# A NEW DCT-BASED DIGITAL IMAGE WATERMARKING ALGORITHM

**Hu Guan, Zhi Zeng, Shuwu Zhang**

Institute of Automation, Chinese Academy of Sciences, Beijing, China
E-mail:{hguan, zzeng, swzhang}@hitic.ia.ac.cn

**Keywords:** digital image watermarking, image scrambling, spread spectrum, Watson perceptual model.

## Abstract

In this paper, we propose a novel digital image watermarking algorithm, which can embed a multi-bit message into the image and realize the blind extraction of the message. To embed the watermark, the image is firstly scrambled to enhance the security. Then, some feature vectors based on the DCT of the scrambled image are extracted. The expanded watermark generated using the spread spectrum technique is embedded into the extracted feature vectors. The Watson perceptual model is also considered to obtain good invisibility. By means of corresponding inverse transformations we get the watermarked image. The watermark extraction process is similar as the embedding process. Unlike previous works, we use a new scheme to extract the image feature vectors and embed the watermark repeatedly into them. And the codebook in spread spectrum watermark generation is orthogonalized and unitized. These steps can dramatically improve the robustness of the watermark. Experiment results demonstrate the robustness of our algorithm against various common attacks such as JPEG compression, additive noise, histogram equalization, low-pass filtering and cropping.

## 1 Introduction

In recent years, with the development of the multimedia and internet technologies, there are a great deal of digital media contents transmitting on the internet and some handheld devices every day. To protect the copyrights of the digital contents, digital watermarks have become more and more important, also in infringing tracking and some other applications. Many research studies have been carried out on this topic, especially in digital images, audios, videos etc.

Most works in this area have been done based on the digital images, according to past research achievements, spread spectrum watermark in image spatial domain or frequency domain is a kind of significant method. In [2], the spread spectrum watermarking is proposed for the first time, the authors of [3] and [4] used this idea to design and realize new watermark algorithms, in their works, multi-bit message is expanded to another spread spectrum type, which is embedded into the image feature vector extracted from the image spatial domain, the main work they have done is to calculate the embedding strength in every position adaptively, meanwhile, eliminate the correlations from the vectors in the codebook which is used in spread spectrum process and the original image vector. The disadvantage of this operation is the large calculated amount to get the inverse matrix of a big intermediate matrix when calculate the embedding strengths, moreover, the inverse matrix may nonexistent sometimes, in the latter condition, the algorithm goes fail. Authors in [5] proposed a watermark embedding and detection method with side-information in the frequency domain, in its realization process, they used trellis coding method to get the expanded watermark aims to achieve better performance, but the number of vectors in codebook they used is larger than the length of each vector, they can't eliminate the correlations between different vectors directly, this is one of the main reasons in decreasing the effectiveness of the algorithm.

In practice, considering the disadvantages of some existing watermark algorithms and some practical applications of digital image watermarks, we should concentrate on the design of image watermarking algorithms which are resistant to many image distortions when the digital images are used or transmitted on the internet or different devices. For example, lossy compression, format conversion, noise jamming, filtering are very common to occur on the digital images.

Besides, in the research field of digital image watermarking algorithms, there are three characteristics of watermark we should pay attention to, that is watermark invisibility, robustness and security, which are often contradictory to each other. We must find a pretty good balance point for these properties. According to many previous works, during watermark embedding or extraction process, to enhance the security, researchers often set a private key to determine the embedding positions or generate random codebooks; to improve the watermark invisibility, researchers have proposed some perceptual masking models, including discrete cosine transformation (DCT) based Watson perceptual model and discrete wavelet transformation (DWT) based pixel-wise masking model, authors of [1] illustrated the former one outperforms the latter; to increase the watermark robustness, many researchers spent their time to explore the properties of different transformations and determine how to embed and extract the watermark.

We use some primary works for reference and present our own algorithm to embed and extract the watermark. Considering many kinds of common distortions, we select DCT domain to embed the watermark, and also use DCT-based Watson perceptual model and private keys to improve performance of our scheme. Our main work is the design of

embedding and extraction scheme, including some modified techniques in the details.

