International Journal of Information System and Engineering
Vol. 4 (No.2), November, 2016
ISSN: 2289-7615
DOI: 10.24924/ijise/2016.11/v4.iss2/01.08

IJISE
www.ftms.edu.my/journals/index.php/journals/ijise

Research Paper

A NOVEL STRUCTURED-LEAST SIGNIFICANT BIT STEGANOGRAPHY TECHNIQUE

Babangida Zachariah
FTMS Global College
Cyber Jaya, Malaysia
daddonyone@gmail.com

Abstract

The need for improved security in data communications has been the driving force behind the several types of research that describes various techniques for implementing security policies and systems. The use of Steganography and Cryptography has made a high impact in advancing the security desired of data communications. However, there is a continuous need to more research that would deal with loopholes inherent in security systems. This research proposes a novel structured-least significant bit steganography technique which embeds secret messages in such a way that retrieval of such message is difficult for a third party, a hacker. The proposed technique was implemented and tested against most relevant existing techniques and was found to perform better in terms of the quality of the cover and Stego-image analysis.

Key Terms: Steganography, LSB, Spatial Domain, Secret Message, Structure-LSB

1. Introduction

Cryptography has long been used to ensure the security of data communication. Though it provides security, it is always faced with attempts to be broken by hackers. This is because the hackers know about the transmitted messages but only that they do not have the key to decrypt the content so as to make meaning out of it. This led to the need for messages to be transmitted in such a way that the hackers are unaware of the existence of the messages. This gave rise to the concept of Steganography. That is, while cryptography scrambles the messages, Steganography hides the existence of the messages (Swain & Lenka, 2012). Thus, having the possibility facing fewer attacks than cryptographic security messages.

In a basic steganography system shown in Fig.1, a message to be transmitted, called the secret message is combined with a secret key and a cover media using an algorithm; and the secret message is embedded inside the cover media. That way, the cover media becomes a Stego-media; when it has been transmitted to the recipient, an algorithm retrieves the secret message from the Stego-media. Therefore, during transmission, the secret message is not seen but the Stego-media which is usually what is known and looks innocent from the hackers (Jain & Kumar, 2012).

The cover media which is also called carrier is usually another file such as an Image, Video, Audio, or Text file. The algorithm may be a distortion method, substitution method, transform domain method, or statistical method. The secret key used for the embedding must be the secret key used for retrieval. The technique has long been in use since the ancient times say 440BC and has also found application in the world of information technology of today.

