Analytical Survey of Image Steganography Algorithms in Spatial Domain

Yamini Joshi
Department of Computer Science and Engineering
Jodhpur Institute of Engineering and Technology
Jodhpur, India
yamini.1691@gmail.com

***Abstract*—This paper presents an overview of basic algorithms in Image Steganography in spatial domain. A novel technique employing Huffman encoding is also discussed. In this technique, spatial domain embedding techniques are chosen to embed the secret image which is initially Huffman encoded. The Huffman encoded image is then embedded on the pixels of cover image. As a result, a secret image which cannot be embedded in a normal LSB embedding technique can be embedded in this proposed technique since the secret image is compressed. Experimental results comparing aforementioned algorithms are also tabulated. Peak Signal to Noise Ratio (PSNR) value is the metric used in quantifying the distortion between images (stego and cover image).**

Keywords—Steganography, PSNR, Huffman, Stego image, Cover image

I. INTRODUCTION

The word Steganography is derived from Greek words *steganos* , meaning "covered or protected," and *graphei* meaning "writing." It is a form of security through obscurity. It is a technique to hide secret information or message in some other data (generally known as cover or host) without any apparent evidence of data alteration.

Steganography differs from cryptography. The latter is an art of secret writing, and is intended to make a message unreadable by a third party. It produces a string of 1s and 0s which can be perceived as gibberish but may give grabbers an impulse to decrypt it. Though grabbers are not able to decrypt the meaningless message due to lack of a secret key, they can simply destroy or delay the transformation process. Steganography has been proposed to fool grabbers from perceiving the existence of secret data and robustness is usually a common factor not taken into consideration.

The capacity of secret data that can be embedded into a host image without degrading its quality is deemed much is more important than robustness. Thus, it does not hide the existence of the secret communication[1][2] .

Although steganography is separate and distinct from cryptography, there are many analogies between the two, and some authors categorize steganography as a form of cryptography since hidden communication is a form of secret writing [2].

II. STEGANOGRAPHY CONCEPTS

A. Problem Formulation

Although steganography is an ancient technique, its modern formulation is often given in terms of the prisoner's problem proposed by Simmons[4], where two inmates Alice and Bob, wish to communicate in secret to chalk out an escape plan. All of their communication passes through a warden who will throw them in solitary confinement should she suspect any covert communication [5]. The warden, who is free to examine all communication exchanged between the inmates, can either be passive (examines and detects covert communication, reports it to some authority and lets the message through without blocking it) or active (alters the communication with the suspected hidden information deliberately, in order to destroy the information)[6].

B. Nomenclature

The file that is used to hide or embed secret data is known as a cover, host or envelop. The aim of steganography is to hide secret information inside the cover so that its presence cannot be detected. The words cover and host file will be used interchangeably henceforward to denote the envelope. The image produced after hiding secret information is known as the stego file.

C. Kinds of steganography

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy far greater than necessary for the object's use and display [6]. Image and audio files especially comply with this criterion.

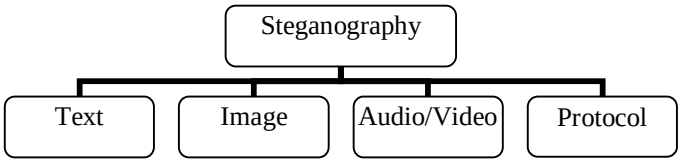

Fig 1. Categorization of Steganography

Figure 1 shows the four main categories of file formats that can be used for steganography. This categorization is based on the type of host or cover file used in the process.

This paper will focus on Image Steganography where images are used to hide secret information. In the following sections, the secret information will also be an image file. Both the images are greyscale files. It not necessary that the secret image is embedded on the host image as is; some transformation can be done on the secret image bit stream but, in this case, the parameters of the transformation must be known by the receiving end to aid the extraction process. For simplicity, this paper has considered the secret image data as is.

III. SPATIAL DOMAIN IMAGE STEGANOGRAPHY TECHNIQUES

In this section, three basic techniques for Image Steganography are introduced: Simple LSB technique, LSB with substitution table technique, and modulus function technique. These techniques focus on pixels' Least Significant Bit (LSB) modification of the host/cover image to hide secret data. In this subsection, the symbol k indicates the number of host bits that are used to embed secret data.

