ISSN:0975-9646

Image Steganography: A Review

Ayush Purohit, P.S.V.S.Sridhar

CIT, CoES, University of Petroleum & Energy Studies, Dehradun, India

Abstract **- One of the reasons that attackers are being successful is that most of the information they acquire from a system is in a form that they can read and access easily. Attackers may then reveal, modify or misuse the information. One of the solutions to this problem is, through the use of steganography. In this paper, we'll discuss an overview of Image steganography, its uses and techniques. We'll also discuss the implementation of these steganographic techniques and evaluate them on the basis of their performance.**

Keywords*:* Steganography, Information Hiding, Image Steganography, Spatial Domain, Transform Domain

I. INTRODUCTION

Due to advances in technology, most of information in now-a-days are kept electronically. Consequently, the security of information has become a major issue.

This is particularly important in the case of confidential data. The need to solve this problem has led to the development of steganography schemes. Steganography is a powerful security tool that provides a high level of security, particularly when it is combined with encryption.

Steganography is one of the methods which are used to exchange the information secretly and it can be defined as the art and science of communicating in a way which hides the existence of the communication. Paired with existing communication methods, steganography can be used to carry out hidden exchanges.

Classical steganography concerns itself with ways of embedding a secret message in a cover message. The embedding is typically secured by a key to increase the security level, without knowledge of this key it is difficult for a third party to detect or remove the embedded material.

Steganography has two primary goals:

1) Security – is the hidden data perceptible by either a person or a computer, and

2) Capacity – how much data can be hidden in a given cover file.

These two goals are often in competition. The more data you hide, the more likely it is to be found, i.e. it has less security and vice versa.

Essentially, the information hiding process in a steganographic system starts by identifying a cover medium's redundant bits. The embedding process creates a *stego medium* by replacing these redundant bits with data from the hidden message.

II. STEGANOGRAPHY VS CRYPTOGRAPHY

Basically, the purpose of cryptography and steganography is to provide secret communication. Most of the people confuse steganography with cryptography as same. However, steganography is different from cryptography.

Cryptography is about concealing the content of messages i.e., to provide secure communications by changing the data into a form so that it cannot be understand by any unauthorized person. Cryptography can also provide authentication for verifying the identity of someone or something. Whereas, steganography is about concealing their existence. Steganography techniques tend to hide the existence of the message itself, which makes it difficult for an observer to figure out where exactly the message is.

In cryptography, the structure of a message is scrambled to make it meaningless and unintelligible unless the decryption key is available. Basically, cryptography offers the ability of transmitting information between persons in a way that prevents a third party from reading it.

In contrast, steganography does not alter the structure of the secret message, but hides it inside a *cover-image* so it cannot be seen. A message in cipher text, for instance, might arouse suspicion on the part of the recipient while an "invisible" message created with steganographic methods will not.

In other words, steganography prevents an unintended recipient from suspecting that the data exists [9]. In addition, the security of classical steganography system relies on secrecy of the data encoding system.

Steganography and cryptography are both ways to protect information from unauthorized users but neither technology alone is perfect and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated. The strength of steganography can thus be amplified by combining it with cryptography.

III. IMAGE STEGANOGRAPHY

An image can be defined as an arrangement of numbers (pixels) and these pixels represents different color intensities in different parts of the image [1]. Images are considered as the most popular file formats used in steganography. They are known for constituting a non-causal medium, due to the possibility to access any pixel of the image at random.

In addition, the hidden information could remain invisible to the eye. However, the Image steganography techniques will exploit "holes" in the Human Visual System (HVS).

IV. IMAGE STEGANOGRAPHY MODEL

The basic model of image steganography consists of *Carrier*, *Message* and *Password*. Carrier is also known as *cover-image*, which the message is embedded and serves to hide the presence of the message.

A basic steganographic model is shown in figure 1 and figure 2. The process of embedding can be explained as:

Let '*C*' represents the cover image and '*Z*' the stego-image.

