Study and Analysis of Various Image Steganography Techniques

[1]Jagvinder Kaur, [2]Sanjeev Kumar

[1,2]Amritsar College of Engineering and Technology, Amritsar, India

Abstract

Steganography is the technique of hiding some data from the knowledge of an unwanted source inside some innocent looking canvas. When a steganographic system is developed, it is important to consider what the most appropriate cover work should be, and also how the stegogramme is to reach its recipient. In the last few years, we have seen many new and powerful steganography techniques reported in the literature. This paper gives the description of various techniques used in steganography. Different applications have different requirements of the steganography technique used. For example, some applications may require absolute invisibility of the secret information, while others require a larger secret message to be hidden. This paper intends to give an overview of image steganography, its uses and techniques. It also attempts to identify the requirements of a good steganographic algorithm and briefly reflects on which steganographic techniques are more suitable for which applications.

Keywords

Least Significant Bits (LSB), Peak Signal-to-Noise Rate (PSNR), Mean Square Error (MSE), Manhattan distance (Mdist), Steganalysis.

I. Introduction

The word steganography comes from the Greek Steganos, which mean covered or secret and Graphy mean writing or drawing. One of the oldest examples of steganography dates back to around 440 BC in Greek History. Herodotus, a Greek historian from the 5th Century BC, revealed some examples of its use in his work entitled "The Histories of Herodotus". One elaborate example suggests that Histaeus, ruler of Miletus, tattooed a secret message on the shaven head of one of his most trusted slaves. After the hair had grown back, the slave was sent to Aristagorus where his hair was shaved and the message that commanded a revolt against the Persians was revealed [14]. In this example, the slave was used as the carrier for the secret message, and anyone who saw the slave as they were sent to Aristagorus would have been completely unaware that they were carrying a message. As a result of this, the message reached the recipient with no suspicion of covert communication ever being raised. The main goal of steganography is to communicate securely in a completely undetectable manner and to avoid drawing suspicion to the transmission of a hidden data [4]. It is not to keep others from knowing the hidden information, but in contrast to cryptography, it is to keep others from thinking that the information even exists.

The basic model for steganography is shown on fig. 1. The model shows the basic process involved in steganography which consists of Carrier, Message and Password [1-3].

Carrier is also known as cover-object, in which message is embedded and serves to hide the presence of the message. The data can be any type of data (plain text, cipher text or other image) that the sender wishes to remain confidential. Password is known as stego-key, which ensures that only recipient who knows the corresponding decoding key will be able to extract the message from a cover-object. The cover-object with the secretly embedded message is then called the stego-object.

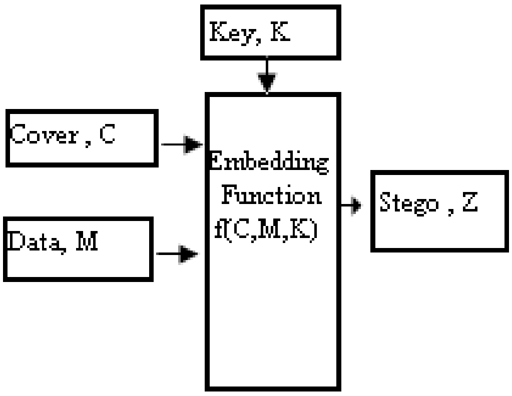


Fig. 1 : Basic Steganography Model

Recovering message from a stego-object requires the cover-object itself and a corresponding decoding key if a stego-key was used during the encoding process. In general, the information hiding process extracts redundant bits from cover-object. The process consists of two steps:

(i). Identification of redundant bits in cover-object. Redundant bits are bits which can be modified without corrupting quality or destroying the integrity of cover-object.

(ii). The embedding process then selects the subset of the redundant bits to be replaced with data from secret message. The stego-object is created by replacing selected redundant bits with message bits.

