ISSN No. 0976-5697

Volume 6, No. 6, July-August 2015

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

# Multistage Image Encryption using Rubik's Cube for Secured Image Transmission

T.Gomathi
Research Scholar,
Department of Computer Science,
Karpagam University, Coimbatore – 641021,
Tamilnadu, India,

B.L. Shivakumar
Director,
Department of Computer Applications,
Sri Ramakrishna Engineering College,
Coimbatore – 641022,
Tamilnadu, India,

***Abstract***: Steganography is a process of hiding one data behind an image. A text data or an image in one format is being hidden in other image or text data of the same format or of the different format. The data transmitted nowadays are being hacked easily by intruders, such that the purpose of secured transmission fails there. There are several traditional ways of transmitting data such as encryption, scrambling, watermarking, steganography, etc; the process of encryption involves changing data in one format to the other and transmitting. When the decryption method is known to the intruders then the data is easily available for them. Most of the encryption techniques are easy to predict. A new method for image encryption called multistage image encryption using Rubik's cube method is,proposed in this paper.The image resolution doesn't change much more and is negligible when a message is embededinto the image and the image is protected.

***Keywords***: Encryption, HVS (Human Visual System), LSB (Least Significant Bit), PSNR (Peak Signal Noise Ratio), Steganography, Rubik's cube algorithm, NPCR (Number of Pixel Change Ratio).

## I. INTRODUCTION

The word **s**teganography is derived from the Greek words stegos meaning cover and grafia meaning writing defining it as covered writing. In image Steganography the information is hidden exclusively in images. **Steganography** is the art and science of secret communication .It is the practice of encoding/embedding secret information in a manner such that the existence of the information is invisible. The original files can be referred to as cover text, cover image, or cover audio. After inserting the secret message it is referred to as stego-medium.

The Least Significant Bit (LSB**)** insertion is the most common spatial domain technique, which consecutively replaces the least significant bit of cover image with the message bits. This method exploits the natural weakness of Human Visual System (HVS) in recognizing the slight difference of colors. The LSB method changes some or all the 8th bit of image's data so that the image's alteration is not perceptible for any human eyes. In this manner, when using a color image the LSB of each of the red, green and blue components can be used. Therefore, the potential capacity for hiding secret data in a color image is triple of the same image size in the Grayscale mode. Furthermore, when the data is embedded subsequently to all bytes of cover image, it would be rather easy to detect and extract the message. A moderately more secure method is to encrypt before performing Steganography[1]**.** The message image which is to be hidden is being encrypted so that the identifying the hidden image presence will become difficult. The protection of images is of particular interest in this paper. Traditional image encryption algorithms such as private key encryption standards (DES and AES), public key standards such as Rivest Shamir Adleman(RSA), and the family ofElliptic-Curve-based Encryption (ECC**)**, as well as the International Data Encryption Algorithm (IDEA), may not be the most desirable candidates for image encryption, especially for fast and real-time communication applications. In recent years, several encryption schemes have been proposed. These encryption schemes can be classified into different categories such as value transformation, pixels position permutation, and chaotic systems.

The security of image encryption has been extensively studied [2]. Almost some encryption schemes based on permutation had already been found insecure against the cipher text-only and known/chosen-plaintext attacks, due to the high information redundancy, and it is quite understandable since the secret permutations can be recovered by comparing the plaintexts and the permuted cipher texts [3]. Generally, chaos-based image encryption algorithms are used more often than others but require high computational cost. Moreover, a chaos system is defined on real numbers while the cryptosystems are defined on finite sets of integers. One-dimensional chaotic cryptosystems are limited by their small key spaces and weak security.

This paper is organized, where Section II describes LSB Steganography algorithm [4] and the ProposedMultistage Image Encryption Using Rubik's cube method given in section III [6]. The Experimental results of the proposed method are discussed in Section IV followed by Conclusion in Section V.

### *Area of Operation*

The ***LSB*** traditional steganography technique is the Least Significant Bits algorithm. It involves converting the data into binary format and then replacing them in the places of binary values of the cover image (the image which hides the data). This technique when used for hiding images behind images failed because of loss in data. The image which is to be hidden is converted into binary form and the Most Significant Bits (MSB) of it is alone replaced in the places of LSB of the cover image. This process will yield some amount of loss in the data at receiving side.

