IMPLEMENTATION OF AN IMAGE STEGANOGRAPHY TECHNIQUE USING X(X-OR)-BOX MAPPING

[1]PRASANNAKUMAR PATIL, [2]SATISH SHET.K

[1]M-Tech (VLSI & EMBEDDED SYSTEMS), [2]Assistant Professor
Dept. of Electronics and Communication, JSS Academy of Technical Education, Bengaluru, INDIA
E-mail: [1]Prasannapatil16@gmail.com, [2]Satish.personal@gmail.com

Abstract- Image steganography is a method of concealing information in to cover image to hide it. Least significant bit (LSB) based approach is most popular stenographic techniques in the spatial domain due to its simplicity & hiding capacity. This paper presents a novel technique for image stenography based on the LSB using X box mapping, where we have used several X boxes having unique data the embedding part is done by this steganography algorithm, where we use four unique X boxes with sixteen different values (expressed by 4 bits) &each value is mapped to 4 LSB's of cover image. This mapping provides sufficient security to payload because without knowing mapping rules no one can extract secret data.

Keywords- Steganography, Lsb Technique, Information Hiding, X-Box

I. INTRODUCTION

It's well known fact that now a days the development of the internet technologies is growing rapidly to the peak level, at present the transmission of the digital media is convenient over the networks. But the transmission of the secret message in the internet suffers from the serious security overhead. Hence protection of the secret message during transmission plays an important role. This idea results in steganography, which is a branch of information hiding by camouflaging secret information within other information. The word steganography in Greek means"covered writing" (Greek words "stegos" meaning "cover" and "grafia" meaning "writing") [2]. The main objective of steganography is to hide a secret message inside harmless cover media in such a way that the secret message is not visible to the observer. Essentially, the information-hiding process in a stenographic system starts by identifying a cover medium's redundant bits (those that can be modified without destroying that medium's integrity). The embedding process creates a stego medium by replacing these redundant bits with data from the hidden message. Steganography differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated. The strength of steganography can thus be amplified by combining it with cryptography. Fig (1) shows simple block diagram of stenography, in which the secret image is hidden in to the cover image of size NxN to form the Stego image of same size.

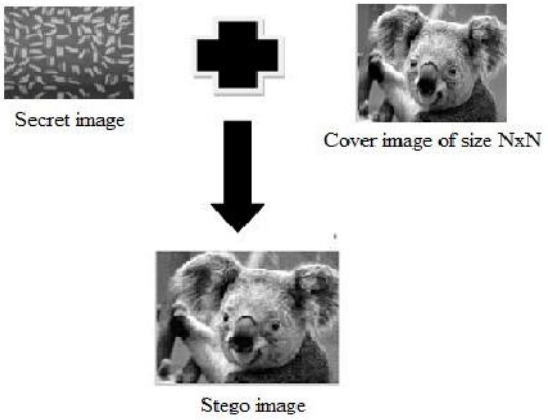

Figure 1: Block Diagram Of Simple Steganography System

II. PARAMETERS USED

2.1 Least Significant Bit (LSB):

Least significant bit (LSB) stenography [2, 3, 4, 5and 6] is common and simple approach to embed information in cover file. It reserves the image quality and requires no complex operation. Capacity, security, robustness are the main aspects, which affects the stenography and its usefulness. Capacity refers to the amount of data bits that can be hidden in the cover medium security relates to the ability of an eavesdropper to the figure the hidden information easily, the robustness is concerned about the resist possibility of modifying or destroying the unseen data

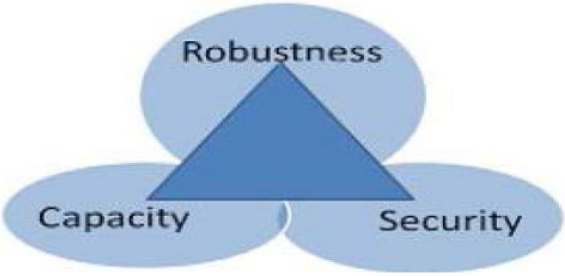

Figure 2: Capacity Security Robustness

2.2PSNR (Peak Signal to Noise Ratio)

