ has been adopted in the DPT construction, we similarly set $s_0(\sigma) = s_1(\sigma)$ and obtain $s_0(\sigma) = s_1(\sigma) = 1/2$.

By summarizing the aforementioned results, we derive the following interpolation functions for scalability:

$$s_0(\sigma) = \begin{cases} \sin\left[\frac{2}{\pi}(\sigma - R_1)\right] / \sin\left[\frac{2}{\pi}(R_0 - R_1)\right], & \frac{\pi}{4} < r < \frac{\pi}{2} \\ \frac{1}{2}, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases}$$

$$s_1(\sigma) = \begin{cases} \sin\left[\frac{2}{\pi}(\sigma - R_0)\right] / \sin\left[\frac{2}{\pi}(R_1 - R_0)\right], & \frac{\pi}{4} < r < \frac{\pi}{2} \\ \frac{1}{2}, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases} \quad (23)$$

Using the steerable and scalable interpolation functions, we can interpolate the deformable filter at arbitrary orientation ϕ and scale σ , say $F^{\phi,\sigma}(r, \theta)$, via the following construction:

$$F^{\phi,\sigma}(r, \theta) = [s_0(\sigma)C_0(r) + s_1(\sigma)C_1(r)] \times [b_0(\phi)G_0(\theta) + b_1(\phi)G_1(\theta)], \quad (24)$$

where (r, θ) are the polar coordinates in the Fourier domain. For convenience, Eq. (24) is called the *deformable interpolation*.

Suppose that $Q_{jk}^l(r, \theta)$ ($j, k \in \{0, 1\}; l = 1, 2, \dots$) denotes the DPT basis subband at the l th pyramid level. The filter response at orientation ϕ and scale σ can then be obtained via the deformable interpolation as:

$$Q^{l,\sigma,\phi}(r, \theta) = s_0(\sigma)[b_0(\phi)Q_{00}^l(r, \theta) + b_1(\phi)Q_{01}^l(r, \theta)] \\ + s_1(\sigma)[b_0(\phi)Q_{10}^l(r, \theta) + b_1(\phi)Q_{11}^l(r, \theta)]. \quad (25)$$

Although both Eqs. (24) and (25) are represented in the Fourier domain, performing the inverse Fourier transform on them leads to a straightforward interpolation expressed in the spatial-frequency domain.

4. Mechanism for geometrical synchronization

In this section, in an attempt to counteract geometrical attacks in robust watermarking, we exploit the characteristics of shift-invariance, steerability, and scalability in the DPT to theoretically derive synchronization mechanisms for translation, rotation, and scaling. The derivation is as follows.

4.1. Synchronization for translation

Let $I(x, y)$ and $I^{x_0, y_0}(x, y)$ be the original image and its translated version, respectively, i.e., $I^{x_0, y_0}(x, y) = \mathcal{T}_{x_0, y_0}[I(x, y)] = I(x - x_0, y - y_0)$, where (x_0, y_0) is the translation distance and $\mathcal{T}_{x_0, y_0}[\cdot]$ is the translation operator. The corresponding Fourier transforms (FTs) are denoted as $I(\omega_x, \omega_y)$ and $I(\omega_x, \omega_y)e^{-j(\omega_x x_0 + \omega_y y_0)}$, respectively.

Assume that $\omega^1 = (\omega_x, \omega_y)$ represents the coordination at the first (finest) level of the DPT pyramid. Its corresponding coordination at the l th ($l \geq 1$) pyramid level is then computed as $\omega^l = (\omega_x^l, \omega_y^l) = (\omega_x/2^{l-1}, \omega_y/2^{l-1})$ (see also Figure 2). Suppose that $Q_{jk}^l(\omega^l)$ and $Q_{jk}^{l, x_0, y_0}(\omega^l)$ ($j, k \in \{0, 1\}; l = 1, 2, \dots$) are the DPT basis subbands in the Fourier domain for $I(x, y)$ and $I^{x_0, y_0}(x, y)$, respectively. According to Figure 2, we have

$$Q_{jk}^l(\omega^l) = I(\omega^l)L_0(\omega^l)(L_1(\omega^l))^{l-1}C_j(\omega^l)G_k(\omega^l), \quad (26)$$

$$Q_{jk}^{l, x_0, y_0}(\omega^l) = e^{-j(\omega_x^l x_0 + \omega_y^l y_0)}I(\omega^l)L_0(\omega^l)(L_1(\omega^l))^{l-1}C_j(\omega^l)G_k(\omega^l). \quad (27)$$

By considering Eqs. (26) and (27), we clearly find that

$$Q_{jk}^{l, x_0, y_0}(\omega_x, \omega_y) = e^{-j(\omega_x^l x_0 + \omega_y^l y_0)}Q_{jk}^l(\omega_x, \omega_y) \Leftrightarrow q_{jk}^{l, x_0, y_0}(x, y) \\ = \mathcal{T}_{x_0, y_0}[q_{jk}^l(x, y)], \quad (28)$$

where $q_{jk}^{l, x_0, y_0}(x, y)$ and $q_{jk}^l(x, y)$ are inverse FTs of $Q_{jk}^{l, x_0, y_0}(\omega_x, \omega_y)$ and $Q_{jk}^l(\omega_x, \omega_y)$ respectively.

Equation (28) implies that the DPT basis subband $q_{jk}^{l, x_0, y_0}(x, y)$ in the spatial-frequency domain for the translated input signal $I^{x_0, y_0}(x, y)$ is also the translated

version of $q_{jk}^l(x, y)$ for the original input signal. This is the essence of shift-invariance in the DPT.

4.2. Synchronization for rotation and scaling

According to the construction of shift-invariance in the DPT, the translation should affect the synchronization of rotation and scaling. To uncouple the translation from rotation and scaling, we adopt the Fourier magnitude of the input signal as the DPT's input, which in turn achieves the real translation invariance. Under such a setting, we derive the synchronization mechanism for rotation and scaling as follows.

Denote $I^{\phi, \sigma}(x, y) = \mathcal{G}_{\phi, \sigma}[I(x, y)]$ as e rotated and scaled version of the original image $I(x, y)$, where $\mathcal{G}_{\phi, \sigma}[\cdot]$ is an operator that rotates counter-clockwise by ϕ and dilates by σ about the origin. Let $M(\omega_x, \omega_y)$ and $M^{\phi, 1/\sigma}(\omega_x, \omega_y)$ be the Fourier magnitude of $I(x, y)$ and $I^{\phi, \sigma}(x, y)$, respectively. Then, we have $M^{\phi, 1/\sigma}(\omega_x, \omega_y) = \mathcal{G}_{\phi, 1/\sigma}[M(\omega_x, \omega_y)]$ according to the property of the FT.

As defined in Section 4.1, let $\omega^l = (\omega_x/2^{l-1}, \omega_y/2^{l-1})$ denote the frequency coordinate at the l th ($l \geq 1$) level of the DPT pyramid. Assume that $M^{\phi, 1/\sigma}(\omega^1)$ and $M(\omega^1)$ are decomposed via DPT into l ($l \geq 1$) pyramid levels to yield the basis subbands $Q_{jk}^{l, \phi, \sigma}(\omega^l)$ and $Q_{jk}^l(\omega^l)$ ($j, k \in \{0, 1\}; l = 1, 2, \dots$), respectively. By virtue of the steerable and scalable properties in Eq. (25), we use $Q_{jk}^l(\omega^l)$ to interpolate the response at orientation ψ and scale λ as:

$$Q^{l, \psi, \lambda}(\omega^l) = \sum_{j=0}^1 \left(s_j(\lambda) \left(\sum_{k=0}^1 b_k(\psi) Q_{jk}^l(\omega^l) \right) \right) \\ = \sum_{j=0}^1 \left(s_j(\lambda) \left(\sum_{k=0}^1 b_k(\psi) M(\omega^l) L_0(\omega^l) (L_1(\omega^l))^{l-1} C_j(\omega^l) G_k(\omega^l) \right) \right) \\ = M(\omega^l) F^{l, \psi, \lambda}(\omega^l) \quad (29)$$

where

$$F^{l, \psi, \lambda}(\omega^l) = \sum_{j=0}^1 \left(s_j(\lambda) \left(\sum_{k=0}^1 b_k(\psi) L_0(\omega^l) (L_1(\omega^l))^{l-1} C_j(\omega^l) G_k(\omega^l) \right) \right).$$

Similarly, we further use $Q_{jk}^{l, \phi, \sigma}(\omega^l)$ to obtain the response at orientation $\phi + \psi$ and scale λ/σ as:

$$Q^{l, \phi + \psi, \frac{\lambda}{\sigma}}(\omega^l) = \sum_{j=0}^1 \left(s_j \left(\frac{\lambda}{\sigma} \right) \left(\sum_{k=0}^1 b_k(\phi + \psi) Q_{jk}^{l, \phi, \sigma}(\omega^l) \right) \right) \\ = \sum_{j=0}^1 \left(s_j \left(\frac{\lambda}{\sigma} \right) \left(\sum_{k=0}^1 b_k(\phi + \psi) M^{\phi, \frac{1}{\sigma}}(\omega^l) L_0(\omega^l) (L_1(\omega^l))^{l-1} C_j(\omega^l) G_k(\omega^l) \right) \right) \\ = M^{\phi, \frac{1}{\sigma}}(\omega^l) F^{l, \phi + \psi, \frac{\lambda}{\sigma}}(\omega^l) \quad (30)$$

where $F^{l, \phi + \psi, \lambda/\sigma}(\omega^l) = \sum_{j=0}^1 \left(s_j(\lambda/\sigma) \left(\sum_{k=0}^1 b_k(\psi + \phi) L_0(\omega^l) (L_1(\omega^l))^{l-1} C_j(\omega^l) G_k(\omega^l) \right) \right)$. In the framework of shiftability [29], $F^{l, \psi, \lambda}(\omega^l)$ represents the filter at orientation

ψ and scale λ in the l th level of the multi-scale DPT (see also Figure 2). This is actually the rotated and scaled version of the kernel $F^{l,0,R_0}(\omega^l) = L_0(\omega^l)(L_1(\omega^l))^{l-1}C_0(\omega^l)G_0(\omega^l)$ at orientation 0 and scale R_0 (see also Eq. (12)). In other words, $F^{l,\psi,\lambda}(\omega^l) = \mathcal{G}_{\psi,\lambda/R_0}[F^{l,0,R_0}(\omega^l)]$ holds and so does $F^{l,\phi+\psi,\lambda/\sigma}(\omega^l) = \mathcal{G}_{\phi+\psi,\lambda/(\sigma R_0)}[F^{l,0,R_0}(\omega^l)]$. Therefore, we have

$$F^{l,\phi+\psi,\lambda/\sigma}(\omega^l) = \mathcal{G}_{\phi,\lambda/\sigma}[F^{l,\psi,\lambda}(\omega^l)]. \quad (31)$$

Taking Eq. (31) and $M^{\phi,1/\sigma}(\omega^l) = \mathcal{G}_{\phi,1/\sigma}[M(\omega^l)]$ into account, Eqs. (29) and (30) essentially imply the following synchronization mechanism for rotation and scaling:

$$\begin{aligned} Q^{l,\phi+\psi,\lambda/\sigma}(\omega^l) &= \mathcal{G}_{\phi,\lambda/\sigma}[Q^{l,\psi,\lambda}(\omega^l)] \Leftrightarrow Q^{l,\psi,\lambda}(\omega^l) \\ &= \mathcal{G}_{-\phi,\sigma}[Q^{l,\phi+\psi,\lambda/\sigma}(\omega^l)]. \end{aligned} \quad (32)$$

Performing the inverse FT leads to the rotation and scaling synchronization mechanism in the spatial-frequency domain:

$$\begin{aligned} q^{l,\phi+\psi,\lambda/\sigma}(\mathbf{s}^l) &= \mathcal{G}_{\phi,\sigma}[q^{l,\psi,\lambda}(\mathbf{s}^l)] \Leftrightarrow q^{l,\psi,\lambda}(\mathbf{s}^l) \\ &= \mathcal{G}_{-\phi,\sigma^{-1}}[q^{l,\phi+\psi,\lambda/\sigma}(\mathbf{s}^l)], \end{aligned} \quad (33)$$

where $\mathbf{s}^l = (x/2^{l-1}, y/2^{l-1})$ is the coordination in the spatial-frequency domain at pyramid level l , and $q^{l,\phi+\psi,\sigma/\lambda}(\mathbf{s}^l)$ and $q^{l,\psi,\lambda/\sigma}(\mathbf{s}^l)$ are the inverse FTs of $Q^{l,\phi+\psi,\lambda/\sigma}(\omega^l)$ and $Q^{l,\psi,\lambda}(\omega^l)$, respectively.

