IEICE TRANS. INF. & SYST., VOL.E96–D, NO.10 OCTOBER 2013

PAPER

Print-and-Scan Resilient Watermarking through Polarizing DCT Coefficients

Chun-Hung CHEN[†], ***Student Member***, **Yuan-Liang TANG**[††a)], ***and*** **Wen-Shyong HSIEH**[†,†††], ***Nonmembers***

SUMMARY Digital watermarking techniques have been used to assert the ownerships of digital images. The ownership information is embedded in an image as a watermark so that the owner of the image can be identified. However, many types of attacks have been used in attempts to break or remove embedded watermarks. Therefore, the watermark should be very robust against various kinds of attacks. Among them, the print-and-scan (PS) attack is very challenging because it not only alters the pixel values but also changes the positions of the original pixels. In this paper, we propose a watermarking system operating in the discrete cosine transform (DCT) domain. The polarities of the DCT coefficients are modified for watermark embedding. This is done by considering the properties of DCT coefficients under the PS attack. The proposed system is able to maintain the image quality after watermarking and the embedded watermark is very robust against the PS attack as well.

key words: *digital watermarking, print-and-scan attack, discrete cosine transform, geometric distortion*

1. Introduction

As multimedia technologies advance, it is very easy to copy, modify, and distribute digital images. However, such convenience also brings about many problems; for example, unauthorized copying may lead to copyright infringement. Currently, watermarking techniques are used to solve the problem by embedding the copyright information into the image as a way to assert its ownership. The embedded information thus should be very robust against any attempts to break or remove the watermark. Among the various attacks, the print-and-scan (PS) attack is very challenging because it not only alters the pixel values but also changes the positions of the original pixels. Most of the modern watermarking systems would fail under such an attack. In this paper, we propose a novel PS-attack resistant watermarking algorithm for color images. The algorithm operates in the discrete cosine transform (DCT) domain and takes advantage of the characteristics of the DCT coefficients under the PS attack. We first investigated the influence of the PS attack on the image pixels as well as the corresponding DCT coefficients. Then, the results of the investigation were used to design a robust watermarking algorithm which can resist the PS attack.

Manuscript received October 23, 2012.
Manuscript revised April 23, 2013.

[†]The authors are with the Department of Computer Science and Engineering, National Sun Yat-sen University, Taiwan.
[††]The author is with the Department of Information Management, Chaoyang University of Technology, Taiwan.
[†††]The author is with the Department of Computer Science and Information Engineering, Shu-Te University, Taiwan.
a) E-mail: yltang@cyut.edu.tw (Corresponding author)

DOI: 10.1587/transinf.E96.D.2208

2. The PS Attack and PS-Resistant Watermarking

2.1 The PS attack

Lin and Chang [1] modeled the PS attack on an image by considering the following two kinds of alterations. (1) Pixel value changes: the brightness, contrast, and colors of image pixels will be affected by the PS attack. (2) Geometric distortions: during the PS attack, the image will experience such geometric distortions as scaling, rotation, and cropping. Most of these distortions result from improper human operations during the printing and/or scanning processes. Geometric distortion destroys the *synchronization* of the watermark, that is, the embedding positions of the watermark are changed so that detection of the watermark using the original positions would fail. These distortions need to be corrected before watermark detection.

2.2 PS-Resistant Watermarking Techniques

A number of researchers have made efforts to tackle the PS attack problem. Guo *et al.* [2] divided an image into blocks and performed the digital Fourier transform (DFT) on those blocks with more complex textures. And then, the values of the coefficients in the mid-frequency areas are increased for watermark embedding. As complex textures are easily smoothed by the PS process, the embedded watermark is very vulnerable; hence, their method is more suitable for high-quality images. Solanki *et al.* [3] used the differential quantization index modulation to embed information in the DFT phase spectrum of the images. The embedding is done by quantizing the phase difference of the adjacent frequency locations. In addition, they estimated the amount of rotation resulting from the scanning process by utilizing the knowledge of the digital halftoning.

Jin *et al.* [4] divided an image into blocks, performed DCT on each block, and then selected the coefficients whose absolute values are greater than some threshold for watermark embedding. The watermark is embedded by setting the relation between the numbers of positive and negative coefficients in each block. Kang *et al.* [5] applied a discrete log-polar transformation on the Fourier magnitude coefficients' Cartesian coordinate to obtain the log-polar coordinate. The watermark is composed of an informative watermark and a

tracking pattern and then embedded to the Fourier magnitude coefficients. The cross phase correlation between the tracking pattern and the log-polar mapped magnitude spectrum of the reproduced image is used to identify the geometric distortion for resynchronization. Their method can achieve a high success rate in extracting multiple watermark bits from the image attacked by a combined operation of JPEG compression and printing-scanning.

