An Improved DCT based Steganography Technique

Deepika Bansal
Department of Computer Science and Engineering
ITM University, Gurgaon
Haryana, India

Rita Chhikara
Department of Computer Science and Engineering
ITM University, Gurgaon
Haryana, India

ABSTRACT
In this paper, a steganographic technique for hiding secret data in image file formats is proposed. This technique uses the Quantized DCT coefficients for hiding the secret information. The proposed steganographic method can provide a high information hiding capacity and successfully increase the security.

Keywords
Steganography, Least Significant Bit Insertion, Discrete Cosine Transform, Steganography Algorithms

1. INTRODUCTION
Steganography is the art and science of hiding the existence of the communication, i.e., it hides the secret message inside the other medium like images, audio, video, text, etc. Steganography word is derived from Greek word steganos, which means covered and graphia means writing [1]. The two files are required to embed the data in any medium. The first one is the cover file medium and the second one is secret message. The secret message can be any plain text, cipher text, or image. After embedding secret message in the cover file we obtains a stego file. The existence of secret message in the stego file cannot be predicted.

Cover Image + Message = Stego Image

There are various steganographic techniques used to hide the secret message. Throughout the history, various steganography techniques were being used, for example wax covered tablets, hidden tattoos, invisible inks, microfilms, microdots, null ciphers, etc [2]. There are two most widely used image steganography techniques:(i) Spatial Domain & (ii) Transform domain, shown in Fig. 1.

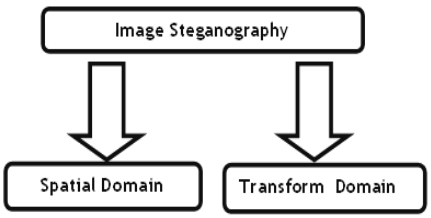

Fig. 1 Image Steganography Techniques

Spatial domain technique embeds secret bits directly in the cover file. The commonly used spatial domain technique is Least Significant Bit Insertion (LSB). In LSB, the secret bits are inserted in the least significant bits of cover image. LSB is of 2 types [3]: LSB Replacement & LSB Matching. In the LSB Replacement, the least significant bit of the carrier is replaced by the message bit directly. But in LSB Matching, if the least significant bit of the cover pixel is same as the message bit, then it remains unchanged, otherwise it is randomly incremented or decremented by one.

Transform domain [4] hides the secret bits in significant parts of the cover file. Transform Domain techniques try to encode message bits in the transform domain coefficients of the image. Data embedding performed in the transform domain is widely used for robust watermarking. Similar techniques can also realize large embedding capacity for steganography. The transform domain techniques include Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and Discrete Fourier Transform (DFT).

2. DISCRETE COSINE TRANSFORM
In Discrete Cosine Transform, for each color component the JPEG image format uses a discrete cosine transform to transform successive 8 x 8 pixel blocks of the image into 64 DCT coefficients each. The DCT coefficients *F(u,v)* of an 8 x 8 block of image pixels *f(x, y)* are given by [5]

$$F(u,v) = \frac{1}{4}C(u)C(v)[\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)cos\frac{(2x+1)u\pi}{16}cos\frac{(2x+1)u\pi}{16}$$

Where,

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}}, & if\ u \leq 0 \\ 1, & if\ u > 0 \end{cases}$$

The algorithm to embed the text message using DCT technique is as follows [6]:-

1. Read cover image.
2. Read secret message and convert it in binary.
3. The cover image is broken into 8×8 block of pixels.
4. Working from left to right, top to bottom, subtract 128 in each block of pixels.
5. DCT is applied to each block.
6. Each block is compressed through quantization table.
7. Calculate LSB of each DC coefficient and replace with each bit of the secret message.
8. Write stego image.

The algorithm to retrieve text message using DCT technique:-

1. Read stego image.
2. Stego image is broken into 8×8 block of pixels.
3. Working from left to right, top to bottom, subtract 128 in each block of pixels.
4. DCT is applied to each block.
5. Each block is compressed through quantization table.

