International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:10, No:6, 2016

An Optimal Steganalysis Based Approach for Embedding Information in Image Cover Media with Security

Ahlem Fatnassi, Hamza Gharsellaoui, Sadok Bouamama

Abstract—This paper deals with the study of interest in the fields of Steganography and Steganalysis. Steganography involves hiding information in a cover media to obtain the stego media in such a way that the cover media is perceived not to have any embedded message for its unintended recipients. Steganalysis is the mechanism of detecting the presence of hidden information in the stego media and it can lead to the prevention of disastrous security incidents. In this paper, we provide a critical review of the steganalysis algorithms available to analyze the characteristics of an image stego media against the corresponding cover media and understand the process of embedding the information and its detection. We anticipate that this paper can also give a clear picture of the current trends in steganography so that we can develop and improvise appropriate steganalysis algorithms.

Keywords—Optimization, heuristics and metaheuristics algorithms, embedded systems, low-power consumption, Steganalysis Heuristic approach.

I. Introduction

NOWADAYS, multiprocessor architectures provide a rich computing environment from which a wide range of problem domains, including real-time applications can benefit [1]. The Internet, for example, has revolutionized the modern world and the numerous Internet based applications that get introduced these days add to the high levels of comfort and connectivity in every aspects of human life [2]. As of September 2009, approximately 1.73 billion people worldwide use Internet for various purposes ranging from accessing information for educational needs to financial transactions, procurement of goods and services [2]. As the modern world is gradually becoming ”paperless” with huge amount of information stored and exchanged over the Internet, it is imperative to have robust security measurements to safeguard the privacy and security of the underlying data.

Cryptography techniques [4] have been widely used to encrypt the plaintext data, transfer the ciphertext over the Internet and decrypt the ciphertext to extract the plaintext at the receiver side. However, with the ciphertext not really making much sense when interpreted as it is, a hacker or an intruder can easily perceive that the information being sent on the channel has been encrypted and is not the plaintext. This can naturally raise the curiosity level of a malicious hacker or intruder to conduct cryptanalysis attacks on the ciphertext (i.e., analyze the ciphertext against the encryption algorithms and decrypt the ciphertext completely or partially) [4]. It would be rather more prudent if we can send the secret information, either in plaintext or ciphertext, by cleverly embedding it as part of a cover media (for example, an image, audio or video carrier file) in such a way that the hidden information can not be easily perceived to exist [3] for the unintended recipients of the cover media. This idea forms the basis for Steganography, which is the science of hiding information by embedding the hidden (secret) message within other, seemingly harmless images, audio, video files or any other media. Indeed, steganography and Steganalysis have many features.

- Steganography protects the intellectual property rights and enables information transfer in a covert manner such that it does not draw the attention of the unintended recipients.
- Steganalysis is the science of detecting the presence of hidden data in the cover media files and is emerging in parallel with steganography.
- Steganalysis has gained prominence in national security and forensic sciences since detection of hidden (ciphertext or plaintext) messages can lead to the prevention of disastrous security incidents.
- Steganalysis is a very challenging field because of the scarcity of knowledge about the specific characteristics of the cover media (an image, an audio or video file) that can be exploited to hide information and detect the same.

The approaches adopted for steganalysis also sometimes depend on the underlying steganography algorithm(s) used. Throughout the paper, the terms ”algorithm”, ”approach”, ”method” and ”technique” are used interchangeably. They mean the same. Also, for discussion purposes, the term ”cover” is used to refer to a media devoid of any hidden secret information and the term ”stego” is used to refer to a media that has hidden secret information.

In this paper, we review the steganalysis algorithms available for the used cover media: Image. Image Steganalysis algorithms explore the strong inter-pixel dependencies that are characteristic of natural images. We present our approach in Section III. Then, we describe the simulation results and discussion in Section IV. Finally, Section V concludes the paper.

Ahlem Fatnassi and Sadok Bouamama are with the National Engineering School of Manouba, Manouba University, Tunisia (e-mail: ahlemfatn@yahoo.fr, Sadok.Bouamama@ensi.rnu.tn).

Hamza Gharsellaoui is with the National Engineering School of Carthage, Carthage University, Tunisia (e-mail: gharsellaoui.hamza@gmail.com).

