Data Security Using LSB & DCT Steganography In Images

Deepak Singla[1], Rupali Syal[2]

Department of Computer Sci. & Engineering[1], Department of Information Technology[2] PEC University of Technology, Sector-12 Chandigarh, India

Abstract

Steganography is a technique of hiding information in digital media. In contrast to cryptography, it is not to keep others from knowing the hidden information but it is to keep others from thinking that the information even exists.In this paper, we propose a LSB & DCT-based steganographic method for hiding the data. Each bit of data is embedded by altering the least significant bit of low frequency DCT coefficients of cover image blocks. There are some techniques to utilizes the idea of SSB-4 technique in modifying the other bits (i.e, 1st , 2nd, 3rd and/or 4th), to obtain the minimum variation between the original and the modified coefficient. The experimental results show that this algorithm has better PSNR value and high capacity in comparison to other techniques such as LSB, modulus arithmetic, SSB4-DCT. It also maintains satisfactory security as secret message cannot be extracted without knowing the decoding algorithm. This is achieved using a Public Private key. It combined both feature of Steganography and cryptography.

Keywords: Authentication, Discrete Cosine Transform, Privacy, Steganography, Zigzag scanning.

1. Introduction

Security of information is one of the most important factors of information technology and communication. Security of information often lies in the secrecy of its existence and/or the secrecy of how to decode it. Cryptography techniques often use the worst approach assuming that only one of these two conditions holds [5]. It was created as a technique for securing the secrecy of communication. Various methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately, it is not enough to keep the content of the information/message secret, it may also be necessary to keep the existence of the information secret. The technique used to implement this, is called steganography.

Steganography is the art of hiding information in plain sight. Looking at data in transmission it is very easy to detect if its encrypted or not. [28]. To hide information, straight message insertion may encode every bit of information in the image or selectively embed the message in noisy areas that draw less attention those areas where there is a great deal of natural color variation. The message may also be scattered randomly throughout the image. Redundant pattern encoding wallpapers the cover image with the message. A number of ways exist to hide information in digital images. We have concentrated on some techniques and methods which are divided into two types: Spatial Domain and Frequency Domain.

1.1. Spatial domain steganography

Spatial domain techniques embed messages in the intensity of the pixels directly [6][7][8]. Least Significant Bit (LSB) is the first most widely used spatial domain steganography technique. It embeds the bits of a message in the LSB of the image pixels [9][10]. But the problem with this technique is that if the image is compressed then the embedded data may be lost. Thus, there is a fear for loss of data that may have sensitive information [11]. LSB has been improved by using a Pseudo Random Number Generator (PRNG) and a secret key in order to have private access to the embedded information [12]. The embedding process starts with deriving a seed for a PRNG from the user password and generating a random walk through the cover image that makes the steganalysis hard. Another recent improvement based on random distribution of the message was introduced by M. Bani Younes and A. Jantan [13]. Modulus arithmetic steganography proposed by Sayuthi Jaafar and Azizah A Manaf has calculated last four bits of each pixel by mod-16 operation. Then these bits are replaced with data bits [8]. In this the amount of the data that can be embedded is more but stego image has less PSNR value than LSB and SSB-4 techniques.

1.2. Frequency domain steganography

In frequency domain, images are first transformed and then the message is embedded in the image [17][18][19]. When the data is embedded in frequency domain, the hidden data resides in more robust areas, spread across the entire image, and provides better resistance against statistical attacks. There are many techniques used to transform image from spatial domain to frequency domain. The most common frequency domain method usually used in image processing is the 2D discrete cosine transform [20][21]. In this technique the image is divided into 8×8 blocks and DCT transformation on each block is performed. DCT arranged the pixel of image according to their frequency value. The data bits are embedded in the low frequency coefficients of DCT. SSB-4 & DCT steganography proposed by Nedal M. S. Kafri and Hani Y Suleiman uses DCT approach with SSB-4 technique [21]. But in this stego image PSNR value is not so high. To improve it, a novel LSB & DCT based steganographic method for is proposed in this paper, which can not only preserve good image quality, but also resist some typical statistical attacks.