Based on Eq. (32), the synchronization for rotation and scaling can be performed as follows: decompose the Fourier magnitude of $I^{\phi,\sigma}(x, y)$ into an l -level DPT pyramid to generate the basis subbands $Q_{jk}^{l,\phi,\sigma}(\omega^l)$ ($j, k \in \{0, 1\}; l = 1, 2, \dots$). Then, interpolate the response at orientation $\phi + \psi$ and scale λ/σ as $Q_{jk}^{l,\phi+\psi,\lambda/\sigma}(\omega^l)$. Finally, successively rotate counter-clockwise by $-\phi$ and dilate by σ the interpolated subband $Q_{jk}^{l,\phi+\psi,\lambda/\sigma}(\omega^l)$ to yield the response $Q^{l,\psi,\lambda}(\omega^l)$ at orientation ψ and scale λ . The $Q^{l,\psi,\lambda}(\omega^l)$ is equivalent to the subband at orientation ψ and scale λ that is synthesized from the DPT basis subbands $Q_{jk}^l(\omega^l)$ of the original image $I(x, y)$. The rotation and scaling synchronization using Eq. (33) is similar to that based on Eq. (32).

5. Proposed robust watermarking scheme

In this section, we present the proposed robust image watermarking algorithm, which is RST-resilient. The translation invariance is achieved by taking the Fourier magnitude of the cover image $I(x, y)$ as the DPT input, and the rotation and scaling are counteracted using the inserted template and the rotation and scaling

synchronization. The details are given below, where only $K=2$ steerable basis filters are adopted to reduce the computational complexity of the rotation and scaling synchronization.

5.1. Template and watermark inserti

Assume that the size of cover image $I(x, y)$ is $H \times W$. To obtain favorable resolution for template matching, we symmetrically pad (crop) the rows/columns of $I(x, y)$ to the size of 1,024 if the height/width, H/W , is smaller (larger) than 1,024. We then calculate its Fourier magnitude $M(\omega_x, \omega_y)$ and phase $\Psi(\omega_x, \omega_y)$, and further decompose $M(\omega_x, \omega_y)$ into a three-level DPT pyramid to generate the spatial-frequency basis subbands $q_{jk}^l(x, y)$ ($j, k \in \{0, 1\}; l = 1, 2, 3$). Among these, the subbands at the first (finest) level, $q_{jk}^1(x, y)$, are used for template insertion, whereas those at the other two levels, $q_{jk}^l(x, y)$ ($l = 2, 3$), are for watermark embedding. We chose to embed in the spatial-frequency domain instead of the Fourier domain because the symmetry of the Fourier magnitude would decrease the number of candidate coefficients for watermarking and thus the embedding capacity. The template and watermark embedding process is illustrated in Figure 3, which is explained as follows.

5.1.1. Template embedding

- (1) Generate, via a secret key KEY_{t1} , a random sequence $P = \{p_i \in \{+1, -1\}, i = 1, \dots, N_t\}$ of length N_t as the template.
- (2) To enhance the security, we tune $q_{jk}^1(x, y)$ to the predefined secret orientation θ_t and scale σ_t and obtain $q^{1,\theta_t,\sigma_t}(x, y)$. According to the steerability and scalability in Eq. (25), we have

$$\begin{aligned} q^{1,\theta_t,\sigma_t} &= \cos\theta_t \cdot \mathcal{F}^{-1}(s_0(\sigma_t)Q_{00}^1 + s_1(\sigma_t)Q_{10}^1) \\ &\quad + \sin\theta_t \cdot \mathcal{F}^{-1}(s_0(\sigma_t)Q_{01}^1 + s_1(\sigma_t)Q_{11}^1), \end{aligned} \quad (34)$$

where $Q_{jk}^1(\omega_x, \omega_y)$ denotes the FT of $q_{jk}^1(x, y)$ and $\mathcal{F}^{-1}(\cdot)$ is the inverse FT. Note that the coordinates in Eq. (34) are omitted for compactness.

- (3) Randomly select N_t template positions from $q^{1,\theta_t,\sigma_t}(x, y)$ using a secret key KEY_{t2} , which is denoted as $PS = \{(x_i, y_i), i = 1, 2, \dots, N_t\}$. As a trade-off between robustness and imperceptibility, we prefer the (x_i, y_i) located in the spatial-frequency region with normalized radius $r \in (\pi/4, \pi/2)$. Then, embed the template in the selected positions using

$$u^{1,\theta_t,\sigma_t}(x_i, y_i) = q^{1,\theta_t,\sigma_t}(x_i, y_i) + \beta_t p_i, \quad (35)$$

where β_t is the embedding strength.

- (4) Tune $u^{1,\theta_t,\sigma_t}(x_i, y_i)$ backward to obtain the watermarked basis subbands $u_{jk}^1(x, y)$. This, however,

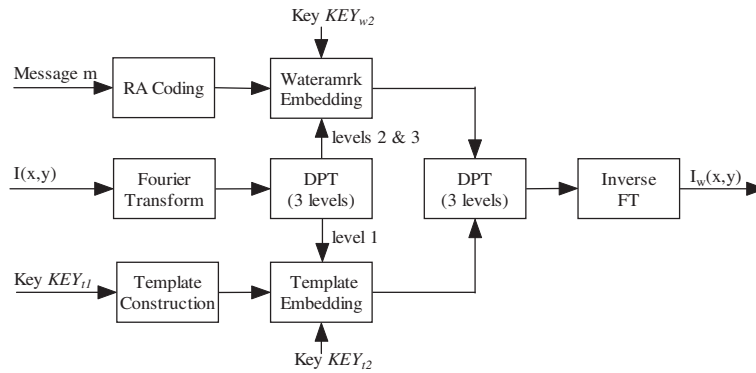


Figure 3 System diagram of template and watermark insertion, where RA, DPT, and FT denote the repeat-accumulate, the deformable pyramid transform, and the Fourier transform, respectively.

is non-trivial for the following two reasons. First, it is difficult to interpolate Eq. (35) backward to yield four embedded basis subbands $u_{jk}^1(x, y)$, $k \in \{0, 1\}$. Second, $s_j(\sigma_t)$ is a piecewise function with respect to $\mathcal{F}^{-1}(\cdot)$, according to Eq. (23), and thus, the interpolation in Eq. (34) cannot be implemented directly in the spatial-frequency domain. The latter situation implies that multiple FTs are required to complete the template insertion. This will significantly degrade the performance of brute-force template matching by the receiver and consequently make the template matching unaffordable.

To simplify the template insertion and template matching, we are motivated to adopt a non-piecewise $s_j(\sigma_t)$, e.g., setting $s_j(\sigma_t)$ to a fixed value u ($u > 0$), which turns Eq. (34) into

$$q^{1, \theta_t, \sigma_t} = u \cos \theta_t (q_{00}^1 + q_{10}^1) + u \sin \theta_t (q_{01}^1 + q_{11}^1). \quad (36)$$

Because $s_j(\sigma_t)$ is piecewise, we determine a suitable u in a piecewise manner. As pointed out in Section 3.2, for $r \in [0, \pi/4]$, $s_j(\sigma_t)$ can be any value. Thus, we merely consider the cases of $r \in (\pi/4, \pi/2)$ and $r \in [\pi/2, \pi]$. For the case $r \in [\pi/2, \pi]$, the setting $s_j(\sigma_t) = 1/2$ is already a fixed value. For $r \in (\pi/4, \pi/2)$, taking Eqs. (16) and (5) into account, we calculate the expression $s_0(\sigma_t)Q_{0k}^1 + s_1(\sigma_t)Q_{1k}^1$ in Eq. (34) as:

$$\begin{aligned} s_0(\sigma_t)Q_{0k}^1 + s_1(\sigma_t)Q_{1k}^1 &= s_0(\sigma_t)ML_0C_0G_k + s_1(\sigma_t)ML_0C_1G_k \\ &= \frac{s_0(\sigma_t) + \sqrt{3}s_1(\sigma_t)}{2}ML_0HG_k \\ &= \frac{s_0(\sigma_t) + \sqrt{3}s_1(\sigma_t)}{1 + \sqrt{3}} \left(\frac{1 + \sqrt{3}}{2}ML_0HG_k \right) \end{aligned} \quad (37)$$

Given that the scale range concerned in our scheme is $[0.5, 2]$ (a broader scale range would degrade the robustness to scaling attacks), the value of $(s_0(\sigma_t) + \sqrt{3}s_1(\sigma_t))/$

$(1 + \sqrt{3})$ is in the range $[0.69, 0.95]$. Thus, we roughly set $s_j(\sigma_t) = u = 0.7$, which approximates the $s_j(\sigma_t)$ in the cases of $r \in (\pi/4, \pi/2)$ and $r \in [\pi/2, \pi]$. Although such an approximation will lead to interpolation errors, it is demonstrated to be feasible by the extensive experimental results in Section 6.

Via the simplified Eq. (37), we equivalently embed the template in the DPT basis subbands $q_{jk}^1(x_i, y_i)$ as follows:

$$\begin{cases} u_{j0}^1(x_i, y_i) = q_{j0}^1(x_i, y_i) + \frac{\beta_i p_i \cos(\theta_t)}{u} \\ u_{j1}^1(x_i, y_i) = q_{j1}^1(x_i, y_i) + \frac{\beta_i p_i \sin(\theta_t)}{u} \end{cases}, j = 0, 1 \quad (38)$$

which avoids both the forward and backward interpolations and solves the problem that exists in the backward interpolation.