He and Sun [6] proposed a watermarking algorithm in which image blocks are classified into smooth or texture types according to magnitude of mid-frequency DFT coefficients. And then, for the texture-type blocks, the watermark is embedded by some method based on DFT; whereas, for the smooth-type blocks, the watermark is embedded by some spatial domain method. Their method can achieve high capacity. Most of the existing techniques deal with grayscale images. However, the method proposed by Chiu and Tsai [7] is designed for color images. They defined certain mid-frequency DFT coefficients as embeddable positions and then modified half of them to achieve a local maximum (peak). These local peaks are used to synchronize the reading of a message arranged in a circular structure. One of the shortcomings of their method is its high computational cost.

Pramila *et al.* [8] took the advantages of different domains to improve the robustness of the embedded watermark. Two synchronizing templates are embedded in the Fourier and spatial domains, respectively, and the watermark is embedded in the wavelet domain. The embedded templates are able to correct geometrical distortions after the print-scan process. Cheng *et al.* [9] found that the averages of DCT coefficients after the PS operation would not change significantly. Therefore they embedded the watermark by changing the signs of the coefficients. The embedding strength of the watermark is determined through the JND model in order not to distort the image too much. They also defined a threshold to determine if the watermark is existent in the image. However, they didn't consider the characteristics of the image when defining the threshold.

Shi *et al.* [10] divided the image into blocks and then performed DCT on each of them. The mid-frequency coefficients are then selected for watermark embedding by setting a set of the DCT coefficients either to be positive or negative for embedding the bit 1 or 0, respectively. Such a method has a severe impact on the image quality if a great number of coefficients change their signs. To overcome, they recorded the relationship between positive and negative coefficients in each block: if the number of positive coefficients is greater than that of negative ones, 1 is recorded; otherwise, 0 is recorded. Therefore, the watermark is embedded by changing the signs of the coefficients with fewer numbers. As a result, the relationship information has to be stored for watermark detection later. This makes their system an informed watermarking system, rather than a more ideal blind watermarking system.

In this paper, we propose a novel watermarking algorithm operating in the DCT domain. The signs of the DCT coefficients are manipulated for watermark embedding which leads to a very robust watermarking system against the PS attack. The image quality is preserved to a high standard as well.

3. The Proposed Method

The design of our method is based on the properties of the DCT coefficients under the PS attack. The watermark is embedded by changing the *polarity* of a set of DCT coefficients. The polarity is defined as: if the number of the positive coefficients is greater than that of the negative ones, it is of *positive polarity*. Otherwise, it is of *negative* polarity. The details of the watermarking algorithm are described as follows.

3.1 Watermark Embedding

Let $X = \{(C_1, C_2, C_3)\}$ be the host color image, and its three color components are denoted as C_1, C_2, and C_3, respectively. Figure 1 briefly delineates the watermark embedding process: (1) The selected channel is first resized to a standard size of $M \times M$, and is divided into blocks of size $N \times N$, which produces $(M/N)^2$ blocks in total. As the border blocks are more vulnerable to the PS attack, only the internal $K = (M/N - 2)^2$ blocks are used for embedding. (2) Perform DCT on each of the K blocks, followed by zig-zag scanning the coefficients to produce a sequence of length N^2. (3) Supposing the watermark W consists of m bits: $W = \{w_i | w_i = 0, 1;\ i = 1 \ldots m\}$, m mid-frequency coefficients are selected out of the N^2 coefficients in each se-

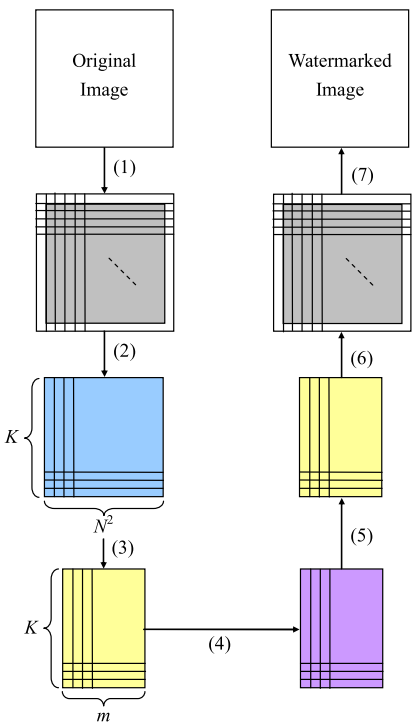


Fig. 1 The embedding process.