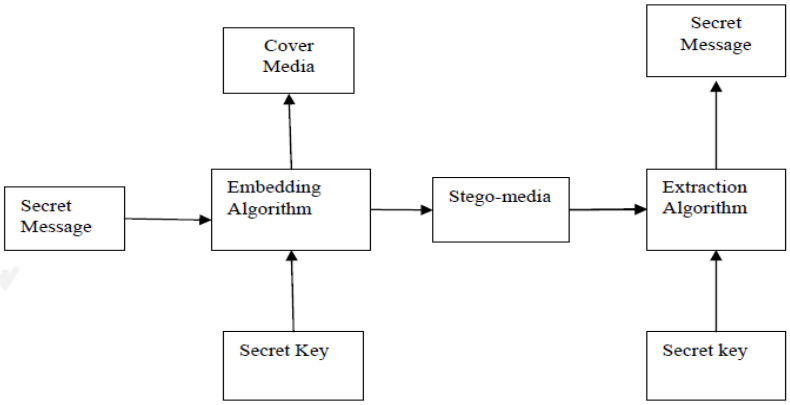

Fig.1: Basic Steganography Model

In Distortion Method, the cover media is distorted and a special technique is used to embed the secret message in such a way as to correct the distorted carrier (Medeni & Souidi, 2010; Gaikwad & Wagh, 2010); Statistical Method where the effort is to overcome the statistical analysis approaches to steganalysis using such properties as statistical features of the cover media file, which include mean, standard deviation, variance, histogram analysis, correlation coefficient, texture similarities, entropy, marginal distribution, second-order statistics, etc. (Sarkar, et al., 2007; Sadkhan, et al., 2009; Sun & Lui, 2010; Seyyedi & Ivanove, 2014); Transform Domain Methods, where the cover media signal such as the frequency domain is transformed using standard mathematical concepts such as Integer Wavelet Transform (IWT), Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), and Arnold Transform are often used (Haynes, 2011; Hemalatha & Renuka, 2013; Hemalatha, et al., 2013); and Spatial-Domain Method, also called Substitution method is the commonly used steganography methods. They are simple to understand and implement; requiring fewer resources during embedding and retrieval of hidden secret messages. The method substitutes bits of the least and/or most/more significant bits of the pixels of the cover media, and mostly done on images.

Our proposed system is based on Spatial-Domain Method. Therefore, the next section discusses a few most relevant types of research in this area.

2. Literature Review

In the work of (Sharmila & Shanthakumari, 2012), an algorithm that works on the edges of JPEG images to hide secret messages and to improve robustness was proposed. It takes advantage of the fact that sharper edges of an image have complicated statistical features and are highly influenced by the content of the image which makes it suitable for hiding secret messages as it is difficult to observe changes in such regions compared to smooth regions of the image. The algorithm uses the Red, Green, and Blue (RGB) component of the image and divides the image into overlapping blocks; using a shared key, one or more component of the image are selected; the blocks are rotated at random degree to generate a secret key; and the resulting image is then rearranged using raster scanning; the secret message is then encrypted and the bits are embedded after calculating the capacity of the image using a threshold.

In (Shanmuga, et al., 2012), an LSB technique that uses a pair of the pixel for embedding of the secret message was proposed. The results approach showed that the Stego-image had less distortion and was resistant to the steganalysis tools. The embedding was done in the sharper regions of the grayscale or color cover images used.

In the work of (Laskar & Hemachandran, 2013), a Pseudo Random Number Generator (PRNG) was used to pixels of the cover image where secret messages would be embedded. The embedding was done only in the Red component of the image. Thus, making it more difficult to detect thereby increasing security and reducing the distortion rate.

In (Thiyagarajan, et al., 2013), 3D geometric models for high steganography capacity was proposed. The algorithm works on the triangular meshes of the cover image, triangulating the meshes and embedding the secret message into newly added positions of the meshes. The resulting Stego-image withstood rotation, cropping, and scaling.

In (Juneja & Singh, 2013), an adaptive LSB technique that embeds a secret message in the Blue and partial green components of random pixels of the cover image was proposed. It integrated AES and embedded the data based on the available Most Significant Bits (MSBs) of the Red, Green, and Blue components of the selected pixels.

Though the spatial-domain methods are mostly used and known for its ease of implementation, it often falls a victim of steganalysis. Once detected, the secret message is often easily retrieved due to the fact that the embedding usually follows a linear approach. Therefore, this research proposes a technique which embeds a secret message in such a way that the cover image has less distortion and the retrieval of the message by a third party is made difficult.

3. Research Design and Methodology

The need for the algorithm was due to the ease of retrieval of information embedded in Stego-images once steganalysis tool identifies the Stego-image. Our proposed algorithm uses the LSB standard approach and embeds secret messages at the sixth (6^{th}), or seventh (7^{th}), or eight (8^{th}) bit planes of the cover image. That is, the embedding is strictly on one-bit plane that may be either of 6^{th}, or 7^{th}, or 8^{th}-bit plane; and not combined, in which case the embedding switches between planes that are used.

3.1 The Header

The Steganography algorithm has a defined structure which is provided as a header information as shown in Table 1.

A. Stego Type:

Stego Type uses a sequence of four (4) bits, the bits planes used for embedding of the secret message is defined. For example, when a sequence 0001 is specified, the embedding is done in the 7^{th} and 8^{th}-bit plane. In our approach, we use a runtime random number generator; for example, to have the 0001 sequence, the random number generator must have returned a one (1) and in that manner. More understanding when the embedding process is explained.