*Bit Plane Complexity Steganography* ***(BPCS***) involves embedding secret information in true color images by replacing the complex areas in the cover image. This method consumes more time to perform the embedding algorithm and it always requires a 24 bit true color image as cover image. And the above mentioned methods, on choosing an improper cover images will show traces when an image is hidden behind it. Also these methods will not attain a ***PSNR*** (Peak Signal to Noise Ratio) as infinity. Only a transmission with PSNR infinity will contain the secret image exactly without a loss.

## II. RELATED WORK

Steganography is done in several ways like hiding text in image, audio in image or image behind image. This work implements a method for hiding image behind image. The system in order to have a secure transmission with no data loss uses two new techniques namely Quad Conceal steganography and Enigma Intermix cube encryption for secured data transmission,Which involves the following steps.

1. Image Importing
2. Pre-processing image
3. Generate Public Key and Private Key
4. Key Based Encrypting image
5. Constructing stegano image
6. Transmitting data
7. Reconstructing Stegano image
8. Key based decrypting image
9. Calculating PSNR

***A. Image Importing:***The data chosen in this method is an image. The image to be hidden and the image which is to hide the secret image are to be chosen. The image chosen are true color RGB images and are processed as such. The format of an RGB image could be in png, jpeg, tiff, etc,

***B. Pre-processing Image:***Basically several images will not be of good quality, perfect size or with good brightness and contrast. Such images when taken for further processing may result unexpectedly. So every image should be enhanced in terms of quality and resized properly. Two Resizing process are done here.

1) The secret image is resized to a size of A x A (for ex, A=128).

2) The cover image is resized to a size of 2A x 2A (for ex, when A=128, 2A=2*128=256).

**C.** ***Generate Public Key and Private Key:*** Public-key refers to a cryptographic mechanism. It has been named public-key to differentiate it from the traditional and more intuitive cryptographic mechanism known as: symmetric-key, shared secret, secret-key and also called private-key.

Symmetric-key cryptography is a mechanism by which the same key is used for both encrypting and decrypting; it is more intuitive because of its similarity of locking and unlocking a door: the same key. This characteristic requires sophisticated mechanisms to securely distribute the secret-key to both parties. Public-key on the other hand, introduces another concept involving key pairs: one for encrypting, the other for decrypting. This concept, which explained below is very clever and attractive, and provides a great deal of advantages over symmetric-key:

• Simplified key distribution
• Digital Signature
• Long term Expression

Public-key is commonly used to identify a cryptographic method that uses an asymmetric-key pair (a public-key and a private-key). Public-key encryption uses that key pair for encryption and decryption. The public-key is made public and is distributed widely and freely. The private-key is never distributed and is kept secret. Given a key pair, data encrypted with the public-key can only be decrypted with its private key; conversely, data encrypted with the private-key can only be decrypted with its public key. This characteristic is used to implement encryption and digital signature.

***D. Encrypting Image:***Encryption is a process of converting data in one form to the other such that the input data which was in an easy understandable form,is converted to less understandable format after encryption. There are several encrypting algorithms for images. Some among them are listed below.

***i) Data Encryption Standard:***The DES algorithm is one of the traditional encryption techniques used earlier. It involves encryption for data in the form of bits. For a 32-bit block it needs 48 bit key to encrypt it and for a 64 bit block of data it needs a 56 bit key to encrypt. Since the number of key bits is less the security is less in this technique. Moreover this method utilizes more time period to encrypt.

***ii) Block Based Standards:***The block based standards for encrypting involves converting the image into smaller blocks and transferring them into another form. These transferred blocks are again replaced in the same location. Since the entire process takes places only in terms of blocks the encrypted image consists of patterns with difference between each block in the image.

***iii) Rubik's Cube Method:***This algorithm involves considering image into six blocks which forms the six faces of a cube and shuffling of the cube takes place. Because of this shuffling of faces the positions of the pixels gets changed. This process when applied in reverse will result in original data. Thus the security level is less in this method.

***D. Enigma Intermix Cube Encryption:***The Encryption technique using in Enigma intermix cube encryption method involves the following steps.

***Step 1:*** Consider a square matrix of image.
***Step 2:*** Perform the sum of elements in all individual rows.
***Step 3:*** If the sum of first row elements are even, perform a right circular shift of that particular row and perform left circular shift if it is odd.
***Step 4*****:** Repeat step 3 for all rows.
***Step 5:*** Now perform column wise sum of all elements.
***Step 6:*** If the sum of first column elements are even, perform a down circular shift of that particular row and perform up circular shift if it is odd.
***Step 7:*** Repeat step 6 for all columns.
***Step 8:*** Convert the finally obtained image matrix data into binary form with each pixel converted into 8 bits.
***Step 9:*** **N**ow perform an XOR operation between binary value of one single key letter say 'k' represented in 8 bits and every element in the binary value matrix.

