International Journal of Electrical and Electronics Engineering (IJEEE)
ISSN 2278-9944
Vol. 2, Issue 4, Sep 2013, 1-8

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

STEGANOGRAPHY USING TWELVE SQUARE SUBSTITUTION CIPHER AND LSB POSITIONS

AUTADE PARIMAL & KATARIYA S. S

Department of Electronics, Pune University, Pune, Maharashtra, India

ABSTRACT

The use of Internet has been extensively increased. Sometimes, it is needed to keep the information secret and secure without attracting the attention of unauthorized person. Here, we proposed Steganography method along with cryptography for secret communication. We are using both cryptography and steganography. Firstly, we encrypt the secret message using our new cipher algorithm called twelve square substitution cipher algorithm, and then embed the cipher text in the carrier image in 6th and 7th bit locations or 7th and 8th bit locations or 6th and 8th bit locations of the different pixels (bytes) of the carrier image depending on the value of an index variable. Using variable LSB position for substitution purpose. After embedding the resultant image should be sent to the receiver and receiver should retrieve the cipher text from the said locations and then decrypt by using the twelve square cipher algorithms to get the secret message. The embedding locations are not same in all pixels, so it is a stronger approach. The algorithm is implemented using MATLAB programming language.

KEYWORDS: Twelve Square Substitution Cipher, Steganography, Index Variable

INTRODUCTION

The Steganography is a technique of hiding the secret data into a carrier, such as a ditital image, audio, video etc. Here, we implemented the text steganography. The word steganography is basically a Greek word. It is the combination of two words stegos and grafia. Stegos means to cover and grafia means writing. We can find many evidents of the Steganography in the history. The people in Rome and Greece used to carve the message on the wooden pieces and this writing would be then covered with the wax.

The secret messages written on thin pieces of silk would be rolled on a small ball and then the ball would be swallowed by the army messenger. The common example of the steganography is, writing secret message on a paper with onion juice or ammonia salts and the secret message can be then exposed by heating the paper. The various type of the Steganography are text, image, audio or video steganography as per the type of the carrier. The steganography should be strong enough to hide the secret data securely and it should not change the basic properties of the carrier.

The proposed method is stronger. Also, there is less threat of changing the basic characteristics of the carrier as we are using the LSB method. It is the age of the internet as we get the service just on a click. We use the internet for searching information, downloading and uploading of multimedia, to check E-mails, E-banking, Online reservations, therefore it is often needed to keep the data secret without attention of hackers. Cryptographic algorithms has been used for security purpose but the main disadvantage of Cryptography is that, it attracts the attention of the third party as the encrypted data is visible to anyone and steganography avoids it, as the data is hidden behind a carrier. The secret data is firstly encrypted and then embedded using the LSB method. Therefore it gives a double layered security to the secret information.

REVIEW OF EXISTING IMAGE STEGANOGRAPHY METHODS

In Image encryption approach we can encrypt the image and embed the secret information in LSBs and after embedding if the entropy and correlation values of stego image and original image are the same then the process is a secure one[1].The most important thing is, while doing the steganography is that, the visible properties of the carrier should not be much changed. Least Significant Bit (LSB) method is the simplest method of steganography. The changes at the LSB positions of the carrier may not be noticeable because of the imperfect sensitivity of the human eyes. The binary secret data replaces the least significant bit of an image.

The embedding of the information at LSB position does not make significant change in the color of the pixel. The LSB method usually does not increase the file size, entropy and correlation values before embedding and after embedding the data therefore the process is a secure one. Ross J. Anderson and Fabien A.P. Petitcolas presented the limitations of steganography and gave contrast of various disciplines of cryptography and traffic security [2]. Inserting the secret message in frequency domain was proposed by Po Yuch Chen and Hung Ju Lin.