II. Background and State of the Art

Steganography is the art of passing information through original files in a manner that the existence of the message is unknown. The term steganography is arrived from Greek word means, "Covered Writing". The innocent files can be referred to as text, image, audio or video as appropriate. After embedding the secret message it is referred to as stego-medium. Also, with the development of digital signal processing (DSP), the boost in computer power, the internet and with information theory and coding theory, steganography has gone "digital". The objective of this section work is to present and describe performance enhancements over the steganography cover mediums and methods proposed in the literature.

A. Image Steganalysis

Algorithms for image steganalysis are primarily of two types: Specific and Generic.

The specific approach represents a class of image steganalysis techniques that very much depend on the underlying steganographic algorithm used and have a high success rate for detecting the presence of the secret message if the message is hidden with the algorithm for which the techniques are meant for.

The Generic approach represents a class of image steganalysis techniques that are independent of the underlying steganography algorithm used to hide the message and produces good results for detecting the presence of a secrete message hidden using new and/or unconventional steganographic algorithms [14]. The image steganalysis techniques under both the specific and generic categories are often designed to detect the presence of a secret message and the decoding of the same is considered complementary not mandatory.

1) Specific Image Steganalysis Algorithms: Image steganography algorithms are more often based on an embedding mechanism called Least Significant Bit (LSB) embedding. Each pixel in an image is represented as a 24-bitmap value, composed of 3 bytes representing the R, G and B values for the three primary colors Red, Green and Blue respectively [14]. A higher RGB value for a pixel implies larger intensity. For instance, a pixel p represented as FF FF FF_{16} is composed of all of these three primary colors at their maximum intensity and hence the color represented by this pixel is white. LSB embedding exploits the fact that changing the least significant bit of each of the three bytes of a pixel would produce only a minor change in the intensity of the color represented by the pixel and this change is not perceptible to the human eye [5]. For example, changing the color values of pixel p to FE FE FE_{16} would make the color darker by a factor of 1/256. Steganography algorithms based on LSB embedding differ on the pattern of modification a modification of randomly chosen pixels or modification restricted to pixels located in certain areas of the image. Images can be represented in different formats, the three more commonly used formats are: GIF (Graphics Interchange Format), BMP (Bit Map) and JPEG (Joint Photographic Exchange Group). Each of these image formats behaves differently when a message is embedded in it. Accordingly, there exist different image steganalysis algorithms for each of these three image formats. We now discuss the algorithms for each of these formats.

2) Palette Image Steganalysis: Palette image steganalysis is primarily used for GIF images. The GIF format supports up to 8 bits per pixel and the color of the pixel is referenced from a palette table of up to 256 distinct colors mapped to the 24-bit RGB color space. LSB embedding of a GIF image changes the 24-bit RGB value of a pixel and this could bring about a change in the palette color (among the 256 distinct colors) of the pixel [14]. The strength of the steganographic algorithm lies in reducing the probability of a change in the palette color of the pixel and in minimizing the visible distortion that embedding of the secret image can potentially introduce. The steganalysis of a GIF stego image is conducted by performing a statistical analysis of the palette table against the image and the detection is made when there is an appreciable increase in entropy (a measure of the variation in the palette colors). The change in entropy is maximal when the embedded message is of maximum length [6].

3) Raw Image Steganalysis: The raw image steganalysis technique is primarily used for BMP images that are characterized by a lossless LSB plane. LSB embedding on such images causes the flipping of the two grayscale values. The embedding of the hidden message is more likely to result in averaging the frequency of occurrence of the pixels with the two gray-scale values. For example, if a raw image has 20 pixels with one gray-scale value and 40 pixels with the other gray-scale value, then after LSB embedding, the count of the pixels with each of the two gray-scale values is expected to be around 30. This approach was first proposed by Westfeld and Pfitzmann [7], and it is based on the assumption that the message length should be comparable to the pixel count in the cover image (for longer messages) or the location of the hidden message should be known (for smaller messages). Dumitrescu et. al [8] proposed another steganalysis algorithm for grayscale images. This algorithm assumes an image to be made up of horizontally adjacent pixels and classifies the set of all such pixel pairs (a, b) into four subsets depending on whether a and b are odd or even and whether a $<$ b, a $>$ b or a = b. The pixel values get modified when message embedding is done in the LSB plane, thereby leading to membership modifications across these four subsets. A statistical analysis on the changes in the membership of the pixels in the stego image leads to the detection of the length of the hidden message. Fridrich et. al. [9] proposed a steganalysis technique that studies color bitmap images for LSB embedding and it provides high detection rates for shorter hidden messages. This technique makes use of the property that the number of unique colors for a high quality bitmap image is half the number of pixels in the image. The new color palette that is obtained after LSB embedding is characterized by a higher number of close color pairs (i.e., pixel pairs that have a maximum difference of one count in either of the color planes). We say that two colors (R1, G1, B1) and (R2, G2, B2) are close if $|R1\text{-}R2| \leq 1$ and $|G1\text{-}G2| \leq 1$ and $|B1\text{-}B2| \leq 1$. Let P be the ratio of the close color pairs