***Step 10:***After performing XOR operation, the matrix is again converted into integers form. And that form the encrypted image.

The key letter can be any character such that it must have an ASCII value. That key value is always confidential between the sender and the receiver. The intruder can't guess that an XOR operation is being done and if known the key will be unknown.

***E. Constructing SteganoImage:***The encrypted image cannot be send as such to the receiver since it may be hacked by the intruders so the encrypted image is next put into a process of steganography. The method followed for embedding encrypted image in the cover image is named as Quad Conceal steganography. The following are the steps for constructing a stegano image.

***Step 1:*** The encrypted image is now converted into binary form such that each pixel value contains 8 bits of value.

***Step 2:*** The cover image is now divided into four equal parts. Such that each sub divided part is of the size of secret image or the encrypted image.

***Step 3:*** Cover image is also converted into binary values each comprising of 8 bits.

***Step 4*:** The encrypted image's binary values are now divided into 4 groups each containing two bits.

***Step 5:*** The first two bits of MSB of the first pixel are replaced as last two bits in the first cover image group's first pixel. Similarly the second two MSB of the first pixel are replaced as last two bits in the second cover image group's first pixel.

***Step 6:*** Repeat step 5 for all pixels.

***Step 7:*** Finally convert the binary values into integers and thus it forms the stegano image.

***F. Transmitting Data:*** The stegano image after all process is transmitted to the receiver.

***G. Reconstruction of stegano image:*** The stegano image received by the receiver is to be reconstructed to extract the information hidden. The reverse operation of the construction process is performed to reconstruct the image. After performing all steps the encrypted image will be obtained.

***H. Decrypting Image*:** The encrypted image obtained is now applied with the steps of encryption in reverse manner. This process will yield the input secret image. The key character used in decryption process is also the same as in encryption.

## III. METHODOLOGY

### *ALGORITHM FOR MULTISTAGE IMAGE ENCRYPTION USING RUBIK'S CUBE*

In order to have a secure transmission a new method is implemented for image encryption called Multi stage image Encryption using Rubik's Cube. In first stage, an original image I of size 810x360 is taken. The entire image I is divided into 9x6 blocks. We are shuffling the pixels in each block based on Rubik's Cube method. Let $I_0$ represents a gray scale image of size M x N. Here M=9 and N=6.Here $I_0$ represent the pixels values matrix of image I. The steps involved in Rubik's Cube Encryption Algorithm are as follows:

Let $Io$ represent a $\alpha$ -bit gray scale image of the size $M$×N. Here, lrepresent the pixels values matrix of image. The steps of encryption algorithm are as follows:

***Step1:*** Generate randomly two vectors $KR$ and $KC$ of length $M$ and N, respectively. Element $KR$ $(i)$ and $KC$ $(j)$ each take a random value of the set $\mathcal{A}$= {0, 1, 2, 2 $\alpha$ −1}. Note that both $KR$ and $KC$ must not have constant values.

***Step 2*:** Determine the number of iterations, ITER max, and initialize the counter ITER at 0.

***Step 3*:** Increment the counter by one: ITER=ITER+1.

***Step 4*:** For each row $i$ of image $I$,

(a) Compute the sum of all elements in the row, this sum is denoted by $(i)$, $(i)$ =$N.j$=1 l$o$ $(i,)$, =1, 2..,

(b) Compute modulo 2 of $\alpha(i)$ , denoted by $M\alpha(i)$ ,

(c) Row $i$ is left, or right, circular-shifted by $KR$ $(i)$ positions (image pixels are moved $KR$ $(i)$ positions to the left or right direction, and the first pixel moves in last pixel.), according to the following:

If $(i)$ =0⟶rightcircularshift Else⟶ left circular shift

***Step 5:*** For each column $j$ of image,

(a) Compute the sum of all elements in the column, this sum is denoted by $(j)$, $(j)$ =$M.i$=1 l$o$ $(i,)$, =1, 2… $N$,

(b) compute modulo 2 of $\beta(j)$ , denoted by $M\beta(j)$ .