Let '*K*' be a stego-key (as a seed used to encrypt the message, which can be set to {ϕ} for simplicity), and let

'*M*' be the message that the sender wishes to send. Then, 'E_m' represents an embedded message and 'E_x' represents the extracted message. Therefore,

$E_m : X \oplus K \oplus M \rightarrow Z$

$\therefore E_x (E_m (x, k, m)) \approx m,$

where, $x \in X$, $k \in K$, $m \in M$

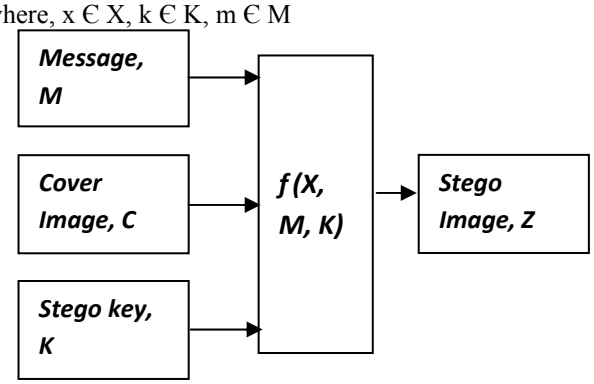

Fig 1: Basic Digital Steganography Encoder[8]

Recovering message from a *stego-image* requires the *cover-image* itself and a corresponding decoding key if a *stego-key* was used during the encoding process. The original image may or may not be required in most applications to extract the message.

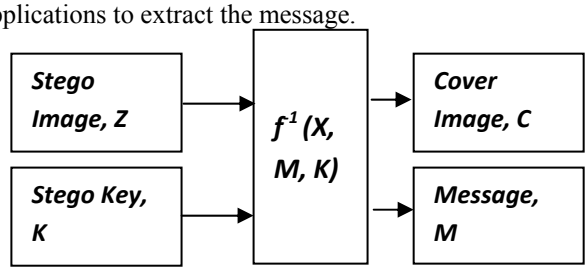

Fig 2: Basic Digital Steganography Decoder

V. STEGANOGRAPHY TECHNIQUES

The Image Steganographic Algorithms proposed in this article can broadly be classified into two categories:

1. Spatial Domain Techniques
2. Transform Domain Techniques

Some of these techniques are covered in detail in the next two subsections.

A. Spatial Domain Techniques

Steganography techniques that modify the cover image and the secret image in the spatial domain are known as spatial domain methods. [2] Spatial domain steganographic techniques, also known as substitution techniques, are a group of relatively simple techniques that embed messages in the intensity of the pixels directly.

This embedding method is basically based on the fact that the significant bits in an image can be thought of as random noise, and consequently they'll not become responsive to any changes on the image.

Some of the spatial domain techniques are given below which are being used worldwide:

i) Least Significant Bit Substitution:

LSB is the most commonly used stenographic technique. The basic concept of LSB Substitution includes the embedding of the secret data at the bits which having minimum weighting so that it will not affect the value of original pixel [6]. In this method binary equivalent of the message (to be hidden) is distributed among the LSBs of each pixel.

The simplest of the LSB steganography techniques is LSB replacement. LSB replacement steganography flips the last bit of each of the data values to reflect the message that needs to be hidden.

ii) Pseudorandom permutation:

Pseudorandom permutation is another substitution technique. This technique includes the embedding of secret message bits over the whole cover in a random manner [10]. It thus increases the complexity, since it is not guaranteed that message bits are embedded in same order. However, in this article we'll discuss LSB substitution implementation in detail, in the next section.

The major drawbacks of these methods are the amount of additive noise that creeps in the image which directly affects the Peak Signal to Noise Ratio and the statistical properties of the image. Moreover, these embedding algorithms are applicable mainly to lossless image-compression schemes.

iii) Image downgrading:

Image Downgrading is a special case of substitution system in which images act as both as secret message and covers. Given the cover image and a secret image of equal dimensions, the sender exchanges the 4 LSBs of the cover's grayscale (color) values with the 4 MSBs of the secret message. The receiver extracts the 4 LSBs out of the stego image, thereby gaining access to the MSBs of secret message.

B. Transform Domain Techniques

Steganography in the transform domain involves the manipulation of algorithms and image transforms. The process of embedding data in the frequency domain of a signal is much stronger than embedding principles that operate in the spatial domain.

These methods hide messages in more significant areas of the cover image, making it more robust to attacks. Data embedding performed in the transform domain is widely used for robust watermarking.

