A Novel Technique for Embedding Data in Spatial Domain

V.Madhu Viswanatham[*1], Jeswanth Manikonda[*2]

*[1]School of Computing Science and Engineering, VIT University, Vellore, India

*[2]School of Computing Science and Engineering, VIT University, Vellore, India

***Abstract*- Steganography is the science of hiding messages in a manner that only the intended users knows the presence of the secret message. Hiding the information within a computer file also comes under Steganography. In digital steganography these media files are suitable because of their large size. LSB insertion is one of the basic techniques prominently used in steganography.**

This paper presents an effective and secure technique of LSB insertion mechanism. The technique involves generation of random numbers and also selecting the region of interest wherein which the required message is embedded along the random pixels that are previously generated. The technique also involves a secret key which has to be provided by the recipient for decoding the message from the image.

Keywords – Image steganography, LSB, Security, Data hiding, Pixels

1. Introduction

Ensuring the confidentiality is one of the biggest challenges while transferring the data. Various methods are used for providing security. One of the methods is the steganography. The word steganography means concealed writing. Steganography, the technique of hiding messages in other files for transmission in a manner that an observer could not identify the occurrence of transmission, is gaining popularity with current industry demands. It includes various techniques of secret communications that veil the message. The various methods include invisible inks, micro dots, character arrangement, digital signatures, convert channels, Least significant Bit insertion, Masking & Filtering and spread-spectrum communications.

Least Significant Bit insertion is one of the widely known, elementary approaches for enclosing data in a cover file. Least significant bit is the bit position in a binary integer that gives the units value. The LSB is also referred to as the right most bit as the less significant bit is written to the right. These least significant bits have an effective property that they change briskly when there was an slight change in the number. An effective medium for hiding the data is an image. Slight variation in its colour in altered image is indistinguishable from the original image. But LSB insertion method is utmost susceptible to attacks, in the manner of image manipulation. Modification of a GIF or BMP format to a lousy confining format like JPEG wipes out the covert data present in the image.

Newer methods embed data in optically unsubstantial parts of an image. Normally an image is a set of pixels where the message can be embedded.

2. Existing Systems:

There were quite few mechanisms proposed earlier in LSB insertion technique. One of them is that secret file is spread out through the image using a secret key which specifies where and how the data was spread out. Though this method provides better security making the attacker inefficient in estimating the presence of message, the receiver must have the password besides the secret key so as to recover the message.

Another proposed system is that it encloses the data in LSB facet of cover image randomly using disorganized system and then makes an effective compensation of the steg-image called dynamic compensation steganography. This model features to be a successful one such that sample pair analysis results in very negligible difference among the original and embedded image.

Another method of LSB insertion technique is that 4LSB insertion. The name itself suggests that the message is embedded in the last four bits rather than only LSB. Since no other protection mechanisms are provided, this is more vulnerable to security attacks.

3. Proposed System

In order to provide much security over the previously mentioned steganography techniques, the proposed system uses a random number generation algorithm and also a password for better authentication of the message. In the proposed system, the image size is of 256 *256. The proposed system is done using MATLAB. Since MATLAB is used, the image is directly displayed

in pixels. A password is selected for authentication purpose during decryption process. The proposed model has two phases

I. Encryption Phase

- Embedding the password
- Random number generation
- Selecting the region of interest

II. Decryption Phase

Encryption Phase:

Embedding the password:

The password chosen which is in ASCII format is first converted into decimal format and then these decimal values are converted into binary format. The pixel values of the image are converted into binary. The password is then nested within the least significant bit position of the pixel. The scenario presents that when the pixels are arranged in a matrix, the password is embedded in the first column of the matrix.

Random number generation:

The proposed model ensures security by embedding the data randomly through a selected region of interest. The random number is generated by using a random function generator such as

K=mod (((k*2) +1), 11)

k corresponds to the bit position. Since image is of size 256 *256, mod 11 is appropriate value for random numbers. This might vary depending on the size of the image. The random function generates random numbers based on the aggregation of bits in the image which are stored in an array of size 9 which can be dynamically modified based on the number of bits.