to the total number of unique colors in the cover image, P be the ratio of close color pairs to the total number of unique colors in a stego image obtained by embedding a new message of particular length in a cover image and P" be the ratio of the close color pairs to the total number of unique colors when the cover image is further embedded in the stego image. If the hidden message is of considerable length, it has been observed that $P^{'} >$ P and $P^{"} \cong$ P. For shorter messages, the values of P and $P^{'}$ will be closer and detection may not be possible [14]. Also, the above technique will not work if the cover image stored in lossless format has a higher number of unique colors (more than half the number of pixels).

4) JPEG Image Steganalysis: JPEG is a popular cover image format used in steganography. Two well-known Steganography algorithms for hiding secret messages in JPEG images are: The F5 algorithm and Outguess algorithm [10]. The F5 algorithm uses matrix embedding to embed bits in the DCT (Discrete Cosine Transform) coefficients in order to minimize the number of changes to a message. However, F5 mutates the histogram of DCT coefficients. Fridrich et al. [6] propose a technique for estimating the unaltered histogram to find the number of changes and length of the secret message. The process involves cropping the JPEG image by four columns and then applying a quantization table to re-compress the image. The resulting DCT coefficient histogram will be a close estimate of the original. Fridrich et al. [6] also propose a technique to attack the Outguess embedding algorithm. The Outguess algorithm makes a random walk and embeds its message bits in the LSB of some of the DCT coefficients. The other DCT coefficients are then adjusted to keep the original histogram intact. As a result, the F5 steganalysis method involving estimation of the original histogram will be useful in the steganalysis of the Outguess algorithm. Also, the process of embedding a message into an unadulterated image introduces noise in the DCT coefficients, leading to increased spatial discontinuities in the 8x8 JPEG image blocks and partial cancellation of the changes made to the LSB of DCT coefficients. Furthermore, when another message is embedded into a stego image, the increase in discontinuities tends to be smaller. The nature of the increase or decrease in discontinuities is widely employed to gauge the size of the hidden message.

B. Generic Image Steganalysis Algorithms

The generic steganalysis algorithms, usually referred to as Universal or Blind Steganalysis algorithms, work well on all known and unknown steganography algorithms. These steganalysis techniques exploit the changes in certain innate features of the cover images when a message is embedded. The focus is on to identify the prominent features of an image that are monotonic and changes statistically as a result of message embedding [13]. The generic steganalysis algorithms are developed to precisely and maximally distinguish these changes. The accuracy of the prediction heavily depends on the choice of the right features, which should not vary across images of different varieties. Avcibas et al. [11] use a set of Image Quality Metrics (IQMs) to develop a discriminator

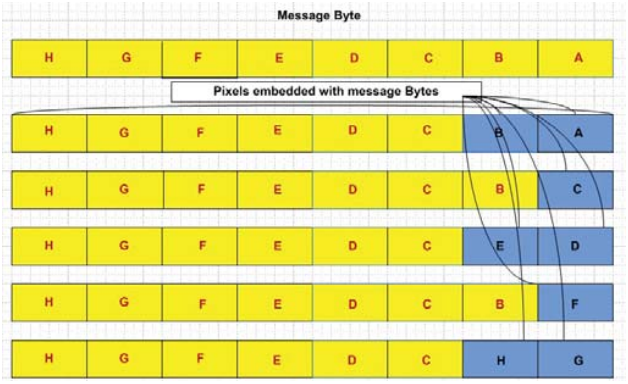

Fig. 1 Proposed OLSB Algorithm

algorithm that differentiates cover images from stego images. The authors use IQMs as a steganalysis tool rather than as an indicator of image quality or algorithmic performance. The ANOVA (Analysis of Variance) statistical test is used to rank the IQMs based on their F-scores and identify the embedding of the message. The success of the approach lies in the identification of IQMs that are very sensitive to steganography and whose changes as a result of message embedding can be measured well. To increase the chances of a successful detection, several IQMs are normally employed to measure the distortions at different levels of sensitivity. For example, the mean square values for the Human Visual System (HVS)-weighted errors demonstrate more sensitivity to pure blur; while the Gradient measure responds to changes in the texture and the image periphery. The message embedding steganography algorithms differ in the changes brought to the different IQMs. Avcibas et al. [12] propose another steganalysis technique that analyzes every seventh and eighth bit planes of an image and measures their binary similarity. The technique measures the correlation between the adjacent bit planes that gets affected as a result of message embedding. The hypothesis is that message embedding decreases the correlation between two contiguous bit planes.

