Optimum Pixel & Bit location for Colour Image Stego- A Distortion Resistant Approach

Rengarajan Amirtharajan
Assistant Professor
School of Electrical & Electronics Engineering
SASTRA University

Aishwarya G
Department of ECE
School of Electrical & Electronics Engineering
SASTRA University
Thanjavur

Madhumita Rameshbabu
Department of ECE
School of Electrical & Electronics Engineering
SASTRA University

John Bosco Balaguru Rayappan
Associate Dean Research
School of Electrical & Electronics Engineering
SASTRA University

ABSTRACT
A conventional Secret key Steganography scheme focuses mainly to reduce the distortion when secret information is embedded into the cover image. On the other hand, the transmitted images may be compressed or faces transmitting errors. If such errors occur, the receiver cannot extract the correct information from the stego-image. Furthermore the three main attributes of steganography are capacity, invisibility and Robustness. In the previous models [3, 4] we mainly concentrated on capacity and invisibility but in this method equal importance will be given to robustness. To increase the stochasticity of information hiding we use pixel indicator techniques which are implemented using three methods. Among these the first method enjoins that red channel steers the other two channels and the second method gives us the liberty to select the steering channel which successively increases the robustness of the shrouded message but its limits when MSE is considered. In third method, the steering channel is selected in a cyclic mode which enhances further the capacity along with security of the shrouded message as the MSE gets equally distributed. To increase the robustness here we introduce a factor E which gives us an option to select the position to plant the message to be concealed. The factor E addresses the bit where the embedding can be started. Once an image is compressed the LSBs of the covered media will get affected which defiles the concealed message. The essence of this method rests in the withstanding capability of the carrier media as the factor E is altered. As the value of E increases the MSE gets stepped up and hence the imperceptibility of the carrier image gets diluted. This can be heightened by using Optimal Pixel Adjustment Process (OPAP).

General Terms
Information Security

Keywords
Modified LSB, Optimal Pixel Adjustment Process (OPAP), Pixel Indicator(PI).

1. INTRODUCTION
Ever since the genesis of life on earth communication has been the integral part of every organism's existence as there has been a need to communicate for its own survival. Man even in this aspect of evolution has proved to be superior by inventing languages and tools for effective communication. However with distances being a barrier his necessity to invent advanced methods of communication augmented. The communication technology thus eroded the distance factor and shrunk the world making it one small global village. With the advent of civilization flourished a myriad of surreptitious and clandestine operations which resulted in a necessity for secrecy and security of veritable information. Information hiding thereby became a basic amenity to maintain the integrity of crucial information and has evolved from sheer primitive methods to the advanced division of information hiding. The discipline of information hiding is stratified into various categories which include cryptography [8], steganography and digital water marking [13, 14].

Cryptography is the ancient art of writing a message in caricature such that only the targeted receiver would be able to decipher it. Steganography differs from cryptography in the sense that cryptography focuses on keeping the contents of the message, a secret and steganography focuses on keeping the existence of the message, a secret. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Steganography is implemented [7, 9, 13,14,15] in digital audio, video and images of which image steganography has gained much impetus and reputation in the recent past. In image steganography the critical data is camouflaged in a cover image with immense dexterity resulting in the stego image. The confidential data that is embedded should be of adequate quality in order to make it imperceptible and indecipherable. In addition the technique employed should facilitate the use of a high payload by the diligent embedding of more data in a given cover image.

Information hiding can be assorted into two major categories as copyright marking and steganography [13, 14, and 15]. Usually when image based information hiding is considered, image steganography is used to increase the shrouding capacity and imperceptibility of an image. Copyright marking techniques are aimed at increasing the powers of copyright protection [13-15] and this paper focuses on data hiding in color images.

