Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 2 Issue 11, November - 2015

LSB Based Audio Steganography Using Pattern Matching

Mr. Ratul Choudhury
Student, Dept of Computer Sc. & Engg Dept.
University of Calcutta
Kolkata, India
ratul.chowdhury@iemcal.com

Prof. Samir Kumar Bandyopadhyay
Professor, Dept of Computer Sc. & Engg. Dept.
University of Calcutta
Kolkata, India
skb1@vsnl.com

***Abstract*— Security of the digital information is becoming primary concern prior to transmitting the information itself via some media. Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction. In this proposed method, the carrier file and the secret message are taken into audio format and a pattern matching algorithm is used which will identify a part of the message from the carrier file and return its location. Then by using the standard LSB method the location will be embedded with the carrier file to form the stego file.**

Keywords— Communication, Security, Steganography

I. INTRODUCTION

Steganography is the art of hiding information imperceptibly in a cover medium. The word "Steganography" is of Greek origin and means "covered or hidden writing". The main aim in steganography is to hide the very existence of the message in the cover medium [1]. 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.

The original steganographic applications used “null ciphers” or clear text. A null cipher conveys that the message has not been encrypted in any way, whether it is using basic character shifting, substitution or advanced modern day encryption algorithm. So, the message is often in plain view but for a reason can either not be detected as being present or cannot be seen once detected. As is common with cryptography, steganography has its roots in military and government applications and has advanced in ingenuity and complexity. The secret information in general is embedded into some media file like image or audio and thus it is transmitted so as to prevent an opponent from guessing that some secret information is being transmitted. So, the main objective of Steganography is not to let the opponent guess that any kind of information apart from the media file itself is transmitted.

In spatial domain of Steganography by using Image as the carrier file, we, in general, invert the Least Significant Bit of a particular byte of the carrier image to embed a particular bit of Secret message [2, 3, 4]. This method is known as LSB (Least Significant Bit) masking method of Steganography.

Least Significant Bit (LSB) coding is the easiest way to embed information in a digital audio file by substituting the least significant bit of each sampling point with a binary message, LSB coding allows for a large amount of data to be encoded.

In this proposed method, the carrier file and the secret message are taken into audio format and a pattern matching algorithm is used which will identify a part of the message from the carrier file and return its location. Then by using the standard LSB method the location will be embedded with the carrier file to form the stego file.

II. RELATED WORK

In the field of image security, Miroslav Dobsicek [1] has developed an interesting application of steganography where the content is encrypted with one key and can be decrypted with several other keys, the relative entropy between encrypt and one specific decrypt key corresponds to the amount of information. Yusuk Lim, Changsheng Xu and David Dagan Feng, 2001, described the web based authentication system consists of two parts: one is a watermark embedding system and the other is authentication system. In case of watermark embedding system, it is installed in the server as application software that any authorized user, who has access to server, can generate watermarked image. The distribution can use any kind of network transmission such as FTP, email etc. Once image is distributed to externally, client can access to authentication web page to get verification of image [2].

Min Wu and Bede Liu, June, 2003, proposed [3] a new method to embed data in binary images, including scanned text, figures, and signatures. The method manipulates “flippable” pixels to enforce specific block based relationship in order to embed a significant amount of data without causing noticeable artifacts. They have applied Shuffling before embedding to equalize the uneven embedding capacity from region to region. The hidden data can then be extracted without using the original image, and can also be

accurately extracted after high quality printing and scanning with the help of a few registration marks.

Rehab H. Alwan, Fadhil J. Kadhim, and Ahmad T. Al- Taani, 2005, has explained a method with three main steps. First, the edge of the image is detected using Sobel mask filters. Second, the least significant bit LSB of each pixel is used.

Finally, a gray level connectivity is applied using a fuzzy approach and the ASCII code is used for information hiding. The prior bit of the LSB represents the edged image after gray level connectivity, and the remaining six bits represent the original image with very little difference in contrast. The given method embeds three images in one image and includes, as a special case of data embedding, information hiding, identifying and authenticating text embedded within the digital images [4].

In 2007, Nameer N. EL- Emam proposed an algorithmic approach to obtain data security using LSB insertion steganographic method. In this approach, high security layers have been proposed through three layers to make it difficult to break through the encryption of the input data and confuse steganalysis too [5].