5.1.2. Watermark embedding

- (1) Generate N_m random bits $\mathbf{b} = \{b_i, i = 1, \dots, N_m\}$ as the message using a secret key KEY_{w1} .
- (2) Encode \mathbf{b} with the repeat-accumulate (RA) code of rate $rate$ [35] to generate the encoded binary sequence $\mathbf{e} = \{e_i, i = 1, \dots, N_m/rate\}$, where RA is a kind of code with excellent codec performance.
- (3) Because there exists a natural quad-tree structure between $q_{jk}^3(x, y)$ ($j, k \in \{0, 1\}$) and $\{q_{jk}^2(2x-1, 2y-1), q_{jk}^2(2x, 2y-1), q_{jk}^2(2x-1, 2y), q_{jk}^2(2x, 2y)\}$, we group the four quad-trees from four different subbands $q_{jk}^1(x, y)$ together to form a 20-element vector tree $T_i = \{T_{i\nu}, \nu = 1, \dots, 20\}$ ($i = 1, \dots, 1024 \times 1024/16$), as illustrated in Figure 4, where the child coefficients of $q_{00}^3(x, y)$ are listed but the other child coefficients are omitted from the figure for compactness. In our scheme, each vector tree is taken as the basic unit for watermarking. This is an attempt to achieve a reasonable trade-off between robustness and embedding capacity.

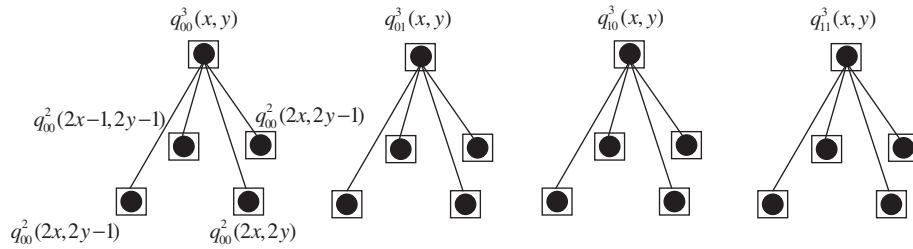


Figure 4 Illustration of a 20-element vector tree.

- (4) In the interest of resisting against cropping, we choose, via a secret key KEY_{w2} , $N_m/rate$ vector trees located in the central region for watermark insertion. Assume that each vector tree T_i is inserted with one encoded bit e_i ($e_i = 0, 1$). We then need to take a 20-element vector to represent e_i . To enhance the watermarking detection performance, we set the 20-element vector $w_0 = \underbrace{\{-1, \dots, -1\}}_{20 \times -1 \text{ 's}}$ for e_0 and

$$w_1 = \underbrace{\{+1, \dots, +1\}}_{20 \times +1 \text{ 's}} \text{ for } e_1, \text{ which achieves the}$$

maximum codeword distance and thus decreases the detection error probability.

- (5) Associate the allocated bit e_i to w_{e_i} and perform the embedding as follows:

$$Y_i = T_i + \beta_w w_{e_i}, \quad (39)$$

where Y_i is the watermarked vector tree and β_w is a non-adaptive embedding strength because, to the best of our knowledge, no suitable human visual model has been reported in the literature for the situation in our scheme. Equation (39) is equivalently written as:

$$u_{jk}^l(x_{iv}, y_{iv}) = q_{jk}^l(x_{iv}, y_{iv}) + \beta_w(2e_i - 1), \quad v = 1, \dots, 20, l = 2, 3, \quad (40)$$

where (x_i, y_i) is the coordination corresponding to the v th element of T_i .

- (6) Embed all bits e_i into the chosen vector trees by iteratively implementing step 5.
 (7) Finally, perform the inverse DPT on the watermark-inserted basis subbands u_{jk}^l ($l = 2, 3$) and the template-embedded ones u_{jk}^1 in Section 5.1.1 to obtain the watermarked Fourier magnitude $M_w(\omega_x, \omega_y)$.
 (8) Multiply $M_w(\omega_x, \omega_y)$ by the original phase $\Psi(\omega_x, \omega_y)$ and perform the inverse FT to obtain the watermarked image $I_w^{pre}(x, y)$ of size $1,024 \times 1,024$.
 (9) Execute the inverse padding (cropping) operation on $I_w^{pre}(x, y)$ to obtain the final watermarked image, $I_w(x, y)$, of size $H \times W$.

5.2. Efficient template-matching algorithm

Because translation invariance has been achieved by taking the Fourier magnitude of the cover image as the DPT input, we merely use the inserted template to estimate the rotation angle and scaling factor. These will be used to correct the rotation and scaling before watermark extraction. Based on the synchronization mechanisms for rotation and scaling in Section 4.2, we develop the efficient template-matching algorithm as follows.

Assume that the received image is $I_r(x, y)$. We first preprocess $I_r(x, y)$ with the same method as in the embedding stage to give an image size of $1,024 \times 1,024$. We then calculate the Fourier magnitude of the preprocessed image and decompose the resulting magnitude into a one-level DPT pyramid. This is because only the template inserted at level 1 is required for template matching. This yields the DPT basis subbands $q_{jk}^1(x, y)$ ($j, k \in \{0, 1\}$). According to Eq. (36), the template matching for rotation and scaling estimation can be performed as follows. The basis subbands $q_{jk}^1(x, y)$ are tuned to any candidate orientation and scale, and the tuned subband is then inversely rotated and dilated. The template is extracted accordingly to compute the correlation with the original template. After all candidate rotation angles and scaling factors have been searched in this way, the orientation and scale corresponding to the maximum correlation are adopted as the estimated parameters for rotation and scaling.

From the process discussed above, it can be seen that only a limited number of template points are involved in template matching. This motivates us to simplify the template matching by merely interpolating the relevant template points, as described below.

- (1) Set the range $[-180, 180]$ with step Δ_ϕ (e.g., $\Delta_\phi = 0.5$) as the search space for the rotation angle, and $[\sigma_1, \sigma_2]$ (e.g., $[0.5, 2.0]$) with step Δ_σ (e.g., $\Delta_\sigma = 0.01$) as that for the scaling factor. Initialize the search parameters as $\phi = -180$ and $\sigma = \sigma_1$.
- (2) For each parameter pair (ϕ, σ) , compute the candidate template position as:

$$\begin{aligned} x'_i &= \text{round}(((x_i - cx) \cdot \cos\phi + (y_i - cy) \sin\phi) / \sigma + cx) \\ y'_i &= \text{round}(((y_i - cy) \cdot \cos\phi - (x_i - cx) \sin\phi) / \sigma + cy), \end{aligned} \quad (41)$$

where $(x_i, y_i)(i = 1, 2, \dots, N_t)$ denotes the original template coordinates determined by key KEY_{t2} , (cx, cy) is the geometrical center, and $\text{round}(\cdot)$ is the rounding operation.

- (3) Obtain, via the steerability and scalability in Eq. (36), the coefficients at location (x'_i, y'_i) as:

$$\begin{aligned} q^{1,\phi,\sigma}(x'_i, y'_i) &= u \cos\phi(q_{00}^1(x'_i, y'_i) + q_{10}^1(x'_i, y'_i)) \\ &\quad + u \sin\phi(q_{01}^1(x'_i, y'_i) + q_{11}^1(x'_i, y'_i)). \end{aligned} \quad (42)$$

- (4) Calculate the correlation between the extracted and original templates as:

$$\text{Corr}(\phi, \sigma) = \sum_{i=1}^{N_t} q^{1,\phi,\sigma}(x'_i, y'_i) \cdot p_i \quad (43)$$

- (5) Increase the candidate scale σ to $\sigma = \sigma + \Delta_\sigma$ while keeping ϕ unchanged. Repeat steps 2 to 4 until $\sigma \geq \sigma_2$.
- (6) Augment the candidate rotation angle ϕ by Δ_ϕ , i.e., $\phi = \phi + \Delta_\phi$ and re-execute steps 2 to 5 until $\phi \geq 180$.
- (7) Find the maximum correlation value $\text{Corr}(\phi, \sigma)_{\max}$ and take the corresponding geometrical parameters $(\phi_{\text{est}}, \sigma_{\text{est}})$ as the estimated rotation angle and scaling factor.
- (8) Calculate the real parameters of rotation and dilation attacks as $\phi_{\text{attack}} = \phi_{\text{est}} - \theta_t$ and $\sigma_{\text{attack}} = \sigma_{\text{est}} / \sigma_t$, respectively. This is because ϕ_{est} and σ_{est} are essentially, according to Section 4.2, equal to $\phi_{\text{est}} = \theta_t + \phi_{\text{attack}}$ and $\sigma_{\text{est}} = \sigma_t \sigma_{\text{attack}}$, respectively, where θ_t , ϕ_{attack} , σ_t , and σ_{attack} correspond to ψ , ϕ , $1/\lambda$, and σ in Section 4.2, respectively.

Although the above template-matching algorithm only addresses symmetrical scaling, i.e., the scaling factors along the x - and y -axes are the same, it can easily be extended to the situation with different scaling factors. To this end, set the parameter space $(\phi, \sigma_x, \sigma_y)$ with $\phi \in [-180, 180]$ and $\sigma_x, \sigma_y \in [\sigma_1, \sigma_2]$, and then search each parameter space successively, similar to the above algorithm. Nevertheless, this would significantly increase the computational complexity.

5.3. Geometrical correction and watermark extraction

Suppose that $M^{\theta,1/\sigma}(\omega_x, \omega_y)$ denotes the Fourier magnitude of the preprocessed image of size $1,024 \times 1,024$. We then correct $M^{\theta,1/\sigma}(\omega_x, \omega_y)$ by rotating counter-clockwise by $-\phi_{\text{attack}}$ and scaling by σ_{attack} about the origin, and obtain the

Fourier magnitude $M^{0,1}(\omega_x, \omega_y)$ corresponding to the original watermarked image at orientation 0 and scale 1. Next, the watermark is recovered from $M^{0,1}(\omega_x, \omega_y)$ as follows.

- (1) Decompose $M^{0,1}(\omega_x, \omega_y)$ into a three-level pyramid via the DPT, which yields the DPT basis subbands $q_{jk}^l(x, y)(l = 1, 2, 3; j, k \in \{0, 1\})$.
- (2) Use $q_{jk}^l(x, y)$ to construct the 20-element vector trees $Z_i(i = 1, \dots, 1,024 \times 1,024/16)$ in the same way as watermark embedding. Choose N_m/rate vector trees located in the central area via the secret key KEY_{w2} .
- (3) For each selected vector tree Z_i , extract the encoded message bits as follows:

$$\sum_{v=1}^{20} Z_{iv} w_{0v} \begin{matrix} > \\ < \end{matrix} \begin{matrix} b=0 \\ b=1 \end{matrix} \sum_{v=1}^{20} Z_{iv} w_{1v}, \quad (44)$$

where $w_b(b = 0, 1)$ are as in Eq. (39).

- (4) After completing the extraction of encoded message bits from all N_m/rate selected vector trees, run the RA decoding to recover the raw message \hat{b} .