(c) Column$j$ is down, or up, circular-shifted by $KC$ $(i)$ positions, according to the following:

If $(j)$ =0⟶upcircularshiftelse⟶downcircularshift. (4)

Steps 4 and 5 above will create a scrambled image, denoted by $I$ SCR.

***Step 6:*** Using vector, the bitwise XOR operator is applied to each row of scrambled image $I$ SCR using the following expressions:

$$\text{IENC1 } (2i-1, \text{j}) = \text{I SCR } (2i-1,) \oplus KC \text{ I } (j),$$

$$\text{ENC1 } (2i,\text{j}) = \text{I SCR.} K(2i,j) \oplus \text{rot180 } C.(j) \quad (5),$$

Where⊕ and rot180 $(KC)$ represent the bitwise XOR operator and the flipping of vector $KC$ from left to right, respectively.

***Step 7:*** Using vector, the bitwise XOR operator is applied to each column of image $I$1 using the following formulas:

$$\text{IENC } (i, 2j-1) = \text{I1 } (i, 2j-1) \oplus KR\text{I } (j),$$

$$\text{ENC } (i, 2j) = \text{I1.} K \ (i, 2j) \oplus \text{rot180 } R.(j) \quad (6),$$

With rot180 $(KR)$ indicating the left to right flip of vector $KR$.

***Step 8*:** If ITER=ITER max, then encrypted image $I$ENC is created and encryption process is done; otherwise, the algorithm branches to step 3.

Vectors and the max iteration number ITER max are considered as secret keys in the proposed encryption algorithm. However, to obtain a fast encryption algorithm it is preferable to set ITER max =1 (single iteration). Conversely, if ITER MAX >1, then the algorithm is more secure because the key space is larger than for ITER MAX =1. Nevertheless, in the simulations presented in Step 3, the number of iterations ITER max was set to one.

## IV RESULTS AND DISCUSSIONS

The algorithm is implemented in CPU Pentium IV using MAT Lab R 2013where large data set was substituted and verified (Shown in the table). The PSNR value where calculated for all Images using the below given formula.

***PSNR Calculation:***

After the decryption process the image at the receiver side and the original image are subjected to calculate PSNR. By the value of PSNR the loss in image recovering can be calculated. PSNR is calculated by using the formula,

$$PSNR = 10 \log \frac{255^2}{MSE}$$

Where,

$$MSE = Mean\left[\sum[(ORG - RECONSTRUCTED)]^2\right]$$

ORG= Original image,
RECONSTRUCTED= Reconstructed image.

Fig 1: Message Image

Fig1.1: Cover Image

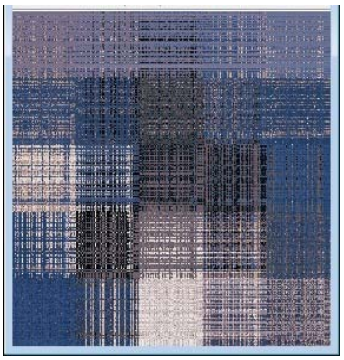

Fig1.2: Encrypted Image (Rubik's Magic Cube Method)

Fig 1.3: Stego Image

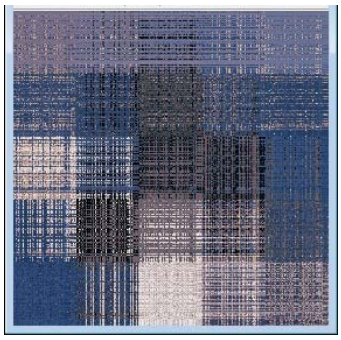

Fig 1.4: Decrypted Image

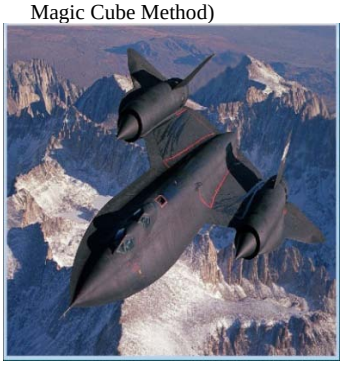

Fig 1.5 Reconstructed Image

Fig 2: Message Image

Fig 2.1: Cover Image

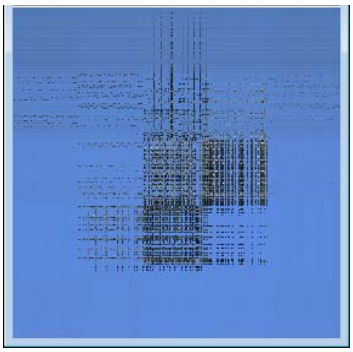

Fig2.2:EncryptedImage (Rubik's Magic cube method)

