An Enhanced Image Watermarking Scheme Using Blocks with Homogeneous Feature Distribution

Kavitha Chandranbabu and Devaraj Ponnaian

Abstract This paper proposes a gray scale image watermarking scheme based on Discrete Cosine Transform (DCT). In this scheme, the host image is divided into non-overlapping blocks. The pixel values within the blocks are permuted and DCT is applied for each block separately. The permutation makes the distribution of the DCT coefficient uniform within the block and hence embedding watermark in any region of the block produces similar result. Since, the embedding is done in blocks, multiple embedding is also possible. Experimental results also show that the performance of the scheme is good against various image processing operations and cropping attack.

Keywords Discrete cosine transform · Block permutation · Frequency domain · Watermark

1 Introduction

Nowadays, digital information is largely used due to the ease of storage and transmission. Digitally stored information such as images, audio and video can be copied, edited and shared easily which makes it difficult to identify the ownership of the content. Digital watermarking techniques are developed to address the copyright issue for digital information. In digital image watermarking, an image that represents the owner identity is embedded in the host image. Embedding can be done either in spatial domain or on frequency domain of the host image. The watermark embedding in the spatial domain is considered to be weak against

K. Chandranbabu (✉) · D. Ponnaian
Department of Mathematics, College of Engineering, Anna University Chennai,
Guindy, Chennai 600025, India
e-mail: kavistha@gmail.com

D. Ponnaian
e-mail: devaraj@annauniv.edu

M. Senthilkumar et al. (eds.), *Computational Intelligence,*
Cyber Security and Computational Models, Advances in Intelligent
Systems and Computing 412, DOI 10.1007/978-981-10-0251-9_9

various image processing operations and hence, most of the algorithms are designed to embed the watermark in the frequency domain which is obtained by using transforms such as discrete cosine transforms (DCT) [1–12], discrete Fourier transforms (DFT), and discrete wavelet transforms (DWT). Frequency domain embedding identifies the region of embedding based on the sensitivity of Human Visual System (HVS). The DCT reserves most of the signal information in the low frequency components. Hence, algorithms [9–12] using DCT, embeds the watermark in the middle frequency band to overcome the tradeoff between the robustness and imperceptibility.

In [9], the watermarking algorithm is designed for image content authentication using DCT where the watermark generated from the original image is embedded in middle frequency range. In [10], the DCT coefficients are modified to embed watermark based on the inter-block DCT coefficient correlation. The modification parameter is computed using the DC coefficient and some low frequency coefficient. The scheme proposed in [11], determines the DCT of each 8 × 8 block and computes the complexity of each block by counting the number of nonzero entries in each block. Based on the complexity the blocks are selected and watermark is embedded in the low frequency band of the selected blocks. An algorithm based on Chinese remainder theorem and DCT was proposed in [13]. In [12], sub sampling technique along with Just Noticeable Distortion (JND) function is used for watermarking where the watermark is embedded in middle frequency domain. Some schemes [14, 15] also use the DC coefficients for marking the watermark to improve the robustness.

In [16], using sub sampling four sub images are formed for which DCT is applied. A permutated binary watermark image is embedded into the DCT coefficients of two of the sub images which are selected according to a random coordinates sequence generated by a secret key. This paper proposes a modified version of the embedding technique discussed in [16] and embeds the watermark into non-overlapping blocks of the host image which have permuted pixel values. When DCT is applied to the permuted blocks, the features within the block are uniformly distributed and hence, the watermark can be embedded in any region of the block. This makes the scheme overcomes the restriction of embedding the watermark in the middle frequency band alone. Moreover, based on the size of watermark the watermark can also be embedded more than once.