6. Experimental results and discussion

In this section, we assess the proposed watermarking scheme via experimental simulations. In the simulations, we test 20 512×512 -grey images with different textures. For each test image, we decompose its Fourier magnitude into a three-level DPT pyramid. The first (finest) level is used for template insertion, and the other two levels are for watermark embedding. The template consists of $N_t = 105$ random bits and is inserted in positions with normalized radiuses $r \in \{0.3, 0.35, 0.4\}$ and angles $\theta \in \{1:1:10, 15:10:95, 100:17:359\}$, where 1, 10, and 17 denote the secret step. The watermark is a sequence of 720 bits that is formed by encoding the $N_m = 60$ random message bits with the RA code of rate $\text{rate} = 1/12$. The embedding strengths β_t and β_w are adjusted image-by-image such that the peak signal-to-noise ratio (PSNR) is 40 dB. Figure 5 illustrates several watermarked images. This figure demonstrates that the images watermarked by the proposed scheme have feasible visual fidelity. The mean and variance of all PSNRs are calculated as 40.01 dB and 0.01, respectively.

For the 20 generated watermarked images, we impose geometrical attacks (e.g., rotation, scaling, cropping) and common signal processing attacks (e.g., JPEG compression, additive white Gaussian noise (AWGN), median filtering, convolution filtering). We then deploy the efficient template-matching algorithm in Section 5.2 to achieve rotation and scaling synchronization, where the search spaces for rotation and scaling are set as $\phi \in [-180, 180]$ with step $\Delta_\phi = 0.5$ and $\sigma \in [0.5, 2.0]$ with



Figure 5 Illustration of watermarked images. From left to right, the images are Baboon (PSNR = 40.00 dB), Lena (PSNR = 40.16 dB), and Tank (PSNR = 40.01dB), respectively.

$\Delta_\sigma = 0.01$, respectively. The performance against these attacks is summarized below.

6.1. Performance against geometrical attacks

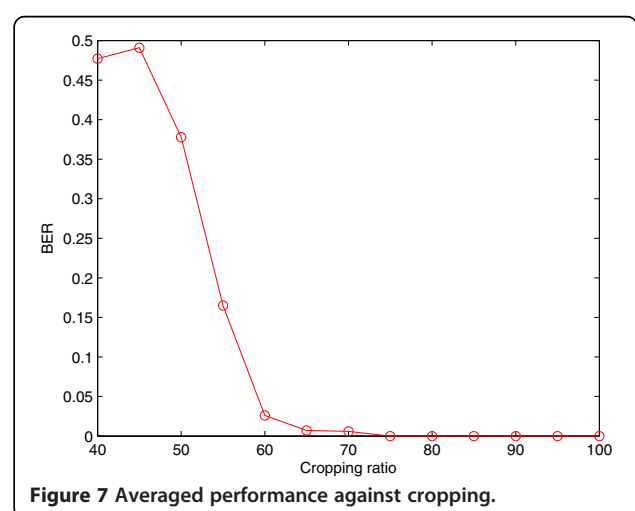
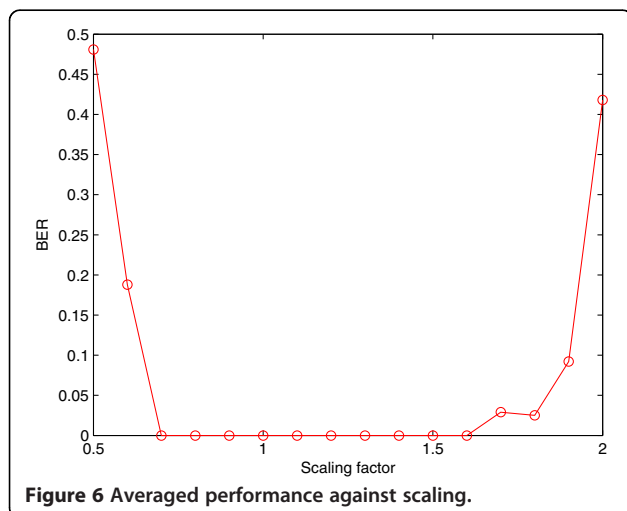
As translation invariance can be theoretically ensured by taking the Fourier magnitude of the cover image as the DPT input, the translation is no longer assessed in this paper. We mainly examine the performance against geometrical attacks such as rotation, scaling, cropping, and row/column line removal, which is practically implemented in StirMark [36,37].

In StirMark, rotation attacks include rotation without auto cropping, rotation with auto cropping, and rotation with auto cropping and scaling. For these three types of attack, we set the rotation angles as $\pm 2^\circ$, $\pm 1^\circ$, $\pm 0.75^\circ$, $\pm 0.5^\circ$, $\pm 0.25^\circ$, 45° , and 90° . We then use the efficient template-matching algorithm in Section 5.2 to estimate the rotation angle and scaling factor, followed by geometrical correction and watermark extraction. The experimental simulation shows that the bit error rates (BERs) for all concerned parameters are exactly 0, which demonstrates the high robustness of the proposed scheme to differently implemented rotations.

For scaling attacks in StirMark, we set the scaling factors in the range [0.5, 2.0] with step 0.1. The performance is shown in Figure 6, where the BER is averaged over all 20 test images. It is observed that the proposed scheme achieves BER = 0 for scaling factors from 0.7 to 1.6 and BER < 0.1 for scaling factors from 1.7 to 1.9. However, it is vulnerable to scaling factors of 0.5 and 2. The failure to counter scaling with a factor of 0.5 can be attributed to the loss of 75% of the image information, and the weakness to scaling with a factor of 2 comes from the interpolation approximation via Eq. (36).

Figure 7 summarizes the averaged performance against cropping attacks in the cropping ratio range [40%, 100%] with step 5%, where the ratio is that of the cropped image to the original image. It is found that the proposed scheme achieves BER = 0 for cropping ratios in the range [75%, 100%] and BER < 0.03 for ratios from 60% to 75%. Nevertheless, it is sensitive to cropping with ratios below 60%, which is to be expected as such cropping would lose more than 40% of the image information. In this sense, our scheme has sufficient robustness to cropping.

In addition, we also examine row/column line removal attacks, which are considered to be a kind of geometrical



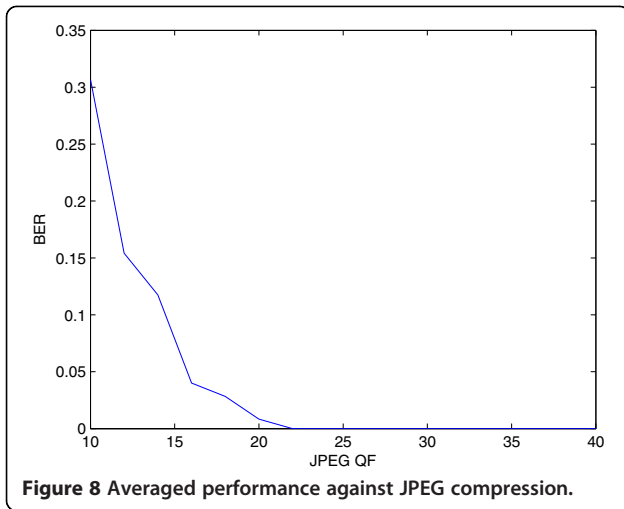


Figure 8 Averaged performance against JPEG compression.

manipulation resulting in local distortions. The frequencies of the removed row and column lines are set in the range [10,100] with step size 10. Simulation results show that the proposed scheme can successfully counteract this attack by achieving BER = 0.

6.2. Performance against common signal processing attacks

Common signal processing attacks include JPEG compression, AWGN, median filtering, and convolution filtering. We first evaluate the performance against JPEG compression. In the simulation, we set the quality factor (QF) range of JPEG compression to be [10,38] with step size 2. Figure 8 plots the performance averaged over 20 test images. It is shown that the proposed scheme has excellent performance against JPEG compression with $QF \geq 22$, as well as favorable robustness for $QF \in [16, 22)$. This preferable performance may be attributed to the redundant representation of the DPT, which is, from a mathematical viewpoint, essentially a frame expansion with promising robustness to added noise.

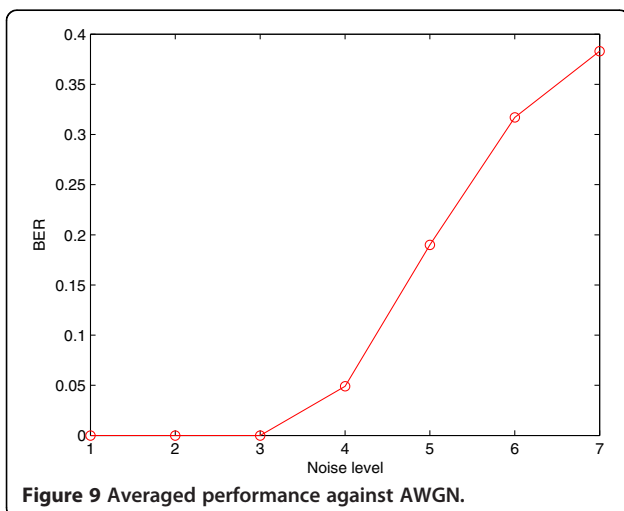


Figure 9 Averaged performance against AWGN.

Table 1 Averaged performance against median filtering

Filter size	2	3	4	5
Average BER	0	0	0.276	0.276

We further impose AWGN on the aforementioned watermarked images. We set the noise levels for AWGN in StirMark from 1 to 7 and then examine the performance in terms of BER. Figure 9 depicts the averaged performance. It is seen that the BER is 0 for noise levels of less than or equal to 3 and smaller than 0.05 for noise level 4, although it is large for the other cases. This indicates that the proposed scheme has sufficient robustness to AWGN.

In examining the performance against median filtering (cut), we set the size of the median filter in the range [2,5] with step 1. The simulation results are summarized in Table 1. It is observed that the proposed scheme has high robustness to median filtering with sizes 2 and 3, but the performance suffers with sizes 4 and 5. This implies that the proposed scheme has feasible but not sufficiently robust performance against median filtering.

Finally, we test the robustness of the proposed scheme to convolution filtering, which includes sharpening and Gaussian filtering. The simulation result shows that the average BER for sharpening is exactly 0 and that for Gaussian filtering is 0.143. This demonstrates that the proposed scheme is totally insensitive to sharpening but is not sufficiently robust to Gaussian filtering.

6.3. Computation time evaluation

In this section, we evaluate the computation time of the proposed scheme. As described in Section 5, the proposed scheme consists of message embedding and

Table 2 Performance comparison between the proposed scheme and PP-AFFINE [21] against the attacks in StirMark 3

Attacks	Scheme in [21]	Proposed
JPEG compression	0.74	0.97
Enhancement	1	1
Row/column line removal	1	1
Rotation	1	1
Random geometrical distortions	0	0
Scaling	0.78	0.68
Cropping	0.89	0.78
Shearing	1	0.20
Flip	1	0

Enhancement includes Gaussian filtering, median filtering, sharpening, and frequency model Laplacian removal, and *Rotation* denotes rotation with auto-cropping and scaling. In the simulation, the default settings in StirMark 3 [36,37] are adopted for all attacks. For each parameter in the default settings, the score is set to 1 if the message can be correctly recovered and 0 if otherwise. These scores are then averaged to yield the results in the table above.