Fig 2.3: Stego Image

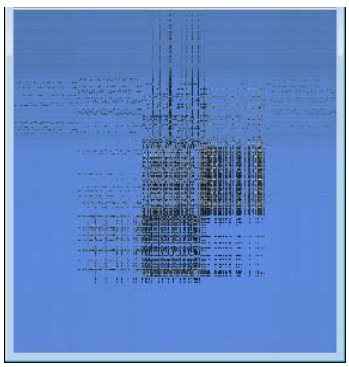

Fig 2.4: Decrypted Image

Fig2.5: Reconstructed Image

Fig 3: Message Image

Fig 3.1: Cover Image

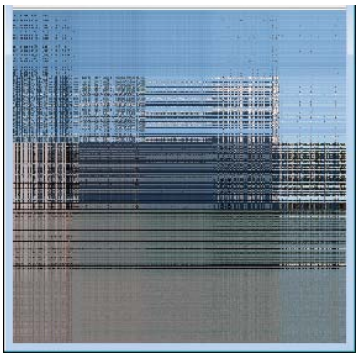

Fig 3.2: Encrypted Image(Rubik's Magic Cube Method)

Fig 3.3: Stego Image

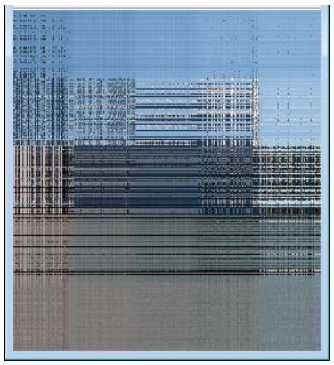

Fig 3.4: Decrypted Image

Fig 3.5: Reconstructed image

TABLE 1: Comparison of Images using Multi-Resolution Magic Cube Methodology (MRMC)

| ***S.NO*** | ***Image No.*** | ***PSNR*** | ***NPCR-R plane*** | ***NPCR- G plane*** | ***NPCR- B plane*** | ***Time Consumed For (secs)*** | |
|---|---|---|---|---|---|---|---|
| | | | | | | ***Encryption*** | ***Decryption*** |
| 1 | Image1 | Infinity | 94.3362 | 95.08005 | 96.22038 | 21.15251 | 8.77106 |
| 2 | Image2 | Infinity | 94.19292 | 95.896 | 95.80159 | 21.4997 | 8.9383 |
| 3 | Image3 | Infinity | 95.2629 | 95.43 | 94.98329 | 21.19 | 8.8253 |

## V .CONCLUSION

The proposed LSB Steganography along with Rubik's cube Encryption method provides high security and protects the image during Transmission. The image resolution does not change much and is negligible when the message is embedded in to image and the image is protected with magic cube rotation. Further the speed of embedding data in to the Image is also high.

## VI.REFERENCES


1. T. Bianchi, A. Piva, and M. Barni, "Encrypted domain DCT based on homomorphic cryptosystems," EURASIP J. Inf. Security, 2009, Article ID 716357.
2. W. Liu, W. J. Zeng, L. Dong, and Q. M. Yao, "Efficient compression of encrypted grayscale images," IEEE Trans. Imag. Process. vol. 19, no. 4, pp. 1097–1102, Apr. 2010.
3. W. Lu, A. L. Varna, A. Swaminathan, and M. Wu, ''Secure image retrieval through feature protection,'' in Proc. IEEE Conf. Acoust., Speech Signal Process., Apr. 2009, pp. 1533–1536.
4. T.GomathiB.L.ShivaKumar," Suspection Less Steganographic Approach Using Enigma Intermix Cube Encryption Technique", IJITEE, ISSN: 2278-3075, Vol -4, Issue-4, Sept.2014, pp. 52-56.
5. V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in Proc. IEEE Int. Workshop Inf. Forensics Security, Dec. 2012, pp. 234–239.
6. K. L. Chiew and J. Pieprzyk, "Binary image steganographic techniques classification based on multi-classsteganalysis," in Information Security, Practice and Experience. Berlin, Germany: Springer-Verlag, 2010, pp. 341–358.
7. M. Guo and H. Zhang, "High capacity data hiding for binary image authentication," in Proc. Int. Conf. Pattern Recognit., Aug. 2010, pp. 1441–1444.