CrossMark

A robust image watermarking scheme in DCT domain based on adaptive texture direction quantization

Han Fang[1] · Hang Zhou[1] · Zehua Ma[1] · Weiming Zhang[1] (ID) **· Nenghai Yu[1]**

© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

This paper proposes a novel robust image watermarking scheme in Discrete Cosine Transform (DCT) domain for active forensics. We investigate the relation between the positions and the modification magnitudes of DCT coefficients and directions of texture blocks. By exploring such relation, a direction-coefficient mapping is designed. First, the texture direction of each image block is estimated by Gabor filter. And then, according to the direction-coefficient mapping, one watermark bit is embedded into each block along its texture direction. Compared with existing schemes, the proposed method utilizes the direction features of texture blocks better . Therefore, the improvements in watermarked image quality and the robustness of watermark signal against image processing attacks are both achieved.

Keywords Active forensics · Robust watermarking · Gabor filter · Texture direction · Direction-coefficient mapping

1 Introduction

Digital forensics is a branch of forensic science encompassing the recovery and investigation of material found in digital devices, often in relation to computer crime. In recent years,

---

✉ Weiming Zhang
   zhangwm@ustc.edu.cn

   Han Fang
   fanghan@mail.ustc.edu.cn

   Hang Zhou
   zh2991@mail.ustc.edu.cn

   Zehua Ma
   mzh045@mail.ustc.edu.cn

   Nenghai Yu
   ynh@ustc.edu.cn

[1] CAS Key Laboratory of Electromagnetic Space Information University of Science and Technology
    of China, Hefei, China

based on needs of cloud computing and various privacy-preserving applications, the issue of forensics in big data becomes vital and has gained increasing attention [22]. Two classes of digital forensics have been developed recently, including active forensics and passive forensics [11]. Hereinto, image watermarking is a branch of active forensics, by which the data embedded in images can withstand various attacks while maintain the visual quality of the image [14]. Robustness and transparency are the two main assessments to the watermarking scheme [2, 16]. Usually, the watermark embedded in spatial domain shows worse robustness than embedded in transform domain. So in order to achieve better robustness, transform domains such as Discrete Wavelet Transform (DWT) domain [5, 10, 19], Discrete Cosine Transform (DCT) domain [1, 7, 17] and Discrete Fourier Transform (DFT) domain [12, 15, 21] are commonly used to embed watermark.

However, usually high robustness is accompanied by large modifications which will seriously reduce the image quality [23]. In order to improve the image quality, the common methods embed watermark into the complex texture areas. Cui et al. [4] generated just perceptual weighting (JPW) standard in DWT domain to define the complexity of the texture. Hsu et al. [8] developed a watermarking algorithm based on backward-propagation neural network and DCT. Before embedding, they used just-noticeable difference (JND) model to define the embedding strength for different texture features. Based on [8], Hsu et al. [9] developed another watermarking scheme ensuring the high robustness while maintaining the high invisibility.

Not only texture complexity, but also texture directionality is important in improving visual quality [13]. The peak signal to noise ratio (PSNR) values of two compared blocks in Fig. 1 are almost the same, but the visual quality of the right image is much higher than that of the left image. So embedding the watermark along the direction of the texture can greatly improve the visual quality. Recently, many schemes for adaptively embedding along the texture direction have been proposed. Lou et al. [13] proposed a multi-bit watermarking scheme based on human visual system (HVS) model in DCT domain. They used HVS model to determine the embedding direction and the embedding position. However, they just considered three orientations. This cannot satisfy the multi-direction situation in natural image. Hamid et al. [6] proposed to embed watermark according to the orientation of the block which is indicated by the most significant DCT coefficients. However, they just roughly divided the direction into five categories. Zhao et al. [24] designed a texture directionality-based digital watermarking method in non-subsample shearlet transform (NSST) domain. The singular values (SVs) of the sub-image with the most significant direction are chosen

Fig. 1 Left image: modifications along the direction; right image: modifications not-along the direction [13]

to embed watermark. However, the original SVs are needed at the receiver side, so it is not a blind extracting scheme.

In general, although the previous methods utilized the direction feature of the texture block, they only carried out a roughly angular division which can not be adapted to the features of the image itself. So how to sufficiently explore the textural directions to improve the performance of watermarking scheme is an interesting problem. In this paper, we investigate the relation between the positions associated with modification magnitudes of DCT coefficients and the texture directions of image blocks. By exploring such relation, a direction-coefficient mapping is designed, based on which we propose an adaptive embedding method. Compared with existing schemes, the proposed method can better utilize texture features, which guarantees the robustness and the transparency of the watermark at the same time.