2. Proposed steganography method

The challenge in this work was to find a way to camouflage a secret message in an image without perceptible degrading the image quality and to provide better resistance against steganalysis process. Therefore, a combination of frequency domain by means of DCT and LSB technique of spatial domain steganography has been used to hide data. Two dimensional DCT converts the image block from spatial domain to frequency domain and then data bits are embedded by altering LSB of DCT coefficients is shown in fig.1.

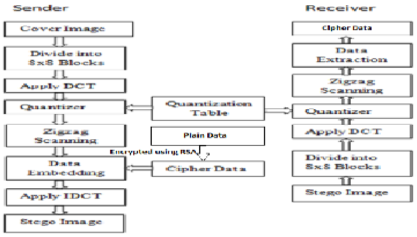


Figure: 1. Block diagram of LSB-DCT steganography

2.1. Discrete Cosine Transform

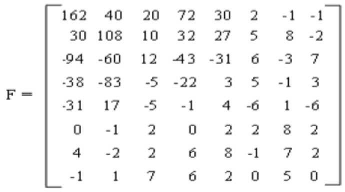
The image of size M×N is divided into 8×8 blocks and two dimensional (2-D) DCT is performed on each block. The DCT is calculated using equation 1:

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

for x=0,..., 7 and y=0,..,7

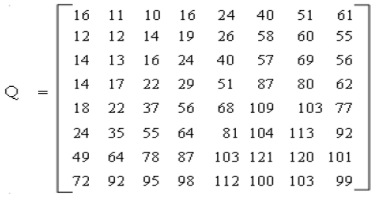
$$\text{where } C(k)=\begin{cases}1/\sqrt{2} & \text{for } k=0\\ 1 & \text{otherwise}\end{cases}$$

In DCT block lower frequency cofficents are at upper left positions and high frequency coefficients are lower right positions. Low frequency coefficients are of larger value than high frequency coeffcients. An example of a 8×8 block of DCT cofficient is shown in fig. 2.



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

Figure: 2. DCT coefficient



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

Figure: 3. Quantization Matrix

2.2. Quantization

The 8 x 8 block of DCT coefficients is compressed by quantization. A useful feature in this process is the image compression and quality is obtainable through selection of specific quantization table. The standard quantization matrix [27] is shown in fig. 3.

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. As eye is not able to discern the change high frequency components so these can be compressed to larger extent. Lower right side components of quantization matrix are of high value so that after quantization high frequency components become zero. The quantized DCT coefficients matrix P is computed by equation (2) and shown in fig. 4.

P (u, v) = F (u, v) / Q (u, v) (2)

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

Figure: 4.Quantized DCT Block

$$\begin{bmatrix} 0 & 1 & 5 & 6 & 14 & 15 & & \\ 2 & 4 & 7 & 13 & 16 & & & \\ 3 & 8 & 12 & 17 & & & & \\ 9 & 11 & 18 & & & & & \\ 10 & 19 & & & & & & \\ 20 & & & & & & & \\ & & & & & & & \end{bmatrix}$$

Figure: 5. Zigzag scan order

2.3. Zigzag Scanning

Although the DCT coefficients have been decorrelated by DCT transform to some extent, DCT coefficients in the same block are still not independent, which is called as intra-block correlation [16]. While neglecting the impact of block edge, the general trend in magnitude of the block coefficients in each block is non-increasing along *zigzag* scan order. After block DCT coefficients are arranged by zigzag scan pattern, dependencies among neighboring coefficients in both horizontal and vertical directions can be conveniently investigated [23]. For example, the sequence pairs (5,6) and (14,15) describe the correlations along horizontal direction, while the sequence pairs (2,3) and (10,20) describe correlations along vertical direction. Zigzag scan converts 8×8 block into 64 elements one dimensional array.

2.4. Data embedding

Data bits are concealed by altering the LSB of elements of zigzag array.

a) If data bit is '0', then make the DCT coefficient even or,

b) If the data bit is '1', then make the DCT coefficient odd.