Steganographic methods proposed recently can be categorized into two types. The methods of the first type employ the spatial domain of a host image to conceal secret data. In other words, secret data are directly embedded into the pixels of the host image [1-7, 9-12, 17, 19]. Steganographic methods of the second type employ the transformed domain of a host image to hide secret data [1, 14, 16, 18]. Transformation functions like the discrete cosine transform (DCT) [6, 14, 16] or discrete wavelet transform (DWT)[18] are first exploited to transform the pixel values in the

spatial domain to coefficients in the frequency domain and then the secret data is embedded in the coefficients.

The frequently experimented steganographic technique in the data hiding field is least-significant-bits (LSBs) substitution [2, 3, 14]. This method embeds the fixed-length secret bits in the same fixed-length LSBs of pixels. Although this technique is simple, it generally causes noticeable distortion when the number of embedded bits for each pixel exceeds three. Several adaptive methods for steganography [5, 9, 15] have been proposed to reduce the distortion caused by LSBs substitution. For example, adaptive methods vary the number of embedded bits in each pixel, and they possess better image quality than other methods using only simple LSBs substitution. However, this is achieved at the cost of a reduction in the embedding capacity and one more option to vary the scanning path has been proposed which gives random Steganography [2, 3, 4, 5, 16].

Many parameters affect Steganography, its design and development. These parameters include security (or perceptual transparency), capacity, robustness, complexity, survivability, capability, and delectability [1, 2, 14]. The relationship between the first three parameters is mostly influential and considerable in most researches in the literature. They consider the following properties: Capacity: This term refers to the amount of data that can be hidden in the medium. Perceptual Transparency/Security: The hiding process should be performed in a way that does not raise any suspicion of eavesdropper. Robustness: It's an ability to withstand attacks and usually applicable to watermarking definitely not for steganography.

If there is an increase in capacity of any cover to store data

with more than certain threshold value, then its transparency will be affected; i.e. with very high capacity,

the steganography is not strong enough to keep data transparent from eavesdroppers. It is required to select the parameters in such a way that steganography can be achieved on the best level accommodating its application need. To summarize in image steganography the critical data is camouflaged in a cover image with immense dexterity resulting in the stego image. The confidential data that is embedded should be of adequate quality in order to make it imperceptible and indecipherable. In addition the technique employed should facilitate the use of a high payload by the diligent embedding of more data in a given cover image.

2. BLOCK DIAGRAM OF THE PROPOSED METHODOLOGY

Normally when data is embedded in an image for concealment, the least significant bits (LSBs) of the pixels of the image are affected and replaced upon. However, when such an image is compressed (especially when stored in .jpg format) and sent over an unsafe channel, then there is very high chance of loss of information because it is the LSBs of every pixel that are going to be affected on compression. In fact the LSB substitution method is a predominant method used and thus message embedded in this method is highly vulnerable owing to the fact that number of message bits embedded in each pixel is same for all pixels, enabling very easy decoding by shrewd analysts. Also payload is not so appreciable in this case.

Thus in this proposed idea, the basic principle of least significant bit (LSB) is employed along with a modified version of the Pixel Indicator Method. The added advantage of this method is the ability to specify which colour of every pixel is to be selected as the indicator, along with the fact that even on compression of image, the embedded data isn't lost. Especially, it is noted that the 5^{th} and 6^{th} bits of the indicator define the channel to be worked upon, and thus the method has a very high stability owing to the fact that on compression even the indicator channel loses no key information. To understand the hidden implications of the concept, consider that every pixel value of a colour image is constituted by three basic bytes of RED, GREEN and BLUE i.e. the total number of bits of the required data to be embedded in every channel is indicated by a certain value and specified by the user.

Furthermore, the strength of this methodology is that the value of the bit position from where the embedding has to start in every channel can be user defined and customizable. Thus these features make the system highly robust, random, very hard to decipher and especially very flexible due to small sizes on compression with intact information hidden in the image. This improved imperceptibility and randomization added to the pliability of the sizes makes it a very fool-proof method of transmission of high profile secret data. The drawback of this method may be attributed to moderate embedding capacity but the high data security along with low memory occupation gives it an edge over most methods. It has been observed through time that there are 3 major procedures that can be followed for data hiding.