Table 3 Performance against common processing attacks in StirMark 4.1

Attack	Tank		Globe		Lena		Man		Zelda	
	DMST	DPT	DMST	DPT	DMST	DPT	DMST	DPT	DMST	DPT
JPEG QF	19	16	12	11	17	15	20	15	13	14
AWGN	4	4	5	5	5	4	5	5	4	4
MedianCut	3 × 3	3 × 3	3 × 3	3 × 3	3 × 3	3 × 3	3 × 3	3 × 3	3 × 3	5 × 5
GaussFilt.	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok
Sharpening	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok

In the table, WNZH-DMST is abbreviated as DMST for compactness. The given values for JPEG compression, AWGN, and MedianCut denote the strongest attacks that the concerned scheme can successfully counteract. The 'Ok' for GaussFilt. (i.e., Gaussian filtering) and sharpening implies that the DMST or DPT survive all default attacks in StirMark 4.1.

extraction processes. As the computation time for the message embedding process is much less than that for the message extraction process, we mainly examine the computation time for message extraction.

The message extraction process includes two stages, namely template matching and message recovery. As the former stage takes up most of the computation time of the message extraction process, below focuses on the analysis on the computation time of template matching. According to Section 5.2, there are total $N_\phi = 360/\Delta_\phi$ candidate rotation angles and $N_\sigma = (\sigma_2 - \sigma_1)/\Delta_\sigma$ candidate scaling factors. For each candidate rotation angle, all candidate scaling factors are required to search. Therefore, the computational complexity, in unit of N_t -dimensional correlation calculation, of template matching can be represented as $O(N_\phi N_\sigma)$.

To illustrate the computation time of template matching, we perform the following experimental simulation. As set in Section 6.1, we adopt $\Delta_\phi = 0.5$, $\sigma_1 = 0.5$, $\sigma_2 = 2.0$, $\Delta_\sigma = 0.01$, and $N_t = 105$ for simulation. Under these settings, we evaluate the computation time of template matching by executing the Matlab code on an Intel personal computer (Intel, Santa Clara, CA, USA) with 2.2-GHz core(TM) 2 Duo CPU and 2-GB memory. The computation time averaged over 10 runs is 4.34 s. This implies that although a brute-force searching approach is employed for template matching, its computation time is not as high as the intuition believes. This is because only a number of template points are

incorporated in template matching, and thus, feasible computation time is achieved.

6.4. Comparison to related schemes

To further evaluate the proposed watermarking scheme, we compare it with those in [21], [38], and [14], which are denoted as PP-AFFINE, WNZH-DMST, and NIKO-SFND for notational convenience, respectively. PP-AFFINE is, as surveyed in [1], a typical template-based watermarking algorithm with high robustness against affine attacks. WNZH-DMST is another template-based watermarking approach incorporating the deformable multi-scale transform (DMST) that is somewhat similar to the DPT. NIKO-SFND is a kind of salient-feature and normalization-based watermarking scheme with excellent performance against both local and global distortions.

We start with the comparison to PP-AFFINE. In this scheme, three 512×512 -grey images, namely Baboon, Lena, and Boat, are adopted for performance assessment. Both a 60-bit message and a 14-point template are inserted in the Fourier domain, and the resulting watermarked images have PSNRs no greater than 38 dB. To ensure a fair comparison, we also embed a 60-bit message via the proposed scheme and adjust the embedding strength adaptively to make the PSNRs close to 38 dB. We then employ the same evaluation as [21] for performance comparison.

Table 2 summarizes the simulation results for PP-AFFINE and the proposed scheme. It can be observed

Table 4 Performances against global geometrical attacks in StirMark 4.1

Attack	Tank		Globe		Lena		Man		Zelda	
	DMST	DPT	DMST	DPT	DMST	DPT	DMST	DPT	DMST	DPT
RotCrp.	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok
RotScL.	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok	Ok
Scaling	0.5 to 1.81	0.6 to 2	0.5 to 1.5	0.6 to 2	0.5 to 1.63	0.7 to 1.9	0.5 to 1.51	0.7 to 1.9	0.5 to 1.57	0.7 to 1.85
Crop	0.62 to 1	0.51 to 1	0.61 to 1	0.55 to 1	0.59 to 1	0.51 to 1	0.76 to 1	0.59 to 1	0.51 to 1	0.51 to 1

In the table, 'RotCrp' and 'RotScL' denote the attacks of rotation with auto-cropping and rotation with auto-cropping and scaling, respectively. The 'OK' has the same meaning as Table 3, and the given values for 'Scaling' and 'Crop' are ranges of scaling factors and cropping ratios that DMST or DPT can successfully resist, respectively.

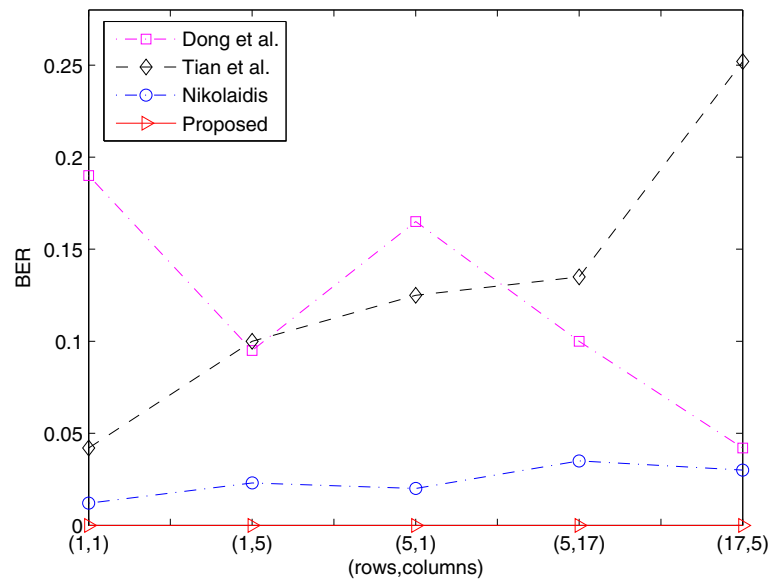


Figure 10 Performance comparison against a row/column line removal attack.

that both schemes have the same high robustness against enhancement, row/column removal, rotation with auto cropping and scaling, and random geometrical distortions. It is interesting to find that the proposed scheme obtains a significant improvement in performance against JPEG compression, where PP-AFFINE takes the Fourier magnitude of the test image as the cover signal and the proposed scheme inputs the Fourier magnitude of the test image to the DPT to generate DPT subbands as the cover signal. This implies that DPT subbands facilitate the improvement in watermarking robustness.

Nevertheless, the proposed scheme is worse than PP-AFFINE in counteracting scaling, cropping, and shearing, which is explained as follows. Compared with PP-AFFINE, the proposed scheme cannot resist against a scaling attack with factor 2, because of the interpolation approximation in Eq. (36). The proposed scheme fails to deal with cropping ratios of 50% and 75%, because at least 75% of the image information has been lost in these two situations. The weakness of the proposed scheme to shearing is to be expected, as it is designed to counteract RST rather than affine transforms.

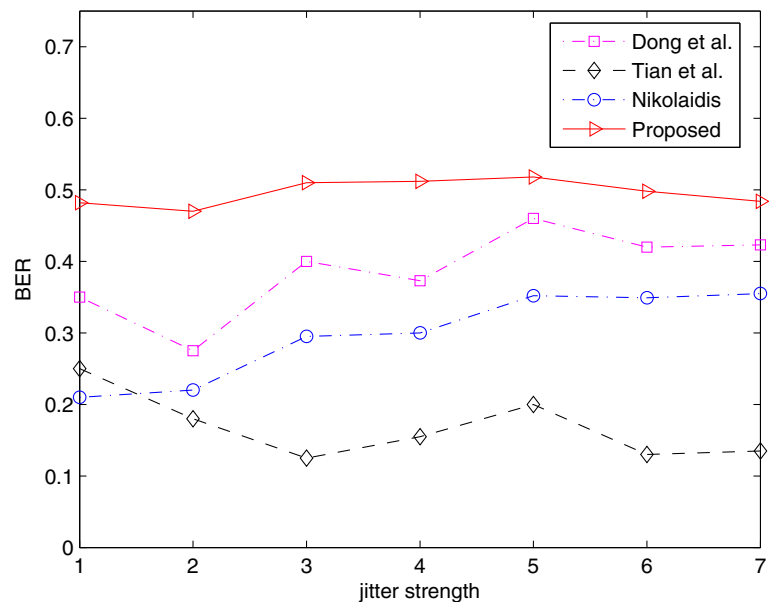
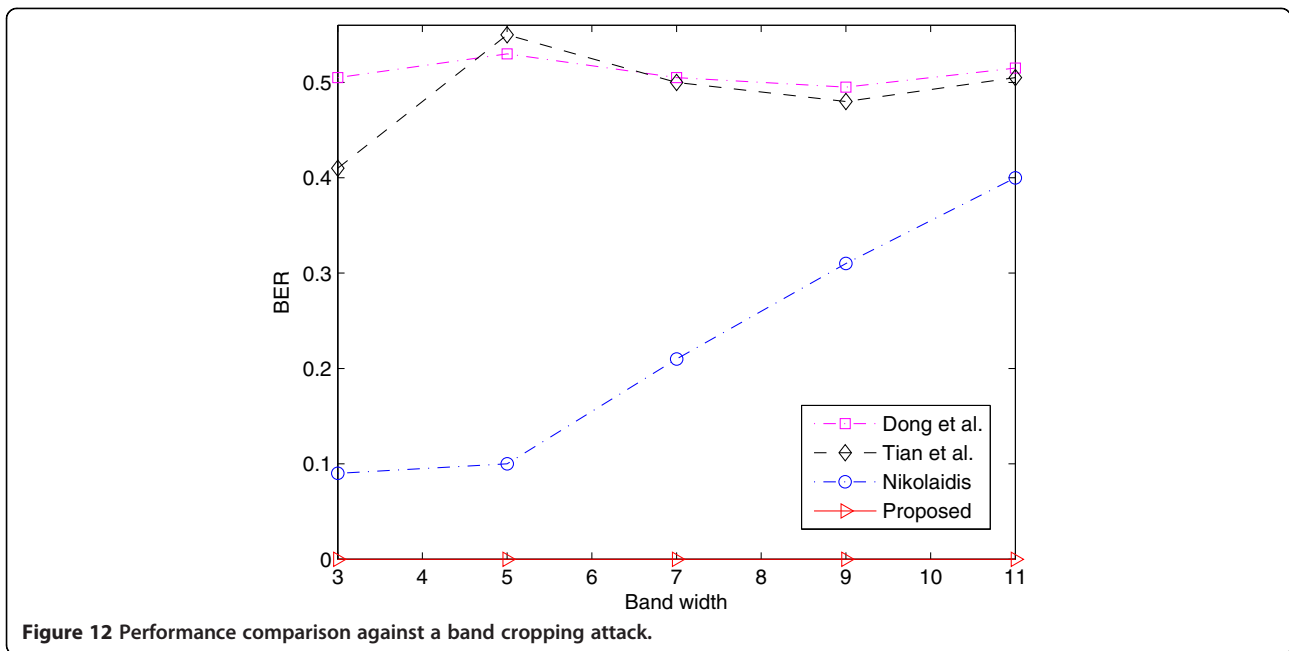


Figure 11 Performance comparison against jitter attacks.