Selecting region of interest:

The region of interest is selected by addition of a required value (say 100) to these random numbers. The text which has to be transferred is also converted into decimal from pixel and then to binary. The text is then embedded into pixel values specified by these random numbers. These binary values are again converted back to the decimals (pixels). Due to the random distribution of the message, the image size and the pixel values may vary. In order to reduce the redundancy, the image is subjected to reshape which produces the image of size 256*256. The size of the image does not have significant change when compared to the original image.

The audio stream can also be transferred inside an image. First the audio wave is subjected to silence removal where the unnecessary nodes the wave are chopped off. The silence removal reduces the number of frames to be processed which in turn reduces the processing time. The audio which is in wave format is read as pixels in MATLAB. Since the values of the sound signal range between -1 and +1, sometimes there might be negative values which would be difficult to process. So a constant value is appended so that the values become positive. The pixels are then transformed into image.

Decryption Phase:

The receiver, on receiving the image has to provide the password provided in order to decode the message. The algorithm works as the reverse process of encoding. When the receiver provides a password, it is first checked for the length of the password in order to reduce the unnecessary calculations. And if the length matches, the password is retrieved from the image and a bitwise EXOR operation is performed on both the retrieved password and the original password. If these two matches then the decoding algorithm comes into light.

Usually in MATLAB the data is stored in a reverse order of the pixel, when stored using binary values. So decoding algorithm mainly generates the random numbers and extracts the message in an reverse fashion. The decoding algorithm produces array of binary values which are converted to decimal values. The end of message is identified when the decoding algorithm reads specific ASCII value of a character which has been designated to denote the end of a message.

While decoding the audio content, the constant number is detained from the pixels values so that original nodes are formed. The audio can formed by applying wavwrite (MATLAB) function to the pixels retrieved.

4. Results and Discussion

This paper clearly explains about hiding text with in an image and sending it securely. The main functionalities explained here are

- Secure transformation of data
- Use of region of interest and random number algorithm.

A modified LSB technique is used to hide data in images. The language MATLAB is used to enclose the data within the image and send it. MATLAB is a fourth generation programming language for numerical computing. The basic data element of MATLAB which is an array does not demand dimensioning. The above image is a default image in MATLAB whose size is 256*256. The pixel values of above image are

156 159 158 155 158 157...

160 154 157 158 157 159...

156 159 158 155 158 156...and so on

The pixel values are converted into binary such as

00111001

00000101

00111001

For example, the ASCII values of password "joker" are

106 111 107 101 114

Once the password entered is converted into decimal form it is then converted into binary. The password 'joker' is of 5 characters and each character has ASCII value.So each character contains 7 bits, so the size of matrix is 5*7. The binary values look like

106--0101011

111--1111011

107--1101011

101--1010011

114—0100111

In MATLAB when decimal value is converted to binary, it is stored in reverse fashion. For example binary value for 118-1110110, but it is stored as 0110111. So least significant bit is 1st bit of a pixel. The 1st bit of each pixel should be replaced with bits of the password. Since the size of password is 5*7, 1st bit in first 35 pixels contain the password.

The password is embedded into the image pixels at the first least significant bits. The binary pixels are modified as

00111001

10000101

00111001...

For example, if the text embedded is 'vit', the ASCII characters of this string are first converted into decimal and then to binary. The ACSII values of given string is

118 105 116 57

where 57 represents the ASCII value of 9 which shows the end of the string. Any character or symbol can be taken as the representation of end of string so that characters appearing after this value are not considered. The binary values of the input text are

0110111

1001011

0010111

1001110

The size of the matrix is 3*7 as the text is of 3 characters length. While using random number generation algorithm, an array of length 9 is used. For example, in the random number generation algorithm k=mod(((i*2)+1),11) if the value of i is taken as '0' then k=mod(((0*2)+1),11) gives k=1 which are then stored in an array say L. Then L (1) =1;

Similarly the random numbers generated are 3,1,4,9,8,6,2,5. Thus the random numbers are generated which represents the pixels. In order to select the region of interest a random number say 100, is appended (L=L+100) to these random numbers which gives the region such as

101 103 107 104 109 108 102 105

In MATLAB when any array is initiated and if any number is added to it after the loop, then it will be added to each and every element of the array.

