International Conference on Technologies for Sustainability- Engineering, Information Technology, Management and the Environment
DAV Institute of Management, Faridabad (SUSTECH-15)
28 November 2015 www.conferenceworld.in ISBN: 978-81-931039-7-5

A SURVEY AND COMPARISON OF STEGANOGRAPHY TECHNIQUES

Preeti[1], Ayushi[2]

[1]M.Tech Student, [2]Assistant Professor, Hindu College of Engineering, Sonepat (India)

ABSTRACT

Steganography is mainly to hide the information for security purpose. In this we mainly combined techniques like data compression, cryptography and spread spectrum to provide the confidentiality for the internet. This paper is study the techniques that are used in steganography and to identify the better technique by comparison of various factors. All the techniques have some pros and cons.

I. INTRODUCTION

Steganography equation is as shown in figure1.

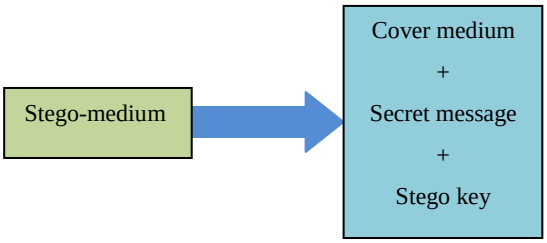

Figure1. Steganography equation

Today everybody wants to send secret message for their privacy. The secret data of message is known as embedded data. The message is mainly hide in message which known as cover-text or cover-image or cover-audio, which results in the stego-text or other stego-object. A secret-key is used to secure and control the hiding process so that any unauthorised person must not detect the hidden data.

Some properties that are used in creating the digital data for hiding message is:

- ✓ Imperceptibility: In this property one cannot define difference between the stego image and original image.
- ✓ Embedding Capacity: It defines how much amount of information can be embedded in the original without changing the quality of an image.
- ✓ Robustness: It defines the degree that is used to demolish embedded information without demolish the original image.

II. HISTORY

The history of Steganography can be traced back from 440 B.C

1) Wax Tablets: In ancient Greece, people wrote secret messages on wood and then covered it with Wax.

2) Shove Heads: This was also used back in ancient Greece. Slave's heads were shoved and secret messages were written on the scalp. Then, the slave's hair was allowed to grow and the secret message was exposed to the recipient after shaving the head again.
3) Invisible Ink: Secret messages were written using invisible ink which became visible only when the paper carrying the message was heated. Liquids such as milk, vinegar and fruit juices were used as invisible inks.
4) Morse code: Secret messages were written in Morse code on the knitting yarn. A cloth was made out of the yarn which was worn by the carrier. Also, Jeremiah Denton blinked his eyes in Morse code to spell the word ―Torture in a Television conference. This ensured the US Military that American POWs were tortured in North Vietnam.

III. STEGANOGRAPHY TECHNIQUES

The techniques which are used in steganography techniques: substitution systems, transform domain techniques, spread spectrum techniques, statistical methods, distortion techniques, and cover generation techniques [9].

3.1 Substitution Systems

In this we mainly find out some insignificant and redundant part of the original file and then embedded the secret message in that insignificant part. Original image contain a large number of redundant bits (e.g. LSB). In this technique, we embedded the secret message in the LSB of the original file without any changes in the original file. LSB technique is a spatial domain technique as the secret message is embedded to the original file. LSB technique is faster and easy, which is mainly used in the digital image.

3.2 Transform Domain Techniques

In this technique, secret data is mainly embedded in the significant part of the original file. So transform techniques are greater robust to attack as compared to substitution technique. Discrete cosine transforms (DCT), discrete wavelet transforms (DWT), and discrete Fourier transform (DFT) are transform in which we embedded the secret data in the original file. When we add the secret data in the original file, the whole image should be changed as the embedded part changing.

3.3 Spread Spectrum Techniques

In this technique we mainly spread the bandwidth of a narrowband signal across a wide band of frequencies". In this the original file is considered as communication channel and the secret message transmitted through like a signal. Since the message is spread out so it is more robust across the hidden file.

3.4 Statistical Techniques

Only one bit of secret data can be add to the secret file. If we hide "1' from the original file, some statistical characteristics (e.g. entropy and probability distribution) must be changed so it define the significant changing in the image. If we hide "0"from the original file, the original file is unchanged. So, it mainly depends on receiver to define the difference between the original and the stego file.