The rest of the paper is organized as follows. The proposed embedding and extracting algorithms are elaborated in Section 2. Experimental results are given in Section 3 to show the advantages of the proposed method, and finally this paper is concluded with a discussion in Section 4.

2 Proposed method

2.1 The embedding algorithm

We will embed the watermark by modifying the DCT coefficients of the image block. To adaptively modifying DCT coefficients, we define the direction-coefficient mapping which describes the appropriate positions and modification magnitudes of DCT coefficients with respect to the texture direction of an image block. For a cover image, it is first divided into non-overlapped blocks of size 16×16, and then a set of Gabor filters with different angles is used to estimate the texture direction of each block. One watermark bit will be embedded into each block by modifying the DCT coefficients according to the direction-coefficient mapping. The embedding procedure is shown in Fig. 2.

2.1.1 Direction-coefficient mapping

As mentioned above, the direction-coefficient mapping that reflects the relationship between direction feature and DCT coefficients is an essential part in our algorithm, based on which the modifications of corresponding DCT coefficients will preserve the feature of the texture direction. Since the blocks with the same direction may correspond to different texture widths, in order to achieve adaptive embedding, we need to fit both the width of the texture and the direction of the texture.
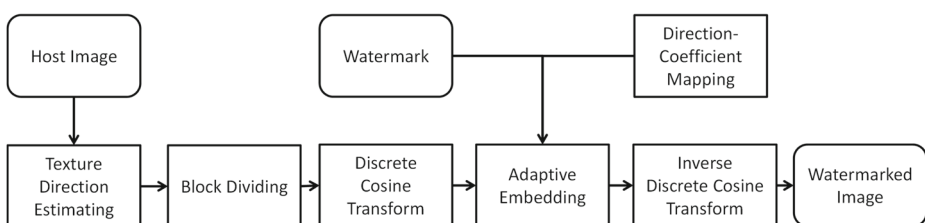

Fig. 2 The framework of the embedding processes

(a) The templates with 6 directions.　　(b) The image constructed by the chosen D-
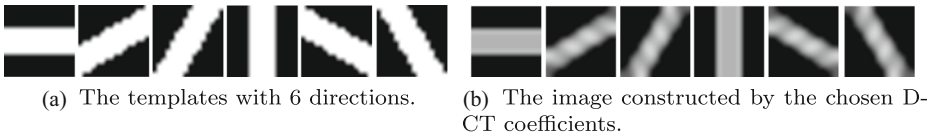　　　　　　　　　　　　　　　　　　　　CT coefficients.

Fig. 3　The template images

To determine the relationship between the DCT coefficients and texture direction, we generate binary template images of size 16×16 with n different texture directions, and then execute DCT to these templates. For each template, the positions of k coefficients with the biggest absolute values are recorded. We think that the coefficients in these k positions have main contribution to exhibit the present texture direction.

In order to reduce the error rate of direction judgement, the angle interval of two adjacent templates should not be too small. On the other hand, to meet the demands of self-adaptability, the angle interval should not be too big. So in this paper, the number of templates n is set to 6, the angle is set to θ ($\theta \in [0°, 30°, 60°, 90°, 120°, 150°]$), and the 6 templates are shown in Fig. 3a. Taking $k = 4$ as an example, we reconstruct the templates by only keeping the k main coefficients. The reconstructed images in Fig. 3b show that the main coefficients we selected for each template can indeed determine the corresponding texture direction. It's obvious that modifications on different DCT coefficients will produce different texture features in the spatial domain. So we need to determine the corresponding DCT coefficients for different texture directions including the position and the ratio of these coefficients. After performing DCT to the n templates, we record the biggest k coefficients with absolute values for a certain template. Modifying along the corresponding position will produce the same texture direction feature just the same as the template.