The measurement of the quality between the cover image f and stego-image g of sizes N x N shown in figure (1) is defined as:

$$PSNR=10 \times \log (255^2/MSE)$$

$$\text{Where } MSE=\sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \left(f(x,y)\ g(x,y)\right)^2 /N^2$$

Where f(x,y) and g(x,y) means the pixel value at the at position (x, y) in the cover-image and the corresponding stego-image respectively. The PSNR is expressed in db. The larger PSNR indicates the higher the image quality i.e. there is only little difference between the cover-image and the stego-image. On the other hand, a smaller PSNR means there is huge distortion between the cover-image and the stego-image.

III. PROPOSED IMAGE STEGNOGRAPHY

Our proposed image stenography technique is based on mapping the different values from X-boxes

Now let's see the block diagram of encoding steganography

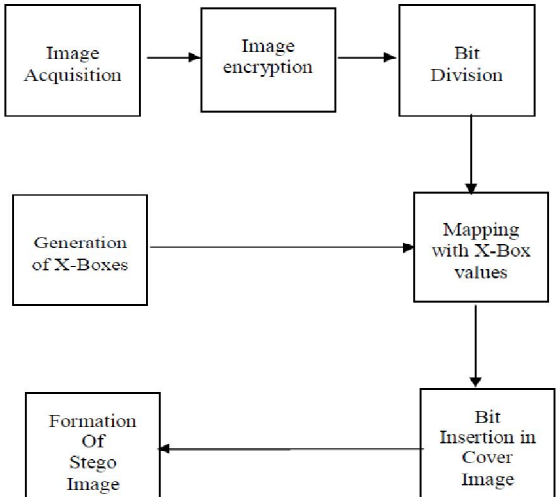

Figure 3: Block Diagram Of Encoding Steganography

3.1 GENERATION OF X-(XOR) BOXES:

To generate the X-boxes we usually use 2x2 matrixes, where 16(0 to 15) values are stored as given below

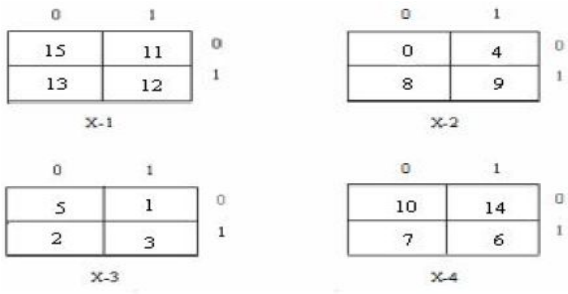

Figure 4: X-Boxes Mapping

To insert the values in X-boxes, we use X-OR property: 0 XOR 0 = 0, 1 XOR 1 =0 and 0 XOR 1 = 1, 1 XOR 0=1 for example 9 is inserted in any one of the x-boxes as follows I.e. 9=1001=10 XOR 01=11

Hence the position of 9 is 1st row 1st column

3.2 BIT DIVISION:

In case of bit division method we need to take the secret encrypted image say with the dimension of 64 x 64.Then after we need to convert the values from decimal to binary. For example, The first pixel value of encrypted image is=152 Then binary value of 152 is (152)10= (10011000)2