Stego Type 4bits	Steganography Switch Indicator 3bits	Password Indicator 1bit	Header Length 8bits
Data Length 16bits			
Data Start Location 8bits			Password Length 8bits
Password Start Location 8bits			App Signature 8bits
Password			
Password			
Data			
Data			

Table 1: Steganography Algorithm Header

B. Switch Indicator:

Switch Indicator uses a sequence of three (3) bits to specify the number of bits to be embedded in a particular sequence of bit plane before changing the plane. For example, if 010 are the bits representing the Switch Indicator, it means that five (5) bits of the secret message are embedded at five sequences of

bits plane before switching to embed the next five bits of the secret message in another sequence of five bit plane of the cover image. To determine the Switch Indicator in our approach, like in the case of Stego Type, we use a random number generator which when the returned value is three (3), the Switch Indicator bits would be 010. The essence of using random number generator is to ensure that given a group of Stego-image, a hacker or anyone for that matter cannot tell how the embedding was done which would make the retrieval of the secret message difficult.

C. Header Length:

This section of the header information stores the number bits used to store header information. It is necessary because, in the general use of the algorithm, a user may decide either or not to use a password which affects the length of the header information.

D. Data Length:

The data length section of the header information stores the length of the number of character making up the secret message. The sixteen (16) bits used would allow 65, 535 characters to be stored in a cover image. However, the allowed length also depends on the capacity of the cover image.

E. Data Start Location:

This section of the header information stores the stores the coordinates of the pixel where embedding of the data was started. This necessary because the desired dynamism and adaptiveness of the proposed algorithm should be readily predicted, thereby making retrieval difficult for a steganalysis tool. This data start location is determined at runtime and in such a way that consideration is given to the size of the hidden message and the size of the picture.

F. Password Length:

The password length section of the header information stores the length of the number of characters used as a password. This section is optional depending on whether the user used a password to add security or not. It is required to ensure that even when a third party to a communication has the software implementing the algorithm, the password when used would make retrieval of the secret message difficult. If not impossible.

G. Password Start Location:

When the password is used, like the data start location, the Password Start Location stores the pixel coordinate where the first bit of the password was stored and the manner of the algorithm, the retrieval is done using the location as a reference. This same as in the Data Start Location.

H. Application Signature:

To ensure adequate and easy retrieval of the hidden secret message from the Stego-image, a "Signature" is used to indicate for the application that the Stego-image has data embedded into it by the same application somewhere. This could be any sequence of bits of length eight (8).

3.2 Message Embedding Process

The uniqueness of every approach to steganography algorithm usually lies in the embedding process. The designed algorithm embeds a secret message in a dynamic and adaptive way so as to reduce distortion of the image against statistical analysis approach to steganalysis. The embedding process is described here.

Assuming at a particular execution of our proposed algorithm, we have a cover image I, a secret message M, Stego Type T, Switch Indicator S and other of the parameter are determined at the runtime. If the sequence of pixels bits of the cover image I starting from the pixel determined as the data start location are presented in their Red, Green, and Blue Planes as in Table 2

Embedding Location	Red Plane	Green Plane	Blue Plane
6th	01001101	00100110	11011010
7th	00011011	01010110	10100010
8th	00100000	10111100	01000011
6th	11110001	01111111	00001111
7th	11000001	11111100	01111110
8th	01000010	00001001	11110011

Table 2: A View of RGB Planes of Cover Image

If M is converted to sequence of bits given as 0 1 0 0 1 0 0 1 1 1 0 1 1 1 0 1 0 1 1 0 1 1…., assume also our Stego type T indicate that we embed the bits of our secret at the 6th, 7th, and 8th bit plane; while the switch indicator S indicate that the embedding of the bits switch after every three (3) bits then the embedding of M bits into the bits of I is as shown in Table 3

Embedding Location	Red Plane	Green Plane	Blue Plane
6th	01001011	00100110	11011010
7th	00011001	01010110	10100001
8th	00100000	10111101	01000011
6th	11110111	01111011	00001111
7th	11000011	11111110	01111101
8th	01000011	00001000	11110011

Table 3: A View of RGB Planes of Stego Image

From Table 5.2 and Table 5.3, it can be observed that when embedding is done in the 6th or 7th-bit plane, a step is taken to ensure that the difference between the integer value of the plane in the cover image and that in the Stego image is reduced. That is, when embedding at such locations, the following bits are checked and alternated accordingly. For example, embedding a 0 at the 6th bit plane of 01001101 would result in 01001011 which should have been 01001001 instead.