Table 1 Percentages of sign-changing coefficients after PS.

Channel	Image	First 10%	First 30%	First 50%	First 70%	First 90%
Red (R)	Lena	3.6	7.5	11.0	15.1	20.1
	Baboon	0.5	2.0	4.5	7.9	13.3
	Peppers	2.4	6.2	10.6	15.3	20.3
Green (G)	Lena	1.9	5.8	9.6	13.5	18.4
	Baboon	0.2	1.0	2.5	4.8	8.7
	Peppers	2.7	6.4	10.7	14.9	19.7
Blue (B)	Lena	1.7	4.2	7.3	11.4	16.5
	Baboon	0.2	0.8	2.3	5.2	10.2
	Peppers	3.0	7.9	13.1	18.2	23.4
			Average	**8.59**		
Hue (H)	Lena	22.4	28.2	30.7	32.9	34.2
	Baboon	10.4	16.1	19.6	23.1	25.9
	Peppers	47.0	46.6	45.7	42.2	39.0
Saturation (S)	Lena	12.2	17.2	18.7	21.1	24.6
	Baboon	3.8	8.2	11.8	15.6	19.8
	Peppers	25.1	24.4	26.9	29.6	32.7
Intensity (I)	Lena	12.2	17.2	18.7	21.1	24.6
	Baboon	3.8	8.2	11.8	15.6	19.8
	Peppers	25.1	24.4	26.9	29.6	32.7
			Average	**23.28**		
Y	Lena	1.9	5.5	8.9	12.5	17.1
	Baboon	0.3	1.0	2.4	4.5	8.5
	Peppers	2.4	5.4	9.0	13.5	18.4
Cb	Lena	2.0	5.5	9.8	14.4	19.5
	Baboon	1.1	2.3	4.5	7.8	13.1
	Peppers	4.1	8.5	13.5	18.2	23.2
Cr	Lena	3.0	7.5	12.0	16.9	21.8
	Baboon	0.6	2.2	4.9	8.7	13.9
	Peppers	4.3	10.3	14.9	19.0	23.6
			Average	**9.38**		

quence for watermark embedding. This produces an $m \times K$ matrix. That is, one bit is embedded in K coefficients. (4) As the coefficients in each row of the matrix is ordered from lower to higher frequencies, the robustness of the embedded watermark bits will be descending. Therefore, we apply random permutation on the coefficients in each row in order to balance their robustness. (5) A secret key is used to generate a sequence of bits as the watermark, and each of the watermark bit is embed by changing the values of the K coefficients such that they are of positive (bit 1) or negative (bit 0) polarity. (6) Apply inverse permutation on the matrix. (7) The final watermarked image is then produced by inversely zig-zag scanning the coefficients, restoring to the original block positions, resizing to the original image size, and finally combining with the other two color channels.

To find an appropriate color models for watermarking embedding, several models such as RGB, HSI, and YCbCr, were chosen for testing the degrees of the polarity invariance after the PS process. We order the DCT coefficients according to their absolute values, conduct experiments based on portions of the leading coefficients (10%, 30%, ...), and then count the percentages of coefficients that change their signs due to the PS process. The results are shown in Table 1. It is obvious that the RGB model has the lowest average percentage of sign changes, thus it is most robust against the PS attack. To find out which channel in the RGB model

Table 2 The variances of sign-changing percentages among images.

Channel	Image	First 10%	First 30%	First 50%	First 70%	First 90%
Red (R)	Lena					
	Baboon	2.4	8.3	13.3	17.8	15.9
	Peppers					
Green (G)	Lena					
	Baboon	1.6	8.8	19.8	29.9	36.1
	Peppers					
Blue (B)	Lena					
	Baboon	2.0	12.6	29.2	42.3	43.6
	Peppers					

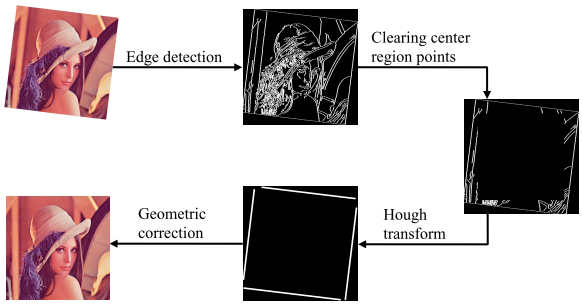


Fig. 2 Geometric correction.

is most appropriate for watermark embedding, we also calculate the variance of sign-changing percentages among the images, as shown in Table 2. It turns out that the R channel is most stable than the G and B channels. Thus, the RGB color model and the R channel are chosen as our watermarking domains.