6. Calculate LSB of each DC coefficient.
7. Retrieve and convert each 8 bits into a character.

The DCT block F consists of 64 DCT coefficients. The top-left coefficients F(0,0) correlates to lower frequency of the original image block, which is called DC coefficient. As we move away from the F(0,0) in all directions the DCT coefficients correlate to higher and higher frequencies, where F(7,7) corresponds to the highest frequency. A sample DCT block is shown in Fig. 2.

$$F = \begin{bmatrix} 162 & 40 & 20 & 72 & 30 & 2 & -1 & -1 \\ 30 & 108 & 10 & 32 & 27 & 5 & 8 & -2 \\ -94 & -60 & 12 & -43 & -31 & 6 & -3 & 7 \\ -38 & -83 & -5 & -22 & 3 & 5 & -1 & 3 \\ -31 & 17 & -5 & -1 & 4 & -6 & 1 & -6 \\ 0 & -1 & 2 & 0 & 2 & 2 & 8 & 2 \\ 4 & -2 & 2 & 6 & 8 & -1 & 7 & 2 \\ -1 & 1 & 7 & 6 & 2 & 0 & 5 & 0 \end{bmatrix}$$

Fig. 2 DCT block

Our 8 x 8 block of DCT coefficients is now ready for compression by quantization. A useful feature in the JPEG process in this step varying image compression and quality is obtainable through the selection of specific quantization table. The standard quantization matrix JPEG uses quality factor (α) 50 that as shown in Fig. 3. In another level of quality and compression desired, scalar multiples of the JPEG standard quantization matrix may be used. The scaled quantization matrix is then rounded and clipped to have positive integer values ranging from 1 to 255. For a quantity level greater than 50, less compression and high image quality are obtained. For a quantity level less than 50, more compression and low image quality are obtained.

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Fig. 3 Quantization Table

Quantization is achieved by dividing each element in the DCT coefficient block by the corresponding value in the quantization matrix, and the result is rounded to the nearest integer. The quantized DCT coefficients F^Q (u, v) is computed by

$$F^Q(u,v) = \left[\frac{F(u,v)}{Q(u,v)}\right]$$

Where Q(u,v) is a 64-element quantization table. The quantized DCT block and dequantized DCT block of Fig. 2 is shown in Fig. 4 and Fig. 5 respectively.

$$F^Q = \begin{bmatrix} 10 & 4 & 2 & 5 & 1 & 0 & 0 & 0 \\ 3 & 9 & 1 & 2 & 1 & 0 & 0 & 0 \\ -7 & -5 & 1 & -2 & -1 & 0 & 0 & 0 \\ -3 & -5 & 0 & -1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Fig. 4 Quantized DCT block

$$F^D = \begin{bmatrix} 160 & 44 & 20 & 80 & 24 & 0 & 0 & 0 \\ 36 & 108 & 14 & 38 & 26 & 0 & 0 & 0 \\ -98 & -65 & 16 & -48 & -40 & 0 & 0 & 0 \\ -42 & -85 & 0 & -29 & 0 & 0 & 0 & 0 \\ -36 & 22 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Fig. 5 Dequantized DCT block

In [7], the LSB, DCT and compression techniques are used to enhance the security of the payload and it is observed that secure images with low MSE and BER are transferred without using any password. The Binary DCT is demonstrated in [8], in which all coefficients are in binary and all multiplications are replaced by shifting and addition operations. Hence, the complexity of the transform is reduced by the Binary DCT algorithm. Integer DCT and Affine Transformations are used in [9] and ensures the stage image to be visually and statistically undetectable even with large payloads. An authentication technique for gray images using DCT is described in [10], in which the embedding algorithm applies DCT on a sub-image block called mask of size 2 x 2 of spatial components in row major order for the whole carrier image.

