Stegnography using 12 Square Cipher Algorithm

Amina Qazi
Asst Professor
BVCOE, Navi Mumbai

Asha Ghodake
Asst Professor
BVCOE, Navi Mumbai

Rupali Patil
Asst Professor
BVCOE, Navi Mumbai

ABSTRACT
Stegnography is the process of hiding one file inside another such that others can neither identify the meaning of the embedded object, nor even recognize its existence. Current trends favour using digital image files as the cover file to hide another digital file that contains the message or information. One of the most common methods of implementation is Least Significant Bit Insertion. This paper is an attempt to introduce an algorithm which uses LSB Stegnography as the basis and randomly disperses the secret message over the entire image to ensure that the message cannot be obtained easily from the image.

General Terms
Cryptography and Stegnography

Keywords
Stegnography, Digital Images, Data hiding, LSB insertion

1. INTRODUCTION
Today we are surrounded by a world of secret communication, where people of all types are transmitting information as innocent as an encrypted credit card number to an online store and as insidious as a terrorist plot to hijackers. The two most popular types of secret communication techniques in use are cryptography and stegnography[1]. Stegnography is the art and science of hiding communication; a stegnographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper's suspicion [7]. It is impossible for someone to extract the hidden message if she/he does not have a special extracting method and a proper secret key. This methodology is gaining popularity with everyday passing because of its unique properties and those days are not far off when it would be adopted by armies of the world for secret message passing.

2. LITERATURE SURVEY
Embedding data, which is to be hidden, into an image requires two files. The first is the innocent-looking image that will hold the hidden information, called the cover image. The second file is the message—the information to be hidden. A message may be plaintext, ciphertext, other images, or anything that can be embedded in a bit stream. When combined, the cover image and the embedded message make a stegoimage. A stego-key (a type of password) may also be used to hide, then later decode, the message. Embedding can be done using many methods, few of which are mentioned below[8-10].

A. Message & Image Manipulation
The secret message to be hidden in the cover, is converted to a bit stream which allows equivalent bit manipulation of the cover, depending on the values in the bit stream of the message. The current system demands that once the bit stream is obtained, it is broken into 8-bit blocks. Similarly, the image is broken into 8-pixel blocks, with each pixel of the block corresponding to a bit in the 8-bit block.

B Least Significant Bit Substitution
LSB substitution is the most popular method used for Stegnography due to its ease of application and less perceptual impact. The current embedding process uses LSB Stegnography as the basis to implement a more robust technique. The secret message is converted to a bit stream and each bit of the message is embedded into the LSB of the pixels of the image. This ensures that the pixel value changes almost by one, which does not result in a significant change in the image quality perceptually. The word *almost* used here is significant as probabilistically there is 50% chance the LSB to be changed is already the one desired, and, hence no change to the image is made.

C Pseudo Random Number Generation
If the message bits are incrementally embedded into the LSB of pixels in the image, it is easy for a cracker, who gains knowledge that Stegnography has been carried out over an image, to obtain the secret message by concatenating the LSBs of the image pixels over the different RGB planes. This vulnerability is solved by using a PRNG to choose the 8-pixel block in which embedding is to be done. This results in distribution of the message over the entire image randomly. Thus, even if LSBs are removed and read, they do not convey any intelligent information.

A user chosen key can be inserted into a pseudo random number generator which will determine a sequence of random numbers. These numbers will indicate the pixel blocks in the image where the least significant bit is to be changed. This makes the system more secure because the reader of the message must know the key in order to determine in which bytes the message bits are hidden. The key must remain unknown to the attacker. If the cover image was known to the attacker, embedding the message in a random way would improve its security.

D Plane Cycling
In order to make detection of hidden data more difficult, the system embeds the message in the three RGB planes randomly. The plane used to embed the data is chosen based on the integer value of each 8-bit message block. That is, if the integer value of the 8-bit message block is between 0 and 85, the embedding of these 8-bits under consideration, takes place in the R plane. Similarly, values ranging between 86 and 170 are embedded in the G plane, while values between 171 and 255 are embedded in the B plane.