Many transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression. Some of the transform domain techniques are given below:

i) Discrete Cosine Transform (DCT):

DCT transforms the image from spatial domain to frequency domain. It separates the image into spectral sub-bands with respect to its visual quality, i.e. high, middle and low frequency components.

In DCT based technique insertion of secret information in carrier depends on the DCT coefficients. Any DCT coefficient value above proper threshold is a potential place for insertion of secret information**.** DCT coefficients are obtained for the given carrier image. The secret data is embedded in the carrier image for DCT coefficients lower than the threshold value.

A lossless and reversible steganography scheme has been introduced that use each block of quantized discrete cosine transformation (DCT) coefficients in JPEG images for embedding secret data [4]. In this scheme, the two

successive zero coefficients of the medium-frequency components in each block are used to hide the secret data. This method results in a high image quality of stego image and successfully achieves reversibility.

ii) Discrete Wavelet Transform (DWT):

Discrete Wavelets Transform (DWT) is used in the image steganographic model as it clearly partitions the high-frequency and low-frequency information on a pixel by pixel basis, which are also favored in approximating data with sharp discontinuities.

The discrete wavelet transform (DWT) method is favored over the discrete cosine transform (DCT) method, owing to the resolution that the WT provides to the image at various levels. Wavelets are mathematical functions that divide data into frequency components hence, makes them ideal for image compression techniques.

We'll discuss the implementation of DCT in detail, in the next section.

VI. STEGANOGRAPHY IMPLEMENTATION

A. Least Significant Bit Substitution

Least significant bit insertion is a common, simple approach to embedding information in a cover image.The least significant bit (in other words, the 8th bit) of some or all of the bytes inside an image is changed to a bit of the secret message. When using a 24-bit image, a bit of each of the red, green and blue color components can be used, since they are each represented by a byte. In other words, one can store 3 bits in each pixel. An 800×600 pixel image, can thus store a total amount of 1,440,000 bits or 180,000 bytes of embedded data.[5]

Algorithm:-

i) Embedding phase:

The embedding process is as follows:

Step 1: Extract all the pixels in the given image and store it in the array called Pixel-array.

Step 2: Extract all the characters in the given text file and store it in the array called Character array.

Step 3: Extract all the characters from the Stego key and store it in the array called Key- array.

Step 4: Choose first pixel and pick characters from Key-array and place it in first component of pixel.

If there are more characters in Key array, then place rest in the first component of next pixels.

Step 5: Place some terminating symbol to indicate end of the key. '0' has been used as a terminating symbol in this algorithm.

Step 6: Place characters of Character- Array in each first component (blue channel) of next pixels by replacing it.

Step 7: Repeat step 6 till all the characters has been embedded.

Step 8: Again place some terminating symbol to indicate end of data.

Step 9: Obtained image will hide all the characters that input.

ii) Extraction phase:

Step 1**:** Read the image to be embedded.

Step 2: Read the image inside which message to be embedded.

Step 3: Set, numSignificantBits = n;
Where, n = 1, 2……8

Step 4: size1 = size (secret); and
size2 = size (coverImage);

Step 5: Set, the "numSignificantBits" n significant bits of each byte of cover image to zero by using bit by AND operation on cover and size1 matrix.

Step 6: Embed the "numSignificantBits" most significant bits of secret image to create the stego image by using:
Stego = (cover zero+ secret)/28-n

Step 7: Recover the embedded image, by using bit by shift operation.

Step 8: Display Figure of cover image, Image to be hidden, stego image and recover image.

