A Robust Adaptive Image Watermarking Algorithm

Shuangyuan Yang Zhengding Lu Fuhao Zou
(College of Computer Science & Technology, Huazhong University of Science & Technology,
Wuhan 430074, China)
E-mail: ysydragon@sina.com

Abstract

As a main method for copyright protection, the watermarking techniques have been widely studied and applied. The proposed algorithm splits the original image into blocks and classifies these blocks based on visual masking. Watermarking components with different strength are inserted into some DCT coefficients according to the classification .The experimental results demonstrate our algorithm is robust against noise and commonly used image processing techniques.

1. Introduction

Although lots of progress has been made in watermarking research and application recently, geometric resistant watermarking remains to be one of the most difficult outstanding areas. A small distortion, such as rotation, scaling , translation, shearing, random bending or change of aspect ratio, can defeat most of the existing watermarking schemes claimed robust. For a strong watermark, There are two parts absolutely necessary to building: the watermark structure and the insertion strategy. Cox[1] proposed:

- The watermark should be placed explicitly in the perceptually most significant components of the data. The stipulation that the watermark be placed in the perceptually significant components means that an attacker must target the fundamental structural components of the data, thereby heightening the chances of fidelity degradation.
- The watermark should be composed of random numbers drawn from a Gaussian distribution.

Although lots of algorithms in the paper[2-6] are very robust to most attacks, They nine tenths need the original unwatermarked image(namely cover image) or the watermark image for the watermark extractor.

For solving this question, we propose an adaptive 2-dimension image watermarking algorithm in spatial domain with gray image's watermark based on the Human Visual System (HVS) by using Podilchuk[2] and Huang[3] for reference.

First, in order to embed the watermark image, we split the watermark image into 8×8 blocks and transform them into DCT domain, and quantize their DCT coefficients with JPEG standard quantization table and adjust them, and Then, take out the top ten lower frequency coefficients from each block to constitute the watermark signal because these lower frequency coefficients contain most of the information carried by image. Then we encrypt this watermark signal to form the final watermark signal by RSA encryption techniques with the private key.

Second, by using the human visual system, we split the cover image into 8×8 blocks and classify them as four classes according to the characteristic value of texture and illumination. Finally, According to the classification, the watermark components with different strength are embedded into the cover image.

2. The Embedding Algorithm

The Embedding algorithm is shown in Fig.1.In order to achieve a sufficiently robust and transparent scheme, we use the technique: RSA encryption, JPEG quantization, adaptive quantization and HVS .

The most variables are defined as:

- The cover image is named as I(size M×N)
- The watermark image is named as W(size R×S)
- The watermark signal after the JPEG quantization is named as Q
- The final watermark signal is named as **W'**
- The luminance plane of cover image is named as **Y**.
- The watermarked image is named as **I'**(size M×N)

If the cover image is the RGB image, we transform the cover image into the YUV image, and then choose the luminance plane Y to embed the watermark signal.

2.1 Watermark decoding

Because the size of the watermark image is very great, we need to compress the watermark image to reduce the size of the watermark signal. Considering that the DCT coefficients are the least affected by the JPEG quantization (in JPEG compression the quantization is the lossy part while the encoding is lossless) , we choose the standard JPEG compression to reduce the size of the watermark signal.

The watermark decoding procedure is described as:

1) Every pixel of the watermark image W subtracts 128,Then W is divided into 8×8 blocks. It is defincd as:

$$B_m = f_m(x,y), \mathrm{m}=0,1,\ldots,\lceil (R\times S)/(8\times 8)-1 \rceil$$

$$\mathbf{W} = \bigcup_{m=0}^{\lceil (R\times S)/(8\times 8)-1 \rceil} fm(x,y), 0 \le x,y \le 7 \qquad (1)$$

2) Transform every block of B_m into DCT domain

$$B_m^{'} = F(u,v) = DCT\{f_m(x,y), 0 \le x,y,u,v \le 7\} \quad (2)$$

3) Quantize the DCT coefficients of $B_m^{'}$ with JPEG standard quantization table

$$B_m^{"} = \lfloor (F(u,v)/Q(u,v)) \rfloor \{0 \le x,y,u,v \le 7\} \qquad (3)$$

4) Take out the top ten lower frequency coefficients from each $B_m^{"}$ block to constitute the watermark signal and the watermark signal is named as P_m ,namely:

$$P = \bigcup_{m=0}^{9} P_m = \begin{Bmatrix} d_{uv}, d_{uv} \in B_m^{"}, \\ u = 0,1,2,3 \quad v = 0,\cdots,3-u \end{Bmatrix} \qquad (4)$$