As mentioned above, the same direction may have different texture widths. To estimate the width of the texture, we generated 3 template images with the line width of 2, 4 and 8 where the size of the template is 16×16. Then we perform DCT on these templates. For each template, the positions of k coefficients with the biggest absolute values are recorded. Figure 4 shows the DCT coefficient matrix corresponding to different width template images. The brighter part of DCT matrix in Fig. 4 corresponds to a larger value of magnitude. It can be seen that the templates with the same angle but different width corresponding to different k main coefficients' positions. So we can judge the width by calculating the sum of the of k coefficients' absolute values at the corresponding positions. For a given angle θ, assume the k DCT coefficients of the image block to be embedded
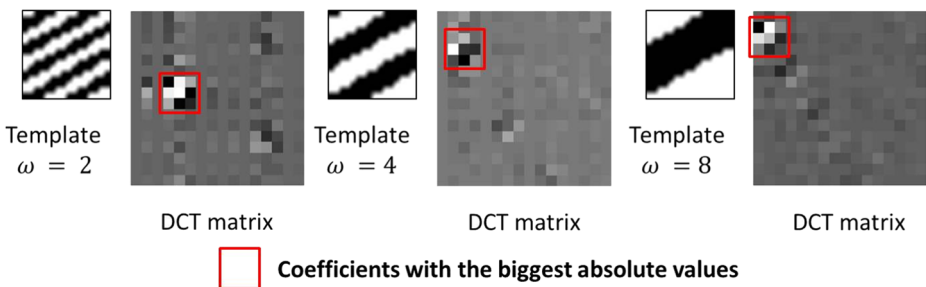

Template　　　　　　　　　　Template　　　　　　　　　　Template
$\omega = 2$　　　　　　　　　　$\omega = 4$　　　　　　　　　　$\omega = 8$

　　　　DCT matrix　　　　　　　　　DCT matrix　　　　　　　　　DCT matrix

　　　☐　**Coefficients with the biggest absolute values**

Fig. 4　The templates and corresponding DCT matrixes with different widths

Table 1 The positions and ratio of the template block

θ	Index i	$\omega = 2$				$\omega = 4$				$\omega = 8$			
		1	2	3	4	1	2	3	4	1	2	3	4
0°	Positions	(4,1)	(6,1)	(8,1)	(10,1)	(2,1)	(4,1)	(6,1)	(12,1)	(3,1)	(7,1)	(11,1)	(15,1)
	R_i^θ	−0.29	−0.40	−1	0.82	0.51	1	−0.61	0.33	1	−0.35	0.23	−0.19
30°	Positions	(8,4)	(7,5)	(8,5)	(8,6)	(4,2)	(4,3)	(5,3)	(4,4)	(2,1)	(3,1)	(2,2)	(3,2)
	R_i^θ	0.90	0.93	1	−0.64	1	−0.48	−0.89	−0.56	−0.59	0.80	1	0.57
60°	Positions	(5,7)	(4,8)	(5,8)	(6,8)	(2,4)	(3,4)	(4,4)	(3,5)	(1,2)	(2,2)	(1,3)	(2,3)
	R_i^θ	0.93	0.90	1	−0.64	1	−0.48	−0.56	−0.89	−0.59	1	0.80	0.57
90°	Positions	(1,4)	(1,6)	(1,8)	(1,10)	(1,2)	(1,4)	(1,6)	(1,12)	(1,3)	(1,7)	(1,11)	(1,15)
	R_i^θ	−0.29	−0.40	−1	0.82	0.50	1	−0.61	0.33	1	−0.35	0.23	−0.19
120°	Positions	(5,7)	(4,8)	(5,8)	(6,8)	(2,4)	(3,4)	(4,4)	(3,5)	(1,2)	(2,2)	(1,3)	(2,3)
	R_i^θ	0.93	−0.90	1	0.64	1	0.48	−0.56	0.89	1	−0.70	0.61	0.95
150°	Positions	(8,4)	(7,5)	(8,5)	(8,6)	(4,2)	(4,3)	(5,3)	(4,4)	(2,1)	(3,1)	(2,2)	(3,2)
	R_i^θ	0.90	−0.93	−1	−0.64	−1	−0.48	−0.89	0.56	−1	−0.61	0.70	−0.95

corresponding to width ω are $\{C_1^\omega, C_2^\omega \ldots, C_k^\omega\}$. So the texture width can be determined by

$$\omega = \underset{\omega \in [2,4,8]}{\arg\max} \sum_{i=1}^{k} |C_i^\omega| \tag{1}$$