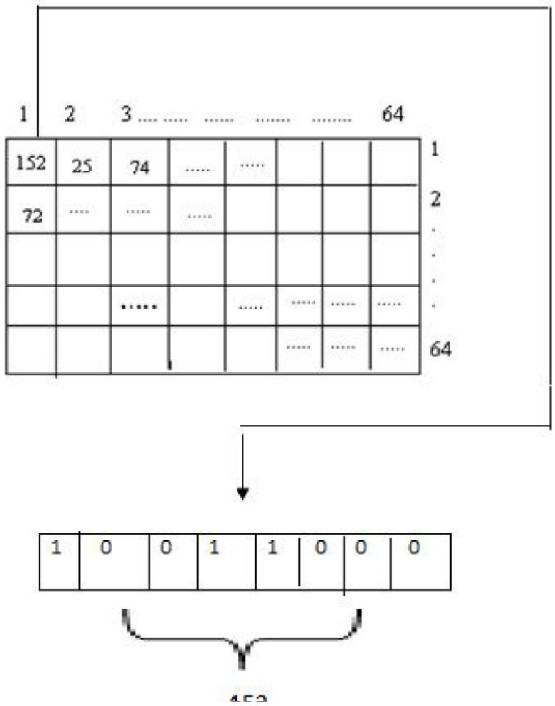

Figure 5: Bit Division

Now we map the values of b1, b2, b3, b4, from the x-box mapping method First we take b1 =10; Then we search the value of 1st row and 0th column of the XI box, after mapping we get the value (13) 10 = (1101)2

similarly we get mapping values for the b2, b3, b4 we get in the same way 4,2,10 sequentially, thus in this way the x-box mapping is done. Now let's take the example of 152, which is to be converted from decimal to the 8 bit binary value

Here the 8 bit value is divided in 4 equal parts by Taking 2 bits

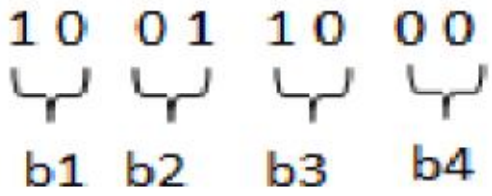

Figure 6: X-box Mapping

3.3BIT INSERTION IN COVER IMAGE:

The newly mapped values that we got from the x-box mapping are said to insert in to the cover image of size NxN, then it is to be placed in to the 4 bit LSB of the cover image sequentially. First we are going to take pixels one by one from the cover image then the 4 LSB bits are replaced by 13, 4,2,10 sequentially

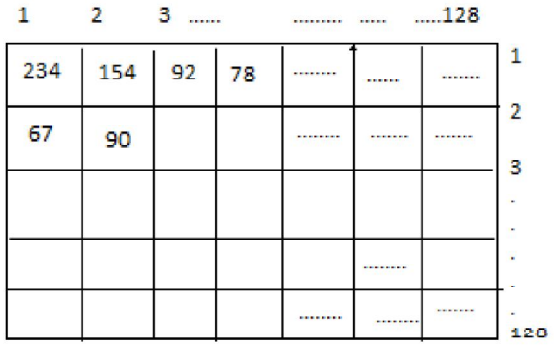

Figure 7: (128 X 128) Cover Image

Here we take the pixel sequentially as
(234)10= (11010110)2;
(154)10= (10011010)2;
(92)10= (01011100)2;
(78)10= (01001110)2;

Hence these pixel values from the cover image of size NxN are replaced by the values 13, 4, 2 and 10 to form the new value, which is nothing but newly formed stego image and for this process we mainly call it as Bit insertion in to cover image. Here in the fig (8) the value 234 of last four bits are replaced by the 13 to get the new value 221 similarly the last four bits of the remaining values are replaced by the 4, 2 and 10 to get the new values and these values which are formed after the bit insertion will form one stego image of same size as the cover image Thus the fig (8) mainly shows how the bit insertion in cover image of size NxN is done.