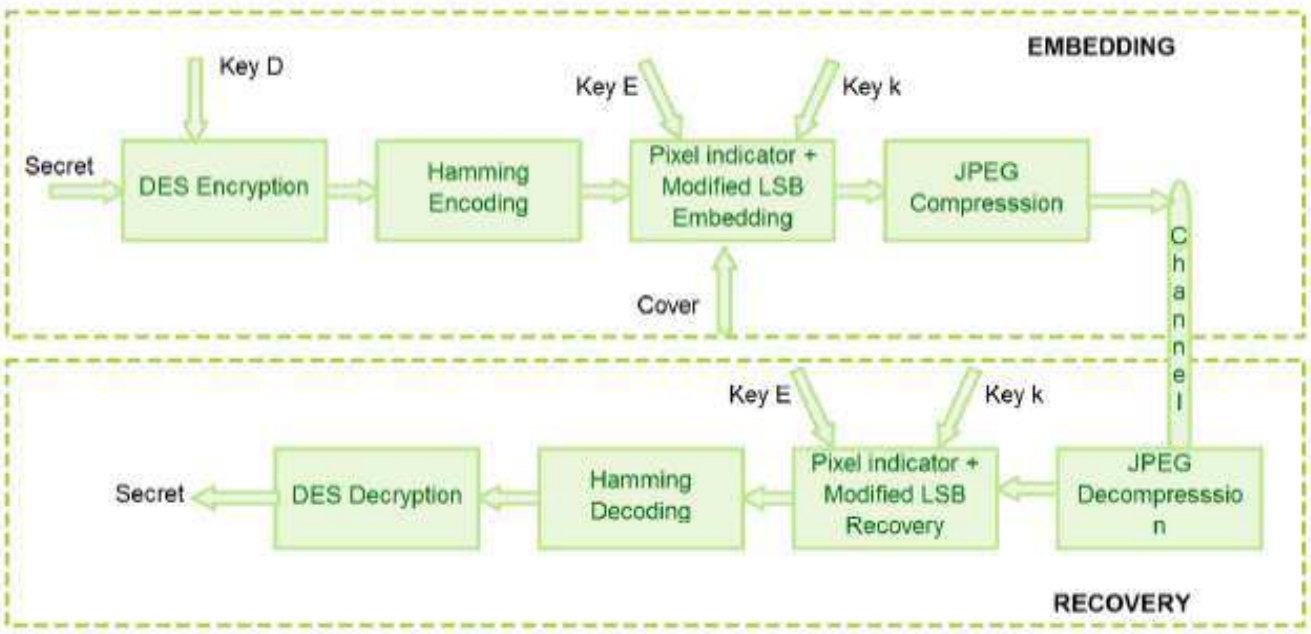

Figure 1. The proposed block diagram schematic for steganography embedding and extraction.

In the first method, RED is being used as the default indicator with the 5^{th} and 6^{th} bits deciding in which channel the data is to be embedded for that particular pixel and k indicating the number of hidden bits in corresponding green and blue channels with position as another option with varying E value given by the user.

For example if position is 1 then k value may be 1 to 4. Instead if the position is 2 then k can take value from 1 to 3, because if the value is more than 3 it will give visual degradation. This case further could be explained for E=3 with k may be 1 or 2 and so on.

In the second algorithm, it is observed the user has the added advantage of selecting the required indicator in every pixel. And in this methodology so as to improve the opaqueness of the program, the preferred channel in an image can be specified which is selected as the pixel indicator.

In fact to improve security and image quality a third method is introduced where the indicator of every pixel is chosen in a cyclic sequence i.e. a continuous change in the colour of the indicator making it a highly secure technique for data transmission as it becomes literally very nettlesome to decipher as the pattern in which data is embedded into the image for camouflage on selecting the particular channel as the pixel indicator.