To embed one watermark bit, supposing there are p positive and n negative coefficients in the K coefficients, respectively, the watermark embedding is amount to changing some of the coefficients' values such that the block is of positive or negative polarity. This is done by setting the resulting numbers of positive and negative coefficients to a predefined ratio r ($r > 1$):

$$p' = rn' or\ n' = rp',$$

where p' and n' are the numbers of positive and negative coefficients after modification, respectively. Let d be the number of coefficients to be modified. For positive polarity, d is obtained by

$$\begin{cases} d = p' - p, & \text{if } p' > p \\ d = 0, & \text{otherwise} \end{cases}$$

For negative polarity, d is obtained by

$$\begin{cases} d = n' - n, & \text{if } n' > n \\ d = 0, & \text{otherwise} \end{cases}$$

To preserve the image quality, the coefficients with small magnitudes will be selected for modification. In addition, the values of them are set to be the average values of the opposite-sign coefficients. After setting the polarity, the positive (or negative) coefficients are multiplied by an embedding strength factor, α ($\alpha > 1$), for further enhancing the

robustness. The embedding process is described as follows:

If $w_i = 1$:

(1) If $p < rn$, achieve a positive polarity such that $p' = rn'$. The values of the d negative coefficients are set to be the average of the positive coefficients.
(2) Multiply the positive coefficients by α.

else ($w_i = 0$):

(1) If $n < rp$, achieve a negative polarity such that $n' = rp'$. The values of the d positive coefficients are set to be the average of the negative coefficients.
(2) Multiply the negative coefficients by α.

After the watermark embedding, perform inverse DCT on each block, rearrange the blocks, resize the original image size, and then combine the three color channels to produce the watermarked image.

3.2 Watermark Detection

Before actually detecting the watermark, we need to perform geometric correction on the image first.

1) Geometric Correction

During the printing and scanning process, the image may experience geometric distortions resulting from human operation errors. The most common are rotation and scaling. The former is due to improper alignment of the paper image during scanning, and the latter is due to different resolution settings between printing and scaling. In our method, the rotation distortion is recovered by first detecting the border lines of the scanned image using the Hough transform, and then rotating the image back accordingly. The scaling distortion is corrected by setting the image to a standard size. In the scanned image, to detect the border lines of the original image, we first detect the edge pixels in the image. And then, the Hough transform is performed to detect the four border lines, whose slopes are used to rotate the image back to its proper orientation. The process is delineated in Fig. 2.

As the Hough transform is very time consuming, it is necessary to develop some means to reduce the computation complexity. Since the Hough transform is computed based on edge pixels, the more edge pixels found in the image, the more time it will spend on computing. Because we are interested in finding only the four border lines, pixels unlikely belonging to the borders should not be taken into account. Thus, if the rotation angle, θ, is supposed to be within some range, say $-10^\circ < \theta < 10^\circ$, which is a reasonable assumption for a human trying to align a paper image properly during scanning, we may consider only those pixels which are in the possible regions for the borders to appear, as shown in gray in Fig. 3. That is, the edge pixels in the center region (shown in white) can be ignored. Such a consideration

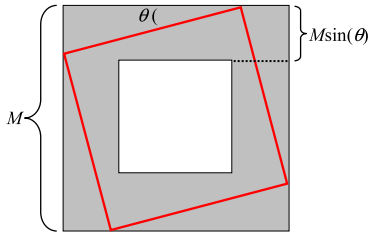


Fig. 3 Possible region (shown in gray) in which borders may appear.

greatly reduces the computational cost.

2) Watermark Detection

The watermark detection proceeds as follows. First, take the R channel of the scanned color image, and then perform geometric correction, followed by scaling the image to the predefined $M \times M$ standard size. The rest of the steps are similar to those in the embedding process: divide the image into blocks of size $N \times N$, select internal K blocks, perform DCT on each block, select m mid-frequency coefficients to form an $m \times K$ array, apply random permutation, and finally detect each watermark bit in each set of K coefficients using the following decision formula to obtain the detected watermark W':

$$\begin{cases} w'_i = 1, & \text{if } p' \geq n' \\ w'_i = 0, & \text{otherwise} \end{cases}$$

The detected watermark, W', is then compared with the watermark, W, generated by the secret key, which produces the error rate E as follows. If E is less than some threshold, the watermark is declared to exist.