5) Reshape P_m into 1-dimension binary sequence according to Zigzag order and the binary sequence (size L) is named as Q:

$$\text{Q=}\{ P_0 P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8 P_9 \} \qquad (5)$$

6) Encrypt Q to form the final watermark signal W'(size L')by RSA encryption techniques with the private key K:

W'=RSA(Q,K)(L'<<R×S×8) (6)

2.2 Block Classification According To HVS

The watermark embedding can be regarded that a weak signal (watermark signal) is added upon the strong background. If this signal is less than the contrast sensitivity threshold, it is imperceptible to HVS[5]. According to the three characteristics of HVS:

- The various grayscale is differently sensitive to our eyesights, usually, medium grayscale has most sensitivity ad hoc, non- linear decrease to low and high grayscale.
- Sensitive to the smooth region, texture region reversly.
- The marginal information is important to our eyesights, so the quality of marginal should be ensured not to be heavily injured.

Based on the entropy and variance of per 8×8 block (if the cover is RGB image, we calculate the luminance plane Y), we segment image into four classes:

- The entropy of the first class is smaller, belonging to the smooth region (S1), which has low-luminance and simple texture, so modification of the pixel value is sensitive to HVS, and the strength of embedded watermarking should be smallest.
- The entropy of the second class is larger, and the variance is smaller, belonging to the marginal region (S2), although it has high-luminance and complex texture, hence, modification of the pixel value is sensitive to HVS, and the strength of embedded watermarking should be smaller.
- The entropy and variance of the second class are larger, whose texture is complex, and belonging to the non-marginal region(S4), so modification of the pixel value is most non-sensitive to HVS, and the strength of embedded watermarking can be largest.
- The rest belongs to the third class(S3).

Where the larger of the entropy should be above the first one third of the entropy sorted ascending, otherwise, the smaller is less than the latter half . At the same time, the boundary of the variance is the first one fifths and the latter half, respectively.

2.3 Watermark Embedding

The embedding procedure is described as:

1) Dividing the cover image I into 8×8 blocks

$$C_m = f_m(x,y), \mathrm{m}=0,1,\cdots,\lceil (M\times N)/(8\times 8)-1 \rceil \qquad (7)$$

2) Transforming every C_m block into DCT domain

$$C_m^{'} = F_m(u,v) = DCT\begin{Bmatrix} f_m(x,y), 0 \le x,y,u,v \le 7 \\ 0 \le m \le \lceil (M\times N)/(8\times 8)-1 \rceil \end{Bmatrix} \quad (8)$$

3) Making the adaptive quantization according to the HVS characteristic of every C_m block:

First, we choose the L'-1 entries of 8×8 blocks by some random way that can be repeated and calculate the entropy and the variance of every C_m block, and then determine the region (S1, S2, S3 or S4) that every C_m block lies in according to the entropy and the variance. To the different region , there is the different quantization magnitude (R1, R2, R3 or R4). R1, R2, R3 and R4 is confirmed by the experiment results, In this paper, R1 is equal to 6, R2 is equal to 10, R3 is equal to 18,R4 is equal to 28.

Second, We transform every C_m block into DCT domain, named as $C_m^{'}$. To every $C_m^{'}$ block, we make adaptive quantization according to its corresponding quantization magnitude. For example, some lower frequency coefficient of $C_m^{'}$ block is equal to 233 and it lies in S2, so the quantization magnitude is R2 , that is equal to 10.

4) Using the paper[4] for reference, we use three lower frequency band of every $C_m^{'}$ block for embedding.

Supposing:

The final watermark signal W' (size L') is defined as

$$W' = \{xi, 0 \le i \le L^{'} - 1\}$$

After the adaptive quantization, the value of every $C_m^{'}$ block is defined as $V_m^{'}$

Then:

The watermark embedding formula is:

$$F_m{'}(u,v) = \begin{cases} V'_m(u,v) \bullet 2\beta + \lambda \bullet \beta \bullet x_i \\ (u,v) \in \{(0,1),(1,0),(1,1)\}, 3m \le i \le 3(m+1) \\ F_m(u,v), \{(u,v) \in the-other-region\} \end{cases} \quad (9)$$

where β is the weighting factor, and the value of β is equal to the half of the quantization magnitude of the corresponding C_m block;

λ is the aspect factor, and the value of λ is equal to 1 or −1. When the remainder after adaptive quantization is less than β, λ is equal to 1, otherwise, λ is given −1.