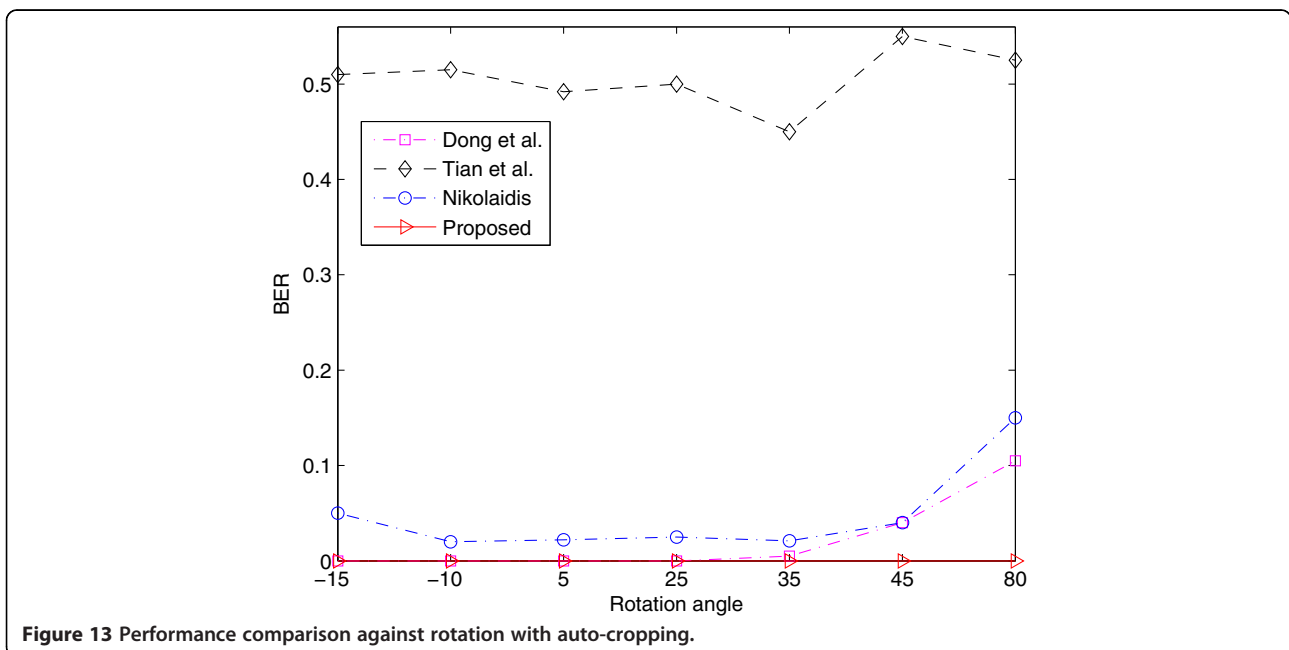


In this sense, we may claim that the proposed scheme is comparable to PP-AFFINE in terms of its performance against RST and common signal processing attacks.

We proceed to compare the proposed scheme with WNZH-DMST [38]. For fair comparison, we adopt the same settings as WNZH-DMST. In particular, we test the same five images (i.e., Tank, Globe, Lena, Man, and Zelda) as those in WNZH-DMST. We also insert a 60-bit message sequence in each test image, and let PSNRs of watermarked images be equal to those in WNZH-

DMST by slightly adjusting the embedding strength. We then perform common signal processing attacks and geometrical attacks on watermarked images via StirMark 4.1 [36,37]. The performance comparison between the proposed scheme (denoted as DPT) and WNZH-DMST is summarized in Tables 3 and 4. It is found that the performance of the proposed scheme is comparable to or better than that of WNZH-DMST.

According to [38], the DMST only has analysis filters and thus is not a pyramid transform. For this sake,



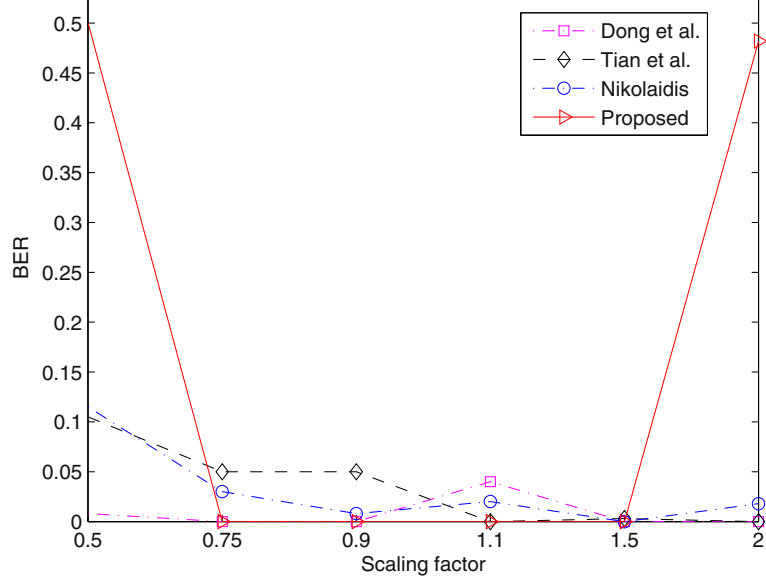


Figure 14 Performance comparison against scaling.

WNZH-DMST uses the SPT for template/watermark insertion and message extraction and adopts the DMST to estimate the rotation angle and scaling factor. As the DMST is similar to DPT's analysis filters, the template matching of WNZH-DMST is thus similar to that in the proposed scheme. By recalling the above performance comparison, it makes sense to claim that the proposed scheme is promising to achieve better performance.

In comparison to NIKO-SFND [14], we adopt the same settings as NIKO-SFND for impartial evaluation. In [14], 10 512×512 -grey images, namely Airplane,

Boat, House, Peppers, Splash, Baboon, Couple, Lena, Elaine, and Lake, were used for performance examination. For each image, a 50-bit message was inserted and the PSNR was held at 40 dB. The watermarked images were then polluted with both local and global geometrical attacks and common signal processing attacks. The performance was evaluated by comparing the NIKO-SFND scheme to state-of-the-art approaches belonging to the same category. It was found that NIKO-SFND demonstrated comparable or even better performance than the schemes it was compared with. To ensure a fair

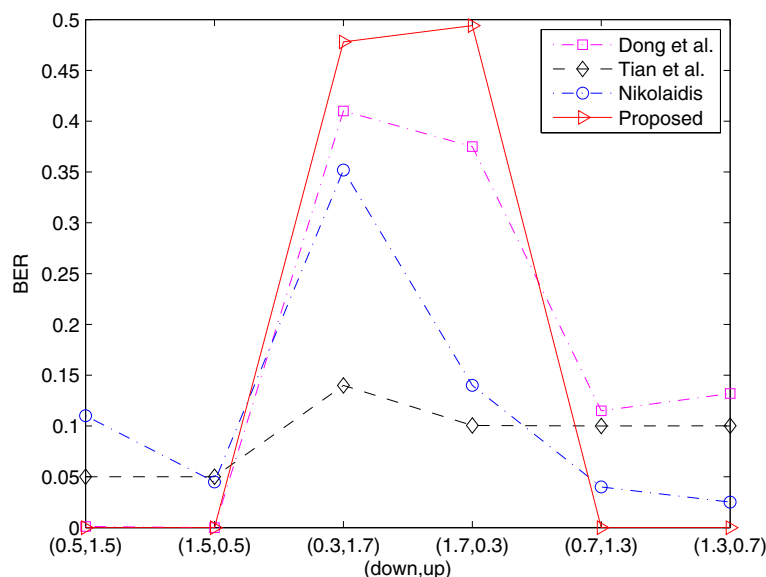


Figure 15 Performance comparison against downsampling followed by upsampling.

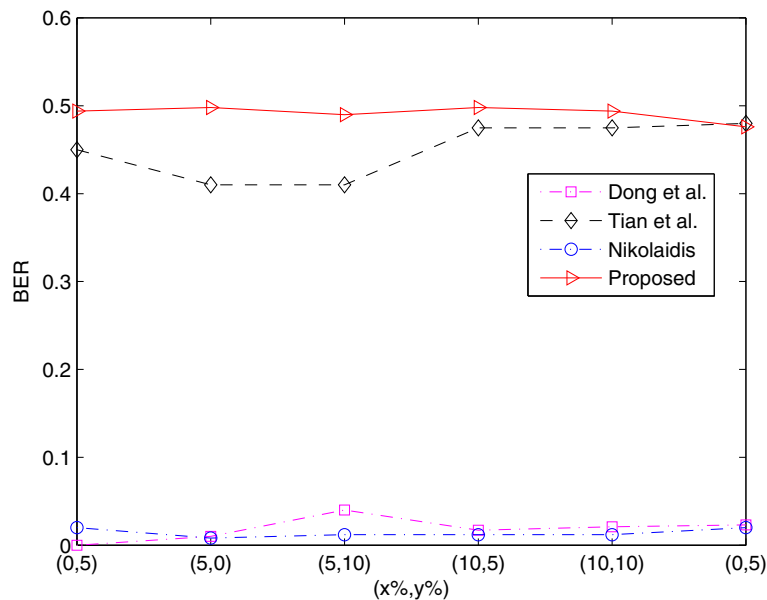


Figure 16 Performance against shearing.

comparison, these settings are similarly applied to our scheme, and the performance comparison is then carried out accordingly. In the simulation, we compare three state-of-the-art schemes, i.e., the SIFT-based NIKO_SFND and the schemes presented by Dong et al. [39] and Tian et al. [40]. The SIFT-based NIKO_SFND is one of the best of its class using different salient features.

1. Local geometrical attacks: According to [14], this type of attack includes row/column line removal, jitter, and cropping. The performance of the proposed scheme against these attacks is summarized in Figures 10, 11,

and 12, respectively, where Nikolaidis denotes the NIKO-SFND in [14], as for all other figures below. The BERs shown in figures are averaged over 10 test images and the search space $(\phi, \sigma_x, \sigma_y)$ (see Section 5.2) is adopted in evaluating the attack of row/column line removal. It can be seen that the proposed scheme significantly outperforms the three comparison approaches in counteracting the attacks of row/column line removal and cropping, whereas it is remarkably weaker than the compared schemes. The weakness to jitter attacks comes from the fact that jitter attacks are outside the scope of the proposed scheme.

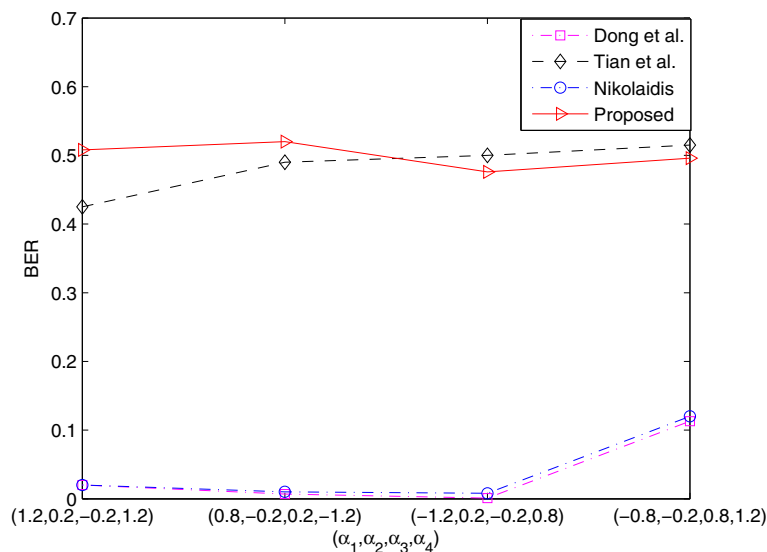


Figure 17 Performance against general affine transform.