The rest of this paper is organized as follows: section 2 presents our novel DCT-based watermark embedding method, and in section 3 we show the corresponding watermark extraction method. Then, in section 4, we concentrate on the experiment results of our algorithm. We illustrate the watermark robustness, invisibility and security in the manner of contrasting the results with some contrastive algorithms using different methods or schemes. We draw our conclusion in section 5 and give some probable improvement measures.

## 2 DCT-based watermark embedding

In this section, we describe the watermark embedding scheme in detail and emphasize the novel ideas we propose. Fig. 1 illustrates the block diagram of the watermark embedding algorithm. Firstly we show how the watermark message is transformed to its spread spectrum form. Secondly we explain how to deal with the image and embed the spread spectrum watermark into it. We explore some significant properties of relevant techniques to guarantee the balance of watermark robustness, invisibility and security.

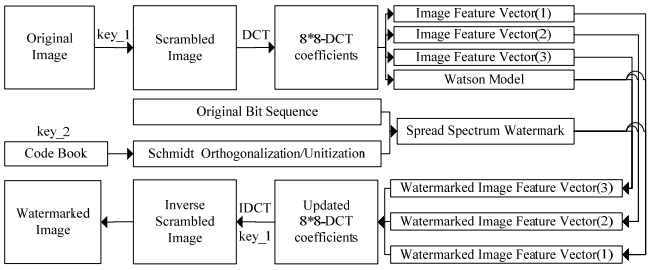


Fig. 1: Watermark embedding process

### 2.1 Spread spectrum watermark generation

Consider the watermark is a multi-bit message $W$ of length $N$.

$$W = [w_1, w_2, \cdots, w_N],\ w_i \in \{0,1\} \tag{1}$$

In order to enhance the robustness of the watermark, we try to transform the message into another form using spread spectrum technique.

As Fig. 1 shows, we first generate a random codebook using the specific private key (key_2 in Fig. 1), which has $N$ vectors of length $M$ for each vector ($M > N$). As mentioned above, $N$ is the length of the original watermark, $M$ can be calculated according to the size of the original image, we will give the computing method in section 4. And then, we do the Schmidt orthogonalization and unitization operation to the vectors in the codebook. In this way, the correlation between arbitrary pair of vectors in codebook, which is harmful for image extraction, is eliminated. Secondly we use spread spectrum technique mentioned in [3] and [4] to generate the expanded watermark $\overline{W}$ of length $M$ which will be embedded into the original image. The spread spectrum process can be showed as follows:

$$\overline{W} = \sum_{i=1}^{N} \beta_i P_i,\ \begin{cases} \beta_i = 1, & w_i = 1 \\ \beta_i = -1, & w_i = 0 \end{cases}, \tag{2}$$

where $P_i$ is the $i$-th vector in the codebook.

### 2.2 Watermark embedding method

After finishing the generation of spread spectrum watermark $\overline{W}$, we embed $\overline{W}$ into the DCT domain of the original image's luminance component. We first do the macroblock based scrambling to the original image using a private key (key_1 in Fig. 1), following by 8×8 DCT. Here, the using of private key_1 is to enhance the security as key_2 above. Secondly, we extract three image feature vectors which are all constituted by the AC coefficients in the specific positions, and for different vectors, we select the different positions, the length of the three vectors are all $M$, the same as the vectors' length in the codebook above. In section 4, we will illustrate the positions where the coefficients are selected to constitute the feature vectors. We also calculate the Watson visual perceptive parameters $V$ using the method detailed in [1], with the same size as original image, to adjust the watermark embedding strength adaptively in different positions in order for better invisibility. We embed the same watermark $\overline{W}$ into the three image feature vectors (distinguished by (1), (2), (3) in Fig. 1). Watermark embedding process can be showed as the following equation:

$$C' = C + \alpha \times V^* \times \overline{W} \tag{3}$$

where $V^*$ is the Watson perceptual model parameter value in the specific position where the AC coefficient is selected and $\alpha$ stands for the global watermark embedding strength, we can select a suitable value according to the balance of watermark invisibility and robustness. $C$ and $C'$ represent the original image feature vectors and the ones after watermark embedding respectively. Thirdly, we update the DCT coefficients using watermarked feature vectors obtained from three different embedding processes, following by the 8×8 IDCT and inverse image scrambling using the same key_1. We, then, get the watermarked image.