Paired with existing communication methods, steganography can be used to carry out hidden exchanges. Secret information is encoding in a manner such that the very existence of the information is concealed. With the Internet offering so much functionality, there are many different ways to send messages to people without anyone knowing they exist. Steganography is often confused with cryptography because the two are similar in the way that they both are used to protect confidential information. The output of steganography operation is not apparently visible but in cryptography the output is scrambled so that it can draw attention. The steganography is such a helpful process that if it is used and implemented properly, the hidden message will not be noticed from unwanted links, who might be trying to attack over it. The others can neither identify the meaning of the embedded object, nor can recognize its existence. It ensures complete security of the data. Only the recipient who must know the technique used, can recover the message and then decrypt it. Thus, the method aims to cause minimum amount of distortion on the cover object. Images are the most popular cover objects used for steganography. If a steganography method causes someone to suspect the carrier medium, then the method has failed. There has been a rapid growth of interest in steganography for two main reasons:

A. The publishing and broadcasting industries have become interested in techniques for hiding encrypted copyright marks and serial numbers in digital films, audio recordings, books and multimedia products.

B. Moves by various governments to restrict the availability of encryption service have motivated people to study methods by which private messages can be embedded in seemingly innocuous cover messages.

II. Stenography Principle

The secret message is embedded inside the cover object by a hiding algorithm and is sent to a receiver. The receiver then applies the reverse process on the cover data and reveals the secret data.

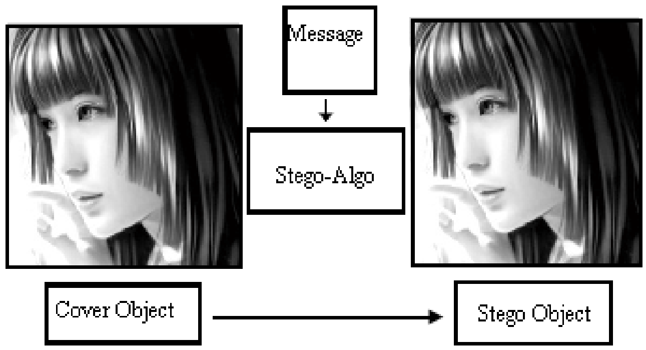


Fig. 2: Simple presentation of the principle of steganography

Fig. 2 shows the principle of steganography. The embedding, i.e. steganography algorithm, tries to preserve the perceptive properties of the original image. A suitable image, called the cover/carrier, is chosen. The secret message is then embedded into the cover using the steganography algorithm, in a way that does not change the original image in a human perceptible way. The result is new image, the stego-image, that is not visible different from the original. From an observer's view, the existence of a secret message is (visibly) hidden. The motive of using image is of no importance, it serves only as a carrier for hidden message. The secret message is embedded inside the cover object by a hiding algorithm and is sent to a receiver. The receiver then applies the reverse process on the cover data and reveals the secret data. The embedding, i.e. steganography algorithm, tries to preserve the perceptive properties of the original image.

A suitable image, called the cover/carrier, is chosen. The secret message is then embedded into the cover using the steganographic algorithm, in a way that does not change the original image in a human perceptible way. The result is new image, the stego-image, that is not visible different from the original. From an observer's view, the existence of a secret message is (visibly) hidden. The motive of using image is of no importance, it serves only as a carrier for hidden message.

III. Types 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 [9]. The redundant bits of an object are those bits that can be altered without the alteration being detected easily [4]. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding. There are four main categories of file formats that can be used for steganography shown in fig. 3.

Since, images are quite popular cover or carrier objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications. For these different image file formats, different steganographic algorithms exist. Here, in this paper, we will discuss about the image domain steganography methods.

In Image Domain methods secret messages are embedded using the intensity of the pixels values directly. The image domain methods are relatively simple compared to the other methods and are sometimes characterized as the "simple systems". However, they are generally more sensitive to small changes on the image such as filtering, resizing and squeezing.

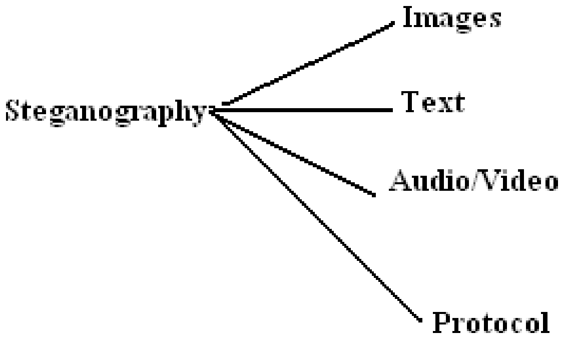


Fig. 3: The four main categories of file formats that can be used for steganography

To hide information in audio files similar techniques are used as for image files. One different technique unique to audio steganography is masking, which exploits the properties of the human ear to hide information unnoticeably. A faint, but audible, sound becomes inaudible in presence of another louder audible sound. This property creates a channel in which to hide information. Although nearly equal to images in steganography potential, the larger size of meaningful audio files makes them less popular to use than images [1, 7].