S. K. Bandyopadhyay, Debnath Bhattacharyya, Swarnendu Mukherjee, Debashis Ganguly, Poulami Das in 2008 has proposed a heuristic approach to hide huge amount of data using LSB steganography technique. In their method, they have first encoded the data and afterwards the encoded data is hidden behind a cover image by modifying the least significant bits of each pixel of the cover image. The resultant stego-image was distortion less. Also, they have given much emphasis on space complexity of the data hiding technique [6].

There is also a good method proposed by G. Sahoo and R. K. Tiwari in 2008. Their proposed method works on more than one image using the concept of file hybridization. This particular method implements the cryptographic technique to embed two information files using steganography. And due to this reason they have used a stego key for the embedding process [7].

Unfortunately, modifying the cover image changes its statistical properties, so eavesdroppers can detect the distortions in the resulting stego-image's statistical properties. In fact, the embedding of high-entropy data (often due to encryption) changes the histogram of colour frequencies in a predictable way. So, in order to obtain more security in our prescribed method, we have embedded the secret information behind an image of 8 times the size of secret information file to hide any remarkable change in the final image and also it helps the secret information remain scattered throughout the carrier image which will make the changes in the histogram look like noise.

III. Detailed method of Encryption

The growth of high speed computer networks and that of the Internet, in particular, has increased the ease of Information Communication. Ironically, the cause for the development is also of the apprehension of the use of digital formatted data. In comparison with Analog media, Digital media offers several distinct advantages such as high quality, easy editing, high fidelity copying, compression etc. But this type advancement in the field of data communication in other sense has hiked the fear of getting the data snooped at the time of sending it from the sender to the receiver. So, Information Security is becoming an inseparable part of Data Communication. In order to address this Information Security, Steganography plays an important role. Steganography is the art and science of writing hidden messages in such a way that no one apart from the sender and intended recipient even realizes there is a hidden message.

A. Preprocessing

It performs bit level manipulation to encode the message. The following steps are

a. Receives the carrier audio file and the secret audio file in the form of bytes and convert it into bit pattern (8 bits).After preprocessing there are n1*8 and n2*8 matrix where n1 and n2 are the number of rows of the cover audio and the secret audio file.

B. Blocking and pattern matching

a. For pattern matching the first two 8 bits block of the cover file are copied into an array index from 1 to 16.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	1	1	0	0	1	0	0	0	1	0	0	1	0	1	0

b. Then each 8 bits block of the message audio file is divided into 2 bits block. Now suppose the 8 bits block of the message audio file are given below:-

01100100

10010010

01010001

11010001

.

.

.

.

.

10110010

11001001

Let us consider the first 8 bit block of the message audio file .It has four two bits block. (01,10,01,00)

01100100

c. This phase will match the pattern of each two bits block of the message audio file

from the array and return the first matching location.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	0	0	1	0	0	0	1	0	0	1	0	1	0

From the above diagram the matching location of 01,10,01,00 are 0,2,0,3 respectively.

d. Each matching decimal location of the array are then converted into 4 bit binary form.

0=0000

2=0010

3=0011

The same procedure is used for each 8 bits block.

C. Size Estimation

This phase will estimate the size of the message audio file.

01 10 01 00

0 2 0 3

0000 0010 0000 0011

Since the standard LSB method is used, so it will add each 0000,0010,0000,0011 into the LSB of each row of the cover file. 16 rows are required to represent a row of the message audio file. If the number of rows of the message audio file is n2 the number of rows of the cover file required to represent the message audio file is 16*n2. In next step it represents the 16*n2 decimal number into 20 bits binary .So the maximum number of row size can be 2^{20}.

D. Probability of Pattern Matching

Considering a 2-bit sequence, the number of possible combinations will be 2^2 =4. So if a properly randomized sample is taken, that is a 4-bit sequence and try to locate a given 2-bit pattern within it then probability of finding it at any location is ¼*4=1. Here the sample space is 4 times larger than the minimum necessary size to fulfil the above criteria. So it will always be able to find the 2-bit sequence within the 16 bit pattern.

E. Message Embedding

a. The first two rows of the cover file are used for pattern matching.

b. In LSB of the next 20 rows the cover file (from 3-22) is used place the number of rows of the message audio file.

c. In LSB of the next 23 to 16*n row is used to accommodate encrypted version of the message audio file.

d. After embedding the whole message file into the cover file the required stego-file will create.

IV. Detailed Method of Decryption

In receiving side the required stego-file is the input. And the receiver will do just the reverse procedure of encryption.

