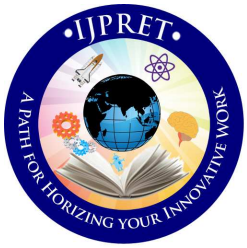
Research Article *ISSN: 2319-507X*
Pranjali Gondse, IJPRET, 2013; Volume 1(8): 476-487 *IJPRET*



INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

DIFFERENT IMAGE STEGANOGRAPHIC TECHNIQUES AND THEIR COMPARISION

PRANJALI G. GONDSE, PROF. ANJALI B. RAUT

1. M.E. (CSE) First Year, H.V.P.M. /C.O.E.T., Amravati.
2. Associate Professor, H.V.P.M. /C.O.E.T., Amravati.

Accepted Date:

27/02/2013

Publish Date:

01/04/2013

Keywords

Steganography,

Spatial domain,

LSB method,

Transform domain

Corresponding Author

Ms. Pranjali G. Gondse

Abstract

Data hiding techniques have taken important role with the rapid growth of intensive transfer of multimedia content and secret communications. Steganography is the art of hiding information in ways that prevent detection. Steganography is the art and science of communicating in such a way that the presence of a message cannot be detected. It includes the concealment of information within computer files. As increasingly more material becomes available electronically, the influence of steganography on our lives will continue to grow. Many confidential information were leaked to a rival firm using steganographic tools that hide the information in music and picture files. The application of steganography is an important motivation for feature selection. Steganography means hiding a secret message (the embedded message) within a larger one (source cover) in such a way that an observer cannot detect the presence of contents of the hidden message. Many different carrier file formats can be used, but digital images are the most popular because of their frequency on the Internet. For hiding secret information in images, there exists a large variety of Steganography techniques some are more complex than others and all of them have respective strong and weak points. Different applications have different requirements of the Steganography technique used. This paper intends to give an overview of image Steganography, its uses and techniques.

INTRODUCTION

Today networks are seriously threatened by network attacks. The network security is becoming more important as the amount of data being exchanged on the Internet is increasing. Securing information is becoming more vital as any form of Internet transmission is subject to unauthorized snooping. There are a number of ways for securing data One is cryptography, where the sender uses an encryption key to scramble the message, this scrambled message is transmitted through the insecure public channel, and the reconstruction of the original, unencrypted message is possible only if he receiver has the appropriate decryption key. The second method is steganography, where the secret message s embedded in another message, thus the existence of message is unknown.

Steganography is the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. The word steganography is derived from the Greek words "*stegos*" meaning "cover" and "*grafia*" meaning "writing" [1] defining it as "covered writing". In image steganography the information is hidden exclusively in images.

The idea and practice of hiding information has a long history. In *Histories* the Greek historian Herodotus writes of a nobleman, Histaeus, who needed to communicate with his son-in-law in Greece. He shaved the head of oneof his most trusted slaves and tattooed the message onto the slave's scalp. When the slave's hair grew back the slave was dispatched with the hidden message [4]. In the Second World War the Microdot technique was developed by the Germans. Information, especially photographs, was reduced in size until it was the size of a typed period. Extremely difficult to detect, a normal cover message was sent over an insecure channel with one of the periods on the paper containing hidden information [5]. Today steganography is mostly used on computers with digital data being the carriers and networks being the high speed delivery channels.

Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Steganography differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret [7]. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated [7]. The strength of steganography can thus be amplified by combining it with cryptography.

Steganography can be used for wide range of applications such as defiance organizations for safe circulation of secret data, intelligence agencies, in smart identity cards where personal details are embedded in the photograph itself for copyright control of materials, medical imaging where patient's details are embedded within image providing protection of information and reducing transmission time.

STEGONOGRAPHY

In stegonography a message is embedded in a digital image (cover image) through an embedding algorithm, with the help of a secret key. The resulting stego image is transmitted over a channel to the receiver where it is processed by the extraction algorithm using the same key. During transmission the stego image, it can be monitored by unauthenticated viewers who will only notice the transmission of an image without discovering the existence of the hidden message. In figure1 the basic principle of stegonography is described, in which data is any secret message which sender want to send to the receiver. Cover image or object is also known as carrier in which message is embedded and serves to hide presence of message. Stego key which ensures that only recipient who knows the corresponding decoding key will be able to extract the message from cover object with secretly embedded message is then called stego object.