The data is embedded throughout the image randomly in the pixels generated by random number generation algorithm. The final pixel

values after embedding data and the password look like

00000101—101

10111001—102

10000101—103

00111001—104 and so on.

The random number generation algorithm uses only 9 values for 9 bits. There might be a case where the data size exceeds the 9 bits. Hence an dynamic modification condition such as

If mod(k,10)==0

L=L+10;

increases the size by 10 bits which can be further extended as required dynamically. In the above example, the data is of size 21 bits. Hence the values 101,103,107,104,109,108,106,102,105...which are up to 9 bits can be extended when an 10^{th} bit arrives such that the array resembles like 111,113,117,114,119,118,116,112,115 so that next 9 bits are embedded in these pixels. Again this can be extended to 121,123,... so that all 21 bits are embedded in the region of interest.

Finally the binary values are converted to decimal format which may change the alignment of the pixels and also a change in size of the image. Hence the image is reshaped which is special function in MATLAB.

5. Conclusion

Steganography provides many different mechanisms to hide the data. This paper presents an image steganography algorithm which uses LSB insertion technique, random number generation algorithm, region of interest selection. These techniques are used for providing better security for efficient data transmission. As this technique is implemented only in an 256*256 image, it can be further extended to larger and better quality image.

REFERENCES

[1] A Secure LSB Steganography System Defeating Sample Pair Analysis Based on Chaos System and Dynamic Compensation, Xiangyang Luo Zongyun Hu Can Yang Shanqing Gao, Zhengzhou Information Science and Technology Institute, Zhengzhou, 450002, China 2006.

[2] Alkhraisat Habes, “Information transmissions in computer network,Information hiding in bmp image Implementation analysis and evaluation” Jan-2006.

[3] Bin Liu, Yongquinag Zhang and Fenlin Liu. A New Scheme on Perturbing Digital Chaotic Systems. Computer Science, Vol.32, No.4, 2005.

[4] Xiangyang Luo, Bin Liu and Fenlin Liu, “Improved RS Method for Detection of LSB Steganography”, In: Proc. The 20005 International Conference on Computational Science and Its Applications, Volume 3481 of Springer LNCS(2005),pp.508-516.

[5] Andrew D Ker, “Steganalysis of LSB matching in Gray scale Images”, IEEE Signal; Processing Letters, Vol. 12, No.6, June 2005, pp.441-444.

[6] Zhang Tao, Ping Xijian,“Reliable Detection of LSB Steganography Based on the Difference Image Histogram”, Proceeding of IEEE ICSAAP, Part III, pp.545-548, 2003.

[7] Jeong Jae Yu, Jae Won Han and et, al. A Secure Steganographic Scheme against Statistical Analysis, eds.: IWDW 2004, LNCS 2939, pp. 497-607, 2004.

[8] Andrew D. Ker, “Improved Detection of LSB Steganography in Gray scale Images”, In: Proc. The6^{th} Information Hiding Workshop. Volume 3200 of Springer LNCS (2004). pp. 97-115.

[9] Sorina Dumitrescu Xiaolin Wu, and Zhe Wang, “Detection of LSB Steganography via Sample Pair Analysis”, IEEE Transcations an Signal Processing, Vol.51, No.7, July 2003.

[10] A. Westfeld: Detecting low embedding rates. In: Proc. Information Hiding Workshop.Volume2578 of Springer LNCS. (2002) 324-339.

[11] J.Fridrich, M. Goljan, “Practical Steganalusis of Digital Images-State of the Art”, In Delp III , E.J., Wong, P.W.,eds.: Security and Watermarking of Multimedia Contents IV. Volume4675 of Proc. SPIE. (2002)1-13.

[12] N Provos, “Defending against Statistical Steganalysis”, 10^{th} USENIX Security Symposium, Washington, DC,2001.

[13] J. Fridirch, M Goljan and R Du, “ Reliable detection of LSB steganography in color and grayscale images”, in Proc. ACM Workshop Multimedia Security, Ottawa, ON, Canada, Oct 5 2001, pp.27-30.

[14] J. Fridrich, R. Du and L. Meng, “Steganalysis of LSB Ecoding in colour Images”, Proc. IEEE Int 1 Conf. Multimedia and Expo, CD-ROM, IEEE Press, Piscataway, N.J.,2000.