In our approach, the header information is strictly embedded at the 8th bit plane of the cover image starting from the very beginning pixel of the cover image.

3.3 Message Retrieval Process

To retrieve a secret message from a stego image, the algorithm retrieves the first sixty-four (64) bits of the 8th bit plane starting from the very beginning pixel of the stego image; it group the bits into the stego type, switch indicator, password indicator, header length, data length, data start location, password length, password start location, and application signature; verifies the application signature, verifies whether password was used to secure the secret image or not. If used, the password indicator would be 1 (one) otherwise 0 (zero). Requests password from the user if the password was 1 indicating it was used and validates the supplied password and then retrieves the secret message based on the given header information. That is, the retrieval of the main secret message follows after the stego type, switch indicator, data length, and data start location.

4. Results and Discussion

The proposed technique was implemented using Visual Basic .NET Framework and experimental results were compared to those of (Rejani, et al., 2015) as a benchmark. The steganography was done on PNG images which are lossless and analysis of the stego images was done using Matlab. The Peak-Signal-To-Noise Ratio (PSNR), Mean Square Error (MSE) of the images were computed. These parameters can be computed using the following equations.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{i=0}^{n-1}[I(i,j) - K(i,j)]^2 \quad ----------(1)$$

$$PSNR = 10.Log_{10}\left[\frac{MAX_I^2}{MSE}\right] \quad ----------(2)$$

Where MAX_I is the maximum possible pixel value of the image; the more MSE tends to diminish, the more less distortive the stego image is, implying more quality, and thus less prone to perceptibility. Also, the greater the value of PSNR the better the technique the steganography.

The results of the experiment are as shown in Table 4.

From the results obtained, Comparing the results obtained with this proposed techniques with that of (Rejani, et al., 2015) which also was compared to a number of other techniques, it is clear that though our approach embedded more data (53 chars for this experiment), change in the size of the image is almost equal. On the general, our approach showed to have less effect on the quality of the image as our PSNR value is generally greater. And finally, on the embedding and retrieval time, our approach is superior always taking less than a second to embed and retrieve secret messages.

Cover Image	Fig.2A	Fig.3A	Fig.4A	Fig.5A
Stego Image	Fig.2B	Fig.3B	Fig.4B	Fig.5B
Cover Image Size (KB)	1,226	421	3,072	247
Stego Image Size (KB)	1,226	447	3,073	248
MSE	5.79e-05	1.50e-03	6.19e-05	2.27e-04
PSNR (dB)	90.54	76.55	90.26	84.81
Embedding Time (ms)	0.9496	1.7356	1.2397	1.177
Retrieval Time (ms)	52.4669	52.6279	50.5897	48.6148
Bits Length Embedded	420	420	420	420
Embedding Planes	7th	7th	7th	6th or 8th
Switch Indicator (bits)	1	5	1	3

Table 4: Experimental Results of Proposed Technique

Fig.2A: Cover Image

Fig.2B: Stego Image

Fig.3A: Cover Image

Fig.3B: Stego Image

Fig.4A: Cover Image

Fig.4B: Stego Image

Fig.5A: Cover Image

Fig.5B: Stego Image

5. Conclusion

The implemented technique which is based on random embedding approach is a vital security technique as not even the sender can predict the particular pattern of embedding used for a particular stego image. This certainly improves the security of data communication.

Also, since the proposed technique has a proposed structure, more different techniques that embed the secret messages using mathematically defined patterns could be used and the information stored in the header. Therefore, we recommend that more research should be carried out to verify more potentials of this proposed technique.

Also, to add more security, compression and encryption are usually used; the purpose of compression is to reduce the amount of data to be embedded, that way, reduced distortion it done to the cover image which is not perceptible to the eyes. Therefore, such techniques to could be combined with the proposed technique for a more advanced security.