The total number of bits of data that has to embedded in every pixel channel needs to be specified as per requirement. And after this procedure the value of E (E here is taken to be the position of the bit from where the embedding starts) to be needs to be assigned. The value of k can take a range varying from 1,2,3,4 wherein E specifies the starting bit from which data needs to be embedded in the pixel. Preferably the value of E should be 3 or higher, that satisfies the aim of the methodology because on compression of the image the last 2 bits even if lost, the concealed data remains intact.

However if the users believes that the channel is secure and data can be transmitted without much distortion and compression then E can assume the value of 1 or 2 wherein it becomes similar to the conventional methods and thus ,then the embedding capacity is not compromised. OPAP (Optimal Pixel Adjustment process) is applied where $(k+1)^{th}$ bit of every pixel is modified if the modified version seems to give better results and thus contributing to a decrease in the MSE value. Before performing the actual embedding, the secret data is pre-processed to improve security and data loss. This pre-processing is performed in two stages namely, DES Encryption and Hamming encoding. DES encryption is performed with the chosen symmetric key D. Hamming encoding is performed to prevent loss of data due to distortion. The actual embedding is characterized by three specific processes namely decomposition, selective injection and compression. In the decomposition stage, the cover image is split into three distinct colour planes. Then, the secret is selectively injected using modified LSB embedding, according to a rule determined by Table 1. The stego image is then compressed using JPEG to save bandwidth. The compressed stego image is then transmitted via a channel.

3. FLOWCHART

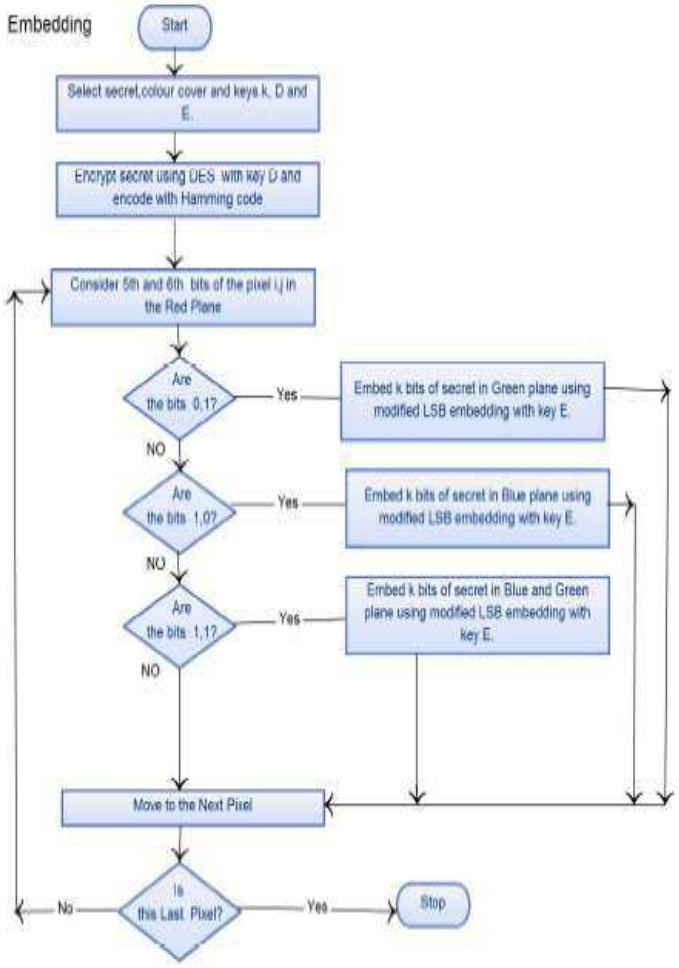

Figure 2: Flowchart for embedding

Table 1. Selective Injection Process

Bit-5 of R	Bit-6 of R	Embed in R	Embed in G	Embed in B
0	0	X	X	X
0	1	X	X	✓
1	0	X	✓	X
1	1	X	✓	✓

X no embedding and ✓ embedding

The recovery is characterized by three processes namely, decompression, decomposition and selective extraction. First, the stego image is decompressed using JPEG decompression. Then, it is split into three planes. Then, data is extracted using table-1. The extracted data is post-processed to get the actual secret.