3.5 Distortion Techniques

Many techniques not required the original file but distortion need the original file to embed the secret data in the cover file. For the receiver the embedded message is the difference between the original and the stego image.

IV. COMPARISON OF LEAST-SIGNIFICANT BIT TECHNIQUE AND PSEUDO-RANDOM ENCODING TECHNIQUE

4.1 Least-Significant Bit (LSB) Technique

In this method the LSB of image should be taken and we change the LSB of the original image. Two types of bits, 8-bit and 24-bit are used in digital images. In 24 bit images we can embed three bits of information in each pixel, one in each LSB position of the three eight bit values. Increasing or decreasing the value by changing the LSB does not change the appearance of the image; so the resultant stego image looks almost same as the cover image. In 8 bit images, one bit of information can be hidden [6].

A stego-image is obtained by applying LSB algorithm on both the cover and hidden images. The hidden image is extracted from the stego-image by applying the reverse process [1, 4]. If the LSB of the pixel value of cover image C(i,j) is equal to the message bit m of secret massage to be embedded, C(i,j) remain unchanged; if not, set the LSB of C(i, j) to m. The message embedding procedure is given below-

S(i,j) = C(i,j) - 1, if LSB(C(i,j)) = 1 and m = 0

S(i.j) = C(i,j), if LSB(C(i,j)) = m

S(i,j) = C(i,j) + 1, if LSB(C(i,j)) = 0 and m = 1

where LSB (C(i, j)) stands for the LSB of cover image C(i,j) and m is the next message bit to be embedded. S(i,j) is the stego image. As we already know each pixel is made up of three bytes consisting of either a 1 or a 0. For example, suppose one can hide a message in three pixels of an image (24-bit colours). Suppose the original 3 pixels are:

(11101010 11101000 11001011)

(01100110 11001010 11101000)

(11001001 00100101 11101001)

A steganographic program could hide the letter "J" which has a position 74 into ASCII character set and have a binary representation "01001010", by altering the channel bits of pixels.

(11101010 11101001 11001010)

(01100110 11001011 11101000)

(11001001 00100100 11101001)

In this case, only four bits needed to be changed to insert the character successfully.

The resulting changes that are made to the least significant bits are too small to be recognised by the human eye, so the message is effectively hidden. The advantage of LSB embedding is its simplicity and many techniques use these methods. LSB embedding also allows high perceptual transparency.

4.2 Pseudo-Random Encoding Technique

In this technique, a random key is used to choose the pixels randomly and embed the message. This will make the message bits more difficult to find and reduce the realization of patterns in the image. Data can be hidden in the LSB of a particular colour plane (Red plane) of the randomly selected pixel in the RGB colour space.

4.3 Embedding Algorithm

In this process of encoding method, a random key is used to randomised the cover image and then hide the bits of a secret message into the least significant bit of the pixels within a cover image. The transmitting and receiving end share the stego key and random-key. The random-key is usually used to seed a pseudo-random number generator to select pixel locations in an image for embedding the secret message [6].

Inputs: Cover image, stego-key and the message

Output: stego image

1. Read character from text file that is to be hidden and convert the ASCII value of the character into equivalent binary value into an 8 bit integer array.
2. Read the RGB colour image (cover image) into which the message is to be embedded.
3. Read the last bit of red pixel.
4. Initialize the random key and randomly permute the pixels of cover image and reshape into a matrix.
5. Initialize the stego-key and XOR with text to be hidden and give message.
6. Insert the bits of the secret message to the LSB of the Red plane's pixels.
7. Write the above pixel to Stego Image File.

4.4 Extraction of Hidden Message

In this process of extraction, the process first takes the key and then random-key. These keys take out the points of the LSB where the secret message is randomly distributed. Decoding process searches the hidden bits of a secret message into the least significant bit of the pixels within a cover image using the random key.

In decoding algorithm the random-key must match i.e. the random-key which was used in encoding should match because the random key sets the hiding points of the message in case of encoding. Then receiver can extract the embedded messages exactly using only the stego-key.

4.5 Message Extraction Algorithm

Inputs: Stego-image file, stego-key, random key.

Output: Secret message.