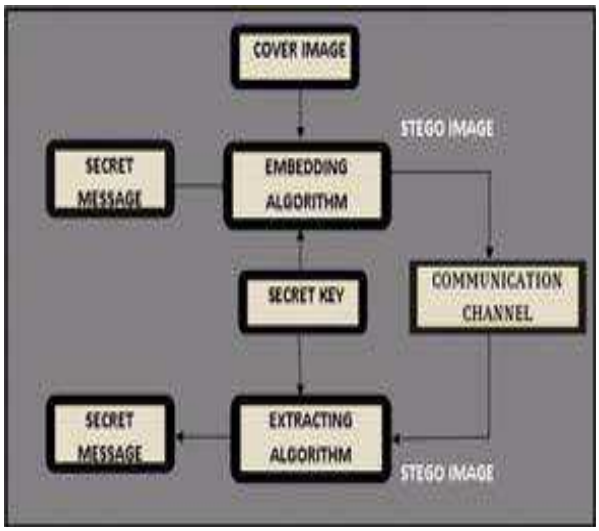


Figure:1 Stgonography principle

.IMAGE STEGANOGRAPHY

As stated earlier, images are the most popular cover 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.

IMAGE STEGANOGRAPHY TECHNIQUE

Image steganography techniques can be divided into two groups: those in the Image Domain and those in the Transform Domain [2]. Image – also known as spatial – domain techniques embed messages in the intensity of the pixels directly, while for transform – also known as frequency – domain, images are first transformed and then the message is embedded in the image [20]. Image domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation and are sometimes characterised as “simple systems” [12]. The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format [10]. Steganography in the transform domain involves the manipulation of algorithms and image transforms [12]. These methods hide messages in more significant areas of the cover image, making it more robust [4]. Many

transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression [10].

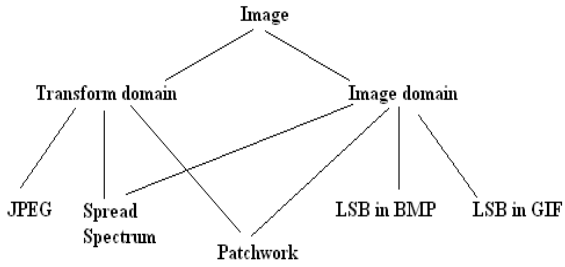


Figure2 Categories of image steganography

Least Significant Bit

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image [14]. 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 colour 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 [19]. For example a grid for 3 pixels of a 24-bit image can be as follows:

(00101101 00011100 11011100)

(10100110 11000100 00001100)

(11010010 10101101 01100011)

When the number 200, which binary representation is 11001000, is embedded into the least significant bits of this part of the image, the resulting grid is as follows:

(00101101 00011101 11011100)

(10100110 11000101 00001100)

(11010010 10101100 01100011)

Although the number was embedded into the first 8 bytes of the grid, only the 3 underlined bits needed to be changed according to the embedded message. On average, only half of the bits in an image will need to be modified to hide a secret message using the maximum cover size [19]. Since there are 256 possible intensities of each primary colour, changing the LSB of a pixel results in small changes in the intensity of the colours. These changes cannot be perceived by the human eye - thus the message is successfully hidden. With a well-chosen image, one can even hide the message in the least as well as second to least significant bit and still not see the difference [14].

In the above example, consecutive bytes of the image data – from the first byte to the end of the message – are used to embed the information. This approach is

very easy to detect [4]. A slightly more secure system is for the sender and receiver to share a secret key that specifies only certain pixels to be changed. Should an adversary suspect that LSB steganography has been used, he has no way of knowing which pixels to target without the secret key [5].

In its simplest form, LSB makes use of BMP images, since they use lossless compression. Unfortunately to be able to hide a secret message inside a BMP file, one would require a very large cover image. Nowadays, BMP images of 800 × 600 pixels are not often used on the Internet and might arouse suspicion [19]. For this reason, LSB steganography has also been developed for use with other image file formats.