In this section, we embed the watermark in a new scheme different from other works. Different feature vectors are selected according to different positions, and the same watermark is embedded for three times. Besides, we bring the Schmidt orthogonalization and unitization operation into the spread spectrum watermark generation process to eliminate the correlations between the vectors in the codebook. These all can increase the watermark robustness even if some image distortions occurred on the watermarked images. By selecting a proper watermark embedding strength, we can get the best balance of watermark robustness and invisibility. We will discuss these deeply in section 4.

## 3 DCT-based watermark extraction

In this section, we describe the watermark extraction approach. Fig. 2 illustrates the block diagram of our proposed watermark extraction algorithm. Similar as watermark embedding above, we also need to generate the codebook first which has the same size and same data as in watermark embedding process (using the same key_2), following with Schmidt orthogonalization and unitization operation to the vectors in it. With the same way and same positions as before, we extract three different feature vectors from detected image

(This image may has gone through some kinds of distortions), which are indicated by $C\_1$ , $C\_2$, $C\_3$. The most significant process in the watermark extraction process is the correlation calculation to get the extracted watermark. We choose the statistical method to decide the extracted bit one by one. The correlation calculation process can be showed as follows:

$$\begin{aligned} N\_1_i &= NC(C\_1, P_i) \\ N\_2_i &= NC(C\_2, P_i) \\ N\_3_i &= NC(C\_3, P_i) \end{aligned}, \ i = 1, 2, \cdots, N, \tag{4}$$

$$\begin{cases} \overline{w}\_1_i = 1, N\_1_i \geq 0 \\ \overline{w}\_1_i = 0, N\_1_i < 0 \end{cases}, \begin{cases} \overline{w}\_2_i = 1, N\_2_i \geq 0 \\ \overline{w}\_2_i = 0, N\_2_i < 0 \end{cases}, \begin{cases} \overline{w}\_3_i = 1, N\_3_i \geq 0 \\ \overline{w}\_3_i = 0, N\_3_i < 0 \end{cases}, \tag{5}$$

where *NC* stands for the correlation of the image feature vector and vector $P_i$ in codebook. We get the final extracted watermark using the following equation:

$$\begin{cases} w'_i = 1, & \overline{w}\_1_i + \overline{w}\_2_i + \overline{w}\_3_i \geq 2 \\ w'_i = 0, & \overline{w}\_1_i + \overline{w}\_2_i + \overline{w}\_3_i \leq 1 \end{cases}, \ i = 1, 2, \cdots, N, \tag{6}$$

where $w'_i$ stands for the *i*-th extracted bit.

By the way, contrasting the extracted bit sequence with the original one, we can get the bit error rate (BER), by means of which we can measure the robustness of the scheme. We will discuss this deeply in section 4.

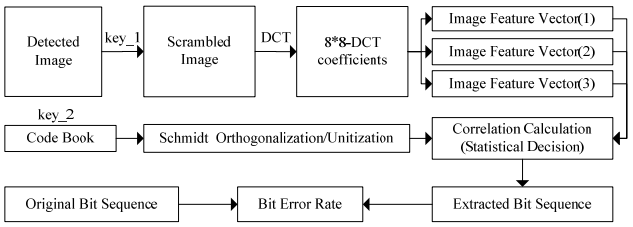


Fig. 2: Watermark extraction process

## 4 Experiment results

In order to signify copyright of digital media, such as images, audios, videos, many watermarking algorithms require multi-bit message length of at least 70 bits [3]. In this section, to test the method we propose in this paper, we embed a 0-1 bit sequence whose length is *N*=70 into the DCT domain of the Lena image which size is 512×512. And the size of macroblock we used for image scrambling is set to be 64×64.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 |
| 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 |