2 Block Permutation

Most of the watermarking algorithms use the concept of creating blocks from the host image and embedding watermark in it. Here, along with the concept of creating blocks, block permutation is also done before embedding watermark. The embedding algorithm in this paper divides the host image into non overlapping block, permutes the pixel values within blocks and applies DCT for each block separately. The DCT has the property of concentrating the low frequency

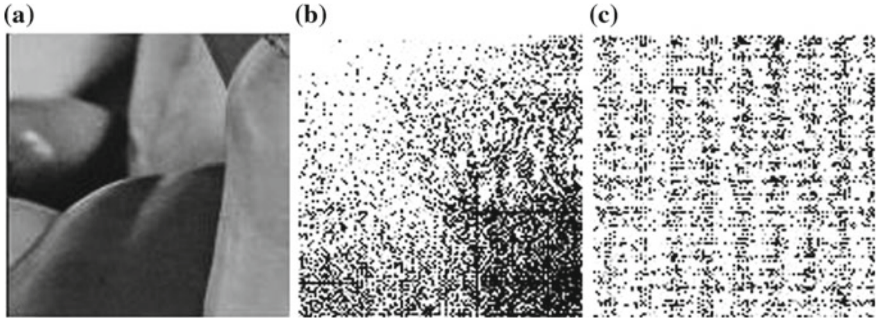

Fig. 1 **a** Original image. **b** DCT of (**a**). **c** DCT of permuted image (**a**)

components in the top left corner which varies diagonally till bottom right corner. The top left most component is the DC coefficient. The remaining components are called AC coefficients. Due to this property of the DCT, embedding the watermark in the low frequency component may result in degradation of the original image whereas the high frequency components may be lost while applying compression functions. Hence, always the middle band serves as the region of embedding. To overcome this restriction, the pixel values within the blocks are permuted and then DCT is applied, which makes the frequency distribution uniform throughout the block. Hence, embedding the watermark in any region has equal chance of being marked in low or high frequency component. Due to this technique the quality of the original image is also maintained up to the desired level as well as the embedding is made robust. The region of embedding is also increased. The permutation technique used in this scheme is first permutating all rows of the block using the function $t = ((p * s) mod\, D) + 1$, where D is the number of rows in the block, p is the prime number, s and t are the actual and destination row locations, respectively. The row permuted image is again permuted columnwise using the same function with different prime number. Figure 1 shows the distribution of the DCT block with and without permutation.

3 Proposed Scheme

Consider a gray scale image, I, of size $H \times W$, where H and W are height and width of the image, respectively. Let the binary watermark, w be of size $H_1 \times W_1$, where H_1 and W_1 represents the height and width of the watermark. Since, the watermark is usually much smaller than the host image, it can be embedded using few blocks of the host image and this embedding algorithm requires the host image to be of size $H \geq 4.H_1$ and $W \geq 4.W_1$, which is sufficient to create blocks.

3.1 Watermark Embedding

1. The watermark, w, is scrambled using the standard map (1) with the secret key x_0, $y_0 \in [0, 2\pi)$ and the parameter $|K| > 0$.

$$\left.\begin{aligned} x_{n+1} &= (x_n + y_n)\, mod\, 2\pi \\ y_{n+1} &= (y_n + K.\sin(x_{n+1}))\, mod\, 2\pi \end{aligned}\right\} \tag{1}$$

2. The scrambled watermark is transformed into a vector, $w^{'}$ of size $H_1 \times W_1$.
3. Divide the host image, I, into maximum possible number non-overlapping blocks of sub images $B_{i,j}$, of size $BD \times BD$ where $BD = \min(2.H_1, 2.W_1)$, $1 \leq i \leq \lfloor \frac{H}{BD} \rfloor$ and $1 \leq j \leq \lfloor \frac{W}{BD} \rfloor$. The part of the image that cannot be divided into blocks of the required size are not used for embedding and are retained as such in the watermarked image.
4. The pixels values within the blocks are permuted using the permutation process discussed in Sect. 2 to make block features uniformly distributed.
5. The permuted blocks rearranged to form a rectangular matrix

$$\begin{gathered} Q = [B_{1,1}, B_{2,1}, \ldots B_{i,j}, B_{i+1,j}, B_{i,j+1}, B_{i+1,j+1} \ldots], \\ i = 1, 3, \ldots, \frac{H}{BD} - 1, j = 1, 3, \ldots, \frac{W}{BD} - 1 \end{gathered} \tag{2}$$

6. In the embedding process, 1/4th of the watermark vector $w^{'}$ is embedded in each block. Hence, for a single embedding of the watermark vector $w^{'}$ four blocks from Q are necessary.
7. A random sequence $Z = (u, v)$ with $1 \leq u, v \leq 4$ of size $H_1 \times W_1 \times \lfloor (size\, of\, Q)/4 \rfloor$ is generated. This sequence is used as a coefficient selector during the embedding process.