Besides the positions of the k main coefficients, we should get to know the proper modification strength on the DCT coefficient in each position as well, for which we define relative modification magnitude (RMM). For a given angle θ and texture width ω, we assume the k main DCT coefficients in the corresponding template images are $\{D_1^\omega, D_2^\omega, \ldots, D_k^\omega\}$. Let $D_{max} = \max\{|D_1^\omega|, |D_2^\omega|, \ldots, |D_k^\omega|\}$, and we denote the RMM in the ith position for angel θ by R_i^θ which is defined as

$$R_i^\theta = \frac{D_i^\omega}{D_{max}}, 1 \leq i \leq k \ . \tag{2}$$

Note that RMM may be negative. Taking $\theta = 30°$, $\omega = 4$ and $k = 4$ as example, we use the second template image in Fig. 4 to get the 4 main coefficients at positions $\{(4, 2), (4, 3), (5, 3), (4, 4)\}$ with values equal to $\{4.07, -1.96, -3.63, -2.29\}$. With (2), the corresponding values of RMM are $\{1, -0.48, -0.89, -0.56\}$. The direction-coefficient mapping consists of the positions of main DCT coefficients and the corresponding RMM values. In Table 1, we list the direction-coefficient mapping for 6 angles when taking $k = 4$.

2.1.2 Texture direction analysis

We use Gabor filter to obtain the texture direction feature of a certain image block. One 2-D Gabor filter is illustrated as

$$g(x, y; \sigma_x, \sigma_y, \omega, \theta) = \frac{1}{2\pi \sigma_x^2 \sigma_y^2} e^{-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_y^2}\right)} \cos(\omega x'), \tag{3}$$

where $x' = x \cos\theta + y \sin\theta$ and $y' = -x \sin\theta + y \cos\theta$. θ indicates the rotate angle, x and y are horizontal and the vertical index of the pixel, σ_x and σ_y are their corresponding standard deviations of Gaussian kernel. The angle intervals for filter are set the same as the direction-coefficient mapping. So we use 6 Gabor filters with the angle of $\theta \in [0°, 30°, 60°, 90°, 120°, 150°]$ to filter the image and then divide them into blocks of size 16×16. Let B denotes a certain block and B_θ denotes the Gabor filtered image block with angle θ. As shown in Fig. 5, when the filter angle matches with the texture direc-

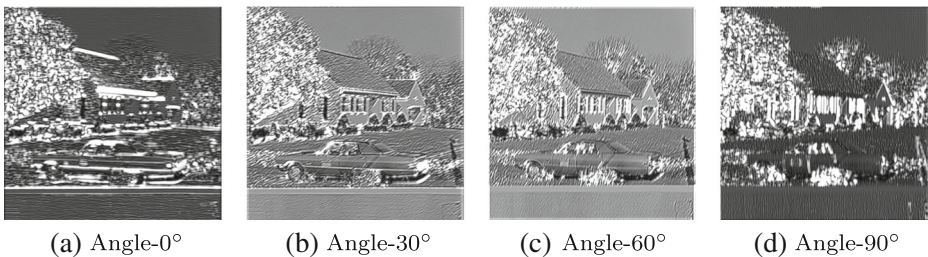| (a) Angle-0° | (b) Angle-30° | (c) Angle-60° | (d) Angle-90° |

Fig. 5 The image filtered by Gabor filter with different angles

tion, the corresponding region in the filtered image will yield high intensity. Therefore, we estimate the texture direction of the image block B with the following equation.

$$\theta = \arg\max_{\theta} \sum_{x, y \in [1,16]} B_\theta(x, y), \tag{4}$$

where $B_\theta(x, y)$ is the pixel intensity at position (x, y) in the image block B_θ.

2.1.3 Adaptively embedding

Firstly we should transform the watermark into one dimensional bit sequence which is to be embedded into the image blocks. Note that each image block only accommodates one-bit message. To embed watermark bit w into the certain block B, we first perform DCT on the block B and estimate the texture direction with (4), which is denoted as θ. According to θ, query the k positions and RMM values in the direction-coefficient mapping table such as Table 1. Denote the i DCT coefficients as D_i with the corresponding RMM values as R_i^θ for $1 \leq i \leq k$. We embed 1 bit watermark w into B by modifying the k DCT coefficients in the following manner [3].

$$\begin{cases} \Delta_i = | \Delta \times R_i^\theta | \\ f_0(D_i) = round(\frac{D_i + \frac{\Delta_i}{4}}{\Delta_i}) \times \Delta_i - \frac{\Delta_i}{4} \\ f_1(D_i) = round(\frac{D_i - \frac{\Delta_i}{4}}{\Delta_i}) \times \Delta_i + \frac{\Delta_i}{4} \end{cases}, \tag{5}$$

$$D_i' = \begin{cases} f_0(D_i), if \ w = 0 \\ f_1(D_i), otherwise \end{cases}, \tag{6}$$

where Δ is the quantization step size. Larger Δ will result in strong robustness but poor image quality. After the coefficients are modified, IDCT is performed to generate the modified image block B'. We repeat the procedure until all the messages are embedded. The embedding processes is described as Algorithm 1.