III. Proposed Approach

Based on [13], a digital image is described using a 2-D matrix of the color intestines at each grid point (i.e. pixel). Typically gray images use 8 bits, whereas colored utilizes 24 bits to describe the color model, such as RGB model. The Steganography system uses an image as the cover, there are several techniques to conceal information inside cover-image. The spatial domain techniques manipulate the cover-image pixel bit values to embed the secret information. The secret bits are written directly to the cover image pixel bytes. Consequently, the spatial domain techniques are simple and easy to implement. The Least Significant Bit (LSB) is one of the main techniques in spatial domain image Steganography. The LSB is the lowest significant bit in the byte value of the image pixel. The LSB based image steganography embeds the secret in the least significant bits of pixel values of the cover image (CVR).

The concept of LSB Embedding is simple. It exploits the fact that the level of precision in many image formats is far greater than that perceivable by average human vision. Therefore, an altered image with slight variations in its colors will be indistinguishable from the original by a human being, just by looking at it [13]. In conventional LSB technique, which requires eight bytes of pixels to store 1 byte of secret data. In contrast, in our optimal Least Significant Bit (OLSB) technique, just five bytes of pixels are sufficient to hold one message byte. Rest of the bits in the pixels remains the same. So, in our optimal proposed example, we need 5 bytes against 8 bytes in conventional LSB technique. For security reasons, only encryption may not be enough, hence our optimal proposed approach includes Steganography wherein encrypted data is hid into the image and then image is transmitted in the network.

Encryption method: Read the secret and cover image and convert them into gray scale images, then check to handle the principle constraint of the size. This constraint mentions that size of the secret image should be less than cover image. Now, we will adopt the [13] method principle for the encryption, decryption process. Encode the secret image into binary using bit gate command and divide it into RGB parts then substitute MSB bits of secret image into LSB bits of cover image. Hide the password with Stego image and send using the network transmission medium.

Decryption method: The reverse process takes place at the receiving end, Stego image can be decrypted using password. We use in our work MATLAB due to its high-performance as a language for technical computing. Matlab function is an easy to use, user interface function (HMI) that guides a user through the process of either encoding & decoding a message into or from the image respectively. In this work, Matlab is implemented for processing OLSB steganography technique with different frame size 256*256, 128*128, 64*64 and simulation results are shown. There are mainly four steps involved in implementing OLSB steganography as shown below.

IV. Simulation Results and Discussion

In this section, we describe all the four steps needed for our proposed approach process. To apply the proposed OLSB algorithm, consider that we have to hide the secret image ”Fig. 3” in cover image ”Fig. 2”: The first step is described in the following subsection paragraph. Fig. 2 shows the cover image used.

A. Conversion of Image to Matrix

In the conversion process of image to matrix, we convert the input cover image into matrix values which is stored in a text file. Firstly an image is read from computer, the original image is in the form of RGB which is converted into grey image. The grey image is resized to a particular size of 256*256. Each image has intensity values for every pixel, here these intensity values are stored into a text file. The secret image is shown in Fig. 3.

Fig. 2 Cover Image

Fig. 3 Secret Image

B. Embedding Process

After completion of image to matrix the next step is to embed a message into an image. The image obtained during this process is called as stegano-embed image. The message is embedded into the intensity values of image obtained during image to matrix conversion. The stegano image is shown in Fig. 4.

C. Conversion of Matrix to Image

In this stage, intensity values are converted back to image. The image obtained has message embedded into it. The cover image and the image obtained here have to be identical. Hence the objective of Steganography is satisfied.

D. Extraction Process

In this process, we extract the message which was embedded during embedding process. At first declare a message byte, here the size of the message is 8 bits. Read a pixel from the array starting from address = 0. Extract the LSB and replace the i^{th} bit in the message byte where i = 1 to 8 Address = address =1. When i = 8, a byte is extracted. Repeat for extracting next byte and to respect the method of 2 bits by one bit in every two successive bytes.