The steps involved in embedding the watermark vector $w^{'}$ is as follows:

a. Choose first four blocks from the matrix Q defined in (2).
b. Apply the DCT for each block separately to obtain the four DCT coefficient sets, D_1, D_2, D_3 and D_4.
c. The watermark is embedded in the DCT coefficient set in the zigzag order leaving the DC coefficient and some of the rows diagonally below the DC coefficient. The watermark bits are embedded starting from the diagonally rth row where $r = \lceil (BD/t) \rceil$, $t \in (1, BD - 1)$.
d. For the mth bit of the watermark vector $w^{'}$, i.e. $w^{'}_{m}$, there are four DCT coefficients in the sets D_1, D_2, D_3 and D_4 out of which a pair is selected using the coefficient selector, $Z_m = (u, v)$. Within the set D_u and D_v the mth DCT coefficient in the zigzag order starting from the diagonally rth row is used for embedding. Let it be denoted by V_u and V_v.

e. Now, compare V_u and V_v. If the bit $w_m^{'} = 1$ and $V_u < V_v$, then swap u and v in Z_m. Similarly, if the bit $w_m^{'} = 0$ and $V_u > V_v$, then swap u and v in Z_m.

f. Using the swapped Z_m, reselect V_u and V_v and compute the following to embed the watermark bit.

$$
\begin{aligned}
& V_a = \frac{|V_u| + |V_v|}{2}, \; D = \frac{V_u - V_v}{2} \\
& \mathit{If}\, |\frac{D}{V_a}| \le \beta \\
& \quad V_u^{'} = V_u + \alpha(2w_m^{'} - 1)|V_a|; \quad V_v^{'} = V_v - \alpha(2w_m^{'} - 1)|V_a| \\
& \mathit{else} \\
& \quad V_u^{'} = V_u; \quad V_v^{'} = V_v
\end{aligned}
$$

where $\alpha \in [0.05, 0.1)$, $\beta = (.1 - a) \times 10$ and α is the watermark embedding strength and β is the threshold.

g. Apply the inverse DCT to the altered set $D_{1,}$ D_2, D_3 and D_4 and update the modified blocks into the matrix Q.

h. Select the next four blocks Q and apply step (b)–step (g), until the number of blocks remaining in Q are less than four or all blocks exhausted.

i. The embedded blocks in the rectangular matrix are again replaced to their corresponding positions in the host image.

4 Watermark Detection

Let $\bar{I}$ be the watermarked image from which the watermark is to be detected. From the image $\bar{I}$ the rectangular matrix Q is created as in embedding process. The swapped coefficient selector $Z = (u, v)$ obtained during embedding process is used for watermark detection.

a. Choose first four blocks from the matrix Q formed from $\bar{I}$.

b. Apply the DCT for each block separately to obtain the four DCT coefficient sets, $\bar{D}_1, \bar{D}_2, \bar{D}_3$ and $\bar{D}_4$.

c. The watermark bits are extracted from the DCT coefficient set in the zigzag same as in embedding process. The mth bit of the extraction vector $\bar{w}^{'}$, i.e. $\bar{w}_m^{'}$, is determined using the DCT coefficients V_u and V_v selected using the swapped coefficient selector $Z_m = (u, v)$.