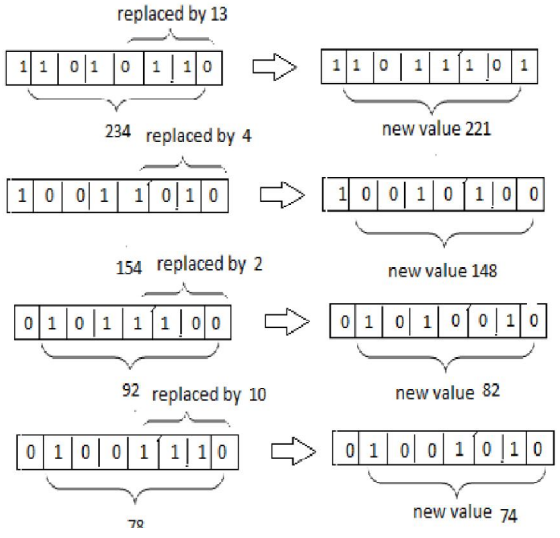

Figure 8: Bit Insertion In To Cover Image

4.2 FORMATIOHN OF STEGO IMAGE:

As soon as we get the new pixel values we form the stego image, which is same as the size of cover image thus the pixel values 221, 148, 81, 78 are placed in to the position of the previous values. Similarly we take the pixel one by one and then insert the secret image in to them and replace them. Finally we get the stego image, which is shown in the figure.

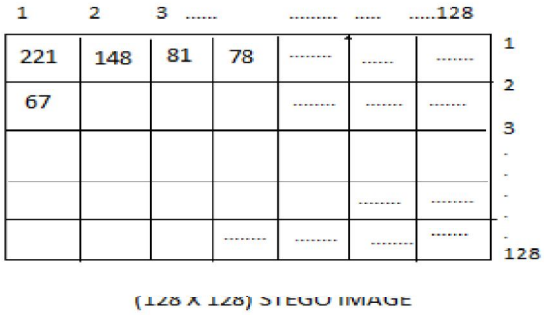

Figure 9: Stego Image

These stego image contains the secret image but it cannot be recognized as the secret image .The change of the pixel value is varied from 0 to 15 which is negligible amount of value. So pixel values or colors will not be changed in great amount

4.3 ENCODING ALGORITHM

Input: A grey-level secret image of size (m x n), A grey level Cover Image of size (2m x 2n); Output: Stego Image of size (2m x 2n); Steps:

1. Divide the each pixel of the secret image into 4 Parts containing 2 bits.
2. Map these 4 parts into the 4 X-boxes and get the new values for each part.
3. Insert these values into the LSB position of the Cover image one by one.
4. End

IV. IMAGE DECODING

To decode the secret image in the receiver side we need to perform the following steps

4.1 GENERATION OF 4 LSB BITS FROM STEGO IMAGE:

Here we take the pixel values one by one then it is transferred in to the binary values and get the 4LSB bits values from it

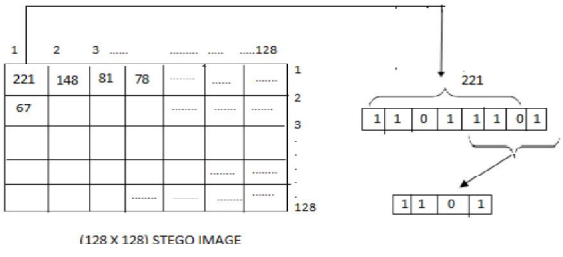

Figure 10: Extraction Of Stego Image

Similarly we take the values of the other three pixels too, thus we have,
(221)10= (11011101)2;
(148)10= (10010100)2;
(81)10= (01010001)2;
(78)10= (01001110)2;
LSB1=1101; LSB2=0100; LSB3=0010; LSB4=1010;

4.3 RETRIEVING THE INSERTED BITS OF SCERET IMAGE:

Here we take the 4 LSB bits of the secret image that are 1101,0100,0001,1110 then we do the x-or operation of first two bits with last two bits i.e. first bit is xored with third bit and second bit is x-ored with fourth bit hence we
LSB1=11 XOR 01 =10;
LSB2=01 XOR 00 =01;
LSB3=00 XOR 10 =10;
LSB4=10 XOR 10 =00;

4.4 CONCATENATION OF RESULT OF X-OR OPERATION:

Now we concatenate the four results of the xor operation as soon as we concatenate the four results we are going to get the 8 bits result then this 8 bit result is transferred in to decimal values ,which is as shown below Concatenated value is,

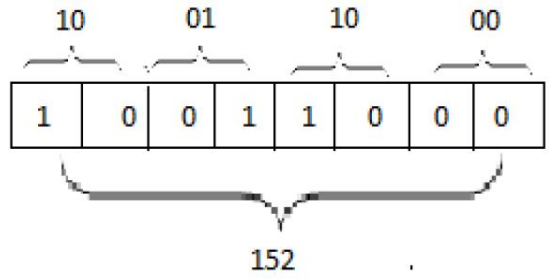

Figure 11: Concatenation Of Results Of X-Or Operation

4.5 GENERATION OF THE SECRET IMAGE:

Now the generated value is placed in to the first position. Similarly we take the next value of the stego image and repeat the steps from 1 to 4 and we get 25, 74 etc. obviously we get the secret image, which is as shown in the figure.