E. Discussion and Interpretations

Here we send secret image + cover Image = stego image from PC to Controller. Then controller decode original image

Fig. 4 Stegano Image

(secret image) from cover image and transmit to PC. We need new powerful Steganalysis techniques that can detect messages without prior knowledge of the hiding algorithm (blind detection). The detection of very small messages is also a significant problem. Finally, we need adaptive techniques that do not involve complex training stages. The comparison between the LSB method and the proposed method using experimental results demonstrate that our original and optimal proposed method keeps distortion low and uses a low memory capacity and a less execution time due to the reduced number of needed bytes, 5 against 8 in conventional LSB technique.

V. Conclusion

In this paper, we have analyzed the steganalysis algorithms available for one used domain of steganography (Image). Image steganalysis algorithms can be classified into two broad categories: Specific and Generic. The enhanced OLSB technique described in this paper helps to successfully hide the secret data into the cover image with minimum distortion made to the cover image and with a minimum of memory space used. This method is essential for construction of accurate targeted and blind steganalysis methods for JPEG, BMP and PNG images. Experimental results of the modified method shows that OLSB is grater than the conventional method of LSB replacement. In summary and as a future work, we will try to propose best and original other steganalysis algorithms for the three commonly used domains of steganography covers (image, audio and video).

References

[1] H. Gharsellaoui, M. Khalgui, S. Ben Ahmed. Preemptive Hard Real-time Scheduling of Reconfigurable OS Tasks on Multiprocessors Embedded Control Systems. *PECCS - Proceedings of the 4th International Conference on Pervasive and Embedded Computing and Communication Systems*, 192–197, Lisbon, Portugal, 7-9 January, 2014.

[2] H. Kekre, A. Athawale, T. Sarode, S. Thepade and K. Sagvekar. *Steganography Using Dictionary Sort on Vector Quantized Codebook*. International Journal of Computer Science and Security (IJCSS), vol. 4, no. 4, pp. 392–402, 2010.

[3] V. Michopoulos, L. Guan, G. Oikonomou, I. Phillips. DCCC6: Duty Cycle-aware congestion control for 6LoWPAN networks. *Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 278–283, 2012.

[4] D. Stinson. *Cryptography: Theory and Practice*. 2nd Edition, Chapman and Hall, CRC, 2002.

[5] N. Johnson and S. Jajodia. Steganalysis of Images Created using Current Steganography Software. *Lecture Notes in Computer Science, vol. 1525*, pp. 32–47, Springer Verlag, 1998.

[6] J. Fridrich, M. Goljan, D. Hogea and D. Soukal. *Quantitative Steganalysis of Digital Images: Estimating the Secret Message Length*. ACM Multimedia Systems Journal, Special issue on Multimedia Security, vol. 9, no. 3, pp. 288–302, 2003.

[7] A. Westfeld and A. Pfitzmann. *Attacks on Steganographic Systems*. Proceedings of the 3rd International Workshop on Information Hiding, pp. 61–76, 1999.

[8] S. Dumitrescu, X. Wu and N. Memon. *On Steganalysis of Random LSB Embedding in Continuous tone Images*. Proceedings of the International Conference on Image Processing, vol. 3, pp. 641– 644, June 2002.

[9] J. Fridrich and M. Long. *Steganalysis of LSB Encoding in Color Images*. Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), vol. 3, pp. 1279–1282, New York, NY, USA, July August 2000.

[10] A. Westfeld. *F5 A Steganographic Algorithm*. Lecture Notes in Computer Science, vol. 2137, pp. 289–302, January 2001.

[11] I. Avcibas, N. Memon and B. Sankur. *Steganalysis using Image Quality Metrics*. IEEE Transactions on Image Processing, vol. 12, no. 2, pp. 221–229, February 2003.

[12] I. Avcibas, N. Memon and B. Sankur. *Image Steganalysis with Binary Similarity Measures*. Proceedings of the IEEE International Conference on Image Processing, vol. 3, pp. 645–648, June 2002.

[13] N. Champakamala, K. Padmini, D.K. Radhika. Least Significant Bit algorithm for image steganography. *International Journal of Advance Computer Technology, vol(3), N. 4*, pages 34–38, 2013.

[14] N. Meghanathan and L. Nayak. Steganalysis Algorithms For Detecting The Hidden Information In Image, Audio And Video Cover Media. *International Journal of Network Security & Its Application (IJNSA), Vol.2, No.1*, pp. 43–55, 2010.