Fig. 3: 8×8DCT coefficients

In this experiment, according to the size of the original image, we calculate the length *M*, it is equal to the number of 8×8 blocks in the image. Because the height and width of the original image are both 512, *M* is equal to 512×512 divided by 8×8. Both the lengths of the image feature vectors and vectors in the codebook are *M*=4096. Using *N* and *M*, we can finish the spread spectrum watermark generation process. In watermark embedding step, we extract the three image feature vectors one by one. For the first vector, we extract the coefficient in row 1 column 3 (the position in Fig. 3 where number "3" is located) from every 8×8 block; for the second, we need the coefficient in row 2 column 2 (the position in Fig. 3 where number "10" is located); and for the third, we use the coefficient in row 3 column 1 (the position in Fig. 3 where number "17" is located). In every block, they are all low frequency coefficients, we can get good robustness selecting these coefficients. Fig. 4(a) shows the original image and Fig. 4(b) shows the watermarked image with global embedding strength $\alpha = 7.5$.

(a) (b)

Fig. 4: (a) original image and (b) watermarked image

No matter the watermarked image be used in any application fields, in practice, watermarked image will be subjected to some distortions before reaching the detector or extractor. To check the performance of the algorithm we propose, we realize some kinds of distortions to the watermarked image showed in Fig. 4(b), including JPEG compression with various quality factors, noise adding (Gaussian noise and Salt&Pepper noise), histogram equalization, Low-pass filtering with different window sizes and cropping with different ratios. Fig. 5 shows some distorted watermarked images with dissimilar image processing forms, including (a) JPEG compression with quality factor equal to 20; (b) Gaussian noise with variance equal to 0.005; (c) Salt&Pepper noise with variance equal to 0.04; (d) Histogram equalization; (e) Low-pass filtering with the window 5×5 and (f) Cropping with the ratio equal to 20%.

(a) (b) (c)

(d) (e) (f)

Fig. 5: Distorted watermarked images

To measure the performance of the proposed algorithm, we do the experiments in different conditions, as table 1 shows, we first realize our algorithm we discussed above (the results can be seen in table 1 line E.a(1)), and get contrastive results using a similar scheme with the embedding strength ("Alpha"

showed in table 1) equal to 13 to get the similar PSNR (Peak Signal to Noise Ratio, in fact, we adjust the parameter "Alpha" to get similar PSNR for all the experiments in our paper to reach a pretty good invisibility and the BERs are all 0.00% when no distortions occur). In the contrastive experiment, we just extract one feature vector from AC coefficients according to the same positions as our method (we use three coefficients per one 8×8 block, as Fig. 3 shows), the difference between the two methods is the latter one get a vector which is three times long as the former, and the corresponding codebook is larger, too. According to the results, we can see that the performances of the two methods in resisting attacks are little difference, but our method is much faster than the latter because of the codebook generation and processing, the codebook generated in the latter algorithm is three time larger than our method's, so the generation process and orthogonalization/unitization operation to the vectors in it will spend much more time. We realize the two algorithms using the same computer, in average, our embedding and extraction algorithm totally takes about 1.038s while the latter algorithm takes about 2.578s.

| Distortion Types | Parameters | E.a(1) Alpha: 7.5 PSNR: 37.98dB | E.b(1) Alpha: 13 PSNR: 37.68dB | E.a(2) Alpha: 0.205 PSNR: 37.95dB | E.b(2) Alpha: 0.205 PSNR: 37.55dB |
|---|---|---|---|---|---|
| No Distortion | / | 0.00% | 0.00% | 0.00% | 0.00% |
| JPEG Compression | 30% | 0.00% | 0.00% | 0.00% | 0.00% |
| | 20% | 0.00% | 0.00% | 0.00% | 0.00% |
| | 10% | 2.86% | 1.43% | 2.86% | 1.43% |
| Gaussian Noise | 0.005 | 0.00% | 0.00% | 0.14% | 0.00% |
| | 0.01 | 0.00% | 0.00% | 0.72% | 0.14% |
| Salt & Pepper Noise | 0.03 | 0.00% | 0.00% | 0.14% | 0.14% |
| | 0.04 | 0.00% | 0.00% | 0.42% | 0.14% |
| | 0.05 | 0.14% | 0.14% | 0.72% | 0.42% |
| Histogram Equalization | / | 0.00% | 0.00% | 0.00% | 0.00% |
| Meaning Filtering | 3*3 | 0.00% | 1.43% | 1.43% | 1.43% |
| | 5*5 | 8.57% | 8.57% | 7.14% | 7.14% |
| Cropping | 10% | 0.00% | 0.00% | 1.43% | 0.00% |
| | 20% | 0.00% | 0.00% | 1.43% | 0.00% |
| | 50% | 1.43% | 2.86% | 1.43% | 2.86% |

