Volume 2, No. 3, March 2011

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

RESEARCH PAPER

IMAGE STEGANOGRAPHY BASED ON POLYNOMIAL FUNCTIONS

Ms.Soniya Vijayakumar
Lecturer,
Department of Computer Science,
New Horizon College of Engineering,
Bangalore, Karnataka, India
soniyaannajoe@gmail.com

Abstract: Steganography hides the text message in the bytes of the cover medium, which is the container, used to hide the message. Currently most of the steganography algorithms work by modifying the Least Significant Bit (LSB) of the consecutive bytes of the cover medium to store the secret data. The main drawback of this algorithm is that hidden message can be retrieved easily through steganalysis since the messages are stored in consecutive bytes. In this paper, two novel methods for selecting the bytes of the cover medium in which the secret data bits to be stored are proposed. In both methods, the byte of the cover medium where the secret data is to be stored is selected randomly. The first method is based on a linear polynomial function and the second method is based on a quadratic polynomial function. The present work compares the efficiency of the two methodologies.

Keywords: Steganography, Data Hiding, Least Significant bit(LSB), Linear polynomial function, Quadratic polynomial function

INTRODUCTION

Steganography is a technology that hides a message within an object, a text, or a picture. Steganography hides either cleartext or encrypted message in the cover medium, which is the container used to hide the message. Image Steganography uses image as the cover medium [1]. Encrypted message can be either based on public key or private key encryption algorithms. It is often confused with cryptography, not in name but in appearance and usage. The easiest way to differentiate the two is to remember steganography conceals not only the contents of the message but also the mere existence of a message. Figure 1 shows a generic steganographic system.

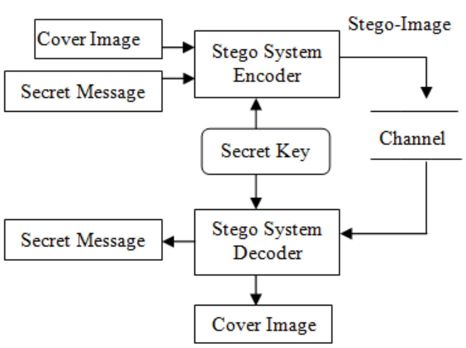

Figure 1. Steganographic Process

In this paper, two novel methods for selecting the bytes of the cover medium, in which the secret data to be stored are proposed. The secret data is stuffed in the least significant bits of the selected bytes. The first method is based on a polynomial function of order one, which is used to determine the byte of the cover medium where the secret data is going to be stored. The second method is based on a Quadratic polynomial function which determines the bytes of the cover medium to be used for storing the secret message. And the paper discusses the advantages of above discussed methods.

Currently most of the steganography algorithms work by modifying the Least Significant Bit (LSB) of the consecutive bytes of the cover medium to store the secret data [2]. The main drawback of this algorithm is that hidden message can be retrieved easily through steganalysis since the messages are stored in consecutive bytes. Steganalysis is the method by which to detect the presence of a hidden message and attempt to reveal the true contents of the message [3] .Steganographic technique embed a message inside a cover. Various features characterize the strength and weaknesses of the methods. The relative importance of each feature depends on the application. They are Capacity, Robustness, Invisibility, and Security.

a) **Capacity**
The notion of capacity in data hiding indicates the total number of bits hidden and successfully recovered by the steganographic system.

b) **Robustness**
Robustness refers to the ability of the embedded data to remain intact if the steganographic system undergoes transformation, such as linear and non-linear filtering; addition of random noise; and scaling, rotation, and loose compression.

c) **Invisibility**
This concept is based on the properties of the human visual system or the human audio system. The embedded information is imperceptible if an average human subject is unable to distinguish between carriers that do contain hidden information and those that do not.

d) **Security**
It is said that the embedded algorithm is secure if the embedded information is not subject to removal after being

discovered by the attacker and it depends on the total information about the embedded algorithm and secret key.

Existing method of data hiding

The simplest approach for hiding data within an image file is called Consecutive Least Significant Bit (LSB) insertion. Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image. 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.
In the existing method, it takes the binary representation of the hidden data and overwrites the LSB of each consecutive byte within the cover image. As an example, suppose that we have three adjacent pixels (nine bytes) with the following RGB encoding:

10010101 00001101 11001001
10010110 00001111 11001010
10011111 00010000 11001011

Now suppose we want to "hide" the following 9 bits of data 101101101. If we overlay these 9 bits over the LSB of the 9 consecutive bytes above, we get the following (where bits in **bold** have been changed):

1001010**1** 0000110**0** 1100100**1**
1001011**1** 0000111**0** 1100101**1**
1001111**1** 0001000**0** 1100101**1**

The technique is providing least security for the secret data.

PROPOSED STEGANOGRAPHY METHODOLOGIES

This section explains new methodologies for data hiding in Images.