Step 9: End

Consider an 8-bitgrayscale bitmap image where each pixel is stored as a byte representing a gray scale value. Suppose the first eight pixels of the original image have the following grayscale values:

{
11010010 01001010
10010111 10001100
00010101 01010111
00100110 01000011
}

To hide the ASCII text 'C' whose binary value is 01000011, we would replace the LSBs of these pixels to have the following new grayscale values:

{
1101001**0** 0100101**1**
1001011**0** 1000110**0**
0001010**0** 0101011**0**
0010011**1** 0100001**1**
}

However, one of its major limitations is small size of data which can be embedded in such type of images using only LSB. LSB techniques implemented to 24-bit formats are difficult to detect contrary to 8-bit format. For example, assume the original three pixels are represented by the three 24-bit words below:

{
(00100111, 11101001, 11001000)
(00100111, 11001000, 11101001)
(11001000, 00100111, 11101001)
}

The binary value for the ASCII text 'A' is (001000001). Inserting the binary value of 'A' into the three pixels, starting from the top left byte, would result in:

{
(0010011**0** 1110100**0** 1100100**1**)
(0010011**0** 1100100**0** 1110100**0**)
(1100100**0** 0010011**0** 1110100**1**)
}

B. Discrete Cosine Transform

DCT coefficients are used for JPEG compression. It separates the image into parts of differing importance. These mathematical transforms convert the pixels in such a

way as to give the effect of "spreading" the location of the pixel values over part of the image.
DCT is used in steganography; Image is broken into 8×8 blocks of pixels. Working from left to right, top to bottom, the DCT is applied to each block [7].
Each block is compressed through quantization table to scale the DCT coefficients and message is embedded in DCT coefficients.
The general equation for a 1-D (N data items) DCT is defined by the following equation:

$$C(u) = \propto (u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

For, u = 0, 1, 2, …… N-1.
The general equation for a 2-D (*N* by *M* image) DCT is defined by the following equation [7]:

$$C(u,v) = \propto (u) \propto (v) \wp$$

Where,

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

For, u, v = 0, 1, 2, . . . , N-1
Here, the input image is of size N X M.
C (i, j) is the intensity of the pixel in row 'i' and column 'j'.
C (u, v) is the DCT coefficient in row u and column v of the DCT matrix.
The image of size M×N is divided into 8×8 blocks and two dimensional (2-D) DCT is performed on each block. Hence, the DCT can be calculated using above equation as:

$$C(u,v) = \frac{1}{4} \propto (u) \propto (v) \wp$$

Where,

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

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

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

i) Algorithm to embed text message:
Step 1: Read cover image.
Step 2: Read secret message and convert it in binary.
Step 3: The cover image is broken into 8×8 block of pixels.
Step 4: Working from left to right, top to bottom subtract 128 in each block of pixels.
Step 5: DCT is applied to each block.
Step 6: Each block is compressed through quantization table.
Step 7: Calculate LSB of each DC coefficient and replace with each bit of secret message.
Step 8: Write stego image.
Step 9: End.

ii) Algorithm to retrieve text message:
Step 1: Read stego image.
Step 2: Stego image is broken into 8×8 block of pixels.
Step 3: Working from left to right, top to bottom subtract 128 in each block of pixels.
Step 4: DCT is applied to each block.
Step 5: Each block is compressed through quantization table.
Step 6: Calculate LSB of each DC coefficient.
Step 7: Retrieve and convert each 8 bit into character.
Step 8: End.

VII. EVALUATION & RESULT

A. Evaluation
For comparing stego image with cover results requires a measure of image quality, commonly used measures are Mean-Squared Error, Peak Signal-to-Noise Ratio and capacity.
i) Mean Square Error:
The mean-squared error (MSE) between two images I1 (m, n) and I2 (m, n) is:

$$MSE = \frac{\sum_k [I_1(M,N) - I_2(M,N)]^2}{M * N}$$

Where,
M and N are the number of rows and columns in the input images, respectively.
ii) Peak Signal-to-Noise Ratio:
PNSR of an image can be calculated as:

$$PNSR = 10 log_{10}\left(\frac{R^2}{MSE}\right)$$

Where,
'R' is the maximum pixel value of the image. In other words, R = 2b − 1, where b is the bit depth of the original image and MSE refers to Mean Square Error.
PSNR is measured in decibels (dB). PSNR is a good measure for comparing restoration results for the same image, but between-image comparisons of PSNR are meaningless. [3]
iii) Capacity:
It is the size of the data in a cover image that can be modified without deteriorating the integrity of the cover image. Capacity depends on total number of bits per pixel & number of bits embedded in each pixel. Capacity is represented by bits per pixel (BPP) and the Maximum Hiding Capacity (MHC) in terms of percentage.
iv) Normalized Coefficient (NC):
Correlation is one of the best methods to evaluate the degree of closeness between the two functions. This measure can be used to determine the extent to which the original image and stego image remain close to each other, even after embedding the data.