3. CONSTRUCTION OF IMAGES
To a computer, an image is a collection of numbers or array of numbers that constitute different light intensities in different areas of the image. This numeric representation forms a grid and the individual points are referred to as pixels. Most images on the Internet consists of a rectangular map of the image's pixels (represented as bits) where each pixel is located and its colour. These pixels are displayed horizontally row by row. The images can be of 8-bits The number of bits

in a colour scheme, called the bit depth, refers to the number of bits used for each pixel . The smallest bit depth in current colour schemes is 8, meaning that there are 8 bits used to describe the colour of each pixel. Digital colour images are typically stored in 24-bit files and use the RGB colour model, also known as true colour. All colour variations for the pixels of a 24-bit image are derived from three primary colours: Red, Green and Blue, and each primary colour is represented by 8 bits. Thus in one given pixel, there can be 256 different quantities of red, green and blue, adding up to more than 16-million combinations, resulting in more than 16-million colours . Through varying the intensity of the RGB values, a finite set of colours spanning the full visible spectrum can be created. Not surprisingly the larger amount of colours. The common image size is 640 * 480 and 256 colours, such image could contain about 300kb of data. where as high resolution 1024*768 image of 24 bits may have size larger than 2 megabits .or 24-bits.[1]

Large images are most desirable for steganography because they have the most space to hide data in .The best quality hidden message is normally produced using a 24-bit bitmap as a cover image.

A 24-bit bitmap is a bitmap header that can be displayed, the larger the file size .A 24-bit bitmap is a bitmap header,

10010101 00001101 11001001 10010110

00001111 11001011 10011111 00010000

followed by the pixels' data. Each pixel is represented by three bytes, representing the red, green and blue color values for that pixel. The higher the number, the more intense that color is for that pixel. The cons to large images are that they are cumbersome to both transfer and upload, while running a larger chance of drawing an "attacker's" attention due to their uncommon size. As a result, compression is often used. [2]

4. ALGORITHM

4.1 Encoder

Block Diagram showing encryption is shown using figure 4.1. The secret data can be embedded inside an image using the algorithm.

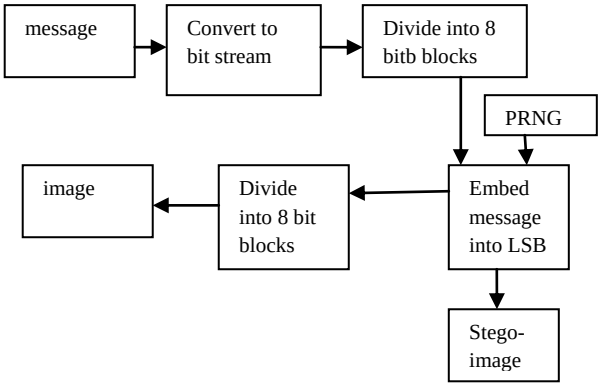

Figure 4.1: Block diagram of encoder

4.2 Decoder

The decryption algorithm can be used to extract the message back from the stego image. The algorithm can be understood using Fig 4.2.

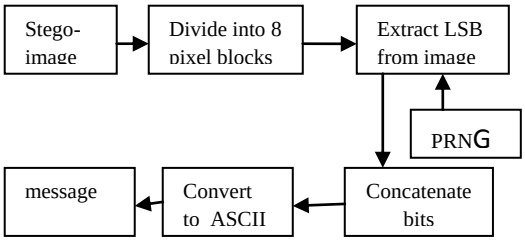

Figure 4.2: Block Diagram of Decoder

5. IMPLEMENTATION OF 12 SQUARE CIPHER ALGORITHM

A cipher is a method for encrypting a message i.e., for transforming the message into one that can't be easily read. The original message is called the plain text and the encrypted message is called cipher text. In cryptography, a substitution cipher is a method of encryption by which plain texts are replaced with cipher text according to a regular system and the receiver deciphers the text by performing an inverse substitution.

The twelve-square substitution cipher encrypts alphabets as well as digits and special characters. It uses total twelve squares, out of which six squares of 5×5 matrices for alphabets(a to z) with omitting ‘q’ so that it can fit into square and another six squares of 6×7 matrices for digits and special characters as shown in table-5.1 and table-5.2.

Table 5.1 Plain Text and Cipher Text for Alphabets

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

Table 5.2 Plain Text and Cipher Text for Digits and Special Character

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 ! & + ; <

For converting plain text to cipher text, if the character is an alphabet it refers to table-4.1, otherwise if it is a number or a special character it refers to table-5.2. 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. Similarly fourth alphabet corresponds to square-1 and square-4, fifth alphabet corresponds to square-2 and square-5, sixth alphabet corresponds to square-3 and square-6 and so on.

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-5.2. 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, fifth special character (including numbers) corresponds to square-8 and square-11, sixth special character (including numbers) corresponds to square-9 and square-12 and so on.

For example:

If plaintext is:

Secret mission in New York on Oct.21, 2012 at 10.30 P.M.