2.5. Dequantization and inverse DCT

After embedding data zigzag array is again converted into 8×8 block. These blocks are dequantized and inverse DCT is performed. The entire 8×8 blocks are combined to form the stego image which is then sent to receiver. Complete embedding algorithm is given as follow:

Input: An M×N size cover image and data to be concealed.
Output: Stego image.
Step 1: Divide the cover image into 8×8 blocks. Step 2: Perform 2-D DCT on each block.
Step 3: Perform quantization on each block.
Step 4: Perform zigzag scan to convert 8×8 block into one dimensional array.
Step 5: Replace the LSB of DCT coefficients with data bits. Step 6: Convert 1-D zigzag array back to 8×8 block.
Step 7: Perform Inverse DCT on each block.
Step 8: Combine all the blocks to form stego image.

2.6. Extraction of secret message

The stego-image is received in spatial domain. Now stego image is divided into 8×8 blocks and DCT is performed on each block. Then scan the DCT block in zigzag way and extract the embedded data. Extraction algorithm is given as follows:

Input: Stego image.
Output: Secret message.
Step 1: Divide the stego image into 8×8 blocks.
Step 2: Perform 2-D DCT on each block.
Step 3: Perform quantization on each block.
Step 4: Perform zigzag scan to convert 8×8 block into one dimensional array.
Step 5: Check the DCT coefficient.
a) If DCT coefficient is even then data bit is 0 or,
b) If DCT coefficient is odd then data bit is 1.
Step 6: Concatenate the bits to obtain secret message and display it on screen.

2.7. Proposed Work

Our work also deals with the security of text messages at the time of sending it over the network. In our algorithm, we have used asymmetric key (RSA method) in cryptography which means public and private keys are needed to encrypt and decrypt the data. Encryption and decryption provide privacy. With RSA, message is encrypted using public and private keys of Sender and receiver. Encrypted message is hide into images using LSB & DCT.

3. Experimental Results

Since the visual detection of stego images is depending on the nature of the image [24], so, varieties of image categories are utilized in the experiments. In order to have significant results. Images are divided into five categories: Tree, Dog, Monkey, Flower and Mountains. The experimental image data set consists of 60 JPEG images, 12 images for each category were used in experiments, which were taken by digital camera. We focused on short messages with length of 1500 bits because they are the most challenging to detect [24]. In addition to proposed steganography and cryptography techniques, for comparison purposes. In order to evaluate the quality of stego file/Images shown in fig. 6, generated by compared techniques.

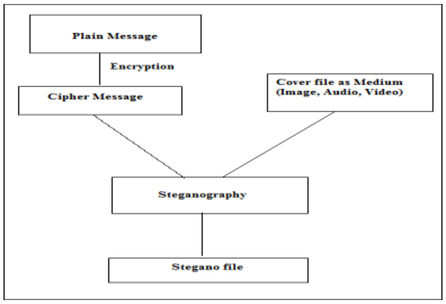


Figure: 6. stego files/Images

	PSNR Value			
	Modulus	***LSB-1st***	***LSB-4th bit***	***LSB-DCT***
Mountain.j	46.32	60.10	52.73	55.43
Dog.jpg	48.54	62.34	53.43	57.48
	47.69	61.34	51.85	55.78
Flower.jpg	48.64	63.46	52.57	56.51
Tree.jpg	47.33	63.59	52.41	57.44

Table 1. Comparative analysis of PSNR values of different steganography techniques

In this paper the stego image qualities are represented by Peak signal to noise ratio (PSNR). The implementations of the compared techniques (Modulus arithmetic (mod-16), 1- LSB, 4-LSB, and DCT with LSB) and the PSNR tests were carried out using MATLAB is a numerical computing environment and fourth-generation programming language. To calculate PSNR, first MSE is calculated using equation 3:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|I(i,j)-K(i,j)\|^2 \qquad (3)$$