A. Simple LSB Technique

It is the simplest technique to embed secret information onto the cover image. Every pixel of the host contains some information of the secret image. For instance, if k LSBs of the host pixel are used to store secret information, then secret information is divided into groups of k and host pixels' k LSBs are simply modified to reflect the secret image.

B. LSB with Substitution Table Technique

Although the LSB substitution works easily, it the quality of the host image is degraded quickly. In 2001, Wang et al. [7] first brought the concept of the substitution table. This substitution table provides a (transformed) value for each secret value so that the difference between k bits of LSBs of the host and respective k bits of the secret information to be embedded is minimal. After transforming the secret value to its corresponding value according to the substitution table, the transformed secret value is embedded to a host pixel. The representation of a substitution table is an N × N matrix $ST_{N \times N}=\{st[i][j] \mid 0<=i, j<=N-1\}$, where the value N is equal to 2^k. The substitution table is a binary matrix: every element of the matrix is either 0 or 1. There is one more constraint, Each row and column has one and only one 1, rest all values are 0. If ST[i][j] is 1, the 2-bit unit with value i will be transformed to value j. For instance, consider the substitution table below:

$$ST_{4 \times 4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

For the matrix above, k=2 and $N=2^k=2^2=4$. Since ST[1][3]=1, two bit unit with value 1 or $(01)_2$ will be transformed to 3 or $(11)_2$.

The main task here is to find a substitution table. As k increases, N increases and probable substitution matrices increase. Various binary matrices satisfy the constraint, but only some can make the transformed bits similar to the host bits, making host image degradation less. Finding a good substitution table for secret information and a host image is crucial. If a substitution table having transformed bits of the secret information most similar to the host image bits is found, then we can obtain the best quality stego-image. Wang et al. used a genetic algorithm to search a substitution table. However, only an approximately optimal substitution table can be found. In 2003, Chang et al. [8] proposed their method to find an optimal substitution table by applying the dynamic programming. This survey used the dynamic programming method proposed by Chang et al to implement the substitution table technique.

C. Modulus Function Technique

This technique was proposed by Thien and Lin [9]. Supposing k bits with decimal value x are to be embedded into the host image's pixel with value y, this scheme proposed to find the value ŷ satisfying ŷ mod 2 k= x. Moreover, the value ŷ must be the closest value to y among all possible values that satisfy the equation ŷ mod 2k= x.

IV. PROPOSED METHOD

A. Huffman Encoded Secret information

In the techniques of Image Hiding discussed above, the size of image file that can be embedded onto the cover image file is restricted. For instance, consider a secret image file of dimensions h x w and cover of size H x W. the size of image file can be hidden in the cover is given by:

$$h * w = k * H * W \quad (1)$$

where * denotes product or multiplication.

The above equation specifies the maximum number of bits that can embedded and does not take into account any key that could also be embedded in order to facilitate the extraction of data.

The proposed system employs Huffman encoding to compress secret images so that images which couldn't be hidden inside the cover due to size constraints can be embedded in the cover.

B. Stego Image Architecture

Since the outcomes of this method is not known, a secret image is embedded inside a host both with and without Huffman Compression to compare the effectiveness of this system. If the stego image containing Huffman Compressed secret image information depicts more distortion or high PSNR, the proposed method will not be considered effective.

This system has also included a key in the stego image architecture to aid the extraction of secret image information. The key is very primitive and is always embedded in the last row of the host using Simple LSB technique with k = 2.

0	1	2	3	……	m-1
1	LSB of every pixel = Secret Image data or Compressed Secret Image data				
P	LSB of every pixel = Substitution table (if used)				
Q	LSB of every pixel = Huffman decoder string(if used)				
R	LSB of every pixel = Substitution table of Huffman decoder string				
. . . .	<unaltered host pixels>				
n-1	LSB of every pixel = Key parameters				

Fig. 2 Stego Image Architecture

The stego key architecture is as follows:

h	w	k	al	S	Tab_row	huff	huff_len	huff_row	Huff_r_sub

Fig. 3 Key Structure

where,
h = Height of secret image
w = Width of secret image
k = no. of bits to be embedded in every pixel's LSB
al = algo used
s = Size of secret image = h*w
tab_row = Row no. of Substitution Table
huff = Huffman Compression choice
huff_row = Row of Huffman Decoder String
huff_r_sub = Row of Substitution Table for Huffman Decoder string

C. Experimental Results

This section provides a comparison of methods discussed in Section III along with the proposed method based on the PSNR value of the stego image and embedding capacity of the cover. The images used as cover and secret information are 8 bit greyscale in .bmp format. The image Tulips with the size as shown in the Fig. 4 is used as the cover. Fig. 5 and Fig. 6. show two secret images Penguins and Colorboard which are used as secret information.