It's cipher text is:

pzxozk jdjpdek dd kzo ylhf ld lxk]^~^ ^56^ gi ~0}-0 f}j}

The encryption does not appear in the image and hence the secret message is successfully embedded.

5.1 Data Embedding

The data embedding process is based on value of index variable. For this first the carrier image is converted into binary form i.e. one pixel value is of 8-bit binary number.Now cipher text of secret message is converted into 8-bit binary number i.e. in the form of bytes. Then calculate the length of cipher text or number of bytes, say it is n. Divide it by 3, and remainder of it is say P. The P called as the index variable.

Table 5.1.1. Index Variable

Present value of Index variable(P)	Bit Locations	Next value of Index variable(P)
0	6 & 7	1
1	7 & 8	2
2	6 & 8	0

As shown in table 4.3, if the value of index variable is P=0, then it corresponds to 6th and 7th bit locations of pixel, P=1 corresponds to 7th and 8th bit locations, P=2 corresponds to 6th and 8th bit locations of any pixel of the carrier image. If present value of P=0 hide the two bits of cipher text in 6th and

7th bit locations of the present pixel, and next value of P = 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, and next value of P is 0 for the next pixel. 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.

Example:

The cipher text of secret message to be embedded is:

11001101 10110110 01011011

Here length of cipher text =3 bytes so,

P=Rem (n/3) =0

Suppose, the different bytes of carrier image are:

	Byte A	Byte B	Byte C
Before Embedding:	01010011	01011101	11010011
	P=0	P=1	P=2
	⇩	⇩	⇩
After Embedding:	01010**11**1	010111**00**	11010**11**1

It can be seen that in byte A of the carrier image, for present value of P=0 we embedded the data bits 11 of cipher text in 6th and 7th bit locations, and next value of P becomes1.We

embed the next data bits 00 into byte B in 7th and 8th bit locations, next value of P becomes 2. Now we embed the next two bits 11 in byte C in 6th and 8th bit locations and so on.

6. SIMULATION AND RESULTS

It can roughly be separated into two parts about estimating the image quality which is altered by data hiding. The first method directly makes use of human vision system. It is simple for a human vision system if there are huge changes in the specific place of the image, such as correction the smooth area in the image. However, it is hard mostly for human eyes to detect the slight difference of the two images in tolerable range. Moreover, the sense of sight in each man is not the same so it is difficult to establish an estimative standard. By using Peak Signal to Noise Ratio (PSNR) to measure the image quality is the other objectively estimative standard. The formula is shown below:

$$\text{PSNR}=10\times\log_{10}\frac{255^2}{MSE}$$

Where the mean square error (MSE) is defined as:

$$\text{MSE}=\frac{1}{N}\sum_{i=1}^{N}(I_i-I_i')^2$$

The MSE value represents the difference between two images I and I' sized *N* pixels. Higher PSNR show the better image quality. In general, it is difficult for human eyes to discriminate the difference between the two images when PSNR is larger than 30dB.The higher PSNR is not enough to show that the image quality is better in detail. Besides PSNR, we also need the human eyes to assist us in deciding the image quality.

The Twelve Square Substitution cipher algorithm using index variable is implemented and simulated using gray scale bitmap Baboon image of size (256×256) pixels as cover image with secret message 'Secret mission in New York on Oct.21, 2012 at 10.30 P.M.' The proposed technique is designed using Matlab 7.0.1 for programming.

6.1 Statistical Analysiss

Statistical analysis is also performed on stego image. Statistical parameters like mean, variance are calculated from image before and after encoding of different sizes of secret message. Statistical analysis is one of the most accurate method for calculating pixel values in image. The values of stego image and cover image are compared. The change in values reveals that something is hidden in the image.

Table 6.1:Statistical analysis results for Baboon cover and stego image

Baboon Image(256×256)				
Embedding Capacity	Mean		Variance	
	Cover Image	Stego Image	Cover Image	Stego Image
2KByte	66.6278	66.6113	86.8562	86.8179
4KByte	66.6278	66.5891	86.8562	86.7429
6KByte	66.6278	66.5657	86.8562	86.6898
8KByte	66.6278	66.5364	86.8562	86.6473
10KByte	66.6278	66.5229	86.8562	86.6433

From the table, it is observed that mean and variance values of stego image are nearly equal to the cover image i.e. the values change only in their decimal positions thereby proving that the proposed method can effectively resist steganalysis. Thus, it is difficult for a steganalyst to find a secret data hidden in the stego image.