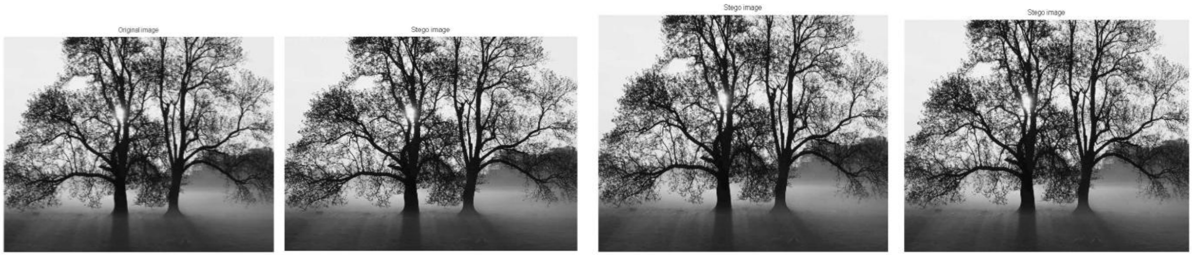
Where MSE is the Mean Squared Error of Original image (I) and stego image (K). Thereafter PSNR value is calculated using equation 4:

$$PSNR = 10.log_{10}\left(\frac{MAX_i^2}{MSE}\right) = 20.log_{10}\left(\frac{MAX_i}{\sqrt{MSE}}\right) \qquad (4)$$

Where, MAX_i is the maximum pixel value of the image. In other words $MAX_i = 2^b - 1$, where b is the bit depth of the original image (e.g., $MAX_i = 255$ in the case of 8 bits depth grayscale images). PSNR computes the peak signal to noise ratio, in decibels, between two images. This ratio is used as a quality measurement between two images. The comparative analysis of PSNR value of different steganography technique, given in table 1, shows that LSB-DCT steganography has better image quality of stego image than other techniques.

4. Discussion

The obtained experimental results is shown in fig 7. indicate that, the proposed method will be a good and acceptable steganogaphy scheme. Random steganography using LSB & DCT with asymmetric keys gives us more security than simple LSB & DCT method, where it is difficult to identify the hidden data in the stego image at specific location.



Original Image After Insertion Using LSB-1st bit After Insertion Using LSB-4th bit After Insertion Using LSB & DCT
Average (PSNR=64.59dB) Average (PSNR = 52.41 dB) Average (PSNR=57.44 DB)

Figure 7. Original tree Image,Stego Images Using 1-LSB,4-LSB and proposed scheme LSB and DCT Steganography

5. Conclusion

In this paper a mixed approach that applies the spatial domain with the frequency domain of steganography techniques and Asymmetric key cryptography. The idea is to utilize a significant bit of the DCT coefficients of a cover image to hide encrypted message bits. Thereafter, the information and the variation of the coefficients, affected by the embedding process, are spread in the stego image by utilizing the inverse of the DCT process. Steganography has its place in security.

References

[1] DES Encryption Standard (DES), National Bureau of Standard (U.S.).Federal Information Processing Standards Publication 46, National Technical Information Service, Springfield, VA, 1997.

[2] Daemen J., and Rijmen, V. “Rijndael: The Advanced Encryption Standard”, Dr. Dobb’s Journal, March 2001.

[3] R. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems”. Communication of the ACM, pp. 120-126, 1978.

[4] Pfitzmann, B. “Information hiding terminology”, Proc. First Workshop of Information Hiding Proceedings, Cambridge, U.K., Lecture Notes in Computer Science, Vol.1174, pp. 347-350, 1996.

[5] H & Wang, S, “Cyber warfare: Steganography vs. Steganalysis”, Communications of the ACM, 47, pp.10-12, October 2004

[6] Chan, C.K. and Cheng. L.M. “Hiding data inimage by simple LSB substitution. Pattern Recognition”, 37, pp. 469 – 474, 2004.

[7] Chang,C.C and Tseng, H.W. “A Steganographic method for digital images using side match”. Pattern Recognition Letters, 25, pp. 1431 –

1437, 2004.