Image steganography using primes

In this method a shared key pair (EvenK, OddK) which are odd and prime are chosen and the ASCII value representing the character is tested for evenness. If it is even, then EvenK is added to the number and if it is not even then OddK is added to it. When we add odd number with an even number results to an odd number and in the same fashion an odd number added with an odd number results an even number. On the decrypting side each number is tested for evenness and if it is even then OddK is subtracted but if it is not even then EvenK is subtracted. Suppose that the shared key is the pair (11, 19). The sender needs to send the message "abcxyz". The ASCII values of (a, b, c, x, y, z) are (97, 98, 99, 120, 121, 122) respectively. Then encryption process calculates for each character as: 97 mod 2 = 1 so C = 97+ 19 = 116, for second character b as 98 mod 2 = 0; so C= 98+11=109. Similarly for rest of the characters as:

99 mod 2 =1 so C= 99+19 =118

120 mod 2 =0 so C= 120+11=131

121 mod 2 =1 so C= 121+19=140

122 mod 2 =0 so C= 122+11=133

These numbers (78, 91, 77, 195, 138, and 199) are the cipher text. Now converting this number to the equivalent binary number, we get (01110100, 01101101, 01110110, 10000011, 10001100, 10000101) respectively. Converting this number to gray code we get (01001110, 01011011,

01001101, 01011001, 10001010, 11000111) respectively. Each bit represents a pixel. Three 8-bit bytes, one byte for each of RGB, is called 24 bit color. Each 8 bit RGB component can have 256 possible values, ranging from 0 to 255. Now a grid of 16 pixels of a 24-bit image can hold all this gray code in the LSB position as:

01010000 01100111 01001100

01101000 01100011 01000101

00111001 01101110 01010100

00101011 01110010 01101001

00101001 01000010 01101111

00111011 01000110 01001101

01101000 01100010 01100101

01111011 01101010 01011101

01010011 01100111 01001100

01101000 01101110 01010010

00101011 01110010 01101101

01001110 01011010 01011010

01010111 01010110 01101011

01101010 01011011 01010111

01011000 01011000 01010010

01101011 01101111 01101001

Changing LSB in each value would allow minor variations in color and unnoticeable to human eye. This image is send to the receiver. In the receiving end, simply extract the appropriate LSB bits from the image and group it by 8 bits i.e. 01110100, 01101101, 01110110, 10000011, 10001100, 10000101. Now convert this into binary number and that is 01110100, 01101101, 01110110, 10000011, 10001100, and 10000101 respectively. Convert this binary number to equivalent decimal number which are 116,109,118,131, 140,133 respectively and it is the cipher text. The decryption process calculates 116 mod 2 as 0 so M = 116 - 19 = 97. Similarly

109 mod 2 as 1 so M=109-11 = 98

118 mod 2 = 0 so M=118-19 = 99

131 mod 2 = 1 so M=131-11 =120

140 mod 2=0 so M=140-19=121and similar for rest of the cipher text. These numbers are the ASCII value of the message. Now converting this value to the equivalent

character we get the message "abcxyz" which is the original message.

JPEG steganography

Originally it was thought that steganography would not be possible to use with JPEG images, since they use lossy compression which results in parts of the image data being altered. One of the major characteristics of steganography is the fact that information is hidden in the redundant bits of an object and since redundant bits are left out when using JPEG it was feared that the hidden message would be destroyed. Even if one could somehow keep the message intact it would be difficult to embed the message without the changes being noticeable because of the harsh compression applied. However, properties of the compression algorithm have been exploited in order to develop a steganographic algorithm for JPEGs.

One of these properties of JPEG is exploited to make the changes to the image invisible to the human eye. During the DCT transformation phase of the compression algorithm, rounding errors occur in the coefficient data that are not noticeable [14]. Although this property is what classifies the algorithm as being lossy, this property mcan also be used to hide messages. It is neither feasible nor possible to embed information in an image that uses lossy compression, since the compression would destroy all information in the process. Thus it is important to recognize that the JPEG compression algorithm is actually divided into lossy and lossless stages. The DCT and the quantization phase form part of the lossy stage, while the Huffman encoding used to further compress the data is lossless. Steganography can take place between these two stages. Using the same principles of LSB insertion the message can be embedded into the least significant bits of the coefficients before applying the Huffman encoding. By embedding the information at this stage, in the transform domain, it is extremely difficult to detect, since it is not in the visual domain.

Spread Spectrum

In spread spectrum techniques, hidden data is spread throughout the cover-image making it harder to detect [4]. A system

proposed by Marvel et al. combines spread spectrum communication, error control coding and image processing to hide information in images [6].

Spread spectrum communication can be defined as the process of spreading the bandwidth of a narrowband signal across a wide band of frequencies [6]. This can be accomplished by adjusting the narrowband waveform with a wideband waveform, such as white noise. After spreading, the energy of the narrowband signal in an one frequency band is low and therefore difficult to detect [6]. In spread spectrum image steganography the message is embedded in noise and then combined with the cover image to produce the stego image. Since the power of the embedded signal is much lower than the power of the cover image, the embedded image is not perceptible to the human eye or by computer analysis without access to the original image [6].