They used Discrete Wavelet Transform method for it. In this method the embedding should be done at high frequency coefficients [5]. Method named as Two way block matching for image in Image steganography was proposed by Ran-Zan Wang and Yeh-shun Chen [12]. Xinpeng Zhang and his colleagues proposed an approach called "multibit assignment steganography for palette images", secret data can be embedded at the same coloured pixels of the gregarious pallette images [13]. Weiming Zhang, Xinpeng Zhang and Shuozhong Wang presented a method for implementing plus minus steganography [4]. When the secret message is hidden into a carrier image, It changes statistics of natural images [5]. If we avoid data embedding at the bits those carry the image features then it would make minimum effect on image features. H. Rifa-Pous and J. Rifa H. Rifa-Pous and J. Rifa presented a steganographic protocol based on hamming code [7]. Mohammad Shirali-Shahreza presented an application of Text steganography for mobile phones for hiding the [5] presented a method of Text steganography using the six square substitution algorithm and it was consisting of only the alphabets. The twelve square substitution algorithm covered alphabets as well as the special characters [8]. but the alphabet „q" was missing and some symbols like µ, Þ, ß, Ø, ×, ð, þ, ø, †, ¯, *f* etc were not considered, in this assign another variable for this missing characters or symbols The proposed technique uses both cryptography and steganography. Therefore, it provides double layered security. The twelve Square Algorithm was implemented for various image formats using variable LSB positions. We compared the data hiding at one LSB and two LSB positions

The Steganography [1]Process can be described using the following block diagram in Figure 1. The encryption of the secret message is done using a new cipher algorithm called extended square substitution cipher algorithm, and then this cipher text is inserted at LSB places. For data embedding at one bit LSB may be at 6th or 7th or 8th position and it is not fixed. Whereas the data embedding at two LSB position uses 7th, 8th or 6th, 7th or 6th, 8th positions as per the variable value. The variable value can be 0 or 1 or 2

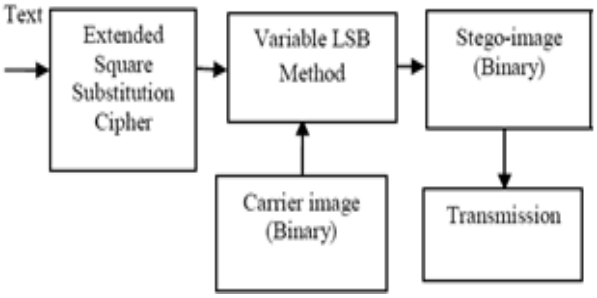

Figure 1: Steganography Process

After embedding the stego image is sent to the receiver and receiver retrieves the cipher text from the said locations and then decrypts by using the extended square cipher algorithm to get the secret message. The entire approach is discussed in the following sections. In section-3, the working of the twelve square substitution cipher is discussed, in section-4 the embedding process, in section-5 the proposed algorithm, in section-6 the results, in section-7 conclusions.

THE TWELVE SQUARE SUBSTITUTION CIPHER

The Six square substitution algorithm could cover only alphabets and its further improved version Twelve Square Substitution Cipher Algorithm covered alphabets as well as some special characters but it could not cover the alphabet q and some special symbols like µ, Þ, ß, Ø, ×, ð, þ, ø, †,(space) , ¯, *f* . The 12 Square Substitution Cipher Algorithm includes numerals and special characters. It encrypts capital as well as small alphabets, digits and special characters. It uses six 5 by 5 matrices each arranged in a square, as shown in table-1. Each of the 5 by 5 matrices contains the letters of the alphabet (usually omitting "Q" to reduce the alphabet to fit into the square) and another six 6 by 7 matrices arranged in squares for digits and special characters, as shown in table 2. All the special characters and digits from your desktop/laptop keyboard are included in this table

Table 1: Plain Text and Cipher Text (Alphabets)

Square-1	Square-2	Square-3
a b c d e f g h i j k l m n o p r s t u v w x y z	f g h i j k l m n o p r s t u v w x y z a b c d e	k l m n o p r s t u v w x y z a b c d e f g h i j
Square-4	**Square-5**	**Square-6**
g m r i t a b c d e f h j k l n o p s u v w x y z	a b c d e f h j k l g m r i t n o p s u v w x y z	a b c d e f h j k l n o p s u v w x y z g m r i t

See, Square-1, we have twenty five alphabets excluding the alphabet q, in each row we arranged five alphabets. Square-2 is created from square-1 by taking the first row of square-1 to fifth row place and other rows one position up. Similarly square-3 is created from square-2 by taking the first row of square-2 to fifth row place and other rows one position up.