For the former example, $F_m(0,1)$ is equal to 233 and it lies in S2, so the quantization magnitude is R2, that is equal to 10. So, β is 5.Because the remainder after quantization with R2 is 3 that is less than β, λ is equal to 1.But ,if $F_m(0,1)$ is equal to 237, the remainder after quantization with R2 is 7 that is not less than β, λ should be given −1. Under the condition, if λ is given to 1 as before, we may not make the watermarked image imperceptible to the HVS because the embedding signal probably exceeds the contrast sensitivity threshold.

To every C_m block, we can give the formula (10) based on the formula (9):

$$F_m^{'}(u,v) = (2 \bullet V_m^{'}(u,v) + \lambda \bullet x_i) \bullet \beta \quad (10)$$

From (10), we can draw the conclusion that when the embedded watermark signal is equal to 0, $F_m^{'}(u,v)$ is even multiples of β and when the embedded watermark signal is equal to 1, $F_m^{'}(u,v)$ is odd multiples of β.

5) Finally, to every $C_m^{'}$ block making the inverse DCT transformation and uniting to form the watermarked image $I^{'}$

$$I' = \bigcup_{m=0}^{\lceil (M \times N)/(8\times8) - 1 \rceil} DCT^{-1}(Fm^{'}(u,v)), \{0 \le u,v \le 7\} \quad (11)$$

3. Watermark Extracting

Because the proposed algorithm is blind, the cover image is not necessary to the watermark extractor. The watermark extracting procedure includes two parts: watermark signal extracting and watermark image extracting.

3.1 Watermark Signal Extracting

Supposing the watermarked image is $I^{'}$ (size P×Q)

1) Dividing the watermarked image $I^{'}$ into 8×8 blocks

$$D_m = f_m(x,y), \mathrm{m} = 0,1,\cdots \lceil (P \times Q)/(8\times8) - 1 \rceil \quad (12)$$

2) Transforming every D_m block into DCT domain

$$D_m^{'} = F_m(u,v) = DCT\{f_m(x,y), 0 \le x,y,u,v \le 7\} \quad (13)$$

3) Making the adaptive quantization according to the HVS characteristic of every D_m block:

First, we calculate the entropy and the variance of every D_m block, and then determine the region (S1, S2, S3 or S4) that every D_m block lies in according to the entropy and the variance. To the different region (S1, S2, S3 and S4), we quantize with the half of the original quantization magnitude(R1,R2,R3 and R4). In this paper, R1 is equal to 6, R2 is equal to 10, R3 is equal to 18 , R4 is equal to 28. So, accordingly, we make the adaptive quantization with 3,5,9 and 14.

Second, We transform the every D_m block into DCT domain, named as $D_m^{'}$. Then, we make the adaptive quantization to every $D_m^{'}$ block,. For example, some lower frequency coefficient (supposing $F_m(0,1)$) of $D_m^{'}$ block is equal to 236 and it lies in S2, so the quantization magnitude is the half of R2. After quantization, $F_m(0,1)$ is equal to 47.

4) Extracting the watermark signal according to the value after the adaptive aquantization

From the formula (10), after the adaptive quantization, if the value of $F_m(u,v)$ is odd, the embedded watermark signal is'1',or else the embedded watermark signal is '0'. In the above example, $F_m(0,1)$ is equal to 47, so the embedded watermark signal should be '1'.

3.2 Watermark Image Extracting

The watermark image can be extracted according to the inverse procedure of the watermark decoding.

Supposing the extracted watermark signal is W^*,

1) Decrypting W^* by RSA decryption techniques with the public key, and the result is named as ***W'***
2) Transform the 1-dimension binary sequence of ***W'*** into decimal sequence grouped by 8 bits and the result is named as ***W''***
3) Constructing the DCT matrix in blocks of 8×8 pixels and making the inverse JPEG quantization and making the inverse DCT transformation to recover the watermark image according to ***W''***.

4.EXPERIMENT RESULTS

In the experiments, we use lena image (size 512×512) as cover image in Fig.2(a) and hust image (size 44×44) as the watermark image in Fig.2(c).The watermarked image is displayed in Fig.2(b).