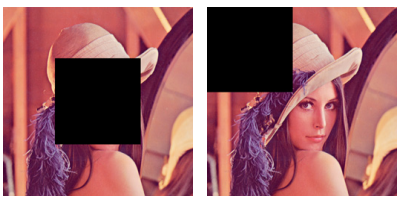
$$E = \frac{1}{m} \sum_{i=1}^{m} \left| w_i - w'_i \right|$$

4. Experimental Results

Six color images were tested in our experiments: *Lena*, *Baboon*, *Peppers*, *Airplane*, *House*, and *Sailboat*. The standard image size $M \times M$ and block size $N \times N$ are set to be 512×512 and 32×32, respectively. The leading 10% to 40% coefficients are considered as mid-frequency coefficients and they are used for watermark embedding in our method. Selecting these mid-frequency coefficients has certain advantages: the watermark will be relatively more robust against the PS attack and the image fidelity may be well preserved. The embedding capacity is defined as $(3 \times N^2)/10$, therefore it will be $(3 \times 16^2)/10 = 76$ (bits) or $(3 \times 32^2)/10 = 307$ (bits) if we set N to be 16 or 32, respectively. As the purpose of watermarking is to identify the owner of a digital image, the watermark is usually some form of the owner's identity (social security number, for example). In general, 307 bits ($\approx$ 38 ASCII characters) are more than enough for storing an identity; therefore, we set $N = 32$. There are thus $(512/32) \times (512/32) = 256$ blocks in total, in which the

Table 3 Error rates under the StirMark attacks.

Attack	L	B	P	A	H	S
JPEG 50	7.7	7.2	7.7	2.6	1.0	5.6
JPEG 60	3.6	4.6	6.2	3.1	0.0	1.5
JPEG 70	1.0	5.1	2.1	0.0	0.0	0.0
JPEG 80	0.0	3.6	0.0	0.0	0.0	0.0
JPEG 90	0.0	0.0	0.0	0.0	0.0	0.0
Median Filter 3×3	0.5	1.5	0.5	0.0	0.0	0.0
Gaussian Filter	0.0	0.0	0.0	0.0	0.0	0.0
Sharpening	0.0	0.0	0.0	0.0	0.0	0.0
Rotation -5°	0.0	0.0	0.0	0.0	0.0	0.0
Rotation 5°	0.0	0.0	0.0	0.0	0.0	0.0
Scaling 25%	8.7	9.2	9.2	7.2	5.1	9.2
Scaling 50%	0.0	0.0	0.0	0.0	0.0	0.0
Scaling 75%	0.0	0.0	0.0	0.0	0.0	0.0
Scaling 150%	0.0	0.0	0.0	0.0	0.0	0.0
Scaling 400%	0.0	0.0	0.0	0.0	0.0	0.0

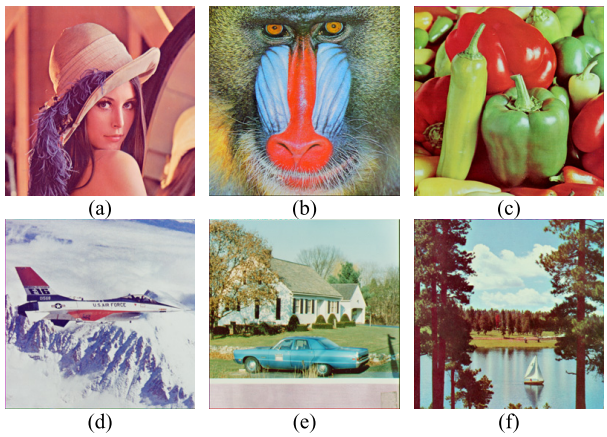


(a) (b)

Fig. 4 Cropped images: (a) Center cropping 20% off, (b) Top-left cropping 20% off.

Table 4 Error rates under cropping attacks.

Attack	L	B	P	A	H	S
Center cropping 5% off	0.0	0.0	0.0	0.0	0.0	0.0
Center cropping 10% off	0.0	0.0	0.0	0.0	0.0	0.0
Center cropping 15% off	0.5	3.1	0.5	1.5	0.0	1.0
Center cropping 20% off	1.0	8.4	4.6	3.1	0.5	5.1
Top-left cropping 5% off	0.0	0.0	0.0	0.0	0.0	0.0
Top-left cropping 10% off	0.0	0.0	0.0	0.0	0.0	0.0
Top-left cropping 15% off	0.0	0.0	0.0	0.0	0.0	0.0
Top-left cropping 20% off	2.1	1.0	0.5	1.5	0.0	1.0



(a) (b) (c)

(d) (e) (f)

Fig. 5 Watermarked images: (a) Lena, (b) Baboon, (c) Pepper, (d) Airplane, (e) House, (f) Sailboat.

internal $K = 14 \times 14 = 196$ blocks are chosen for watermark embedding. The length, m, of the watermark is 195, which means m mid-frequency coefficients are selected in each block, and a watermark bit is embedded in a set of K coefficients by changing its polarity. The ratio r and the embedding strength α are set to be 2 and 1.1, respectively, in the experiments to achieve high robustness.