IV. Image Steganography Techniques

Image steganography techniques can be classified into two broad categories: Spatial-domain based steganography and Transform-domain based steganography.

A. Spatial Domain Method

In spatial domain scheme, the secret messages are embedded directly. Here, the most common and simplest steganography method is the least significant bits (LSB) insertion method. In the LSB technique, the least significant bits of the pixels are replaced by the message bits which are permuted before embedding.

- **Least Significant Bit Insertion Method**

Most steganography software hide information by replacing only the least-significant bits (LSB) of an image with bits from the file that is to be hidden. This technique is generally called LSB encoding. One of the most common techniques used in steganography. The following example shows how the letter A can be hidden in the first eight bytes of three pixels in a 24-bit image.

Pixels: (10101111 11101001 10101000)
(10100111 01011000 11101001)
(11011000 10000111 01011001)

Secret message: 01000001

Result: (10101110 11101001 10101000)
(10100110 01011000 11101000)
(11011000 10000111 01011001)

The three bold bits are the only three bits that were actually altered. Since the 8-bit letter A only requires eight bytes to hide it in, the ninth byte of the three pixels can be used to begin hiding the next character of the hidden message. A slight variation of this technique allows for embedding the message in two or more of the least significant bits per byte. This increases the hidden information capacity of the cover-object, but the cover-object is degraded more, and therefore it is more detectable.

Other variations on this technique include ensuring that statistical changes in the image do not occur. Some intelligent software also checks for areas that are made up of one solid color. While LSB insertion is easy to implement, it is also easily attacked. Slight modifications in the color palette and simple image manipulations will destroy the entire hidden message. Some examples of these simple image manipulations include image resizing and cropping. Since the steganalysis of LSB method is easier. Therefore, it is suggested that the image should be first manipulated before the embedding of the message into it.

- **Hiding Gray Images Using Blocks Method**

As vast channels for communication such as the Internet are becoming popular, the security of digital media becomes a greater concern. The hiding of a message will reduce the probability of detecting this message. This method hides a gray image in one another. The cover is divided into blocks of equal sizes. Each block size equals the size of the embedding image.

Compare each pixel in embedding image with all the corresponding pixels in the blocks of the cover image (assume there are C blocks).i.e. pixel (i,j) in the embedding image is compared with the pixel (i,j) in all C blocks of cover image. Select the best pixel to be embedding in. Best pixel is the pixel that gives minimum difference between it and the pixel to embed. For Example, if pixel (i,j) to embed has a value 250, and corresponding pixels values are: 248, 230, 249, 252, 255, 260, 270, and 262 (assume cover is divided into 8 blocks). Then the pixel with value 249 will be selected to embed 250.

- **Hiding Secret Message In Edges Of The Images Method**

Edge Based Steganography is in which only the sharper edge regions are used for hiding the message while keeping the other smoother regions as they are. It is more difficult to observe changes at the sharper edges than those in smoother regions. In this method Enhanced Least Significant Bit algorithm is used which can reduce the rate of pixel modification thereby increasing the security both visually and statistically.

- **Grey Level Modification Steganography Method**

This steganography method is based on image layers. This method divides the host image into blocks and embeds the corresponding secret message bits into each block using the layers which are made by the binary representation of pixel values. It then performs a search on the rows and columns of the layers for finding the most similar row or column. The location of row/column and its differences from the secret message is then marked by modifying minimum number of bits in the least significant bits of the blocks.

B. Transform Domain Method

The transform domain steganography technique is used for hiding a large amount of data with high security, a good invisibility and no loss of secret message. The idea is to hide information in frequency domain by altering magnitude of all of discrete cosine transform (DCT) coefficients of cover image. The 2-D DCT converts image blocks from spatial domain to frequency domain. The carrier image is divided into non overlapping blocks of size 8×8 and applies DCT on each of blocks of cover image using forward DCT [8].