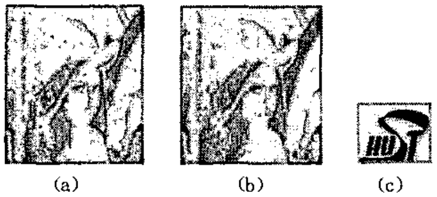
(a) (b) (c)

Fig. 2: (a) cover image (b) watermarked image (c) watermark image

The following default test cases from Stirmark3.1 are performed, and the results are tabulated in Table 1.The bit error rate (BER) in the table is calculated as the number of incorrectly decoded bits divided by the total number of embedded bits in the watermarked image.

cases attackes	1	2	3	4	5	6	7
Removal	0	0.02	0.02	0.02	0.02	0.03	0.03
scaling	0.35	0.33	0.31	0.31	0.32		
Aspect ratio	0.28	0.31	0.32	0.31	0.31	0.3	0.32
Rotation	0.26	0.21	0.26	0.19	0.23	0.25	0.21
sharpening	0						
LR attack	0.08						
Linear trans.	0.1	0.1	0.1				
Centered cropping	0.1	0.35	0.5	0.55			
Median filter	0.05	0.03	0.06				
Gaussian filter	0.1						
RBA	0.58						
Horizontal flip	0.1						

Table 1: Test results of StirMark3.1 attack

(1) Line and column removal: (1,1), (5,1), (1,5), (10,5), (5,10), (8,17), (17,8). The first and second component is the number of rows and columns removed respectively.

(2) Scaling: scaling by factors 0.25,0.5,0.75,0.9, 1.2.

(3) Aspect ratio change: (0.8,1), (0.9,1), (1.1,1), (1.2,1), (1,0.8), (1,0.9), (1,1.1). The first component is the scaling in X direction, and the second is the scaling in Y direction.

(4) Rotation by a small angle followed by cropping: -1, -0.75, -0.5, -0.25, 0.25, 0.5, 0.75.

(5) Sharpening by: {0 -1 0; -1 5 –1;0 -1 0}.

(6) Frequency mode Laplacian removel.

(7) General linear geometric transformation followed by: {1.010 0.013;0.009 1.011}, {1.007 0.010;0.010 1.012}, {1.013 0.008;0.011 1.008}.

(8) Centered cropping: 1%, 2%, 5% and 10%.

(9) Median filtering: 2×2, 3×3, 4×4.

(10) Gaussian filtering by: {1 2 1; 2 4 2; 1 2 1}.

(11) Random bending attack (RBA).

(12) Horizontal flip.

To the different JPEG compression quality, The proposed algorithm is very robust. The results are shown in Fig.3.

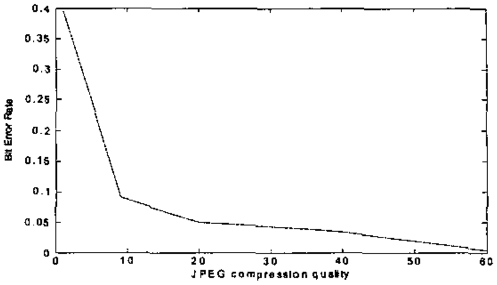

Fig.3: BER vs. JPEG quality

5.Conclusion

In this paper, we proposed a new public robust watermarking scheme. Based on HVS and image compression technique, we realize a 2-dimension image watermarking system.

From the experiment results, we can see that the proposed algorithm is very robust to most attacks, except RBA and centered cropping attack.

References

[1].Cox I J, Killian J, Leighton F T et al. "Secure spread spectrum watermarking for multimedia".IEEE Trans Image Processing, 1997, 6(12): 1673~1687

[2].Yeung M.M. "Digital Watermarking", Editorial for communications of the ACM , 1999,41(7):31-33

[3].Christine I. Podilchuk, Wenjun Zeng, "Digital Image Watermarking using Visual Models". In: Proceeding of SPIE on Human Vision and Electronic Imaging, San Jose, 2000, 3016:100~111

[4].J.W. Huang, Y.Q. SHI "An adaptive image watermarking algorithm". Acta automatica sinica, 1999, 25 (4): 477~482.

[5].Jayant N, Johnston J, Saftanek R. "Signal compression based model of human perception" . Proceedings of the IEEE, 1993,81(10):1385-1421

[6].Hui Xiang,et al. "Digital Watermarking Systems with Chaotic Sequences". In: Proceeding of SPIE on Security and Watermarking of Multimedia Contents, San Jose, 1999, 3657:449~457