[8] Sayuthi Jaafar, Azizah A Manaf, Akram M Zeki, “Steganography Technique using Modulus Arithmetic”, 9th International Symposium on Signal Processing and Its Applications, pp. 1 – 4, April 2007.

[9] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, “Techniques for Data Hiding”, I.B.M. Systems Journal, 35(3-4): pp. 313-336, 1996.

[10] N. Nikolaidis, and I. Pitas, “Robust Image Watermarking in the Spatial Domain”, Signal Processing, 66(3), pp. 385-403, 1998

[11] T. Morkel, J. Eloff, and M. Olivier, “An overview of image steganography”, In Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), (Sandton, South Africa, Jun/Jul. 2005).

[12] J. Fridrich, M. Goljan, “ Steganalysis of JPEG Images: Breaking the F5 Algorithm”, Publisher: Springer Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 2578, pp 310-323, 2003.

[13] M. A. Bani Younes, A. Jantan, “A New Steganography Approach for Image Encryption Exchange by Using the Least Significant Bit Insertion”, IJCSNS, International Journal of Computer Science and Network Security, Vol. 8 No. 6, June 2008.

[14] J. Rodrigues, J. Rios, and W. Puech “SSB-4 System of Steganography using bit 4”, In International Workshop on Image Analysis for Multimedia WIAMIS, May, 2005.

[15] J. Fridrich, and M. Goljan, “Practical steganalysis: state-ofthe-art”, In Proceeding of SPIE Photonics West, Electronic Imaging 2002, volume 4675, pp. 1-13, 2002.

[16] Tu C. and Tran T D. “Context based entropy coding of block transform coefficients for image compression”, IEEE Transaction on Image Processing, Vol.11, No.11, November, 2002.

[17] Wenqiong Yu, “Blind Detection for JPEG Steganography”, International Conference on Networking and Information Technology , pp. 128-132, July 2010.

[18] Chung, K.L., Shen, C.H. and Chang, “A novel SVD- and VQ-based image hiding scheme”.Pattern Recognition Letters, 22, pp. 1051 – 1058, 2001.

[19] Iwata M., Miyake K., and Shiozaki, “Digital Steganography Utilizing Features of JPEG Images”, IEICE Transfusion Fundamentals, E87-A, 4, pp. 929 – 936, 2004.

[20] M. Kharrazi06, H. Sencar, and N. Memon, “Performance study of common image steganography and steganalysis techniques,” Communications of the SPIE and IS&T, 15, No.4, pp. 1017-9909, Oct- Dec., 2006.

[21] Nedal M.S. Kafari, Hani Y. Suleiman, “Bit-4 of Frequency Domain DCT Steganography Technique”, FIRST NATIONAL CONFERENCE ON NETWORKED DIGITAL TECHNOLOGIES, PP. 286-291, 2009.

[22] Dr. Ekta Walia, Payal Jain, Navdeep, “An Analysis of LSB & DCT based Steganography”, Global Journal of Computer Science and Technology, Vol. 1, pp. 4-8, April, 2010.

[23] Zhiping Zhou and Maomao Hui, “Steganalysis for Markov Feature of Difference Array in DCT Domain”, Proceedings of Sixth International

Conference on Fuzzy Systems and Knowledge Discovery, pp. 581- 584 , Aug. 2009.

[24] L. Davidson, and P. Goutam, “Locating secret message in images”, In ACM SIGKDD international conference on Knowledge discovery and data mining, (Seattle, Washington, Aug.22-25. ACM 1-58113-888-1, 2004.

[25] Gonzalez, R.C. and Woods, R.E., Digital Image Processing using MATLAB, Pearson Education, India,2006.

[26] Jayaraman, S., Esakkirajan, S. and Veerakumar, T. Digital Image Processing, Tata McGraw Hill Education Private Limited, India, 2009.

[27] www.vision.arc.nasa.gov/dctune/

[28] S. Dickman, An Overview of Steganography, Research Report JMU- INFOSEC-TR -2007-002, James Madison Univer- sity, July, 2007.