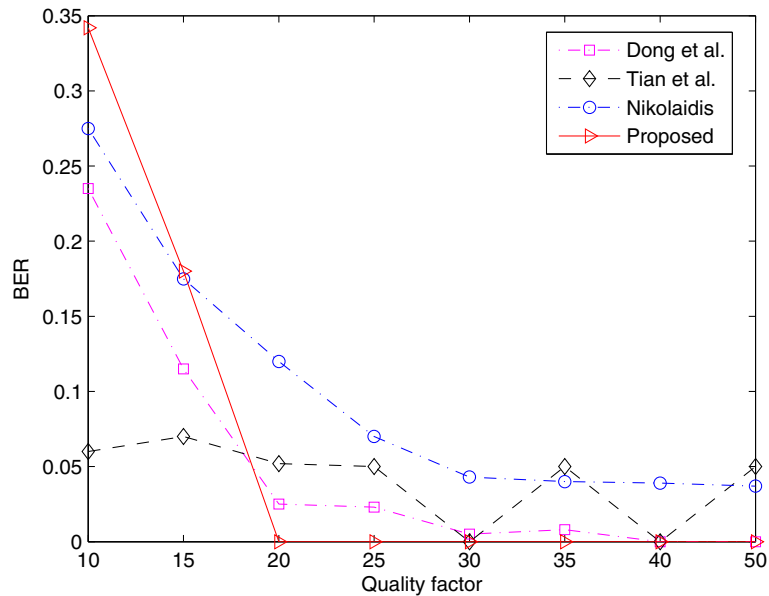


Figure 18 Performance comparison against JPEG compression.

2. Global geometrical attacks: In [14], the author evaluated the performance of NIKO-SFND against global geometrical attacks such as rotation, scaling, down-sampling followed by up-sampling, shearing, and general affine transforms. Performance comparisons are given in Figures 13, 14, 15, 16, and 17, respectively. It can be observed from Figure 13 that the proposed scheme successfully estimates and corrects all checked rotation angles. This outperforms the method of Dong et al.,

obtains a remarkable improvement over NIKO-SFND, and has superiority over Tian et al.

Based on the performance against scaling shown in Figure 14, the proposed scheme exhibits a considerable improvement over the three compared schemes for scaling factors of 0.75, 0.9, 1.1, and 1.5. However, it is much worse for scaling factors of 0.5 and 2.

Figure 15 shows that the proposed scheme has high robustness to the downsampling and up-sampling pairs

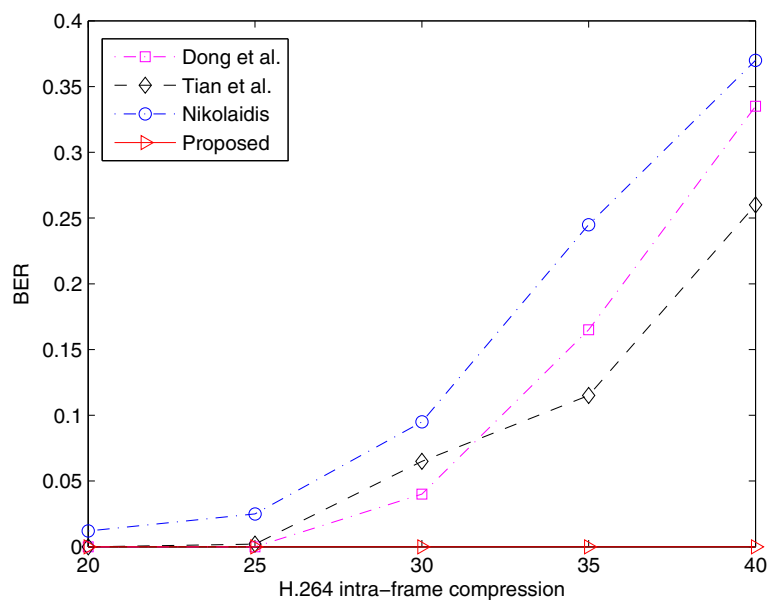
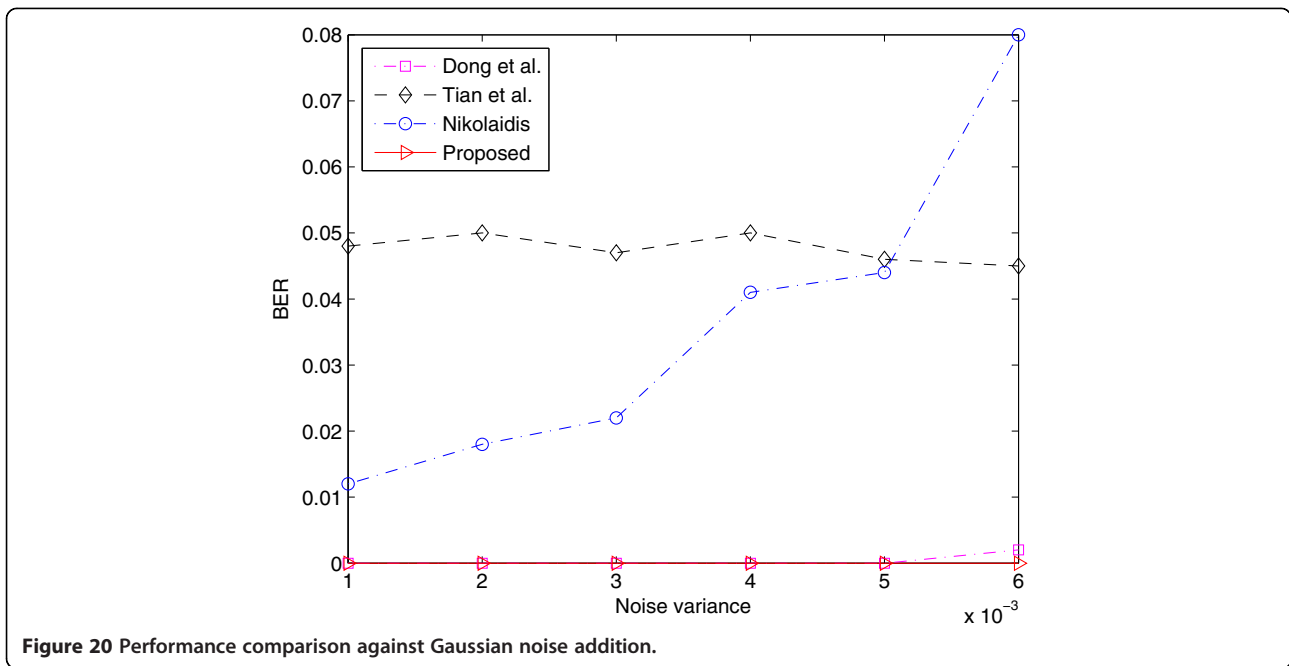


Figure 19 Performance against H.264 intra-frame compression.



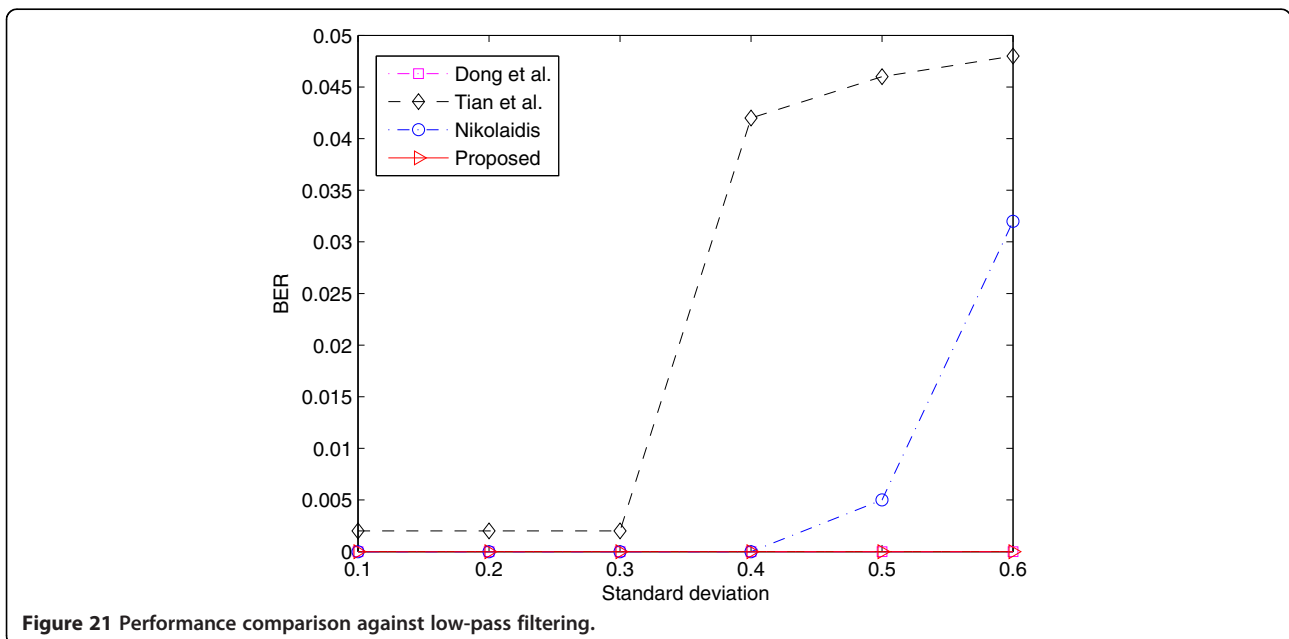
(0.5, 1.5), (1.5, 0.5), (0.7, 1.3), and (1.3, 0.7), outperforming the approaches of Dong et al., Tian et al., and Nikolaidis. Nevertheless, these three approaches outperform the proposed scheme for other cases with a high ratio of information loss.

It is shown from Figures 16 to 17 that the proposed scheme is vulnerable to the attacks of shearing and general affine transform. This is because that the proposed scheme is designed to counteract RST attacks rather than to handle affine transform. Also, it is found that the proposed scheme and the Tian et al.'s approach have

similar poor performance, and both of them are significantly worse than NIKO-SFNK and Dong et al.

3. Signal processing attacks: The signal processing attacks considered in [14] are JPEG compression, H.264 intra-frame compression, Gaussian noise addition, and low-pass filtering. The performance against these attacks is illustrated in Figures 18, 19, 20, and 21, respectively.

It is found from Figure 18 that the proposed scheme has better performance than the compared schemes for JPEG QFs from 20 to 50 but is weaker for other cases. Because situations with $QF < 20$ seldom occur in practice, the



proposed scheme is more favorable than the three compared approaches in counteracting JPEG compression.