Table 1: Simulation results for various attacks

Results in Table 1 line E.a(2) and E.b(2) indicate another pair of results from the algorithms we propose and the contrastive one we mentioned above respectively, but both excluding the orthogonalization and unitization operation. By contrasting the results in line E.a(1) and E.a(2) we can conclude that orthogonalization and unitization operation plays a significant role in watermark embedding and extraction processes we propose. Also we know, in contrastive experiment, because the vectors in codebook are much longer, the correlation between each pair of vectors will decrease, that is why the results in line E.b(2) are generally better than in line E.a(2).

In addition, according to the experiments we do in this paper, we can see that, the method we propose reveals the best performance in contrast with others mentioned in our paper, not only the resistance to the image distortions and invisibility, but also the efficiency and security.

## 5 Conclusions

We present a novel approach for multi-bit watermark embedding and extraction. We extract several feature vectors from the image instead of only one, aim to enhance the extracted accuracy and embedding/extraction efficiency. We also bring Schmidt orthogonalization and unitization operation into the spread spectrum watermark generation process, aim to increase the watermark robustness. Results illustrate that the approach we propose has pretty good resistance to many kinds of attacks, and also shows its good properties in efficiency and security. According to the good performance, we can try to apply this approach to frame-based video watermarking algorithms directly in future works. The disadvantage of this method is that it doesn't resist to some kinds of geometric distortions, such as translation, flipping, rotation, scaling, we can using this approach with template-based watermarking algorithm[7], image-normalization-based watermarking algorithm [6] or feature-points-based watermarking algorithm [8] together to enhance its performance to such attacks.

## Acknowledgements

This work has been supported by the National Key Technology R&D Program of China under Grant No. 2012BAH04F02, 2011BAH16B01 and 2011BAH16B02.

## References

[1] G. Xie, M. N. S. Swamy, M. O. Ahmad. "Perceptual-shaping comparison of DWT-based pixel-wise masking model with DCT-based Watson model", *IEEE International Conference on Image Processing, ICIP*, vol. 2, pp. 1381-1384, (2006).

[2] I. J. Cox, J. Kilian, F. T. Leighton, T. Shamoon. "Secure spread spectrum watermarking for multimedia", *IEEE Transactions on Image Processing*, vol. 6, pp. 1673-1687, (1997).

[3] J. Mayer, J. C. M. Bermudez. "Multi-bit informed embedding watermarking with constant robustness", *IEEE International Conference on Image Processing, ICIP*, vol. 1, pp. 669-672, (2005).

[4] J. Mayer, R. A. Silva. "Efficient informed embedding of multi-bit watermark", *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol. 1, pp. 389-392, (2004).

[5] M. L. Miller, G. J. Doerr, I. J. Cox. "Dirty-paper trellis codes for watermarking", *IEEE International Conference on Image Processing, ICIP*, vol. 2, pp. 129-132, (2002).

[6] P. Dong, N. P. Galatsanos. "Affine transformation resistant watermarking based on image normalization", *IEEE International Conference on Image Processing, ICIP*, vol. 3, pp. 489-492, (2002).

[7] S. Pereira, T. Pun. "Robust template matching for affine resistant image watermarks", *IEEE Transactions on Image Processing*, vol. 9, pp. 1123-1129, (2000).

[8] T. Jen-Sheng, H. Win-Bin, K. Yau-Hwang. "On the Selection of Optimal Feature Region Set for Robust Digital Image Watermarking", *IEEE Transactions on Image Processing*, vol. 20, pp. 735-743, (2011).