Permutation Based Invisible Digital Watermarking Technique using DCT Domain

Manish Choubisa
Research Scholar
Gyan Vihar University
Jaipur, India

Kamal Hiran
Research Scholar
Gyan Vihar University
Jaipur, India

S. K. Singh
Associate Professor
Gyan Vihar University
Jaipur, India

ABSTRACT
Digital watermarking is the modern idea in digital media for copyright protection. Many watermarking algorithm has been developed in recent years, but context of the purposes, as they serve, they contrast from each other. Here we propose algorithm of digital watermarking technique based on DCT (Discrete Cosine Transformation) using permuting the image. Through adjusting the block DCT coefficient of the image the watermarks are invisible. The images are first permuted and then converting into block allowing to 8×8 pixel and thus the watermark images are embedded through adjusting their DCT coefficient. The proposed paper also describes the experimental results that the method has strong robust.

General Terms
Watermarking, Image processing, Watermark embedding and Extracting, Permutation.

Keywords
Digital watermarking, DCT coefficient, PSNR and SM, MBEC

1. INTRODUCTION
In digital technology all types of multimedia digital products such as text, image, video, audio, digital repositories and libraries, web publishing image, video frequency and audio frequency, are released by network mode. Watermarks used to protect digital products with different algorithms and so research of digital watermark technology is rapidly developed.
The watermark is a digital code embedded in the image and used for the embedded transmission of binary information such that the watermark signal is unobstructed and secure in the digital mixture and never remove [1].
An efficient digital watermarking algorithm is the one which finds a good balance between invisibility and robustness [2]. The DCT is common transform for image process. The important feature of DCT that removing correlation makes it the perfect solution to our problem.
We apply DCT to permuted image and then embed the watermark. After this we can achieve a better balance between robustness and invisibility. To ensure robustness, the watermark image or information is usually redundantly distributed over many samples (pixels) or sub-images of the host data, thus providing "Holographic" robustness which means that the watermark can usually be recovered from a small fraction of the watermarked data. Usually robustness, imperceptibility and watermark capacity have to be traded against each other.
In general, watermark systems use one or more cryptographically secure keys to ensure security against manipulation and erasure of the watermark. Figure 1 shows the basic principal block diagram of watermarking.

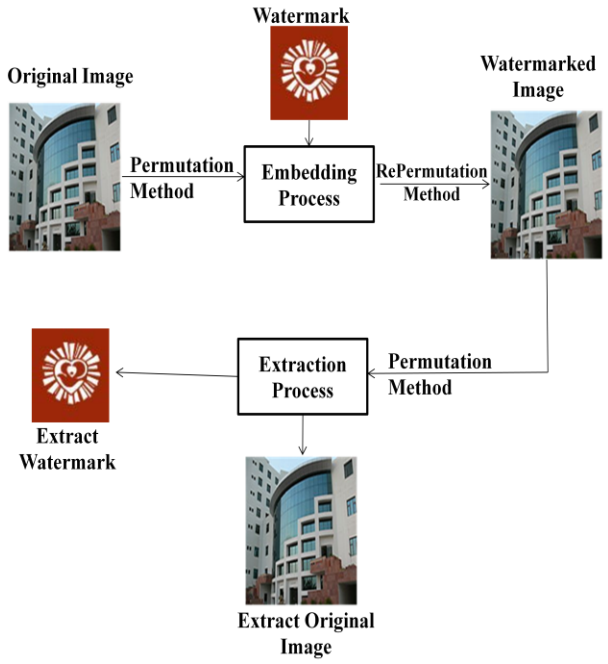

Figure 1: block diagram of watermarking process

Watermarking is easy to manipulate multimedia products and make its unauthorized duplication and distribution. This has resulted in copyright protection of digital contents on the internet and protecting rights of buyer and establishing his ownership of legal copy.

2. CONCEPT AND PRINCIPLE OF DCT BASED ALGORITHM
The DCT based on Mid Band Exchange Coefficient (MBEC) algorithm. DCT-based methods divide image into 8×8 sized blocks and then transform's image of size N×N into the DCT coefficients matrix with the same size [3]. MBEC use the one bit of binary watermark image. The MBEC watermarking algorithm encodes one-bit of the binary watermark image into one 8x8 DCT sub-block of the original image. If the difference of two mid-band coefficients is positive in case of the encoded value is '1' means the first coefficient is small then second coefficient then we encoded value is '1'. Otherwise, these two mid-band

coefficients are exchanged. Classical middle-band based algorithm is quite robust against JPEG compression and common image manipulation operations. The basic idea of the classical MBEC scheme was discussed in [4].