A. Preprocessing

It performs bit level manipulation to encode the message. The following steps are

a. Receives the stego file as input.

b. Convert it into bytes.

c. Block it into 8-bits pattern.

Rows estimation

Store the first two 8 bits block into an array index from 1 to 15 for pattern matching.
In next step it will collect the LSB bits from row number 3 to 22 and convert it into decimal form. Now suppose if the decimal value is n then number of rows of the cover file where the secret message is hidden will be 16*n .The decryption algorithm is applied for row number (23 to 23+16*n.)

Pattern matching and hidden message identification

Suppose the 16 corresponding LSB bits of the cover file are
0000001000000011
Four bit grouping- 0000 0010 0000 0011
Decimal 0 2 0 3
Representation
The first two blocks store in the array are

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	1	0	0	1	0	0	0	1	0	0	1	0	1	0

0,2,0,3 are array location .We have to collect the 2 consecutive bits starting from the location i.e
0-01
2-10
0-01
3-00
Adding these four bits will create one row of the message i.e
01100100.
From row 23 to 16*n it will do the same procedure .From the LSB of the each 16 consecutive block of the cover file it will create a row of the message audio file.
The message is now a n2*8 binary matrix.
In last step we have convert it .wav file.

V. Algorithm

A. Encryption Algorithm

Step 1: Start

Step 2: Read the cover audio file of length a1 and secret message of length a2

Step 3: Convert the cover audio file and secret message into digital form.

Step 4: The amplitude value of the cover file and secret message are normalized into 8 bits block and it will get two matrix of n1*8 and n2*8 where n1 and n2 are the rows of cover file and secret message respectively and n1 >=16*n2+22

Step 5: Store the first two 8 bits block of the cover file into an array index from 1 to 16.

Step 6: Represent the number of rows of the secret message n2 into 20 bits binary form.

Step 7: From row numbers 3 to 22, Replace the LSB of the cover file with these 20 bits number.

Step 8: Divide each rows of the secret message (8 bits block) into four 2 bits block.

Step 9: Search each 2 bit block of the secret message from the array and identify its starting location. If there is more than one match, then take the first match.

Step 10: Convert the array index at the starting location into 4 bit binary.

Step 11: From 23rd of the cover file store all these 4 bit binary numbers in the LSB position of each block column-wise.

Step 12: END

B. Decryption Algorithm

The receiver will do just a reverse job.

1. Start

2. Convert the stego file into digital form.

3. Covert it into 8 bit blocks.

4. Store the first two 8 bit block into an array indices from 1 to 16 for pattern matching.

5. From row number 3 to 22 collect all the LSB and convert it into decimal form. Multiply 16 with the decimal value. This represents the number of rows of the cover file where the secret message is hidden.

6. From row number 23 select all the LSB bits of the next 16 rows of the cover file and group it into a 4 bit block.

7. Each decimal value of the 4 bit block is basically the array index which identifies the starting address of a 2 bit blocks of the secret message.

8. According to these array index receiver will fetch 2 consecutive bits from the array and four two bits block will create a row of the secret message.

9. Repeat step 6, 7 and 8 until it goes to the decimal value of step 5 which is the number of rows of the cover file.

10. Covert the n2*8 into audio file.

11. Stop

VI. RESULT ANALYSIS

The method proposed in this paper hides two target images behind one cover image thereby increasing the data hiding capacity greatly. For the cause of security only the final stego-image is sent over the network. This approach is also free from the constraint of size.