To test the robustness of the proposed method, StirMark [11] operations were performed to attack our watermarked images. Table 3 lists various StirMark attacks and their corresponding watermark detection results. The error rate is defined as the number of successfully detected watermark bits divided by the total number watermark bits. As most of the detection error rates are zero and the others are all under 10%, it is obvious that our method is very robust against both pixel-value and geometric attacks. For further robustness testing, we performed center and top-left cropping attacks, as shown in Figs. 4(a) and 4(b). As our method is designed to embed watermark bits in many distributed blocks, even if some of them are cropped off, the remaining blocks are still capable of recovering the embedded information. So we expect that the cropping attacks will not have much impact on our method. This can be seen from Table 4, in which the error rates are all under 9%. One thing worth noting is that the low error rates shown in Tables 3 and 4 can be easily decreased down to zero if the watermark is encoded in advance using error-correcting codes.

To test the robustness of the proposed method against the PS attack, the *HP LaserJet Pro CP1525nw* color laser printer and the *HP F4280 All-in-One* scanner were used to print and scan the images, respectively. The watermarked images were printed with the resolution of 600 dpi and then scanned back with resolutions of 600, 300, and 100 dpis, respectively. The performance of our method is compared against those of Shi *et al.*'s, Chiu and Tsai's, and Digimarc's methods. The lengths of the watermark are defined to be 126 and 64 bits in Shi *et al.*'s and Chiu and Tsai's method, respectively. Digimarc has four levels of durability (*dur*) settings. Durability 4 stands for highest robustness and lowest fidelity, whereas Durability 1 means lowest robustness and highest fidelity. We set both $dur = 1$ and 4 in our experiments.

The watermarked images produced by the proposed method are shown in Fig. 5. As can be seen, these images all have high visual quality. Their corresponding print-and-scanned versions are shown in Fig. 6. Figure 7 shows the watermarked and print-and-scanned images produced by Shi *et al.*'s, Chiu and Tsai's, and Digimarc's methods. All of the print-and-scanned images in Figures 6 and 7 were printed with the resolution of 600 dpi and then scanned back with 100 dpi. Table 5 lists the PSNR values of the water-

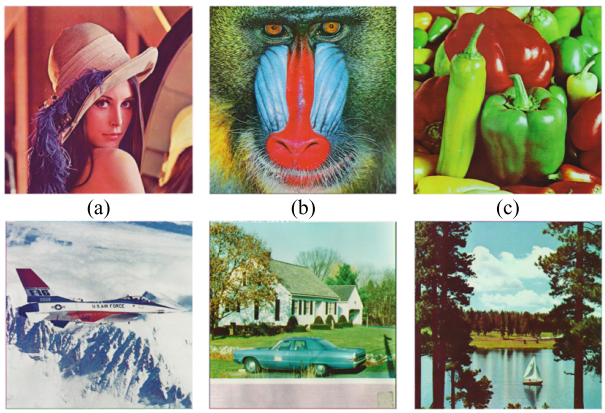


Fig. 6 Print-and-scanned images: (a) Lena, (b) Baboon, (c) Pepper, (d) Airplane, (e) House, (f) Sailboat.

Fig. 7 Watermarked (left column) and print-and-scanned (right column) produced images by: (a) Shi *et al.*'s, (b) Chiu and Tsai's, (c) Digimarc (*dur* = 1), and (d) Digimarc (*dur* = 4) methods.

marked images produced by various methods, in which (and in the following tables) symbols *L*, *B*, *P*, *A*, *H*, and *S* denote Lena, Baboon, Peppers, Airplane, House, and Sailboat images, respectively. From Table 5, it is obvious that our PSNR levels are quite acceptable and our method outperforms the others.

The results of watermark detection on 600-, 300-, and 100-dpi scanning resolutions are shown in Tables 6, 7, and 8 respectively. As Digimarc does not have error rate measurements, "Y" and "N" are used to denote if the watermark is successfully detected or not. The experimental results clearly show that if the scanning resolution is 600 or 300 dpi, the watermarks are all correctly detected (i.e., error rate = 0%) by our method, and the watermarked images have high PSNR values as well. Even when the scanning resolution is as low as 100 dpi, the error rates of our method are still under 6% and our method outperforms all the others. Therefore, our method is capable of producing robust watermarks, and the watermarked images also have high visual quality.