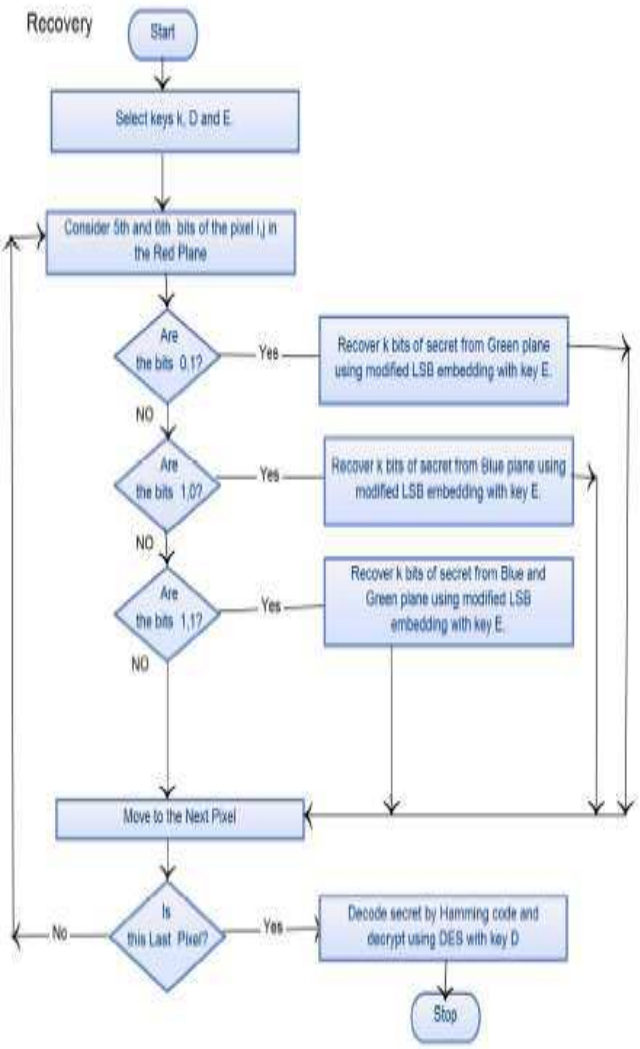

Figure 3: Flowchart for Extraction.

Post processing is done in two stages namely, decoding and decryption. Hamming decoding is performed to correct error, if any. Decryption is performed using DES with symmetric key D used during embedding. Thus, a secret data can be securely communicated by this method.

3.1 The Proposed Algorithm.

Method I: Embedding Algorithm:

Inputs:

Secret Data(S), Cover Image (I), Key D for DES
No of secret bits embedded per pixel k∈{1,2,3,4}
Key E for Modified LSB embedding E∈{1,2,3,4}

Output:

Stego image (O) with secret data embedded in it.

1. Convert the Secret Data (S) into binary format.
2. Encrypt S by using DES with key D.
3. Encode S using Hamming codes.
4. Split the cover image I into Red, Green and Blue Planes.(A, B and C respectively)
5. For each pixel in A, do the following:
   5.1. Let b[0]=5th bit of the current pixel in A
   5.2. Let b[1]=6th bit of the current pixel in A

5.3. If b=00 then
Go to next pixel.
Else if b=01 then
Embed k bits of secret data in current pixel of C using modified LSB embedding with key E.
Else if b=10 then
Embed k bits of secret data in current pixel of B using modified LSB embedding with key E.
Else
Embed k bits of secret data in current pixel of both A and B using modified LSB embedding with key E.
5.4. If S has no more data then
Go to step-6
6. Apply OPAP on the resulting stego image O.
7. Compress O using JPEG and transmit.