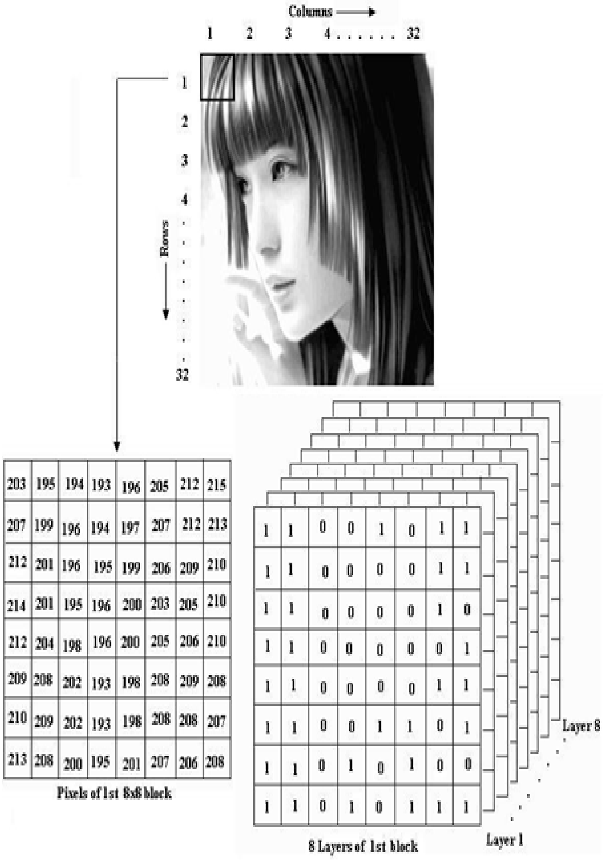
Now perform Huffman encoding on the 2D secret image of size M2xN2, to convert it into 1D stream. Huffman code is decomposed in 8-bits blocks. The least significant bit of all of the DCT coefficients inside 8×8 block is changed to a bit taken from each 8 bit block from left to right. Now, perform the inverse block DCT using inverse DCT and obtain a new image which contains secret image. At the receiver side, the stego-image is received, which is in the spatial domain.

C. Improved Steganography Algorithm

The new algorithm divides the cover image into fixed size blocks and embeds the secret bits into each block. The number of message bits to be embedded in the blocks is dependent on the size of the block. The algorithm tries to match the secret bit sequence with the rows and columns of the block's layers using the binary values of pixels. Each block is separated into layers using the digits in binary values of pixels. Considering a gray level image whose pixels are defined by a byte, the image block is separated into eight layers, where each layer contains the bits in one of eight digits. The fig. 4 shows an example block of an image and its corresponding layers. For the given a secret message bit sequence, the algorithm applies a sequential search on the layers to find the most similar row or column.

Except the last layer, all layers are used in the search process. The last layer contains the LSBs of pixels, and it is not used in search process. The search space for the particular image block is created by the rows and columns of the search layers, shown in fig. 4. The LSB layer is used for marking the differences between the most similar row and the secret message bit sequence, or we can say, that it is used to mark the hiding information of the secret data. The process of searching similar row or column starts from the very first layer and runs through all the layers except the last layer. As the most similar row is detected by the algorithm, it marks its position i.e. its layer and number of row or column in that particular layer. In marking, the rows of LSB layer are used as indexing the search layer in which the match is found. The row or column information of the match, on the other hand, is marked using the columns of LSB layer. The secret bit sequence which is matched in a row is indicated by an increment operation in the pixel value at that index.

And if the match is found in the column of any layer, then a decrement operation is performed at that index. Note that the last row of the LSB layer remains idle in marking the location information. The number of search layers is less than the number of rows in the layer because the LSB layer is not used in the search process. If the most similar row or column is exactly matched with the secret data, then only its location information will be marked. However, if it doesn't matches exactly, then the unmatched bits of the row or column will also be marked in the LSB layer. For this purpose, the last row of LSB layer is used to mark the indexes of the unmatched bits in the secret data. The position of unmatched bit is noted and at the same position the pixel in that block is modified accordingly.