In square-4, we have used a word gmrit in the first row which comprises of the five alphabets and the remaining twenty alphabets are arranged in other four rows continuously excluding the alphabets of the word “gmrit”. Square-5 is made from square-4 by taking the first row to third row place. Similarly square-6 is made from square-4 by taking the first row to fifth row place. See in table-2 In square-7, the numerals and special characters from a standard laptop are arranged in six rows and seven columns. Square-8 is made from square-7 by taking the first row to sixth row place. Similarly square-9 is created from square-8 by taking the first row of square-8 to sixth row place. Square-10 is created from square-7 by arranging the row elements in columns. Square-11 is created from square-10 by taking the first row of square-10 to third row place. Similarly square-12 is constructed from square-10 by taking the first row into sixth row place.

The plain text is read from left to right. If the character is an alphabet it refers to table.1, otherwise if it is a number or a special character it refers to table-2. While scanning the plain text the first alphabet’s plain text is in square-1

and its cipher is in same row and column location of square-4. The second alphabet, its plain text is in square-2 and cipher text is in same row and column location of square-5. The third alphabet, its plain text is in square-3 and cipher text is in same row and column location of square-6.

Table 2: Plain Text and Cipher Text (Digits and Special Characters)

Square-7	Square-8	Square-9
0 1 2 3 4 5 6 7 8 9 ` ~ ! @ # $ % ^ & * ( ) _ - + = { [}] ; : " ' \ | < , > . ? /	7 8 9 ` ~ ! @ # $ % ^ & * ( ) _ - + = { [}] ; : " ' \ | < , > . ? / 0 1 2 3 4 5 6	# $ % ^ & * ( ) _ - + = { [}] ; : " ' \ | < , > . ? / 0 1 2 3 4 5 6 7 8 9 ` ~ ! @
Square-10	**Square-11**	**Square-12**
0 6 ! & + ; < 1 7 @ * = : , 2 8 # ({ " > 3 9 $) [' . 4 ` % _ } \ ? 5 ~ ^ -] | /	1 7 @ * = : , 2 8 # ({ " > 0 6 ! & + ; < 3 9 $) [' . 4 ` % _ } \ ? 5 ~ ^ -] | /	1 7 @ * = : , 2 8 # ({ " > 3 9 $) [' . 4 ` % _ } \ ? 5 ~ ^ -] | / 0 6 ! & + ; <

The secret message is combination of alphabets, numbers and special characters. While scanning the secret message, for the special characters and digits it refers to table-2. The first special character (including digits), its plain text is in square-7 and cipher text is in same row and column location of square-10. For second special character (including digits), the plain text is in square-8 and cipher text is in same row and column location of square-11.

For the third special character (including numbers) the plain text is in square-9 and cipher text is in same row and column location of square-12. Similarly fourth special character (including numbers) corresponds to square-7 and square-10, 5th special character (including numbers) corresponds to square-8 and square-11, 6th special character(including numbers) corresponds to square-9 and square-12 and so on.

For example if the plain text is:

Professor Khurana will arrive at 10. 30 P.M.

It's cipher text is:

nmeazjplh fclovdg oihh vomivz gs 65} &5 g}c]

THE EMBEDDING PROCESS AND INDEX VARIABLE

The process of embedding message into an image can be explained as: Firstly, the carrier image is transformed into binary form. Each pixel becomes 1 byte. The cipher text of the secret message is converted into bytes. Now calculate the number of bytes, say it is n. Divide it by 3, say it is p. The p called as the index variable. The value p=0, corresponds to 6th and 7th bit locations, p=1 corresponds to 7th and 8th bit locations, p=2 corresponds to 6th and 8th bit locations of any pixel (byte) of the digital image.