Fig. 4. Tulips.bmp 700 x 800

Fig. 5. Penguins.bmp 200 x 200

Fig. 6. Colorboard.bmp 600 x 500

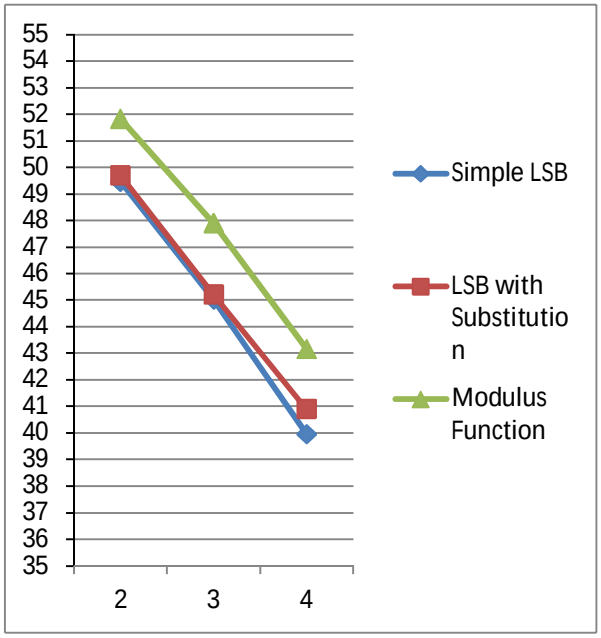

Fig. 7. PSNR values for secret image Penguins embedded in cover

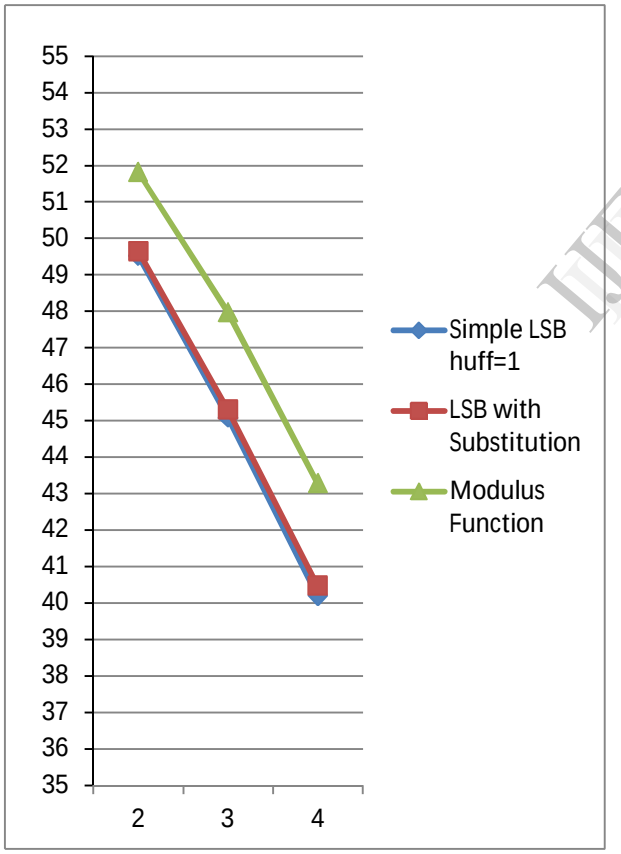

Fig. 8. PSNR values for secret image Penguins embedded in cover with Huffman Compression

The graphs depicted above indicate that as k increases, the PSNR value of the stego image decreases and hence the distortion in the cover image increases. It can also be concluded that if k is kept constant, the Modular function technique introduces the least amount of distortion (highest PSNR) in the cover image. Also, the Huffman compressed secret image data does not affect the PSNR values significantly. Thus, the proposed method of Huffman Compressing secret image information is an efficient way to hide images in the cover which couldn't be hidden otherwise due to their size constraints.

The image Colorboard cannot be embedded in the cover, even with k= 4, due to its size constraints. But when Huffman Encoding is applied prior to embedding, the secret information compresses and is hidden in the cover.