Table 5 Image quality measurements based on the PSNR.

Method	L	B	P	A	H	S	Avg
The proposed	50.1	46.8	49.8	48.1	45.1	47.7	**47.9**
Shi *et al.*'s	49.0	38.3	47.1	46.2	42.0	44.2	**44.5**
Chiu and Tsai's	33.0	26.9	33.0	32.4	31.3	30.3	**31.2**
Digimarc (*dur*=1)	39.8	36.1	39.7	41.0	39.8	37.8	**39.0**
Digimarc (*dur*=4)	34.3	34.1	33.8	35.0	35.0	34.2	**34.4**

Table 6 Error rates under the PS attack of 600-dpi printing and scanning resolutions.

Method	L	B	P	A	H	S	Avg
The proposed method	0.0	0.0	0.0	0.0	0.0	0.0	**0.0**
Shi *et al.*'s method	3.6	2.1	5.1	2.1	0.0	2.1	**2.5**
Chiu and Tsai's method	0.0	0.0	0.0	0.0	0.0	0.0	**0.0**
Digimarc (*dur*=1)	N	Y	N	Y	Y	Y	**4/6**
Digimarc (*dur*=4)	Y	Y	Y	Y	Y	Y	**6/6**

Table 7 Error rates under the PS attack of 600-dpi printing and 300-dpi scanning resolution.

Method	L	B	P	A	H	S	Avg
The proposed method	0.0	0.0	0.0	0.0	0.0	0.0	**0.0**
Shi *et al.*'s method	4.1	14.4	5.1	9.2	3.6	4.6	**6.8**
Chiu and Tsai's method	6.3	9.4	3.1	6.3	3.1	3.1	**5.2**
Digimarc (*dur*=1)	N	Y	N	Y	Y	Y	**4/6**
Digimarc (*dur*=4)	Y	Y	Y	Y	Y	Y	**6/6**

Table 8 Error rates under the PS attack of 600-dpi printing and 100-dpi scanning resolutions.

Method	L	B	P	A	H	S	Avg
The proposed method	6.7	8.2	6.7	5.6	2.1	4.1	**5.6**
Shi *et al.*'s method	20.5	21.5	17.4	19.5	3.6	15.4	**16.3**
Chiu and Tsai's method	17.2	29.7	21.9	26.6	25.0	18.8	**23.2**
Digimarc (*dur*=1)	N	Y	N	Y	Y	N	**3/6**
Digimarc (*dur*=4)	Y	Y	Y	Y	Y	Y	**6/6**

5. Conclusions

In this paper, we proposed a watermarking system which operates in the DCT domain and is very robust against the PS attack. The watermark is embedded by setting the polarity

of DCT coefficients. By analyzing the PS characteristics, carefully choosing the DCT coefficients for modification, and effectively setting the embedding strength, the system is able to achieve a high rate of watermark detection. The qualities of the watermarked images are also well preserved.