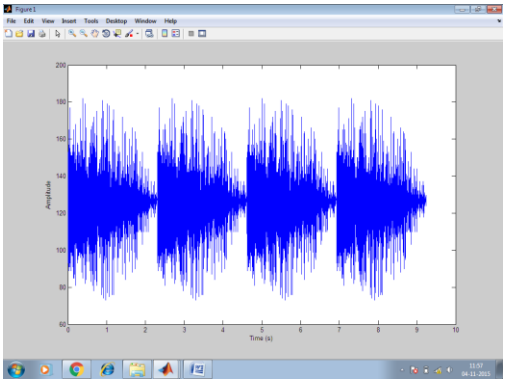

Fig-1a-Cover file

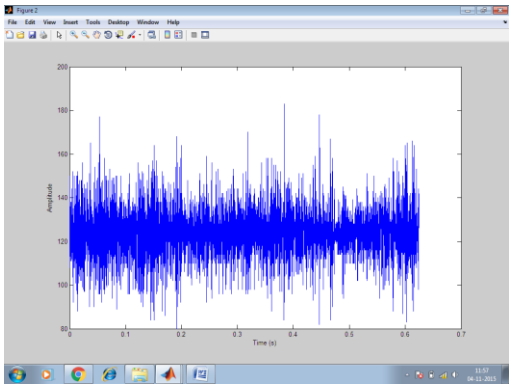

Fig1b-Message file

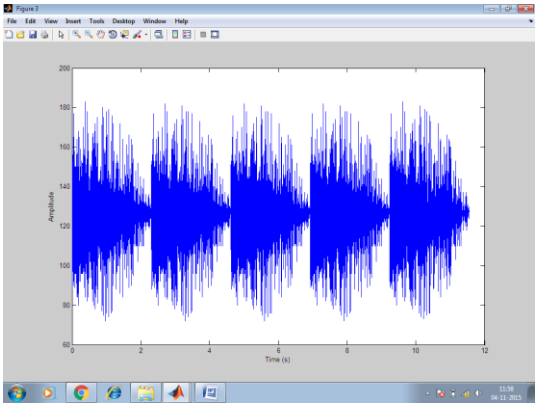

Fig1c- Encrypted Stego-file

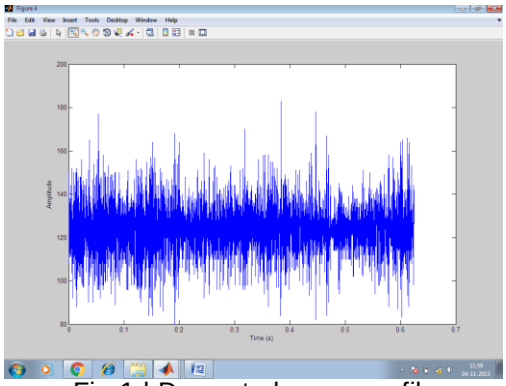

Fig-1d-Decrypted message file

It is observed that the stego-file hasn't audible be modified. For clear identification here two sets of files are graphically plotted.

a. 1a is cover file.
b. 1b is message audio file.
c. 1c is stego-file that was created after embedding the message audio file into the cover file.
d. 1d is decrypted message audio file in the receiving end.

The graphical representation shows that there is reasonably no change between input carrier file and output stego-file i.e 1a, and 1c graph is reasonably same.

In decryption end the graphical representation shows that the decrypted audio is fully same as with the input message audio file.

VII. Conclusion

The main advantage of our algorithm is that the final image can be derived only from the Stego Image. The original cover image is not needed for decoding the stego image. This algorithm is free from size constrains i.e. it performs well on any size of the cover image or target image.

References

[1] Dobsicek, M., "Extended steganographic system.", In: 8th Intl. Student Conf. On Electrical Engineering, FEE CTU 2004, Poster 04.

[2] Yusuk Lim, Changsheng Xu and David Dagan Feng, "Web based Image Authentication Using Invisible Fragile Watermark", 2001, Pan-Sydney Area Workshop on Visual Information Processing (VIP2001), Sydney

[3] Min Wu, Member, IEEE, and Bede Liu, Fellow, IEEE, "Data Hiding in Binary Image for Authentication and Annotation", IEEE Trans. Image Processing, volume 6, Issue 4, Aug. 2004 Page(s): 528 - 538

[4] Rehab H. Alwan, Fadhil J. Kadhim, and Ahmad T. Al- Taani, "Data Embedding Based on Better Use of Bits in Image Pixels", International Journal of Signal Processing Vol 2, No. 2, 2005, Page(s): 104 – 107

[5] Nameer N. EL-Emam "Hiding a large amount of data with high security using steganography algorithm", Journal of Computer Science. April 2007, Page(s): 223 – 232

[6] S.K.Bandyopadhyay, Debnath Bhattacharyya, Swarnendu Mukherjee, Debashis Ganguly, Poulumi Das, "A Secure Scheme for Image Transformation", August 2008, IEEE SNPD, Page(s): 490 – 493

[7] G. Sahoo, R. K. Tiwari, "Designing an Embedded Algorithm for Data Hiding using Steganographic Technique by File Hybridization", January 2008, IJCSNS, Vol. 8, No. 1,Page(s): 228 – 233.