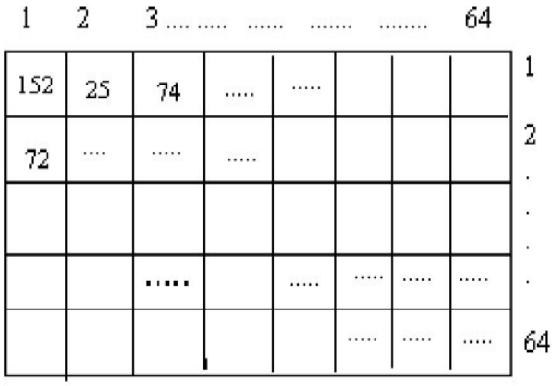

Figure 12: 64 X 64 Secret Image

Hence these are the overall process of the x-box steganography, which is implemented in this paper, now let's see the algorithm of that particular method.

4.6 DECODING ALGORITHM:

Input: Stego Image of size (2m x 2n); Output: A grey-level Cipher image of size (m x n); Steps:

1. Select each pixel of the Stego-image and take 4 Bits from LSB position of stego image.
2. Then Perform the XOR operation of that 4 bit LSB and concatenate the four results.
3. Ultimately we get the pixel value of the secret Image and place one by one to get a secret Image.
4. End

V. EXPERIMENTAL RESULTS

The experimental results in Mat lab is discussed below
5.1 Experimental results in mat lab:
This embedding technique is no doubt a strongest Steganography technique than normal LSB encoding technique. Because, we embed each 2 bits of secret Image into the 4 bit of Cover Image. Again before insertion we code these two bits by some mapping box into another form. Hence one can understand that something is embedded in it, but the mapping will be totally unknown to him. Thus to extract the image is really a tough job.

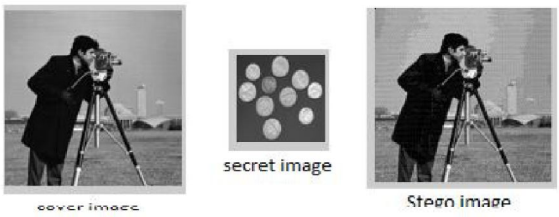

Figure 13: Cover, Secret And Stego Image Of Camera Man Of X-Box Mapping

As we see here in the Stego Image there is no such abroad distortion. Seeing this image no one can recognize that some secret image is embedded in it. We can say that just seeing its PSNR table given below.

Table: PSNR of different images

IMAGE NAME	SIZE (PIXEL)	PSNR in db
Camera man.jpg	64	+31.8
Plane.jpg	64	+31.6
Baboon.jpg	64	+32.5
Lena.jpg	64	+31.2

5.2 Experimental results of Xilinx:

Hardware implementation of image steganography using x-box mapping is done by writing Verilog code in the Xilinx for the various values hence the hardware implementation for encoding part is shown below

Results for encoding part:

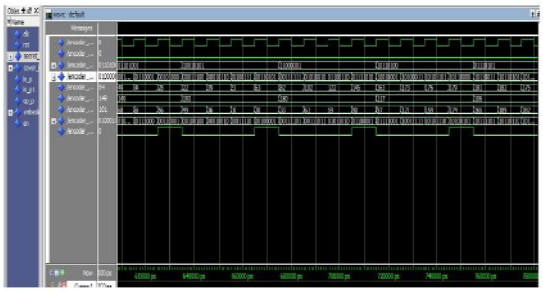

Figure 14: Simulation Results Of Encoding Part

Similarly the hardware implementation for the decoding part is as shown

Results for the decoding part:

Figure 15: Simulation Results Of Decoding Part

CONCLUSION

In this paper, we propose a mapping based steganography Process using x-box method to improve security and image quality compared to the existing algorithms. Compare to other approaches our approach is better because Without stego key, no one can extract the original information from the stego-image, for purposes of secret communication which is more important.