$$\bar{w}_m^{'} = \begin{cases} 1, & if\ V_u \geq V_v, \\ 0, & if\ V_u < V_v. \end{cases}$$

d. The extraction vector is converted into a two dimensional array of size $H_1 \times W_1$, for which inverse permutation is applied using the standard map with same key. The image produced is the watermark $\bar{w}$, that is embedded the set of blocks.
e. The Normalised Correlation (NC) of the extracted watermark is computed using formula (3) given below.

$$NC = \frac{\sum_i \sum_j w(i,j) \cdot \bar{w}(i,j)}{\sum_i \sum_j [w(i,j)]^2} \tag{3}$$

where $w(i,\ j)$ and $\bar{w}(i,j)$ represents the $(i,\ j)$th pixel value in the original watermark and extracted watermark, respectively.
f. Select the next four blocks Q and apply step (b)–step (e), until the number of blocks remaining in Q are less than four or all blocks exhausted.
g. Finally, using $\bar{w}$ and their corresponding NC from each block the extracted watermark w_e is obtained by taking weighted average as given in (4).

$$w_e = \sum_i \frac{NC(i)}{Total} * \bar{w}_i \tag{4}$$

where $NC(i)$ is the normalized correlation of ith extracted watermark, $Total$ is the sum of $NC's$ of all extracted watermark and $\bar{w}_i$ is the ith extracted watermark from the blocks.

5 Experimental Results

The proposed scheme is tested using a gray scale image of size 512 × 512 and a binary watermark of size 64 × 64. The performance of the scheme is analysed using two types of measure: peak signal to noise ratio (PSNR) and normalized correlation (NC). PSNR is used to test the quality of the watermarked image against the original image. For the 8 bit gray scale image the PSNR is computed using formula (5):

$$PSNR(dB) = 10.\log_{10}\left(\frac{255^2}{MSE}\right) \tag{5}$$

where $MSE = \frac{1}{D \times D} \sum_i \sum_j [I(i,j) - \bar{I}(i,j)]^2$, I and $\bar{I}$ are the original and watermarked image. Figure 2 shows the original image, marked image and extracted watermark.

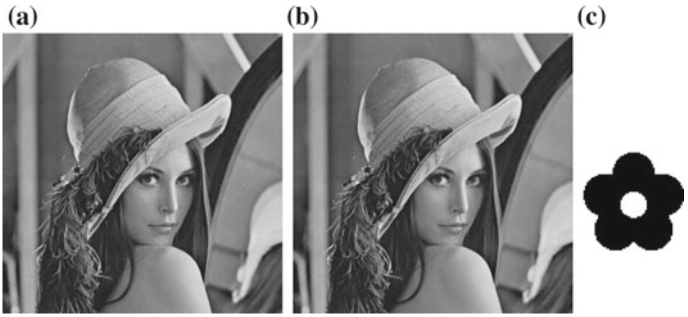

Fig. 2 **a** Original image. **b** Watermarked image with t = 1.5 (PSNR = 56.1787 dB). **c** Extracted watermark (NC = 1)

The PSNR value of the marked image is 55.8328 dB for $\alpha = .09$ and $t = 1.5$ which is sufficiently more than the desired value of 30 dB to make watermark imperceptible. The NC gives the correlation between the original and extracted watermark and it is 1 for Fig. 2c. Hence, the extracted watermark is same as the original one.

The comparison of the proposed scheme with and without block permutation is given in Table 1. From the table, it is clear that the detection of the watermark under various attack is equally good irrespective of the region of embedding when embedded in permuted blocks whereas without permutation the performance of the scheme varies when embedded in different region. Moreover, when watermark is embedded without permuting values in the block, the PSNR and NC values in the table shows, the quality of the watermark extracted from middle frequency region is low when compared to low frequency region. Table 1 also shows the comparison of the scheme with [16]. From the NCs of extracted watermark without permutation $(t = 127)$ and that of [16], it can be seen that multiple embedding and weighted average improves the performance of the extraction process. From the table, it is also clear that the proposed scheme preforms better than [16] under most of the attacks and in some cases it is equally good.