Algorithm 1 Embedding Algorithm

Require: Image I, direction-coefficient mapping, watermark W, quantization step size Δ.
Ensure: Watermarked image I'.
  1: Generate the bit sequence to be embedded from watermark W.
  2: **for all** $\theta \in (0°, 30°, 60°, 90°, 120°, 150°)$ **do**
  3:     Perform Gabor filter with θ to the image I to obtain filtered image I_θ;
  4: **end for**
  5: Divide the image I into 16×16 non-overlapped blocks.
  6: Estimate the texture direction of each block B with (4).
  7: Estimate the texture width of each block B with (1).
  8: **for all** B **do**
  9:     Modify the DCT coefficients with (5, 6);
 10:     Generate the modified block B';
 11: **end for**
 12: Generate the watermarked image I' by collecting all the modified blocks.
 13: **return** Watermarked image I'.

2.2 Extracting algorithm

The extracting algorithm is an inverse procedure of the embedding algorithm. We first divide the watermarked image I' into 16×16 non-overlapped blocks and estimate the texture direction of each block. For each block B', do DCT and get the k main DCT coefficients and RMM values according to the estimated direction θ and direction-coefficient mapping table. And then we can extract one bit from the block B' with (5), (7) , (8) and (9).

$$
\begin{cases}
t_{0i} = (D_i' - f_0(D_i'))^2 \\
t_{1i} = (D_i' - f_1(D_i'))^2
\end{cases}
\tag{7}
$$

$$
w_i =
\begin{cases}
-1 \times R_i, \, if \, t_{0i} \geq t_{1i} \\
1 \times R_i, \, otherwise
\end{cases}
\tag{8}
$$

$$
w =
\begin{cases}
1, \, if \, \sum_{i=1}^{k} w_i \geq 0 \\
0, \, otherwise
\end{cases}
\tag{9}
$$

The extracting processes is described in Algorithm 2.

Algorithm 2 Extracting Algorithm

Require: Image I', direction-coefficient mapping, quantization step size Δ.
Ensure: Watermark W.
  1: **for all** $\theta \in (0°, 30°, 60°, 90°, 120°, 150°)$ **do**
  2:     Perform Gabor filter with θ to the image I' to obtain filtered image I'_θ;
  3: **end for**
  4: Divide the image I' into 16×16 non-overlapped blocks.
  5: Estimate the texture direction of each block B' with (4).
  6: Estimate the texture width of each block B' with (1).
  7: **for all** B' **do**
  8:     Extract one bit from B' with (5), (7), (8) and (9);
  9: **end for**
 10: Generate the watermark W from the extracted bit sequence.
 11: **return** Watermark W.

---

3 Results and analysis

3.1 Setting of the parameter in direction-coefficient mapping

In the direction-coefficient mapping, we record the k contributed DCT coefficients for each texture direction. It is worth noting that the number k will affect the transparency and robustness of the watermark. The increase of k is accompanied by an enhancement in robustness but a decrease in visual quality. To determine the optimal number, we perform a series of experiments with different k in the image adtabase [18] which consists of 96 images. When the embedding process is finished, the attack by combining Gaussian noise and JPEG compression is performed to the images. Then we extract the watermark from the attacked image to calculate the bit error rate (BER) value.
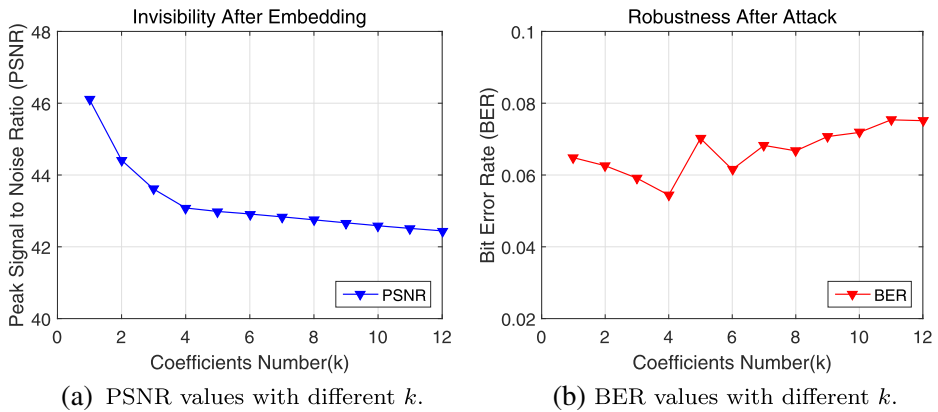
(a) PSNR values with different k.   (b) BER values with different k.