Linear polynomial function

In linear polynomial function method, a polynomial function of order one is used to determine the byte of the cover medium where the secret data is going to be stored. The general format of the 1st degree polynomial function is Q = a.X + c. Based on this equation, the subsequent byte where the secret bit is to be kept is determined by

$$X_{i+1} = (a.X_i + c) \bmod m$$

where a, c and m are constants. X_i determines the current byte position in the cover medium where the secret bit is stored and X_{i+1} determines the next byte position where secret data can be stored.
Assume that a = c = X_1 = 3 and m= 5. And assume that the secret message that we need to hide is **101**. And the source image data is

10100011 00001010 10001000
10000011 10101011 10100011
10001011 10011001 10100001
10101000 11101110 11110001

The first secret bit will be stored in X_1 +1, which is the 4th position of the source file. So the 4th byte after inserting the secret bit will be 10000011. Now, the next bit of the secret data is to be stored based on the following calculation $X_2 = (a.X_1 + c) \bmod 5$. And it is 2 after calculation. So the next secret bit is to be stored in the byte position which is the sum of $X_1 + 1$ and $X_2 +1$ (addition of 1 is to get a unique cover medium byte since the above function can return 0) and it is 7th position. So the 7th byte after adding the secret bit is 10001010. Now the 3rd secret bit should be stored in the byte position which can be calculated based on $X_3 = (a.X_2 + c) \bmod 5$. And the value of X_3 is 4. So the 3rd secret bit should be inserted in the 12th position which is X_3 +1 position from the current position.

10100011 00001010 10001000
1000001**1** 10101011 10100011
1000101**0** 10011000 10100001
10101000 11101111 1111000**1**

Quadratic polynomial function

In this method, a 2nd degree Polynomial function is used to determine the byte of the cover medium where the secret data is going to be stored. The general format of the quadratic polynomial function is $Q = a.X^2 + b.X + c$. Based on this equation, the subsequent byte where the secret bit is to be kept is determined by

$$X_{i+1} = (a.X_i^2 + b.X_i + c) \bmod m$$

where a, b, c and m are constants. X_i determines the current byte position in the cover medium where the secret bit is stored and X_{i+1} determines the next byte position where secret data can be stored.
Assume that a = b = c = X_1 = 2 and m= 5. And assume that the secret message that we need to hide is **101**. And the source image data is

10100011 00001010 10001000
10000011 10101011 10100011
10001011 10011001 10100001
10101000 11101110 11110001

The first secret bit will be stored in X_1 +1, which is the 3rd position of the source file. So the 3rd byte after inserting the secret bit will be 10001001. Now, the next bit of the secret data is to be stored based on the following calculation
$X_2 = (a.X_1^2 + b.X_1 + c) \bmod 5$. After calculation X_2 is 4. So the next secret bit is to be stored in the byte position which is the sum of $X_1 + 1$ and X_2+1 (addition of 1 is to get a unique cover medium byte since the above function can return 0) and it is 8th position. So the 8th byte after adding the secret bit is 10011000. Now the 3rd secret bit should be stored in the byte position which can be calculated based on $X_3 = (a.X_2^2 + b.X_2 + c) \bmod 5$. And the value of X_3 is 2. So the 3rd secret bit should be inserted in the 11th position which is X_3 +1 position from the current position.

10100011 00001010 1000100**1**
10000011 10101011 10100011
10001011 1001100**0** 10100001
10101000 1110111**1** 11110001

Advantages of Proposed Methods

The proposed algorithm selects the bytes randomly. So this is much stronger than existing algorithms where the bytes are selected consecutively.
These proposed methods embed the secret data without affecting the quality of the cover image as it modifies only the randomly selected LSBs.

RESULTS AND DISCUSSION

In this section, a comparison is made with illustration after applying the steganographic techniques proposed in the paper. Figure 2 shows the input image without any text data embedded which is used for the steganographic process

Figure 2.Input Image

Figure 3 shows the input image in figure 2 after applying Linear Polynomial function steganographic technique. From the illustration, it is clear that there is no difference that can be captured with naked eye.

Figure 3.Stego-image (Linear polynomial function method)

Figure 4 shows the input image in figure 2 after applying Quadratic Polynomial function steganographic technique.

Figure 4.Stego-image (Quadratic polynomial function method)

CONCLUSION

A detailed overview is given on the steganography techniques in the beginning of the paper. After that, a detailed explanation has been given for the newly proposed steganographic algorithms based on polynomial functions and their advantages over the existing method. Also discussion has been carried out to illustrate the new algorithms with examples and it concludes that the proposed algorithms which select the cover medium bytes randomly to store the secret text is better and stronger than the conventional steganographic techniques.

REFERENCES

[1] Neil F. Johnson, Sushil Jajodia, "Exploring Steganography: Seeing the Unseen," Computer, vol. 31, no. 2, pp. 26-34, Feb. 1998
[2] R. Chandramouli, Nasir Memon, “Analysis of LSB Based Image Steganography Techniques” Proc. IEEE ICIP 2001 pp1019-1022.
[3] Li Zhi, Sui Ai Fen, “Detection of Random LSB Image Steganography” IEEE 2004 pp2113-2117.