3. THE PROPOSED ALGORITHM – SHIELD ALGORITHM

The application is designed using steganography method that consists of two main processes, concealing and extraction of secret message from the image. The proposed method hides secret information in JPEG compressed images according to the Quantized DCT coefficients. Data embedding requires three steps: selecting DCT coefficients, information hiding and modification of quantization table. Here, not all DCT coefficients will be used for hiding secret data. We can observe that many DCT coefficients in high frequency areas tend to be zero after the quantization step in JPEG compression. The DCT coefficients that turn out to be non zero after quantization is selected for embedding secret information. Each block has 64 DCT coefficients, as F0 to F63.

The proposed algorithm (Shield Algorithm) is summarized below.

3.1 Concealing Algorithm

Input: Cover Image I, Secret Message
Input Parameters: Quantization Matrix (Q)
Output: Stego Image S
Begin

1. Read the cover image, I.
2. Divide the cover image, I into blocks of size 8 x 8.
3. Find the Discrete Cosine transformation of I.
4. Obtain the Quantized DCT blocks by dividing the DCT of I by the quantization matrix.
5. Hide the secret message in the Quantized DCT blocks according to the embedding capacity shown in the Table 1.
6. Obtain the dequantized matrix and inverse DCT.
7. Restructure the 8 x 8 blocks into a single array.
8. Stego image is formed.

End Shield Algorithm (SD)

Table 1. Embedding Capacity Range

Quantized DCT value	Number of Bits
2 – 8	1
9 – 16	2
17 – 31	3
32 – 64	4
>65	5

3.2 Extracting Algorithm

Input: Stego Image S
Input Parameters: Quantization Matrix (Q)
Output: Cover Image I, Secret Message
Begin
Read the stego image, S.

1. Divide the stego image, S into blocks of size 8 x 8.
2. Find the Discrete Cosine transformation of S.
3. Obtain the Quantized DCT blocks by dividing the DCT of S by the quantization matrix.
4. Extract the secret message from the quantized DCT blocks and concatenate DCT LSB to secret message.
5. Obtain the dequantized matrix and inverse DCT.
6. Restructure the 8 x 8 blocks into a single array.
7. Cover image and secret message are obtained.

End Shield Algorithm (SD)

The layout for the Shield algorithm is given in Fig. 6.

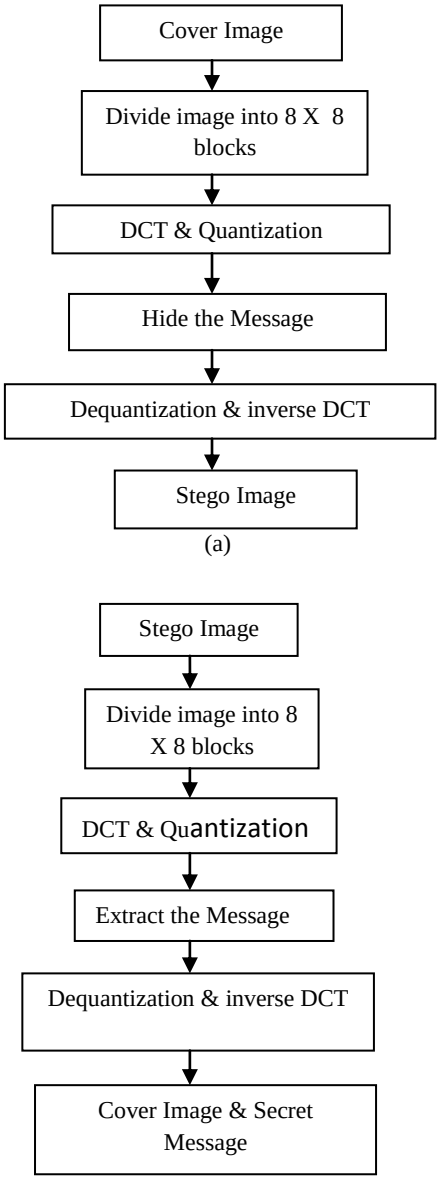

Fig. 6 (a)Layout for concealing the message using Shield Algorithm
(b)Layout for extracting the message using Shield Algorithm