The algorithm is tested against various attacks such as jpeg compression, cropping, and by applying various noises and filters. The results are compared with [10] in Table 2, which shows that the performance is good against attacks such as cropping and jpeg compression whereas for other attacks the results show it is resistant against these attack. Figure 3 shows the results of the cropping attack where it can be seen that the extracted watermark is same as original watermark. The watermark extracted after distorting the watermarked image under various attacks is given in Fig. 4.

Table 1 Comparison of the proposed scheme with [16] (t = 127 and t = 3 represents the low and middle frequency regions, respectively)

	With permutation				Without permutation				Reference [16]
	t = 127		t = 3		t = 127		t = 3		
PSNR of watermarked image	55.7029 dB		50.0422 dB		52.2830 dB		83.2314 dB		
Attacks	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	NC
JPEG 80	59.1356	0.9182	59.4832	0.9350	68.4566	0.9898	67.0116	0.9844	0.9312
JPEG 40	56.8508	0.8682	57.2301	0.8730	61.2878	0.9495	61.0322	0.9416	0.8278
JPEG 20	55.9678	0.8321	55.8975	0.8478	58.2775	0.8977	57.5986	0.8851	0.7521
Gaussian low pass filter (7 × 7)	67.6268	0.9880	67.0116	0.9898	81.2441	0.9988	81.2441	1	0.7883
Average filter (5 × 5)	55.1375	0.7954	55.4178	0.8261	52.8368	0.6330	52.1272	0.5734	0.8034
Median filter (5 × 5)	57.7708	0.8863	57.8100	0.8911	58.5959	0.8989	58.0739	0.8941	0.8540
Gaussian noise (0, 0.01)	53.1284	0.7443	53.2645	0.7136	54.2849	0.8087	53.9041	0.7924	0.7804
Gaussian noise (0, 0.02)	52.8809	0.6721	52.8211	0.6703	53.6854	0.7768	53.6061	0.7232	0.5797
Salt and Pepper noise (0.01)	54.2327	0.8087	54.1472	0.8039	56.5459	0.8598	55.9937	0.8448	0.6458

Table 2 Comparison of the proposed scheme (t = 127) with [10] under various attacks

Attacks	Proposed scheme	[10]	Attacks	Proposed scheme	[10]
JPEG compression (CR = 7.92)	0.9681	1	Cropping (top-left corner)	1	0.9954
JPEG compression (CR = 12.2)	0.9561	0.9810	Cropping (top-right corner)	0.9994	0.9973
JPEG compression (CR = 15.54)	0.9278	0.9280	Cropping (bottom-left corner)	1	0.9924
JPEG compression (CR = 18.67)	0.9140	0.8847	Cropping (bottom-right corner)	0.9994	0.9981
Salt and pepper noise (0.01)	0.8087	0.8122	Median filtering 3 × 3	0.9368	0.9118
Gaussian noise, mean = 0, variance = 0.001	0.8670	0.8816	Histogram equalization	0.9537	0.9253

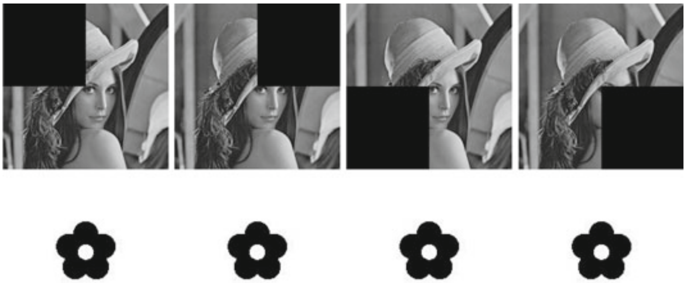

Fig. 3 The cropped images and the corresponding extracted watermark

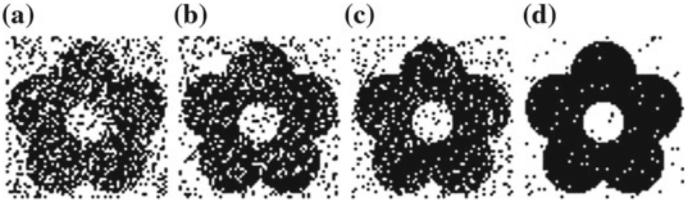

Fig. 4 The extracted watermark after **a** Salt and Pepper noise (0.01). **b** White Gaussian noise, mean = 0, variance = 0.001. **c** Median filter (3 × 3). **d** Histogram equalization