RESULT

Comparative analysis of LSB based and DCT based steganography has been done on basis of parameters like PSNR, MSE, NC, Processing time & Capacity on different images and the results are evaluated. If PSNR ratio is high then images are of best quality.

Two images are taken on which Steganography is implemented. The images are taken as gray scale, shades of gray changes very gradually between the palette entries.

This increases the ability to hide information. The Cover Gray scale Images along with their histograms are shown below:

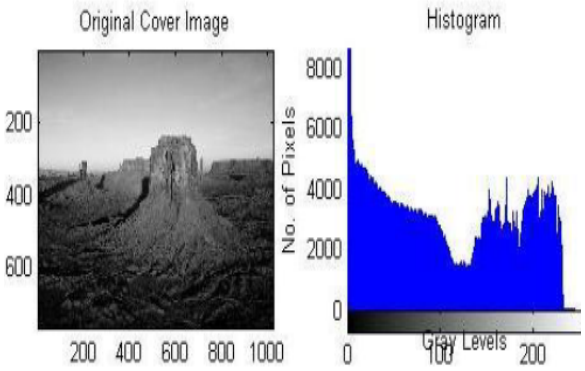

Image 1

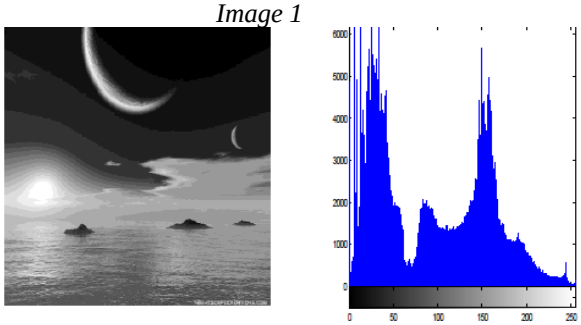

Image 2

Figure 3: Original Carrier Image and its histogram

TABLE 1: PARAMETERS OF LSB SUBSTITUTION

Image	MSE	PSNR	NC	Processing Time
Image 1	.000228	84.53	1	1.356234
Image 2	.000133	86.88	1	1.317199

i) LSB Substitution Technique:

The LSB Substitution technique is implemented on the two images and various parameters are evaluated. The Stego images and parameters are shown in Figure below:

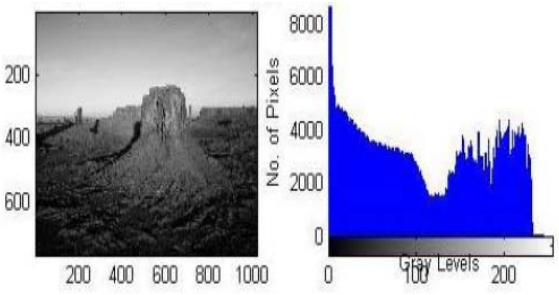

Image 1

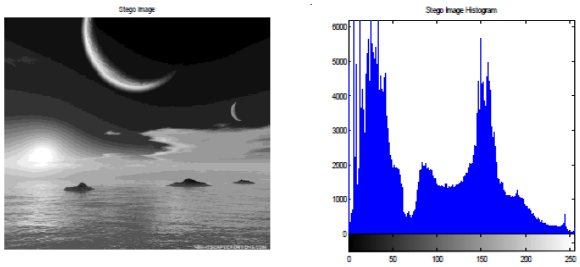

Image 2

Fig 4: Stego Image and its histogram (LSB)

ii) DCT Substitution Technique:

The DCT Substitution technique is implemented on the two images and various parameters are evaluated. The Stego images are shown in Fig 6. The values of various parameters are shown in Table 2. :