REFERENCES

[1] Amitava Nag', Saswati Ghosh2, Sushanta Biswas, Debasree Sarkar, Partha Pratim Sarkar, "Image stenography technique using x-box", ISBN: 978-81-909042-2-3 ©2014 IEEE

[2] Moerland, T, "Steganography and Steganalysis", Leiden Institute of Advanced Computing Science, www.liacs.nl/home/ trnoerl/privtech.pdf

[3] c.-C. Chang, T.D. Kieu, A reversible data hiding scheme using complementary embedding strategy, Inform. Sci. 180 (16) (2010) 3045-3058.

[4] c.-c. Chang, W.-L. Tai, c.-c. Lin, A reversible data hiding scheme based on side match vector quantization, IEEE Trans. Circ. Syst. Video Technol. 16 (10) (2006) 1301-1308.

[5] W. Luo, F. Huang, J. Huang, Edge adaptive image steganography based on Isb matching revisited, IEEE Trans. Inf. Forens. Security 5 (2) (2010) 201-214.

[6] J. Mielikainen, LSB Matching Revisited, IEEE Signal Process. Lett. 13 (5) (2006) 285-287.

[7] Chen, B. and G.W. Wornell, 2001. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding.lEEE Trans.Inform. Theor., 47: 1423-1443. DOl: 10.1109118.92372

[8] Amitava Nag, Saswati Ghosh, Sushanta Biswas, Debasree Sarkar, Partha Pratim Sarkar,"An Image Steganography Technique using X-Box Mapping," in the proceedings of the International Conference On Advances In Engineering, Science And Management (ICAESM 2012),pp.709-713,Nagapattinam,India,March 2012.

[9] A. D. Ker, "Steganalysis of LSB matching in grayscale images," IEEE Signal Processing Letters, vol. 12. No. 6, pp.441-444, 2005.

[10] W. Luo, F. Huang, J. Huang, "Edge adaptive image steganography based on LSB matching revisited," IEEE Transaction on Information Forensics Security, vol. 5, no. 2, pp.201-214, 2010.

[11] J. Mielikainen, "LSB matching revisited," IEEE Signal Processing Letters, vol. 13, no. 5, pp.285-287, 2006.

[12] M. Shobana, R. Manikandan, "Efficient method for hiding data by pixel intensity," International Journal of Engineering and Technology, vol. 5, no. 1, pp.75-81, 2013.

[13] A. Herrigel, J. J. K. O Ruanaidh, H. Petersen, S. Pereira, and ´ T. Pun, "Secure copyright protection techniques for digital images." In Aucsmith [148], pp. 169–190, ISBN 3-540-65386-4.

[14] M. D. Swanson, B. Zu, and A. H. Tewfik, "Robust data hiding for images." In 7th Digital Signal Processing Workshop (DSP 96), pp. 37–40, IEEE, Loen, Norway, Sep. 1996.

[15] G. C. Langelaar, J. C. A. van der Lubbe, and R. L. Lagendijk, "Robust labeling methods for copy protection of images." In Sethin and Jain [149], pp.298–309, ISBN0-8194-2433-1.

[16] Mr. Vikas Tyagi, ― Data Hiding in Image using least significant bit with cryptography‖, in International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 4, April 2012

[17] Shailender Gupta, Ankur Goyal , Bharat Bhushan, ― Information Hiding Using Least Significant Bit Steganography and Cryptography‖ , in I.J.Modern Education and Computer Science, june 2012

[18] Joyshree Nath, Asoke Nath,‖Advanced Steganography Algorithm using Encrypted secret message‖, in (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No.3, March 2011

★★★