6 Conclusion

This paper introduces the techniques of embedding the watermark in the DCT of the permuted block. A watermarking scheme is proposed based on this technique. The simulation results show that embedding the watermark after permutation performs alike irrespective of the region of embedding. This overcome the restriction in selecting the region of embedding watermark and increases the region of embedding. Moreover, the performance of the scheme against JPEG compression, cropping, Gaussian noise, salt and pepper noise and median filter is equally good when compared with the existing schemes.

References

1. Chu, W.C.: DCT-based imagewatermarking using subsampling. IEEE Trans. Multimedia **5**(1), 34–38 (2003)
2. Eyadat, M., Vasikarla, S.: Performance evaluation of an incorporated DCT block-based watermarking algorithm with human visual system model. Pattern Recogn. Lett. **26**, 1405–1411 (2005)
3. Al-Haj, A.: Combined DWT-DCT digital image watermarking. J. Comput. Sci. **3**(9), 740–746 (2007)
4. Jiansheng, M., Sukang, L., Xiaomei, T.: A digital watermarking algorithm based on DCT and DWT. In: Proceedings of the International Symposium on Web Information Systems and Applications (WISA'09), pp. 104–107 (2009)
5. Zhu, G., Zhang, S.: Research and implementation of DCT-based image digital watermarking algorithm. In: IEEE Proceedings of the International Symposium on Electronic Commerce and Security, pp. 195–198 (2009)
6. Wei, Z, Ngan, K.N.: Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain. IEEE Trans. Circuits Syst. Video Technol. **19**(3), 337–346 (2009)
7. Xu, Z.J., Wang, Z.Z., Lu, Q.: Research on image watermarking algorithm based on DCT. Procedia Environ. Sci. **10**, 1129–1135 (2011)
8. Yesilyurt, M., Yalman, Y., Ozcerit, A.T.: A new DCT based watermarking method using luminance component. Elektronika ir Elektrotechnika **19**(4), 47–52 (2013)
9. Rosales-Roldan, L., Cedillo-Hernandez, M., Nakano-Miyatake, M., Perez-Meana, H., Kurkoski, B.: Watermarking-based image authentication with recovery capability using halftoning technique. Signal Process. Image Commun. **28**, 69–83 (2013)
10. Das, C., Panigrahi, S., Sharma, V.K., Mahapatra, K.K.: A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation. Int. J. Electron. Commun. (AEU) **68**, 244– 253 (2014)
11. Lin, S.D., Shie, S., Guo J.Y.: Improving the robustness of DCT-based image watermarking against JPEG compression. Comput. Stand. Interfaces **32**, 54–60 (2010)
12. Chang, C., Lin, P., Yeh, J.: Preserving robustness and removability for digital watermarks using subsampling and difference correlation. Inf. Sci. **179**, 2283–2293 (2009)
13. Patra, J.C., Phua, J.E., Bornand, C.: A novel DCT domain CRT-based watermarking scheme for image authentication surviving JPEG compression. Digital Signal Process. **20**, 1597–1611 (2010)

14. Taherinia, A.H., Jamzad, M.: A robust image watermarking using two level DCT and wavelet packets denoising. In: IEEE Proceedings of the International Conference on Availability, Reliability and Security, pp. 150–157 (2009)
15. Ali, M., Ahna, C.W., Pant, M.: A robust image watermarking technique using SVD and differential evolution in DCT domain. Optik **125**, 428–434 (2014)
16. Lu, W., Lu, H., Chung, F.: Robust digital image watermarking based on subsampling. Appl. Math. Comput. **181**, 886–893 (2006)