Table 1. MSE and PSNR values for stego image obtained by hiding Colorboard in the cover.

HUFF	Algo	Colorboard.bmp(600x500)	
		MSE	PSNR
0	LSB	-	-
0	SUBS TABLE	-	-
0	MOD	-	-
1	LSB	34.7723821429	32.7184591706
1	SUBS TABLE	31.7359571429	33.1152875989
1	MOD	17.2101392857	35.7729597568

Table 2. MSE and PSNR values for stego image obtained by hiding Penguins in the cover for k=2

HUFF	Algo	Penguins.bmp(200x200)	
		MSE	PSNR
0	LSB	0.739553571429	49.4411072218
0	SUBS TABLE	0.698373214286	49.6899278697
0	MOD	0.427460714286	51.8218415374
1	LSB	0.725378571429	49.5251563902
1	SUBS TABLE	0.705208928571	49.6476255865
1	MOD	0.427371428571	51.8227487633

Table 3. MSE and PSNR values for stego image obtained by hiding Penguins in the cover for k=3

HUFF	Algo	Penguins.bmp(200x200)	
		MSE	PSNR
0	LSB	2.048075	45.0173450452
0	SUBS TABLE	1.96221964286	45.2033274195
0	MOD	1.05431071429	47.9011174087
1	LSB	2.01516964286	45.0876874868
1	SUBS TABLE	1.91883214286	45.3004337603
1	MOD	1.03564821429	47.9786810019

Table 4. MSE and PSNR values for stego image obtained by hiding Penguins in the cover for k=4

HUFF	Algo	Penguins.bmp(200x200)	
		MSE	PSNR
0	LSB	6.56844464286	39.9561781673
0	SUBS TABLE	5.2859625	40.8995628306
0	MOD	3.14430714286	43.1555539859
1	LSB	6.21054642857	40.1995054805
1	SUBS TABLE	5.82231428571	40.4798471613
1	MOD	3.0532875	43.2831266113

One potential problem of the approaches listed above is that there is no way of finding if the stego image is tampered with. If an intruder makes changes in the stego image, it will not be detected by the receiving end. In this aspect lies the major difference between Cryptography and Steganography, the former stresses on robustness while the later tries to hide the existence of secret information transmission.

REFERENCES

[1] International Journal of Computer and Information Technology (ISSN: 2279 – 0764) Volume 01– Issue 02, November 2012, Steganography and Its Applications in Information Dessimilation on the Web Using Images as SecurityEmbeddment: A Wavelet Approach, pg 159

[2] Chi-Shiang CHAN and Chin-Chen CHANG, Department of Information Science and Applications, Asia University, Wufeng, Taiwan, A Survey of Information Hiding Schemes for Digital Images Chi-Shiang CHAN and Chin-Chen CHANG, IJCSES International Journal of Computer Sciences and Engineering Systems, Vol.1, No.3, July 2007

[3] Bauer, F. L. Decrypted Secrets: Methods and Maxims of Cryptology, 3rd ed. Springer-Verlag, New York, 2002

[4] Simmons, G., "The prisoners problem and the subliminal channel", CRYPTO, 1983

[5] Chandramouli, R., Kharrazi, M. & Memon, N., "Image steganography and steganalysis: Concepts and Practice", Proceedings of the 2nd International Workshop on Digital Watermarking, October 2003

[6] T. Morkel , J.H.P. Eloff , M.S. Olivier, "AN OVERVIEW OF IMAGE STEGANOGRAPHY", Information and Computer Security Architecture (ICSA) Research Group, Department of Computer Science, University of Pretoria, 0002, Pretoria, South Africa

[7] C. C. Chang, J. Y. Hsiao, and C. S. Chan, "Finding Optimal Least-significant-bit Substitution in Image Hiding by Dynamic Programming Strategy," Pattern Recognition, vol. 36, no. 7, July 2003, pp. 1583-1595

[8] R. Z. Wang, C. F. Lin, and J. C. Lin, "Image Hiding by Optimal LSB Substitution and Genetic Algorithm," Pattern Recognition, vol. 34, no.3, March 2001, pp. 671-683.

[9] C. C. Thien, and J. C. Lin, "A Simple and High-hiding Capacity Method for Hiding Digit-by-digit Data in Images Based on Modulus Function," Pattern Recognition, vol. 36, no. 12, Dec. 2003, pp. 2875-2881