Fig. 6 The results in different settings

Figure 6a indicates the PSNR values for different k. With the increase of k, more coefficients of the original image are modified and thus the image quality will decrease. It is easy to see that PSNR does decrease with the increase of k, but slope of PSNR slows down when $k \geq 4$.

Figure 6b shows that the BER values do not increase continuously and a minimum value appears at $k = 4$. The reason can be summarized as that with increase of the number of contributed DCT coefficients, the quantization step size Δ_i's of these increased coefficients are getting smaller and the ability against attacks is even worse. So the increase of the coefficient number does not necessarily enhance the robustness.

Therefore, for both visual quality and robustness, $k = 4$ is the best choice.

3.2 Comparison with previous methods
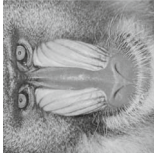
In this section, we show and discuss the experiments results. The proposed method is applied on a variety of images. A random binary message of length 1024 bits was embedded into each image. Specifically, we set the quantization step Δ as 55. Figure 7 shows the examples of original images and the corresponding watermarked images. We use PSNR and the structural similarity (SSIM) index [20] to measure the quality of the embedding image, the PSNR and SSIM values of 8 watermarked images are shown in Table 2. To measure the robustness of watermarking schemes, BER was computed as the assessment for comparison with other schemes under various of attacks.

We compare the proposed scheme with three state of the art watermarking schemes (Lou et al. [13], Hamid and Wang [6], Zhao et al. [24]). For fair comparison with other schemes,


Fig. 7 Upper image: original images; bottom image: watermarked images

Table 2 The PSNR and SSIM of watermarked images

| Watermarked image								
PSNR values	44.5951	44.4212	44.3554	44.5296	44.6787	44.8370	44.6709	44.6950
SSIM values	0.9967	0.9901	0.9886	0.9925	0.9910	0.9963	0.9923	0.9914

PSNR values of the embedded images are set to the same level of 44 ± 0.1dB. Five kinds of attacks, Gaussian noise with variance ranging from 0 to 0.003, JPEG compression with quality factor (QF) varies from 20 to 100, upper-left cropping with size of {1/16, 1/8, 1/4, 1/2}, scaling attack with proportion {0.5, 0.75, 1, 1.25, 1.5} and different kinds of filter attack include Gaussian filter, median filter, average filter and wiener filter are used to estimate the performance of the proposed scheme.
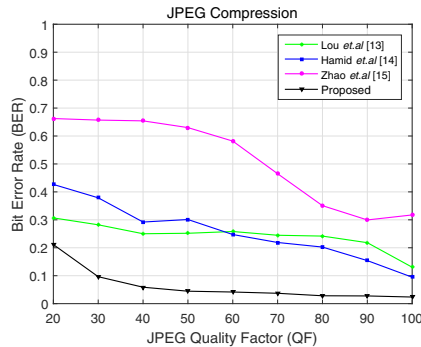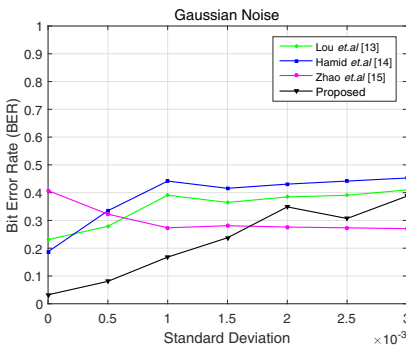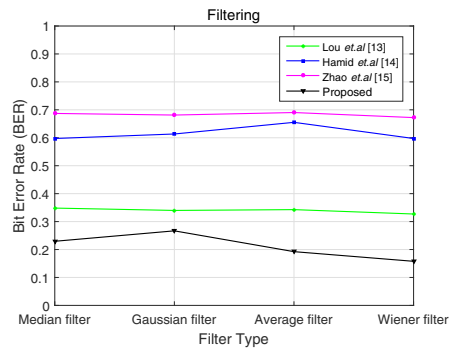