203	195	194	193	196	205	212	215
207	199	196	194	197	207	212	213
212	201	196	195	199	206	209	210
214	201	195	196	200	203	205	210
212	204	198	196	200	205	206	210
209	208	202	193	198	208	209	208
210	209	202	193	198	208	208	207
213	208	200	195	201	207	206	208

Pixels of 1st 8x8 block

1	1	0	0	1	0	1	1
1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	0
1	1	0	0	0	0	0	1
1	1	0	0	0	0	1	1
1	1	0	0	1	1	0	1
1	1	0	1	0	1	0	0
1	1	0	1	0	1	1	1

8 Layers of 1st block

Fig. 4 : The 1st block of the image and its eight layers

The probability of finding the same bit sequence in this whole search space is very high. For instance, in an 8 by 8 block of a grey level image, each layer contains 8 possible bit sequences, i.e. 8 rows and 8 columns. Since there are 7 search layers in a gray level image block, the total number of bit sequences to be searched becomes 112. If the algorithm finds a match among 112 alternatives, an 8 secret bits is embedded by changing only one bit in the LSB layer. Since the rows and columns of all the search layers create a large number of alternatives for the exact match. We can have large search space to find the most similar bit sequence.

Let's explain the above process with a simple example. Assume that we have a host image block with 8x8 pixels and a secret data with 8 bits given as;

$$T_i = \begin{pmatrix} 193 & 208 & 200 & 195 & 201 & 212 & 206 & 208 \\ 209 & 208 & 202 & 214 & 198 & 208 & 209 & 208 \\ 210 & 209 & 202 & 193 & 199 & 208 & 208 & 207 \\ 195 & 204 & 199 & 205 & 200 & 205 & 206 & 210 \\ 214 & 201 & 195 & 206 & 200 & 203 & 205 & 210 \\ 207 & 199 & 194 & 195 & 200 & 207 & 212 & 213 \\ 212 & 201 & 203 & 195 & 213 & 206 & 209 & 210 \\ 203 & 195 & 194 & 193 & 214 & 205 & 212 & 215 \end{pmatrix}$$

Let the secret message bit sequence is given as;
$MS_i = [1\ 1\ 0\ 0\ 0\ 1\ 0\ 1]$

The algorithm searches for the secret bit sequence [1 1 0 0 0 1 0 1] in the first seven layers. It finds the most similar sequence at the 1st row of the 5th layer. In other words, the result of equation (1) is (5, 1). It then marks the pixel value at the index (5, 1) of the original image block Ti. The pixel value at 214 is increased to 215. In addition, the unmatched bit of secret data is found at the 1st and 2nd index. This is marked at location of the 1st and 2nd element of 8th row which is the reserved LSB layer. Thus, Stego-Image block Ti ' becomes;

$$T_i{}' = \begin{pmatrix} 193 & 208 & 200 & 195 & 201 & 212 & 206 & 208 \\ 209 & 208 & 202 & 214 & 198 & 208 & 209 & 208 \\ 210 & 209 & 202 & 193 & 199 & 208 & 208 & 207 \\ 195 & 204 & 199 & 205 & 200 & 205 & 206 & 210 \\ \underline{215} & 201 & 195 & 206 & 200 & 203 & 205 & 210 \\ 207 & 199 & 194 & 195 & 200 & 207 & 212 & 213 \\ 212 & 201 & 203 & 195 & 213 & 206 & 209 & 210 \\ \underline{204} & \underline{196} & 194 & 193 & 214 & 205 & 212 & 215 \end{pmatrix}$$

The steganalysis of the above methods goes exactly in the reverse process of the given algorithm. It is briefly explained as follows; In recovering the secret message, the stego-image, T_i' is compared with the host image T_i block by block and the degradation at the index (5, 1) for this particular block is recognized. The bit sequence [0 0 0 0 0 1 0 1] in the 1st row of the fifth layer is extracted. The last row of layer 8 is checked and the modification in the 1st and 2nd index is recognized.

Then the 1st and 2nd index of [1 1 0 0 0 1 0 1] is converted and the secret data [0 0 0 0 0 1 0 1] is found.