The middle-band frequencies (F_M) of an 8*8 DCT block can be shown below in figure 2. In 8x8 DCT block the middle-band frequencies region is denoted by FM. The lowest frequency component of the block is denoted by FL, while FH is used to denote the higher frequency components. FM is chosen as the embedding region as to provide additional resistance to lossy compression techniques, while avoiding significant modification of the cover image. Next, from the FM region, two locations P_i (u_1, v_1) and P_i (u_2, v_2) are chosen in the i^{th} DCT block for comparison.

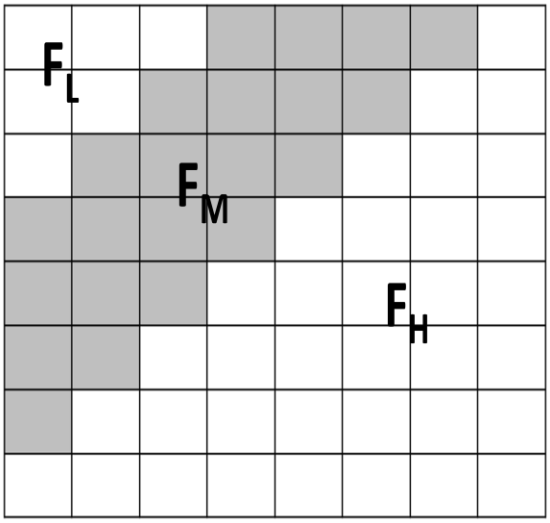

Figure 2: Definition of DCT Region

The choice in selecting the two locations is dependent on the content of the JPEG quantization table given below in table1. The two locations which have identical quantization values are chosen for embedding one watermark bit of information [5].

From the table the coefficients at (5, 2) and (4, 3) have value 22 or (2, 3) and (4, 1) have value 14, would make suitable candidates for comparison, as their quantization values are equal. The DCT block will encode a "0" if the value of the first pixel position is greater than or equal to the second pixel position other vise it will encode a "1". In other word if $P_i(u_1, v_1) \geq P_i(u_2, v_2)$ then DCT block a value "0" and if $P_i(u_1, v_1) < P_i(u_2, v_2)$ it will encode a "1". The coefficients are then swapped if the relative size of each coefficient does not agree with the bit that is to be encoded.

Table 1: Pixel Matrix used in transforms Domain

16	11	10	16	24	40	51	61
12	12	14	19	26	48	16	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	108	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

DCT-based methods which is used in this thesis, divide image into 8 × 8 sized blocks and then transforms image of size N × N into the DCT coefficients matrix with the same size [6][7]. The equations for two-dimension DCT and Inverse DCT (IDCT) are:

$$C(u,v) = a(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} a(u)a(v)C(u,v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

Here C is the DC transformed block, f is the inverse DCT block and u = 0,1,2, ...,K −1,v = 0,1,2, ...L−1; K and L are length and breadth of the image. a(u) and a(v) are defined in the following equation.

$$a(u) = \begin{cases} \frac{1}{\sqrt{K}}, & u=0 \\ \sqrt{\frac{2}{K}}, & 1 \leq u \leq K-1 \end{cases}$$

$$a(v) = \begin{cases} \frac{1}{\sqrt{L}}, & v=0 \\ \sqrt{\frac{2}{L}}, & 1 \leq v \leq L-1 \end{cases}$$

3. WATERMARK EMBEDDING AND EXTRACTING PROCESS

In Embedding process insert or embed the watermark information within the original image by modifying all or selected pixel values (spatial domain); or coefficients (frequency domain), in such a way that the watermark is undetectable to human eye and is achieved by minimizing the embedding distortion to the host image [8].
The system block diagram for the embedding process is shown in Figure 3.

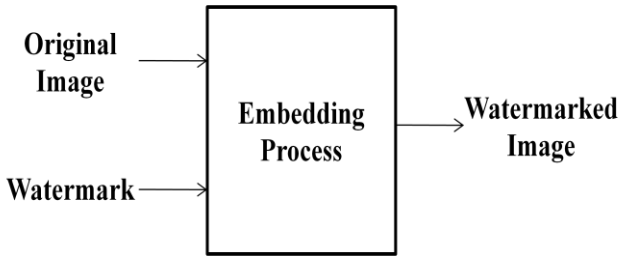

Figure 3: Watermark Embidding Process

The watermark extraction follows a reverse embedding algorithm, but with a similar input parameter set [9]. In this paper we used the DCT domain for Watermark embedding and extracting process on permuted image.

4. WATERMARK EMBEDDING ALGORITHM

The algorithm which is used to embed a watermark on an image is given below. We take input as Original Image and Watermark data and produce output as Watermarked Image.