(a) JPEG Compression
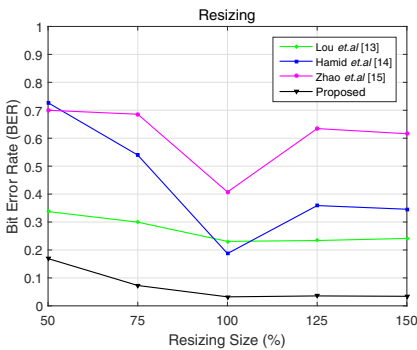
(b) Resizing
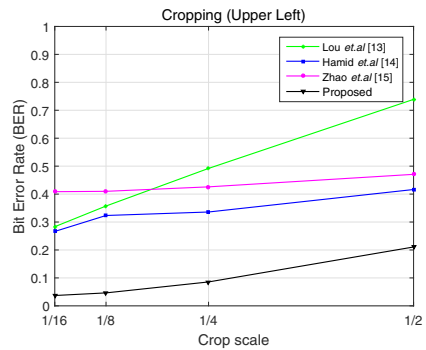
(c) Upper-left Cropping

(d) Gaussian Noise

(e) Filtering

Fig. 8 Robustness against different attacks

Figure 8a indicates the average BER values of different schemes. It's easy to see that our algorithm has good performance than other methods when quality factor (QF) is above 20. The BER value is at least 0.15 lower than the other algorithms. When the quality factor (QF) is beyond 50, the BER value of the extracted watermark is smaller than 0.05. So it is obviously that the influence of JPEG compression on the proposed method is very small.

Figure 8b shows the BER values of different schemes after rescaling operation. The rescaling size range from 0.5 to 1.5. The BER value of proposed scheme is about 0.1 lower than other algorithms. When the rescaling size is bigger than 0.75, the average BER value of proposed method is below 0.1. But the small size rescaling will affect a lot to our scheme.

The robustness in resisting upper-left cropping attack of different schemes is shown as Fig. 8c. The cropping size is doubled as a step increase. We can see that when the cropping size is small than 1/8, the proposed scheme is almost unaffected. But when the cropping size is bigger than 1/8, the extraction will be greatly affected.

Figure 8d indicates the ability to resist Gaussian noise attack of different schemes. We can see that the proposed scheme is robust to Gaussian noise attack. The BER value of proposed scheme is about 0.05 lower than other algorithms when the standard deviation is small than 0.15%. But when the standard deviation is bigger than 0.15%, the proposed scheme gets bad performance compared with other schemes. Because the Gaussian noise is heavily influence the texture feature of the block, so both the direction and the texture width are badly affected which result in the big BER values.

Figure 8e shows the BER values of different schemes with different filter attacks. We can see that although the proposed scheme is about 0.05 lower than other algorithms, the BER value is higher than 0.15 which means the proposed algorithm cannot be very effective when facing filter attack.

4 Conclusion and discussion

This paper proposes a new scheme of digital watermarking for active forensics which can make full use of the direction feature of the images. We design a direction-coefficient mapping which reflects the relations between direction features and DCT coefficients. Based on the mapping, the image blocks will keep the desired directional features when modifying the corresponding DCT coefficients. This is the key to ensure that the image still has a good visual quality after embedding the message. The experiments show that under the premise of good visual quality, the proposed algorithm performs better against many types of distortions than other schemes in most cases. It is noteworthy that the proposed algorithm is relatively weak in terms of resisting Gaussian noise attack. So our future work will be improving the robustness against noise attack.