If present value of p=0 hide the two bits of cipher text in 6th and 7th bit locations of the present pixel (byte), and next value of p is 1 for the next pixel. If present value of p=1 hide the two bits of cipher text in 7th and 8th bit locations of the present pixel (byte), and next value of p is 2 for the next pixel. If present value of p=2 hide the two bits of cipher text in 6th and 8th bit locations of the present pixel (byte), and next value of p is 0 for the next pixel.

Table 3: Byte Selection Using Index Variable

Carrier File Byte	Operation	location	Index Variable, P
Byte A	Embed (11)	6th and 8th	2
Byte B	Embed (00)	6th and 7th	0
Byte C	Embed (10)	7th and 8th	1
Byte D	Embed (11)	6th and 8th	2
Byte E	Embed (01)	6th and 7th	0
Byte F	Embed (11)	7th and 8th	1
Byte G	Embed (10)	6th and 8th	2
Byte H	Embed (10)	6th and 7th	0
Byte I	Embed (10)	7th and 8th	1
Byte J	Embed (10)	6th and 8th	2
Byte K	Embed (10)	6th and 7th	0
Byte L	Embed (10)	7th and 8th	1
so on			

Example*:* Suppose the cipher text to be sent is: 11001011 01111010 10101010 10011001 01010101. This data is five bytes. So n=5 and p=2. Suppose the different bytes of the digital image are A, B, C, D, E etc. From table-3 we can see that in byte A of the carrier file we embedded the data bits 11 in 6th and 8th bit locations, and next value of p becomes 0. We embed the next data bits 00 into byte B in 6th and 7th bit locations, next value of p becomes 1. Now we embed the next two bits 10 in C in 7th and 8th bit locations and so on. See table-3. In every image there will be some bytes representing the image features which should not be altered.

In JPEG images of size more than one Mega Bytes, there will be a maximum of hundred bytes carrying the image characteristics, if we modify these bytes the image will be disturbed. So these bytes should not be altered. For different image formats like BMP, JPG, TIF it is different. For JPG it is around 100 bytes. Normally these are the first 100 bytes of the image.

The Table-4 illustrates the scheme for Steganography at one bit LSB position of carrier image pixel. The binarised ASCII value of alphabet ‘A ‘is inserted at one LSB position of eight carrier image pixels. Now it is clear that the data embedding at one LSB position takes eight consecutive pixels of the carrier image whereas the text data embedding at two LSB positions need only four consecutive pixels two hide the alphabet “A”.

Table 4: LSB Selection Using Index Variable for One Bit Data Embedding

Cipher Text	Binary pixels	Variable x	Insert	LSB positions
A (01000001)	p	1	0	7^{th}
	q	0	1	6^{th}
	r	2	0	8^{th}
	s	1	0	7^{th}
	t	0	0	6^{th}
	u	2	0	8^{th}
	v	1	0	7^{th}
	w	0	1	6^{th}

THE PROPOSED ALGORITHM

Step 1: Transform the carrier image to binary

Step 2: Apply the Twelve Square Cipher to get the cipher text of the secret message.

Step 3: Convert the cipher text to binary.

Step 4: Make sure that the length of carrier image is sufficient enough to conceal the cipher text.

Step 5: Embed the cipher into the cover image as discussed in the embedding process.

Step 6: Now sends the resultant image to receiver.

Step 7: Receiver applies the reverse process what sender has done and gets the hidden information.

Now let us discuss what should be the required length of the carrier file to conceal a specific amount, say n bytes of the secret information. It is very clear that in each pixel we are hiding two bits of cipher. So one byte of cipher can be accommodated in 4 bytes of image. Thus for n byte cipher text we need a carrier image of 4n bytes length.