Recovery Algorithm:
Input :
Stego Image(O), Key D for DES
No of secret bits to be recovered per pixel k∈{1,2,3,4}
Key E for Modified LSB embedding E ∈ {1,2,3,4}
Output:
Secret Data (S)
1. Split the decompressed Stego image O into Red, Green and Blue Planes.(A, B and C respectively)
2. For each pixel in A, do the following:
2.1. Let b[0]=5th bit of the current pixel in A
2.2. Let b[1]=6th bit of the current pixel in A
2.3. If b=00 then
Recover nothing from current pixel.
Else if b=01 then
Recover k bits of secret data from current pixel of C using modified LSB recovery with key E.
Else if b=10 then
Recover k bits of secret data from current pixel of B using modified LSB recovery with key E.
Else
Recover k bits of secret data from current pixel of both A and B using modified LSB recovery with key E.
3. Store the recovered secret data as S.
4. Decode S using Hamming code.
5. Decrypt S using DES with key D.
6. Store S as secret data and process it.

Method II : Embedding Algorithm:
Inputs :
Secret Data(S), Cover Image (I), Indicator plane(In)
Key D for DES
No of secret bits embedded per pixel k∈{1,2,3,4}
Key E for Modified LSB embedding E∈{1,2,3,4}
Output:
Stego image (O) with secret data embedded in it.
1. Convert the Secret Data (S) into binary format.
2. Encrypt S by using DES with key D.
3. Encode S using Hamming codes.
4. Split the cover image I into Red, Green and Blue Planes. (A, B and C respectively)
5. If In=1 then,
P[1]=A, P[2]=B, P[3]=C
Else if I=2,
then P[1]=B, P[2]=A, P[3]=C
Else if I=3,
then P[1]=C, P[2]=A, P[3]=B
6. For each pixel in P[1], do the following:
6.1. Let b[0]=5th bit of the current pixel in P[1]
6.2. Let b[1]=6th bit of the current pixel in P[1]
6.3. If b=00 then
Go to next pixel.
Else if b=01 then
Embed k bits of secret data in current pixel of P[3] using modified LSB embedding with key E.
Else if b=10 then
Embed k bits of secret data in current pixel of P[2] using modified LSB embedding with key E.
Else
Embed k bits of secret data in current pixel of both A and B using modified LSB embedding with key E.
6.4. If S has no more data then
Go to step-6
7. Apply OPAP on the resulting stego image O.
8. Compress O using JPEG and transmit.

Recovery Algorithm:
Input :
Stego Image(O), Key D for DES, Indicator plane(In)
No of secret bits to be recovered per pixel k∈{1,2,3,4}
Key E for Modified LSB embedding E ∈ {1,2,3,4}
Output:
Secret Data (S)
1. Split the decompressed Stego image O into Red, Green and Blue Planes.(A, B and C respectively)
2. If In=1 then,
P[1]=A, P[2]=B, P[3]=C
Else if I=2,
then P[1]=B, P[2]=A, P[3]=C
Else if I=3,
then P[1]=C, P[2]=A, P[3]=B
3. For each pixel in P[1], do the following:
3.1. Let b[0]=5th bit of the current pixel in P[1]
3.2. Let b[1]=6th bit of the current pixel in P[1]
3.3. If b=00 then
Recover nothing from current pixel.
Else if b=01 then
Recover k bits of secret data from current pixel of p[3] using modified LSB recovery with key E.
Else if b=10 then
Recover k bits of secret data from current pixel of p[2] using modified LSB recovery with key E.
Else
Recover k bits of secret data from current pixel of both A and B using modified LSB recovery with key E.
4. Store the recovered secret data as S.
5. Decode S using Hamming code.
6. Decrypt S using DES with key D.
7. Store S as secret data and process it.

Method 3:
Embedding Algorithm:
Inputs :
Secret Data(S), Cover Image (I), Key D for DES
No of secret bits embedded per pixel k∈{1,2,3,4}
Key W for Modified LSB embedding W∈{1,2,3,4}
Output:
Stego image (O) with secret data embedded in it.