V. Conclusions

In the given paper, various methods of steganography were discussed along with their analysis. All the techniques under the image domain methods are focus which mostly works on the least significant bits of the pixel values. The improved steganography method which works through the block scheme, tries to found a match in the higher sub blocks of original image. The LSB based methods do gives good image quality but block method using the indexing technique performed on both row and column search outperforms them.

Also the improved method benefits from the possibility of matching whole secret bit sequence with any row and column of the image block. As a result, it makes minimum amount of degradation of the host image. Hence image quality is preserved at much higher level which is the main concern of this research.

References

[1] T. Sharp, "An implementation of key-based digital signal steganography", in Proc. Information Hiding Workshop, Springer LNCS 2137, pp. 13–26, 2001.

[2] Jarno Mielikainen, "LSB Matching Revisited", Signal Processing Letters, IEEE, Publication Date: May 2006 Volume : 13, Issue : 5, pp. 285- 287.

[3] K.M. Singh, L.S. Singh, A.B. Singh, K.S. Devi, "Hiding Secret Message in Edges of the Images", Information and Communication Technology, 2007. ICICT '07, pp. 238-241.

[4] S.Atawneh, "A New Algorithm for Hiding Gray Images Using Blocks", Information and Communication Technologies, 2006. ICTTA '06. 2nd, Volume: 1, pp. 1484- 1488.

[5] M.A.Khan, V.Potdar, E.Chang, "An Architecture Platform for Grey Level Modification Steganography", Industrial

Electronics Society, 2004. IECON 2004, 30th Annual Conference of IEEE, Vol. 1, pp. 463- 471.
[6] P.D. Khandait, S.P. Khandait, "LSB Technique for Secure Data Communication", pp.1
[7] T. Morkel , J.H.P. Eloff , M.S. Olivier , "An Overview Of Image Steganography", [Online] Available : http://mo.co.za/open/stegoverview.pdf., pp.2.5
[8] Johnson, N.F.; Jajodia, S., "Exploring Steganography: Seeing the Unseen", Computer Journal, February 1998
[9] Lee, Y.K.; Chen, L.H., "High capacity image steganographic model", Visual Image Signal Processing, 147:03, June 2000
[10] R. Chandramouli, Nasir Memon, "Analysis of LSB Based Image Steganography Techniques", IEEE 2001.
[11] Jithesh K , Dr. A V Senthil Kumar, " Multi layer information hiding -a blend of Steganography and visual cryptography", JTAIT, 2010
[12] Debnath Bhattacharyya, Poulami Das, Samir Kumar Bandyopadhyay, Tai-hoon Kim, "Text Steganography: A Novel Approach", IJAST Vol. 3, February, 2009
[13] S.K.Bandyopadhyay, Debnath Bhattacharyya, Poulumi Das, S. Mukherjee, D. Ganguly, "A Tutorial Review on Steganography", IC3 Noida, pp. 106-114, August 2008
[14] Robert Krenn, "Steganography, steganalysis".
[15] Chandramouli, R., Kharrazi, M.; Memon, N., "Image steganography an steganalysis: Concepts and Practice", Proceedings of the 2nd International Workshop on Digital Watermarking, Oct,2003
[16] Sutaone, M.S., Khandare, M.V, "Image based steganography using LSB insertion technique", IEEE WMMN, January 2008.
[17] Sanjive Tyagi, Ajay Agarwal, "Multi Layers Security Scheme for Embedding Secrets In Stego Image", IJAEST, Vol No.3, Issue No. 1, pp. 029 – 033
[18] Juan Jose Roque, Jesus Maria Minguet, "SLSB: Improving the Steganographic Algorithm LSB"

Jagvinder Kaur received her B.E. degree in the field of Electronics and Communication Engineering from Jammu University in 2008 and the M.Tech (ECE) from Amritsar College of Engineering and Technology, Amritsar, India. Her research interests include image processing, computer security, neural and fuzzy logic applications, and wireless communication. She has represented various research papers in International as well as National Level.

Er. Sanjeev Kumar is currently working as an Assistant Professor in Department of Electronics and Communication Engineering in Amritsar College of Engineering and Technology, Amritsar, India. He has more than 12 years of experience in teaching as well as industry. He has represented 12 papers in International journals and 25 papers in international as well as national conferences. His area of research is wireless communication, Image processing and Neural/fuzzy logic. He is a reviewer of various international journals.