EVALUATION OF DIFFERENT TECHNIQUES

All 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. As defined before, there are several parameters to measure the performance of the steganographic system. Fridrich in Fig. 2 shows the relationship between three parameters These parameters are as follows:

- **Undetectability (imperceptibility):** this parameter is the first and the primary requirement; it represents the ability to avoid detection, i.e., where the human eye fail to notice it. However, the techniques that do not alter the image in such a way to be perceptible to the human eye may still alter the image in a way that it is detectable by the statistical tests. Truly secure steganographic techniques should be undetectable neither by the human eye nor by the statistical attacks.

- **Robustness:** it is the second parameter that measures the ability of the steganographic technique to survive the attempts of removing the hidden information. Such attempts include, image manipulation (like cropping or rotating), data compression, and image filtering.

• **Payload capacity:** it is the third parameter that represents the maximum amount of information that can be hidden and retrieved successfully. When compared with watermarking, that requires embedding only a small amount of copyright information, steganography is seen to hide communication and consequently a sufficient embedding capacity is required. Accordingly and by using this parameter, small amounts of data could be hidden without being detected by the human eye. Larger amounts of information, on the other hand, may detect artifacts by the HVS or statistical tests.

	Spread Spectrum	LSB in BMP	LSB in GIF	JPEG Compression
Invisibility	high	high	Medium	high
Payload capacity	Medium	high	Medium	Medium
Robustness	high	low	low	Medium

Table1: comparison of different image steganographic algorithms

CONCLUSION

Although only some of the main image steganographic techniques were discussed in this paper, one can see that there exists a large selection of approaches to hiding information in images. All the major image file formats have different methods of hiding messages, with different strong and weak points respectively. Where one technique lacks in payload capacity, the other lacks in robustness. For example, the patchwork approach has a very high level of robustness against most type of attacks, but can hide only a very small amount of information.

REFERENCES

1. Silman, J., "Steganography and Steganalysis: An Overview", *SANS Institute*, 2001

2. Jamil, T., "Steganography: The art of hiding information is plain sight", *IEEE Potentials*, 18:01, 1999

3. Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", *Communications of the ACM*, 47:10, October 2004

4. Anderson, R.J. & Petitcolas, F.A.P., "On the limits of steganography", *IEEE Journal of*

selected Areas in Communications, May 1998

5. G. Simmons, The prisoners problem and the subliminal channel, CRYPTO, 1983

6. Souvik Bhattacharyya. and Gautam Sanyal. An Image Based Steganography Model for Promoting Global Cyber Security. In Proceedings of International Conference on Systemics, Cybernetics and Informatics, Hyderabad, India, 2009.

7. K. Ahsan and D. Kundur, Practical Data hiding in TCP/IP, Proceedings of the workshop on Multimedia security at ACM Multimedia, 2002

8. J. Silman, Steganography and Steganalysis: An Overview, SANS Institute, 2001

9. Y.K. Lee and L.H. Chen, High capacity image steganographic model, Visual Image Signal Processing,147: 03, June 2000

10. R. Krenn, Steganography and Steganalysis, www.krenn.nl/univ/cry/steg/article.pdf

11. K. B. Raja, S. Sindhu, T. D. Mahalakshmi, S. Akshatha, B. K. Nithin, M. Sarvajith, K. R. Venugopal,

12. L. M. Patnaik, "Robust Image Adaptive Steganography using Integer Wavelets" *International conference on Communication Systems Software*, pp. 614-621, 2008.

13. Jan Kodovsky, Jessica Fridrich "Influence of Embedding Strategies on Security of Steganographic Methods in the JPEG Domain" *Proceedings of SPIE, the International Society for Optical Engineering*, vol. 6819, pp. 681902.1-681902.13, 2008.

14. L. Y. Por, W. K. Lai, Z. Alireza, T. F. Ang, M. T. Su, B. Delina, "StegCure: A Comprehensive Steganographic Tool using Enhanced LSB Scheme," *Journal of WSEAS Transctions on Computers*, vol. 8, pp. 1309-1318, 2008.

15. Gaetan Le Guelvouit, "Trellis-Coded Quantization for Public-Key Steganography," *IEEE International conference on Acostics, Speech and Signal Processing*, pp.108-116, 2008.

16. Mohammed Ali Bani Younes and Aman Jantan, "A New Steganography Approach for Images Encryption Exchange by Using the Least Significant Bit Insertion," *International Journal of Computer Science and Network Security*, vol. 8, no. 6, pp.247-257, 2008.