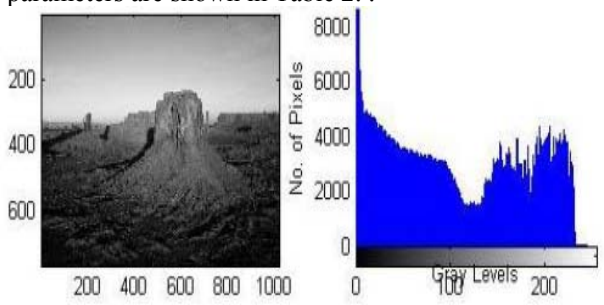

Image 1

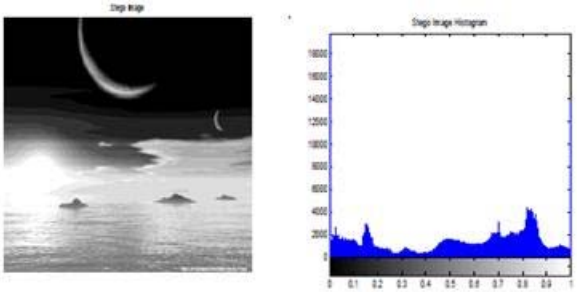

Image 2

Fig 5: DCT stego image and its histogram

TABLE 2: PARAMETERS OF DCT SUBSTITUTION

Image	MSE	PSNR	NC	Processing Time
Image 1	.00107	77.822	.9964	1.687751
Image 2	.04701	61.409	.8163	1.885138

VIII. CONCLUSION

In this paper, we have thus discussed the various techniques which are used in steganography. Analysis of LSB and DCT methods has been successfully implemented. In this paper, implementation of the major algorithms which are used in steganography has been discussed.

TABLE 3: PARAMETERS ANALYSIS OF STEGANOGRAPHY METHODS

Features	LSB	DCT
Invisibility	Low	High
Payload Capacity	High	Low
Robustness	Low	High
PSNR	High	Low
MSE	Low	High

DCT based Steganography scheme works perfectly with minimal distortion of the image quality in comparison to LSB based Steganography. Even though the amount of secret data that can be hidden by using this technique is smaller as compared to LSB based Steganography, DCT based Steganography scheme is being recommended by us as it ensures minimum distortion of image quality. LSB insertion is more vulnerable to even the most harmless and usual transformations.

The PSNR shows the quality of image after hiding the data. PSNR ratio of LSB based Steganography scheme is higher than Frequency domain based Steganography scheme for all types of images Gray scale as well as Color.

REFERENCES

[1] N.F. Johnson and S. Jajodia. (1998, Feb.). "Exploring steganography: seeing the unseen." IEEE Computer Journal. [On line]. 31(2), pp. 26-34.Available:http://www.jjtc.com/pub/r2026.pdf [Jun. 2011].

[2] P. Kruus, C. Scace, M. Heyman, and M. Mundy. (2003), "A survey of steganography techniques for image files." Advanced Security Research Journal. [On line], 5(1), pp. 41-52.Available: http://www.isso.sparta.com/documents/asrjv5.pdf#page=47 [Oct., 2011].

[3] A. Shaddad, J. Condell, K. Curran, and P. Mckevtt. "Biometric inspired digital image steganography." In Proc. of the 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, 2008. Pp. 159-168.

[4] C.-C. Chang et al., "Reversible hiding in DCT based compressed images", Information Sciences 177 (2007) 2768–2786.

[5] Krenn, R., "Steganography and Steganalysis", Available: http://www.krenn.nl/univ/cry/steg/article.pdf

[6] R. Chandramouli, N. Memon, "Analysis of LSB Based Image Steganography Techniques", IEEE pp. 1019-1022, 2001.

[7] Gurmeet Kaur and Aarti Kochhar "Transform Domain Analysis of Image Steganography" International Journal for Science and Emerging Technologies with Latest Trends" 6(1): 29-37 (2013).

[8] Steganographic Techniques and their use in an Open-Systems Environment- Bret Dunbar, The Information Security Reading Room, SANS Institute 2002.Availavle:http://www.sans.org/reading-room/whitepapers/covert/677.php

[9] F.A.P Peticolas, R.J. Anderson and M.G. Kuhn, "Information Hiding – A Survey", in *proceeding of IEEE*, pp. 1062-1078, July 1999.

[10] Y.K. Lee and L.H. Chen, "A Secure Robust Image Steganographic Model*", in Tenth National Conference on Information Security*, Hualien, Taiwan, pp. 275-284, May 5- 6, 2000.