This steganographic approach should have the following characteristics: (i) should not be vulnerable to exhaustive search attacks, (ii) the degradation in, quality of the stego image should not be noticeable and (iv) should provide atleast two levels of security. This proposed algorithm possesses all these characteristics.

RESULTS

The plain text in example ii has been considered for encryption and the graphical user interface for two LSB and one LSB. Here we are not embedding the plain text directly; we are embedding the cipher text. The encryption algorithm is 12-square substitution cipher algorithm. Due to its multi-substitution ability it is less susceptible to frequency analysis attack and known plain So compared to most of the algorithms it is very robust and secure. text attacks. From this analysis it is clear that inserting data at variable LSB positions hardly changes the parameters and the quality of the image is retained clearly shows in figure 2

CONCLUSIONS

This secret communication system is based on both Cryptography and Steganography. We successfully verified the steganography at one LSB and two LSB positions and it is clear from the study that inserting the data at two LSB position does not change image parameters. It retains the image quality similar to that of one bit LSB scheme. The amount of data sent through steganography at two LSB is double than that of one LSB scheme.

This system is able to conceal all types of alphabets (small as well as capital), special characters and mathematical symbols. The variable x takes values as 0, 1, 2and 3. Embedding the cipher at two LSBs is decided by variable x. Also it provides two levels of security. One at the cryptography level and the other at the steganography level. If at all the intruder suspects it is very difficult for him to steal the data

ACKNOWLEDGEMENTS

It is the great pleasure that we acknowledged the enormous assistance and excellent of this technology, extended to use of in various applications. Both the authors work under Pune University. Prof. S.S. Katariya, for her valuable suggestions and guidance throughout course of study and timely help given in the completion of this work.

Thankful to the teachers and management of A.V.C.O.E of Engineering and College of for permitting them to pursue their research work, And also guidance of my teacher. (please refer to the paper of Steganography Using Least Signicant Bit Algorithm.

REFERENCES

1. Mohammad Ali Bani Younes and Aman Jantan, “A New Steganography Approach for Image Encryption Exchange by using the LSB insertion”, International Journal of Computer Science and Network Security, Vol 8, No 6,2008, pp. 247-254.

2. Ross J. Anderson and Fabian A.P. Petitcolas, "On The Limits of steganography", IEEE Journal of selected Areas in communication, Vol.16, No.4, 1998, pp. 474-481

3. Mohammed A.F Al-Husainy, "Image Steganography by mapping Pixelsto letters", Journal of Computer science, Vol.5, No.1, 2009, pp. 33 38.

4. Hardik J. Patel and Preeti K. Dave, "Least Significant Bits Based Steganography Technique", International Journal of Electronics Communication and Computer Engineering (IJECCE-2012) pp.44-50.

5. Gandharba Swain, Saroj Kumar Lenka, "Better Steganography using the Six Square Cipher Algorithm", Proc. of International Conference on Advances and Emerging Trends in Computing Technologies (ICAET-2010), Chennai, India, 2010, pp.334-338.

6. Po Yuch Chen and Hung Ju Lin, "A DWT Based Approach for Image Steganography", International journal of Applied Science and Engineering, Vol.4, No.3, 2006, pp. 275-290.

7. Joachim J. Eggers, R.Bauml and Bernd Girod, "A Communications Approach to image steganography", Proc. Of SPIE Volume 4675, San Jose, Ca, 2002, pp. 1-12.

8. Gandharba Swain and Saroj Kumar Lenka, "Steganography Using the Twelve Square Substitution Cipher and an Index Variable", IEEE transactions on Image Processing, 2011, pp. 84-88.

APPENDICES

Figure 2: Steganography at Two LSB Graphical User Interface Consisting of Plain Text and its Cipher Text of Example

AUTHOR DETAILS

Parimal Autade, M.E Electronics in Digital System Degree (appear), A.V.C.O.E Sangamaner. And Prof.S.S Katariya guide to me for my work in A.V.C.O.E Sangamaner