Step1: Start
Step2: Read input gray scale Image in 256 × 256 standard and Watermark Data.
Step3: If the Watermark is small then it is padded with ones (1's) so the small watermark image scale up to the max message length for original image.
Step4: Permuted the original image using pseudo random sequence.
Step5: Perform DCT on Each 8*8 block of image and Embed the watermarking information into the (3,3) and (4,4) pixel of the 8x8 DCT coefficient block by classical coefficient exchange scheme. DCT of each block is calculated.
Step6: DCT co-efficient at the position say (3,3) and (4,4) are compared for every block. The DCT block will encode a "0" if pixel at position (3,3) is greater than or equal to the pixel at the position (4,4) otherwise it will encode a "1". The coefficients are then swapped if the relative size of each coefficient does not agree with the bit that is to be encoded.
Step7: Re-permuted the image.
Step8: Stop.

The insertion of the watermark in the mid band of the coefficient block of each averaged DCT block gives extra robustness to the watermark.

5. WATERMARK EXTRACTING ALGORITHM

For check the original image is watermarked or not, after embedding the watermark into original image we apply the watermark extracting algorithm. The algorithm which is used to extract a watermark is given below.

Step1: Start
Step2: Permuted the Watermarked Image pseudo random sequence.
Step3: Subdivide the Watermarked image into 8x8 sub-images using DCT domain.
Step4: DCT co-efficient at the position say (3,3) and (4,4) are compared for every block. If pixel at position (3,3) is greater than or equal to the pixel at the position (4,4) then Watermark bit hidden would be black or DCT block will encode a "0" else white or DCT block will encode a "1".
Step5: Stop.

6. SIMULATION RESULTS AND THEIR ANALYSIS

Two metrics for quality of watermarked images have been used which are Peak Signal to Noise Ratio (PSNR) and Similarity Factor(SM).
In order to test the performance of this watermarking scheme, we have used 256×256 gray scale images which are Lighthouse, Girl and Pepper. The original watermark is shown in figure 4. The watermarked images and the extracted watermark are shown in figure 5-7.
For image with 255 gray levels, the PSNR is defined as:

$$PSNR = 10\log_{10}\left(\frac{(255)^2}{MSE}\right)db$$

Where MSE is the mean square error of two images of N x N pixels is defined as

$$MSE = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(p_{ij} - p'_{ij}\right)^2$$

Where P_{ij} is the original pixel value and p'_{ij} is the reconstructed pixel value.
The similarity factor has value [0,1] calculated using following equation . If SM = 1 then the embedded watermark and the extracted watermark are same. Generally value of SM>.75 is accepted as reasonable watermark extraction.

$$SM = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N} W_M(i,j)W_M^*(i,j)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N} W_M(i,j)^2 \times \sum_{i=1}^{M}\sum_{j=1}^{N} W_M^*(i,j)^2}}$$

Where W_M is Original Watermark and W_M^* is detected watermark.

Figure 4: Original Watermark and extract watermark

Figure 5: Permuted Lighthouse image and Watermarked Lighthouse image

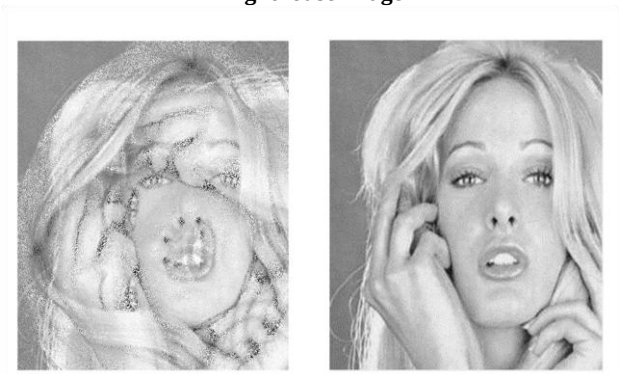

Figure 6: Permuted Girl image and Watermarked Girl image

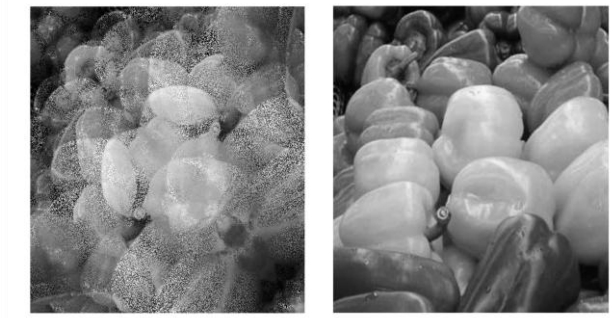

Figure 7: Permuted Pepper image and Watermarked Pepper image

The following Table 2 shows the PSNR of the different watermarked images and the Similarity factor (SM) of their extract watermarks.