References

[1] C.Y. Lin and S.F. Chang, "Distortion modeling and invariant extraction for digital image print-and-scan process," Int. Symp. Multimedia Information Processing, 1999.
[2] C. Guo, G. Xu, X. Niu, Y. Yang, and Y. Li, "A color image watermarking algorithm resistant to print-scan," IEEE Int. Conf. Wireless Communications, Networking and Information Security, pp.518–521, 2010.
[3] K. Solanki, U. Madhow, B.S. Manjunath, S. Chandrasekaran, and I. El-Khalil, "Print and Scan Resilient Data Hiding in Images," IEEE Trans. Information Forensics and Security, vol.1, no.4, pp.464–478, 2006.
[4] J. Jin, Y. Xiong, and H. Hou, "A practical DCT based blind image watermarking scheme for print-and-scan process," SPIE Image Processing: Machine Vision Applications III, 2010.
[5] X. Kang, X. Zhong, J. Huang, and W. Zeng, "An efficient print-scanning resilient data hiding scheme based on a novel LPM," IEEE Int. Conf. Image Processing, pp.2080–2083, 2008.
[6] D. He and Q. Sun, "A practical print-scan resilient watermarking scheme," IEEE Int. Conf. Image Processing, vol.1, pp.257–260, 2005.
[7] Y.C. Chiu and W.H. Tsai, "Copyright protection against print-and-scan operations by watermarking for color Images using coding and synchronization of peak locations in frequency domain," Journal of Information Science and Engineering, vol.22, no.3, pp.483–496, 2006.
[8] A. Pramila, A. Keskinarkaus, and T. Seppänen, "Multiple domain watermarking for print-scan and JPEG resilient data hiding," Int. Work. Digital Watermarking, pp.279–293, 2007.
[9] D. Cheng, X. Li, W. Qi, and B. Yang, "A statistics-based watermarking scheme robust to print-and-scan," Int. Symp. Electronic Commerce and Security, pp.894–898, 2008.
[10] D. Shi, Q. Wang, and C. Liang, "Digital watermarking algorithm for print-and-scan process used for printed matter anti-counterfeit," Congr. Image and Signal Processing, vol.5, pp.697–701, 2008.
[11] A. Fabien and P. Petitcolas, "Watermarking schemes evaluation," IEEE Signal Processing Magazine, vol.17, no.5, pp.58–64, 2000.
[12] R. Villán, S. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun, "Tamper-proofing of electronic and printed text documents via robust hashing and data-hiding," Proc. SPIE, vol.6505, 65051T, 2007.
[13] Yu, X. Niu, and S. Sun, "Print-and-scan model and the watermarking countermeasure," Image and Vision Computing, vol.23, no.9, pp.807–814, 2005.
[14] H. Xu and X. Wan, "Watermarking algorithm for print-scan based on HVS and multiscale error diffusion," Int. Conf. Computer Science and Software Engineering, vol.6, pp.245–248, 2008.
[15] L. Yu and S. Sun, "Image authentication in print-and-scan scenario," Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing, vol.1, pp.295–298, 2007.
[16] P. Chen, Y. Zhao, and J.S. Pan, "Image Watermarking robust to print and generation copy," Int. Conf. Innovative Computing, Information and Control, vol.1, pp.496–500, 2006.
[17] Y. Zhao, Z. Fan, and M. Hoover, "Frequency domain infrared watermarking for printed CMYK image," IEEE Int. Conf. Image Processing, pp.2725–2728, 2011.
[18] A. Keskinarkaus, A. Pramila, and T. Seppänen, "Image watermarking with feature point based synchronization robust to print-scan attack," Journal of Visual Communication and Image Representation, vol.23, no.3, pp.507–515, 2012.
[19] S. Wang, S. Huang, X. Zhang, and W. Wu, "Hologram-based watermarking capable of surviving print-scan process," Applied Optics, vol.49, no.7, pp.1170–1178, 2010.
[20] S. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun, "Visual communications with side information via distributed printing channels: extended multimedia and security perspectives," Proc. SPIE, vol.5306, pp.428–445, 2004.
[21] J. Zhou and M. Pang, "Digital watermark for printed materials," IEEE Int. Conf. Network Infrastructure and Digital Content, pp.758–762, 2010.
[22] A. Eliasi, M. Eliasi, and Z. Yaghoubi, "Digital images watermarking robust to print & scan for electronic evidence," IEEE Int. Conf. Computer Applications and Industrial Electronics, pp.453–458, 2011.
[23] G. Nian, X. Tang, D. Wang, and H. Liu, "Print-scan resilient data hiding scheme applied in certificate verification," Int. Congr. Image and Signal Processing, vol.3, pp.1161–1165, 2010.
[24] A. Poljicak, L. Mandic, and D. Agic, "Discrete Fourier transform–based watermarking method with an optimal implementation radius," Journal of Electronic Imaging, vol.20, no.3, 033008, 2011.

Chun-Hung Chen received the M.S. degree in Information Management from Chaoyang University of Technology, Taiwan, in 2003. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, National Sun Yat-sen University. His research interests include multimedia security and data hiding.

Yuan-Liang Tang received the M.S. and Ph.D. degrees in Electrical Engineering and Computer Engineering from The Pennsylvania State University, USA, in 1991 and 1994, respectively. He is currently an associate professor at the Department of Information Management, Chaoyang University of Technology, Taiwan. His research interests include image processing, computer vision, information hiding, and watermarking.

Wen-Shyong Hsieh is a distinguished professor of Department of Computer Science and Information Engineering, Shu-Te University, Taiwan. He is also a professor of Department of Computer Science and Engineering, National Sun Yat-sen University, Taiwan. He received the B.S., M.S. and Ph.D. degrees in Department of Electrical Engineering from National Cheng-Kung University, Taiwan, in 1972, 1974 and 1992, respectively. His research interests include computer networks, multimedia system, digital watermarking, digital image compression and network security.