Reference

Gaikwad, D. & Wagh, S., 2010. Colour Image Restoration For An Effective Steganography. *Imanager's Journal on Software Engineering, Vol.4 .No.3,* pp. 65-71.

Haynes, K. L., 2011. Using Image Steganography to Establish Covert Communication Channels. *International Journal of Computer Science and Information Security, Vol 9, No.9,* pp. 1 - 7.

Hemalatha, S., Acharya, U., Renuka, A. & Kamnath, P. R., 2013. A Secure and High Capacity Image Steganography Technique. *Signal & Image Processing – An International Journal Vol.4, No.1,* pp. 83 - 89.

Hemalatha, S. A. U. & Renuka, A., 2013. Comparison of Secure and High Capacity Color Image Steganography Techniques in RGB and YCBCR domains. *International Journal of Advanced Information Technology, Vol.3, No.3,* pp. 1 - 9.

Jain, R. & Kumar, N., 2012. Efficient Data Hiding Scheme Using Lossless Data Compression and Image Steganography. *International Journal of Engineering Science and Technology (IJEST). Vol. 4 No.08,* pp. 3908 - 3915.

Juneja, M. & Singh, P. S., 2013. A New Approach for Information Security Using an Improved Steganography Technique. *Journal of Info.Pro.Systems, Vol 9, No:3,* pp. 405 - 424.

Laskar, S. A. & Hemachandran, K., 2013. Steganography Based on Random Pixel Selection For Efficient Data Hiding. *International Journal of Computer Engineering and Technology. Vol.4, Issue 2,* pp. 31 - 44.

Medeni, M. & Souidi, E. M., 2010. Steganography and Error Correcting Codes. *International Journal of Computer Science and Information Security, Vol.8.No.8,* pp. 147-149.

Rejani, R., Murugan, D. & Deepu, V. K., 2015. PIXEL PATTERN BASED STEGANOGRAPHY ON IMAGES. *ICTACT JOURNAL ON IMAGE AND VIDEO PROCESSING, VOLUME: 05, ISSUE: 03,* pp. 991 - 997.

Sadkhan, S. B., Al-Barky, A. M. & Muhammad, N. N., 2009. "An Agent based Image Steganography using Information Theoretic Parameters. *MASAUM Journal of Computing, Vol. 1, No. 2,* pp. 258 - 268.

Sarkar, A. et al., 2007. *SECURE STEGANOGRAPHY: STATISTICAL RESTORATION OF THE SECOND ORDER DEPENDENCIES FOR IMPROVED SECURITY.* Honolulu, HI, IEEE, pp. II-277 - II-280.

Seyyedi, A. S. & Ivanove, N., 2014. Statistical Image Classification for Image Steganography Techniques. *International Journal of Image, Graphics and Signal Processing,* pp. 19 - 24.

Shanmuga, S. P., Mahesh, K. & Kuppusamy, K., 2012. Efficient Steganography Method To Implement Selected Least Significant Bits in Spatial Domain. *International Journal of Engineering Research and Applications, Vol2, Issue 3,* pp. 2632 - 2637.

Sharmila, B. & Shanthakumari, R., 2012. "Efficient Adaptive Steganography For Colour Images Based on LSBMR Algorithm. *ICTACT Journal on Image and Video Processing, Vol. 2, Issue:03,* pp. 387 - 392.

Sun, Y. & Lui, F., 2010. *Selecting Cover for Image Steganography by Correlation Coeffiecient.* s.l., s.n., pp. 159 - 162.

Swain, G. & Lenka, S. K., 2012. A Technique for Secret Communication Using a New Block Cipher with Dynamic Steganography. *International Journal of Security and Its Applications; Vol. 6 No. 4,* pp. 13 - 24.

Thiyagarajan, P. et al., 2013. Pattern Based 3D Image Steganography. *3D Research center, Kwangwoon University and Springer 2013, 3DR Express,* pp. 1 - 8.

IJISE is a FTMS Publishing Journal