Table 2: PSNR and SM values for different images

Image	Degree of Permutation	PSNR(db)	SM(db)
Lighthouse	100	40.36	0.89
	200	39.93	0.90
	400	36.13	0.97
Girl	100	43.94	0.99
	200	43.22	0.99
	400	40.10	1.00
Pepper	100	44.87	0.99
	200	40.79	0.99
	400	34.62	1.00

When PSNR is higher than 30, Watermarked image has a very good quality and the eye could hardly tell the difference between the original and the Watermarked image. While when SM is higher than 0.75, the extracted Watermarked is considered as valid one. From the above Table we can safely say that the watermarking schema discuss in this paper has a good invisibility and can extract the marks correct. Figure 8 show the graph between SM and different degree of permutation.

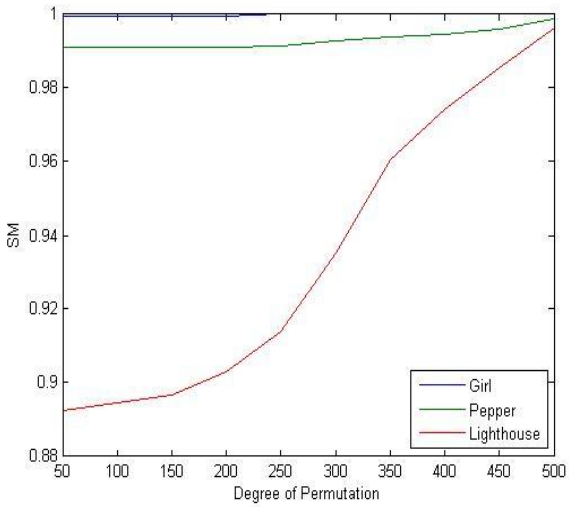

Figure 8: SM values of different watermarked images are increases using permutation

7. CONCLUSION

Digital Watermarking is the process of inserting watermark data into original images in a way that the degradation of quality is minimized and remain in an invisible level. Many digital watermarking algorithms have been proposed in special and transform domains. The techniques in the spatial domain still have relative low-bit capacity. On the other hand, frequency domain-based techniques can embed more bits for watermark

and are more robust to attack. In this paper, we use DCT transform watermark algorithms based on robustness.
We use the permutation method for more secure watermarking algorithm. The robustness of the watermarking methods has been measured by computing the Peak Signal to Noise Ratio (PSNR) of the Watermarked Image and Similarity between original Watermark and extract Watermark using Similarity Factor (SM).

8. ACKNOWLEDGEMENTS

We would like to express our gratitude to experts Professor Naveen Hemrajani, Dean (Engineering), Associate Prof. Vibhakar Pathak (HOD, IT Department) and Inder Pratap Singh (Asst.Professor, IT) for their guidance and contributions. We would also like to thank for the valuable informations they provided us. We would like to thank our family members for the love and care.

9. REFERENCES

[1] J. R. Hemandez, M. Amado, "DCT domain watermarking techniques for still images as detector performance analysis and a new structure," in IEEE Transactions on Image Processing, 2000, vol. 9, pp. 55-68.

[2] Q. Du, "Color image digital watermarking algorithm based on DCT and quantifying," in Journal of Soochow university, 2006, vol. 26, pp. 47-51.

[3] Z. M. Zhang, L. Wang, "Semiblind image watermarking algorithm in DCT domain with chaotic encryption," in Computer Engineering, 2003, vol. 29, pp. 10.

[4] L. S. Liu, R. H. Li, Q. Gao, "Method of embedding digital watermark into the green component of color image," in Journal of XianJiaotong university, 2004, vol. 38, pp. 1256-1259.

[5] N. Bourbakits and C. Alexopoulos, "Picture data encryption using scan patterns", Pattern Recognition, Vol. 25, No. 6, 1992, pp. 567–581.

[6] L. Wei, H. T. Lu, F. L. Chung, "Robust digital image watermarking based on subsampling," in Applied Mathematics and Computation, 2006, vol. 181, pp. 886-893.

[7] S. Z. Yu, "A color image-adptive watermark based on wavelet transform," in Computer Simulation, 2006, vol. 23, pp. 132-134.

[8] C. C. Chang and H. M. Tsai, "A generalized secret sharing scheme", Journal of Systems and Software, Vol. 36, No. 3, 1997, pp. 267–272.

[9] C. C. Chang, J. Y. Hsiao, and C. S. Chan, "Finding optimal LSB substitution in image hiding by using dynamic programming strategy", Pattern Recognition, Vol. 36, No.7, 2003, pp.1583–1595.