1. Convert the Secret Data (S) into binary format.
2. Encrypt S by using DES with key D.
3. Encode S using Hamming codes.
4. Split the cover image I into Red, Green and Blue Planes.(A, B and C respectively)
5. Let index i=1.
6. For each pixel in P[1], do the following:
   6.1. If (i mod 3) =1 then,
        In[i]=1
      Else if (i mod 3)=2 then,
        In[i]=2
      Else In[i]=3
6.2. Set i=i+1
7. Let index j=0
8. For each pixel in P[1], do the following:
   8.1. If In[j]=1 then,
        P[1]=A[i], P[2]=B[i], P[3]=C[i]
      Else if I[j]=2, then
        P[1]=B[i], P[2]=A[i], P[3]=C[i]
      Else if I[j]=3, then
        P[1]=C[i], P[2]=A[i], P[3]=B[i]
9. For each pixel in P[1], do the following:
   9.1. Let b[0]=5th bit of the current pixel in P[1]
   9.2. Let b[1]=6th bit of the current pixel in P[1]
   9.3. If b=00 then
        Go to next pixel.
      Else if b=01 then
        Embed k bits of secret data in current pixel of P[3] using modified LSB embedding with key E.
      Else if b=10 then
        Embed k bits of secret data in current pixel of P[2] using modified LSB embedding with key E.
      Else
        Embed k bits of secret data in current pixel of both P[2] and P[3] using modified LSB embedding with key E.
   9.4. If S has no more data then
        Go to step-10
      Else j=j+1
10. Apply OPAP on the resulting stego image O.
11. Compress O using JPEG and transmit.

Recovery Algorithm:

Input :

Stego Image(O)
Key D for DES
No of secret bits to be recovered per pixel k∈{1,2,3,4}
Key E for Modified LSB embedding E ∈ {1,2,3,4}

Output:

Secret Data (S)

1. Split the decompressed Stego image O into Red, Green and Blue Planes. (A, B and C respectively)
5. Let index i=1.
6. For each pixel in P[1], do the following:
   6.1. If (i mod 3) =1 then, In[i]=1
      Else if (i mod 3)=2 then, In[i]=2
      Else In[i]=3
6.2. Set i=i+1
7. Let index j=0
8. For each pixel in P[1], do the following:
   8.1. If In[j]=1 then,
        P[1]=A[i], P[2]=B[i], P[3]=C[i]
      Else if I[j]=2, then
        P[1]=B[i], P[2]=A[i], P[3]=C[i]
      Else if I[j]=3, then
        P[1]=C[i], P[2]=A[i], P[3]=B[i]
9. For each pixel in P[1], do the following:
   9.1. Let b[0]=5th bit of the current pixel in P[1]
   9.2. Let b[1]=6th bit of the current pixel in P[1]
   9.3. If b=00 then
        Recover nothing from current pixel.
      Else if b=01 then
        Recover k bits of secret data from current pixel of P[3] using modified LSB recovery with key E.
      Else if b=10 then
        Recover k bits of secret data from current pixel of P[2] using modified LSB recovery with key E.
      Else
        Recover k bits of secret data from current pixel of both P[2] and P[3] using modified LSB recovery with key W.
10. Store the recovered secret data as S.
11. Decode S using Hamming code.
12. Decrypt S using DES with key D.
13. Store S as secret data and process it.

4. RESULT AND DISCUSSION

In this present implementation Lena, baboon, Gandhi and Temple of 256 × 256 color digital images has been taken as cover images as shown in Figure 4 a, b, c & d and tested for full embedding capacity and the results are given in Figure 5. The effectiveness of the stego process proposed has been studied by calculating MSE and PSNR for all the four digital images in RGB planes using the proposed methods I, II and III

The MSE is calculated by using the equation,

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(X_{i,j} - Y_{i,j})^2$$

where $X_{i,j}$ is Stego value and $Y_{i,j}$ is the cover object.