1. Open the Stego image file in read mode and from the Image, read the RGB colour of each pixel.
2. Extract the red component of the host image.
3. Read the last bit of each pixel.
4. Initialize the random-key that gives the position of the message bits in the red pixel that are embedded randomly.
5. For decoding, select the pixels and Extract the LSB value of red pixels.
6. Read each of pixels then content of the array converts into decimal value that is actually ASCII value of hidden character.
7. ASCII values got from above are XOR with stego-key and gives message, which we hide inside the cover image.

4.6 Appraisal of Above Two Techniques

We have implemented the above two techniques in MATLAB and the above mentioned algorithms with respect to image steganography are not void of weak and strong points. Consequently, it is important to decide the most suitable approach to be applied.

Table 1: Comparison of characters of above two techniques

Techniques	*Imperceptibility*	*Robustness*	*Capacity*
Simple LSB	High*	Low	High
Pseudo-Random Encoding	Higher*	Low	High

*: Indicates dependency on the used cover image

4.7 Performance Analysis

As a performance measure for image distortion due to hiding of message, the well-known peak-signal-to noise ratio (PSNR), can be applied to stego images. It is defined as:

PSNR = 10 log $(Cmax)^2$/MSE:

MSE = mean - square - error;

which is given as:

MSE = 1/MN ($(S-C)^2$):

Cmax = 255:

Where M and N are the dimensions of the image,

S is the resultant stego-image, and C is the cover image.

PSNR values below 30 dB indicate low quality (i.e., distortion caused by embedding is high).

A high-quality stego image should strive for a PSNR of 40 dB, or higher [7].

Table 2: PSNR of Pseudo-Random Encoding

S. No	Cover Image	Secret Message	Stego-Image	MSE	PSNR(dB)
1	Gray image	Text message	Gray image	0.0449	61.6065
2	RBG image	Text message	sisbr	0.0111	67.6835
3	RBG image	Image	Image	0.0911	58.5346

Table 3: PSNR of Least Significant Bits Encoding

S. No	Cover Image	Secret Message	Stego-Image	MSE	PSNR(dB)
1	Gray image	Text message	Gray image	0.0463	61.4733
2	RBG image	Text message	sisbr	0.021	67.6697
3	RBG image	Image	Hydrang	0.0912	58.5311

V. CONCLUSIONS

On the basis of this paper it is found that which technique is best for hiding secret information. The LSB Technique and Pseudo-Random Encoding Technique on images to obtain secure stego-image which shows that PSNR of Pseudo random encoding is higher than PSNR of LSB encoding. Our results indicate that the LSB insertion using random key is better than simple LSB insertion in case of lossless compression. The image resolution doesn't change much when we embed the message into the image and the image is protected with the secret key. So, it is not possible to damage the data by any unauthorized person. The algorithm is usage for both 8 bit and 24 bit image of the same size of cover and secret image .This paper mainly increasing the security of the message and increasing PSNR and reducing the distortion rate.

REFERENCES

[1] R.Anderson and F. Petitcolas, "On the limits of steganography" IEEE Journal of Selected Areas in Communications, Vol. 16, No. 4, May 1998.

[2] P. Kruus, C. Scace,M. Heyman, and M. Mundy. (2003), "A survey of steganography techniques for image les." Advanced Security Research Journal.

[3] Jaswinder Kaur, Inderjeet & Manoj Duhan, (2009) "A Comparative Analysis of Steganographic Techniques",International Journal of Information Technology and Knowledge Management.

[4] A. Cheddad, J. Condell, K. Curran and P.M. Kevitt. (2010). "Digital image steganography: survey and analysis of current methods." Signal Processing Journal.

[5] C.P.Sumathi, T.Santanam2 and G.Umamaheswari "A Study of Various Steganographic Techniques Used for Information Hiding".

[6] P.sanjay kumar jena, Kshetrimayum Jenita Devi (2013)"A Sesure Image SteganographyUsing LSB Technique and Pseudo Random Encoding Technique".

[7] "Detecting LSB Steganography in Color and Gray-Scale Images" Jessica Fridrich, Miroslav Goljan, and Rui Du State University of New York, Binghamton.

[8] Ravi K Sheth, Rashmi M. Tank "Image Steganography Techniques" International Journal of Computer Engineering and Sciences (IJCES) Volume-1 Issue-2, 2015

[9] Dr.Adal almohammad, E. Karthikeyan "Steganography-Based Secret and Reliable Communications: Improving Steganographic Capacity and Imperceptibility" Sponsored National Conference on Advanced Networking and Applications, august 2010.