6.2 Performance Analysis

The visual quality of a stego-image is important in evaluating the performance of any steganographic technique, peak-signal-to-noise ratio (PSNR) was adopted for these simulations. PSNR is a well-known measurement for evaluating the similarity of a stego-image to its original image. That is, a high PSNR value implies that the stego-image is most similar to its original image. In contrast, a small PSNR value implies that a stego-image is observably different from its original image. Generally, the human eye finds it hard to distinguish any distortion in a stego image with a PSNR value higher than 30 dB.

The performance in terms of capacity and PSNR are demonstrated in the table.

Table 6.2: Performance analysis results for Baboon image

Baboon Image(256×256)		
Embedding Capacity	PSNR(dB)	MSE
2KByte	51.8620	0.4235
4KByte	48.6663	0.8840
6KByte	46.971	1.3058
8KByte	45.6328	1.7775
10KByte	44.6381	2.2349

According to simulation results the PSNR is still a satisfactory value even the highest capacity case is applied. So the quality of stego image is high and unintended observers will not be aware of the existence of hidden important data. Indeed, it is impossible to distinguish cover image and stego image using naked eye, which indicates that hidden data is preserved. The high value of PSNR indicates that both images are of acceptable quality.

7. CONCLUSION

Stegnography is in nascent stage of development. In the Modified technique, the message to be hidden is embedded in the LSBs of the pixels of the image. Furthermore, an element of randomness is added by use of the Pseudo Random Number Generator. This helps in dispersing the message over the entire image and makes retrieving the hidden message from the image an extremely difficult task if the key of the PRNG is not known. More dispersion is guaranteed by the cyclic plane technique, as Stegnalysis of no single plane can reveal an intelligent message.

The resulting stego image obtained contains minimal visually perceptible changes. This is a positive sign as the goal of steganography is to achieve a system where a stegoed-image may pass unnoticed without any challenge to its integrity. The size of the message that can be embedded in an image is only restricted by the size of the image. Although, it is technically more secure to hide smaller messages in an image as statistical analysis in such cases yields very little information regarding any change to the original image.

8. ACKNOWLEDGMENTS

It is great pleasure to express our deep sense of gratitude to our colleagues for suggesting this interesting topic. Their constant guidance, support and encouragement contributed very much in our efforts for completing this paper. The lab facility provided by Bharati Vidyapeeth College of

Engineering, Navi Mumbai helped us to implement the above proposed idea.

9. REFERENCES

[1] Beenish Mehboob and Rashid Aziz Faruqui, "A Steganography Implementation", Bahria University, Department of Computer Science and Engineering IEEE, 2008.

[2] Neil F.Johnson, Sushil Jajodia, "Exploring Stegnography: Seeing the Unseen", George Mason University, IEEE Computer, Feb 1998, pp 26-34

[3] Kh.Manglem Singh, LShyamsudar Singh, A.Buboo Singh, "Hiding Secret Message in Edges of the image", International Conference on Information and Communication Tech, ICICT, 2007Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd.

[4] Mehdi Hussain, Mureed Hussain, " Pixel Intensity Based High Capacity Data Embedding Method," Shaheed Zulfikar Ali Bhutto Institute of Science and Technology (SZABIST)Islamabad, Pakistan.IEEE2010

[5] Gandharba Swain and Saroj Lenka, "Steganography Using the Twelve Square Substitutio Cipher and an Index Variable," Dept of I T, GMR Institute of Technology, Rajam-53212 Andhra Pradesh, India., IEEE 2011.

[6] WangYan and Ling-di Ping, "A New Steganography Algorithm Based on Spatial Domain",Second International Symposium on Information Science and Engineering,IEEE 2009,pp 171- 176.

[7] Hamed Modaghegh, Seyed Alireza Seyedin, "Active Steganalysis of Transform Domain Steganography Based on Sparse Component Analysis" Journal of Information Systems and Telecommunication, Vol. 3, No. 2, April-June 2015.

[8] Ratul Choudhury, Samir Kumar Bandyopadhyay, "LSB Based Audio Steganography Using Pattern Matching" Journal of Multidisciplinary Engineering Science and Technology (JMEST) ISSN: 3159-0040 Vol. 2 Issue 11, November – 2015.

[9] Dipanwita Debnath, Suman Deb, Nirmalya Kar, An Advanced Image Encryption Standard Providing Dual Security: Encryption Using Hill Cipher & amp; RGB Image Steganography International conference on Computational Intelligence and Networks (CINE), 12-13 Jan.2015.

[10] Subarna Shakya1 , Sanjita Lamichhane, "Secured Crypto Stegano Data Hiding Using Least Significant Bit Substitution And Encryption", Journal of Advanced College of Engineering and Management, Vol. 2, 2016