4. EXPERIMENTAL RESULTS AND ANALYSIS

The proposed work is implemented in Matlab R2010a on 1000 natural images downloaded from www.1000pictures.com [11] and then resized to 640 x 480. They span an extensive range of landscape, architecture, animal, and people. For the set of 1000 natural images, we use our proposed algorithm and the two already available steganography methods, i.e., F5 and PQ, to produce different sets of stego-images. In all the results presented below, for all sets of stego-images and corresponding cover image sets, 750 images per set are chosen to train SVM classifier, and the remaining 250 images in each set are used to test. LibSVM [12] tool using radial basis function was employed for classification. Table II shows the classification accuracy results of our method for features extracted using SVM. The accuracy of Shield algorithm is 98.2% which is better than F5

having 98.6%. As can further be observed SVM classifies PQ with an accuracy of almost 90%, which is better than our proposed algorithm. Although from Table III showing PSNR values of the cover image shown in Fig. 7 and the corresponding stego images of F5, PQ and Shield algorithm in Fig. 8, 9 and 10, we can observe that Shield algorithm is giving better PSNR value 29.77 in comparison to the value 17.26 of F5 and 25.66 of PQ.

Fig. 7 Cover Image

Fig. 8 F5 Stego Image

Fig. 9 PQ Stego Image

Fig. 10 Shield Stego Image

Table 2. Classification accuracy obtained from various steganography tools

Tools	Classification Accuracy
F5	98.6
PQ	90
Shield	98.2

Table 3. PSNR values obtained from various steganography tools

Tools	PSNR Value
F5	17.26
PQ	25.36
Shield	29.77

5. CONCLUSION

Steganography is the art and science of hiding secret messages in the other file formats, so that the existence of the secret message is not revealed. In this paper, a steganography algorithm using DCT domain is proposed, Shield Algorithm. The two different tools are used to perform the analysis of images using the classification accuracy and PSNR, in comparison to the proposed algorithm. On the basis of above analysis, we can conclude that the Shield algorithm is giving better PSNR results from F5 and PQ, and better classification accuracy is obtained than F5 steganography tool.

6. REFERENCES

[1] N.F.Johnson, S.Jajodia, Exploring Steganography: Seeing the Unseen IEEE Computer 31(2) (1998)26-34.

[2] Kh. Manglem Singh, S.Birendra Sigh and L. Shyam Sundar Singh, Hiding Encrypted Message in the Features of Images, IJCSNS, Vol.7 No. 4, April 2007, pp 302-307.

[3] W.-N. Lie and L.-C. Chang, Data hiding in images with adaptive numbers of least significant bits based on human visual system, in Proc.,IEEE Int. Conf. Image Processing, 1999, Page(s): 286–290.

[4] S. Katzenbeisser and F. A. P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking. Norwood, MA: Artech House, 2000.

[5] D.R. Denslin Brabin, Dr.V.Sadasivam, QET Based Steganography Technique for JPEG Images.

[6] Stuti Goel, Arun Rana & Manpreet Kaur, "A Review of Comparison Techniques of Image Steganography", Global Journal of Computer Science and Technology Volume XIII Issue IV Version I, 2013, pp. 8-14.

[7] K.B.Raja, C.R.Chowdary, Venugopal K R, L.M.Patnaik, "A Secure Image Steganography using LSB, DCT and Compression Techniques on Raw Images".

[8] Wei Zheng, Yanchang Liu, "Research in a Fast DCT Algorithm Based on JPEG", IEEE, 2011, pp.551-553.

[9] Xianhua Song, Shen Wang, Xiamu Niu, "An Integer DCT and Affine Transformation Based Image Steganography Method", Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2012, pp. 102-105.

[10] Anirban Goswami, Dipankar Pal, Nabin Ghoshal, "Authentication Technique for Gray Images Using DCT (ATGIDCT)", 2012 Third International Conference on Emerging Applications of Information Technology (EAIT), pp. 421-424.

[11] Image Source: www.1000pictures.com

[12] LibSVM ToolBox Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/