Publisher's Note   Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Budimir L, Miloš D, Srdjan S, Irena O (2017) An algorithm for robust image watermarking based on the DCT and Zernike moments. Multimed Tools Appl 76(22):23333–23352
2. Cao Y, Zhou Z, Sun X, Gao C (2018) Coverless information hiding based on the molecular structure images of material. Computers Materials Continua 54(2):197–207
3. Chen B, Wornell GW (2001) Quantization index modulation: a class of provably good methods for digital watermarking and information embedding. IEEE Trans on Inf Theory 47(4):1423–1443
4. Cui L, Li W (2011) Adaptive multiwavelet-based watermarking through JPW masking. IEEE Trans Image Process 20(4):1047–1060
5. Falgun N, Vinay K (2017) A blind medical image watermarking: DWT-SVD based robust and secure approach for telemedicine applications. Multimed Tools Appl 76(3):3669–3697
6. Hamid M, Wang C (2016) A simple image-adaptive watermarking algorithm with blind extraction. In: IEEE IWSSIP 2016 - 2016 international conference on systems, signals and image processing, pp 1–4
7. Horng S (2013) A blind image copyright protection scheme for e-government. J Vis Commun Image 24(7):1099–1105
8. Hsu LY, Hu HT (2015) Blind image watermarking via exploitation of inter-block prediction and visibility threshold in DCT domain. J Vis Commun Image 32:130–143
9. Hsu LY, Hu HT (2017) Robust blind image watermarking using crisscross inter-block prediction in the DCT domain. J Vis Commun Image 46:33–47
10. Hu HT, Hsu LY (2017) Collective blind image watermarking in DWT-DCT domain with adaptive embedding strength governed by quality metrics. Multimed Tools Appl 76(5):6575–6594
11. Hu W, Chen W, Huang D, Yang C (2016) Effective image forgery detection of tampered foreground or background image based on image watermarking and alpha mattes. Multimed Tools Appl 75(6):3495—3516
12. Kang X, Huang J, Zeng W (2010) Efficient general Print-Scanning resilient data hiding based on uniform log-polar mapping. IEEE Trans Inf Forensics Security 5(1):1–12
13. Lou O, Li S, Liu Z, Tang S (2014) A novel multi-bit watermarking algorithm based on HVS. In: 2014 Sixth international symposium on parallel architectures, algorithms and programming, pp 278–281
14. Ma Y, Luo X, Li X, Bao Z, Zhang Y (2018) IEEE Trans Circuits Syst Video Technol. https://doi.org/10.1109/TCSVT.2018.2799243
15. Matthieu U, Dalila G, Florent A (2014) Perceptual DFT watermarking with improved detection and robustness to geometrical distortions. IEEE Trans Inf Forensics Security 9:1108–1119
16. Meng R, Rice SG, Jin W, Sun X (2018) A fusion steganographic algorithm based on faster R-CNN. Computers, Materials and Continua 55(1):1–16
17. Parah SA, Sheikh JA, Loan NA, Bhat GM (2016) Robust and blind watermarking technique in DCT domain using inter-block coefficient differencing. Dig Signal Process 53:11—24
18. Related images of the experiments, [Online]. Available:http://decsai.ugr.es/cvg/dbimagenes/g512.php.
19. Tayebe A, Mohsen EM (2016) A new visual cryptography based watermarking scheme using DWT and SIFT for multiple cover images. Multimed Tools Appl 75(14):8527–8543
20. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
21. Wang C, Kong X, Li C (2017) Process color watermarking: the use of visual masking and dot gain correction. Multimed Tools Appl 76(15):16291–16314
22. Wang J, Li T, Shi Y, Lian S, Ye J (2017) Forensics feature analysis in quaternion wavelet domain for distinguishing photographic images and computer graphics. Multimed Tools Appl 76(22):23721–23737
23. Zhang Y, Qin C, Zhang Y, Qin C, Zhang W, Liu F, Luo X (2018) On the fault-tolerant performance for a class of robust image steganography. Signal Process 146:99–111
24. Zhao J, Fan S, Jia J (2017) Texture directionality-based digital watermarking in nonsubsample shearlet domain. Math Probl Eng 2017:1–14

Han Fang received his B.S. degree in 2016 from Nanjing University of Aeronautics and Astronautics (NUAA). He is currently pursuing the Ph.D. degree in Information Security in University of Science and Technology of China (USTC). His research interests include image watermarking, information hiding and image processing.
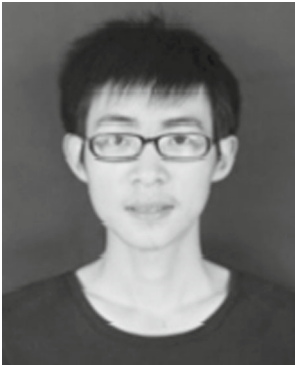
Hang Zhou received the B.S. degree in School of Communication and Information Engineering, Shanghai University, in 2015. He is currently pursuing the Ph.D. degree in Information Security in University of Science and Technology of China (USTC). His research interests include information hiding, image processing and computer graphics.

Zehua Ma is currently pursuing the B.S. degree in Information Security in University of Science and Technology of China (USTC). His research interests include image watermarking, information hiding and image processing.

Weiming Zhang received his M.S. degree and Ph.D. degree in 2002 and 2005 respectively from the Zhengzhou Information Science and Technology Institute, P.R. China. Currently, he is 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.

Nenghai Yu received his B.S. degree in 1987 from Nanjing University of Posts and Telecommunications, M.E. degree in 1992 from Tsinghua University and Ph.D. degree in 2004 from University of Science and Technology of China, where he is currently a professor. His research interests include multimedia security, multimedia information retrieval, video processing and information hiding.