Figure 19 presents the performance comparison against H.264 intra-frame compression. It can be observed that the proposed scheme has similarly high robustness as the compared three approaches for quality factor values below 25, but it has significantly better performance for other cases.

As shown in Figures 20 and 21, the performance against Gaussian noise addition and low-pass filtering is somewhat similar. According to Figure 20, the proposed scheme has significant superiority over the schemes of Tian et al. and Nikolaidis. It is similar to the scheme of Dong et al. for noise variances from 0.001 to 0.005 but is slightly better for a noise variance of 0.006. The low-pass filtering results shown in Figure 21 demonstrate that the robustness of the proposed scheme is higher than that of the schemes presented by Tian et al. and Nikolaidis, while it is equivalent to that of Dong et al.'s scheme.

7. Conclusions

In this paper, we have presented a DPT-based robust image watermarking scheme resilient to rotation, scaling, and translation. We first constructed a DPT with shift-invariance, steerability, and scalability by extending an SPT represented in a closed and polar-separable form. The radial component of the SPT's basis filters was taken as the kernel for designing the scalable basis filters. These were further combined with the steerable basis filters corresponding to the angular components of the SPT's basis filters, resulting in joint scalability and steerability. The shift-invariance was inherited from the SPT by retaining undecimated high-pass and band-pass basis subbands. We also derived interpolation functions for steerability and scalability. These allow the interpolation of any filter (response) at an arbitrary orientation and scale via a linear combination of the DPT's basis filters (responses). By exploiting the characteristics of shift-invariance, steerability, and scalability, we further derived the theoretical synchronization mechanisms for translation, rotation, and scaling.

Based on the constructed DPT with preferable characteristics, we developed a robust image watermarking scheme that is resilient to translation, rotation, and scaling. The translation invariance is achieved by taking the Fourier magnitude of the cover image as the DPT input. The resilience to rotation and scaling is obtained via the synchronization mechanisms for rotation and scaling. At the transmitter, the template and watermark are inserted in the first level of the DPT pyramid and the other two levels, respectively. At the receiver, the rotation angle and scaling factor are estimated via an efficient template-matching algorithm, and these are further used to correct the rotation and scaling attacks on the

received image followed by watermark extraction from the corrected image. Extensive simulations show that the proposed scheme is highly robust to geometrical attacks, such as rotation, scaling, translation, cropping, and row/column line removal, as well as common signal processing attacks such as JPEG compression, AWGN, median filtering, and convolution filtering. In addition, the comparison to some excellent related schemes demonstrated that the proposed scheme has a comparable performance against rotation, scaling, translation, cropping, and row/column line removal attacks, whereas it generally achieves a higher robustness to JPEG compression, AWGN, and low-pass filtering.

Competing interests

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by NSFC (nos. 61202467 and 61100170), the National Research Foundation for the Doctoral Program of Higher Education of China (no. 20120171110037), the Key Program of Natural Science Foundation of Guangdong (no. S2012020011114), and the Scientific Research Foundation for Returned Overseas Chinese Scholars (State Education Ministry).

Received: 18 March 2013 Accepted: 25 May 2013

Published: 24 June 2013

References

1. D Zheng, Y Liu, J Zhao, AE Saddik, A survey of RST invariant image watermarking algorithms. *ACM Comput. Surv.* **39**(2, Article 5), 1–91 (2007)
2. A Kumar, V Santhi, A review on geometric invariant digital image water-marking techniques. *Int. J. Comp. Appl.* **12**(9), 31–36 (2011)
3. P Bas, JM Chassery, B Macq, Geometrically invariant watermarking using feature points. *IEEE Trans. Image Processing* **11**(9), 1014–1028 (2002)
4. X Wang, C Wang, Y Yang, P Niu, A robust blind color image watermarking in quaternion Fourier transform domain. *J. Syst. Softw.* **86**, 255–277 (2013)
5. Y Wang, JF Doherty, RE Van Dyck, A rotation, scaling and translation resilient image watermarking algorithm using circular Gaussian filters, in *Proc. of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing* (Baltimore, MD, 2001)
6. J Lichtenauer, I Setyawana, T Kalker, R Lagendijka, Exhaustive geometrical search and the false positive watermark detection probability, in *Proc. of the SPIE-Security and Watermarking of Multimedia Contents V*, vol. 5020 (Santa Clara, CA, 2003), pp. 203–214
7. M Barni, Effectiveness of exhaustive search and template matching against watermark desynchronization. *IEEE Trans. Signal Proc. Letters* **12**(2), 158–161 (2005)
8. JJK O'Ruanidh, T Pun, Rotation, scale and translation invariant spread spectrum digital image watermarking. *Signal Process.* **66**(3), 303–317 (1998)
9. HS Kim, H-K Lee, Invariant image watermark using Zernike moments. *IEEE Trans. Circuit Syst. Video Technol.* **13**(8), 766–775 (2003)
10. C-W Tang, H-M Hang, A featured-based robust digital image watermarking scheme. *IEEE Trans. Signal Processing* **51**(4), 1123–1129 (2003)
11. M Teague, Image analysis via the general theory of moment. *J. Opt. Soc. Am.* **70**(8), 920–930 (1980)
12. H Zhang, H Shu, G Coatrieux, J Zhu, J Wu et al., Affine Legendre moment invariants for image watermarking robust to geometrical distortions. *IEEE Trans. Image Process.* **PP**(99), 1055–1068 (2011)
13. S Xiang, H-J Kim, J Huang, Invariant image watermarking based on statistical features in the low-frequency domain. *IEEE Trans. Circuit Syst. Video Technol.* **18**(6), 777–790 (2008)
14. A Nikolaidis, Local distortion resistant image watermarking relying on salient feature extraction. *EURASIP J. Adv. Signal Processing* **2012**, 97 (2012)
15. M Kutter, Watermarking resistance to translation, rotation, and scaling, in *Proc. of the International Society for Optical Engineering (SPIE): Multimedia Systems Applications*, vol. 3528 (Boston, MA, 1998), p. 423

16. S Voloshynovskiy, F Deguillaume, T Pun, Content adaptive watermarking based on a stochastic multiresolution image modeling, in *Proc. of the 10th European Signal Processing Conference (EUSIPCO2000)* (Tampere, Finland, 2000), pp. 5–8
17. S Voloshynovskiy, F Deguillaume, T Pun, Multibit digital watermarking robust against local nonlinear geometrical distortions, in *Proc. of the International Conference on Image Processing*, vol. 3 (Thessaloniki, Greece, 2001), p. 999
18. Z Zheng, S Wang, J Zhao, RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes. *IEEE Trans. Image Process.* **18**(5), 1055–1068 (2009)
19. J Tsai, W Huang, Y Kuo, On the Selection of optimal feature region set for robust digital image watermarking. *IEEE Trans. Image Process.* **20**(3), 735–743 (2011)
20. J-S Tsai, W-B Huang, Y-H Kuo, M-F Horng, Joint robustness and security enhancement for feature-based image watermarking using invariant feature regions. *Signal Process.* **92**, 1431–1445 (2012)
21. S Pereira, T Pun, Robust template matching for affine resistant image watermarks. *IEEE Trans. Image Processing* **9**(6), 1123–1129 (2000)
22. X Kang, J Huang, YQ Shi, Y Lin, A DWT-DFT composite watermarking scheme robust to both affine transform and JPEG compression. *IEEE Trans. Circuit Syst Video Technol* **13**(8), 776–786 (2003)
23. J Ni, C Wang, J Huang, A RST-Invariant robust DWT-HMM watermarking algorithm incorporating Zernike moments and template, in *KES2005: Knowledge-Based Intelligent Information & Engineering Systems*, ed. by R Khosla, R Howlett, LC Jain, vol. 3681 (Springer, Heidelberg, 2005), pp. 1233–1239
24. J Ni, R Zhang, J Huang, C Wang, Q Li, A rotation-Invariant secure image watermarking algorithm incorporating steerable pyramid transform, in *IWDW2006: Digital Watermarking*, ed. by YQ Shi, B Jeon, vol. 4283, 5th edn. (Springer, Heidelberg, 2006), pp. 446–460. *Int'l Workshop on Digital Watermarking, Lecture Notes Computer Science*
25. D Bogumi, An asymmetric image watermarking scheme resistant against geometrical distortions. *Signal Processing: Image Communication* **21**, 59–66 (2006)
26. YG Fu, R Shen, H Lu, Watermarking scheme based on support vector machine for color images. *IEE Electronics Letters* **40**(16), 986–7 (2004)
27. HH Tsai, DW Sun, Color image watermark extraction based on support vector machines. *Inf. Sci.* **177**(2), 550–69 (2007)
28. H Peng, J Wang, W Wang, Image watermarking method in multi-wavelet domain based on support vector machines. *J. Syst. Softw.* **83**(8), 1470–7 (2010)
29. EP Simoncelli, WT Freeman, EH Adelson, DJ Heeger, Shiftable multi-scale transform. *IEEE Trans. Information Theory* **38**(2), 587–607 (1992)
30. WT Freeman, EH Adelson, The design and use of steerable filters. *IEEE Trans. PAMI* **13**(9), 891–906 (1991)
31. P Perona, Deformable kernels for early vision, in *Proc. of the third Int. Conf. on Computer Vision and Pattern Recognition (CVPR)* (Lahaina, Maui, 1991), pp. 222–227
32. P Perona, Deformable kernels for early vision. *IEEE Trans. Pattern Anal. Machine Intelligence* **17**(5), 488–499 (1995)
33. A Karasiris, EP Simoncelli, A filter design technique for steerable pyramid transform, in *Proc. of the 21th International Conference on Acoustics, Speech, and Signal Processing*, vol. 4 (Atlanta, GA, 1996), pp. 2387–2390
34. J Portilla, V Strela, MJ Wainwright, EP Simoncelli, Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Processing* **12**(11), 1338–1351 (2003)
35. D Divsalar, H Jin, RJ McEliece, Coding theorems for turbo-like codes, in *Proc. of the 36th Annual Allerton Conf. on Communication, Control and Comp* (Monticello, IL, 1998), pp. 525–539
36. FAP Petitcolas, Watermarking schemes evaluation. *IEEE Trans. Signal Processing* **17**(5), 58–64 (2000)
37. FAP Petitcolas, M Stir, *IOP Publishing PhysicsWeb*, 2012, 2013. <http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/>. Accessed 16 Feb. 2013
38. C Wang, J Ni, H Zhuo, J Huang, A geometrically resilient robust image watermarking scheme using deformable multi-scale transform, in *Proc. of the Intl. Conf. on Image Processing 2010* (Hong Kong, 2010), pp. 3677–3680
39. P Dong, JG Brankov, NP Galatsanos, Y Yang, F Davoine, Digital Watermarking Robust to Geometric Distortions. *IEEE Trans. Image Process.* **14**(12), 2140–2150 (2005)
40. H Tian, Y Zhao, R Ni, J-S Pan, Spread spectrum-based image watermarking resistant to rotation and scaling using radon transform, in *Proc of the Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2010)* (Darmstadt, 2010), pp. 442–445

doi:10.1186/1687-6180-2013-119

Cite this article as: Wang et al.: Counteracting geometrical attacks on robust image watermarking by constructing a deformable pyramid transform. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:119.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com