The PSNR is calculated using the equation

$$PSNR = 10\log_{10}\left(\frac{I_{\max}^2}{MSE}\right)dB$$

where I_{max} is the intensity value of each pixel which is equal to 255 for 8 bit gray scale images. Higher the value of PSNR better the image quality

Figure 4 a. Lena b. Baboon c. Gandhi d. Temple

The corresponding ouput for k=1,2,3,4 with E = 4, 3,2,1 method III is given in Fig 5 as follows.

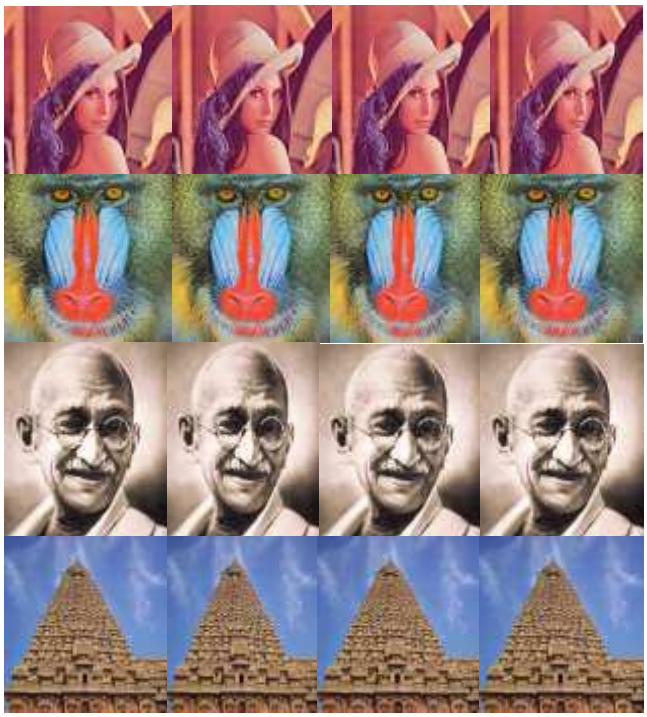

Figure 5 ouput for k=1,2,3,4 with E = 4, 3,2,1 method III.

As a sample histogram RED, GREEN and BLUE E=4 and k=1 for temple, Method I, II and III are given in Figure 6.

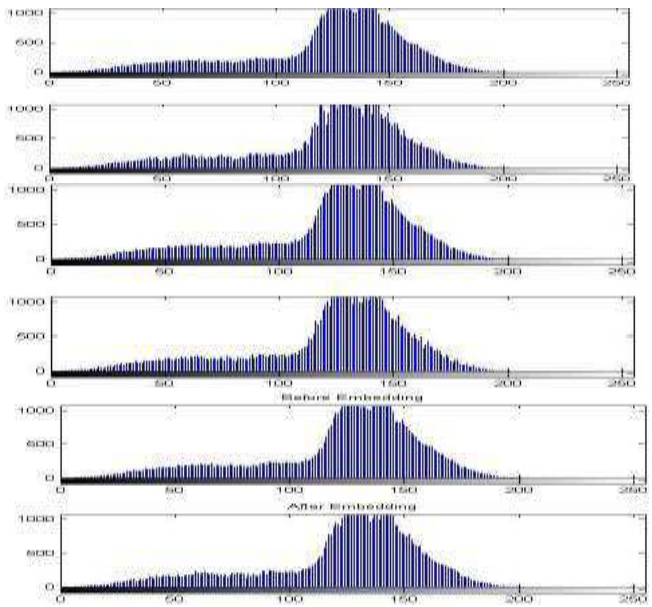

Figure 6 Histograms RED, GREEN and BLUE E=4 and k=1 for temple, Method I, II and III

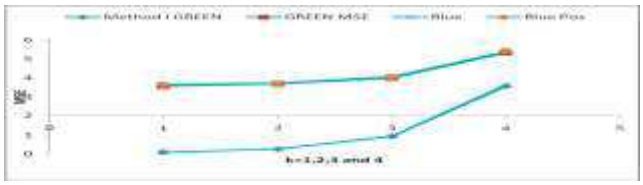

Fig 7 a. The performance MSE of G,B channel for the method I, RED as Indicator

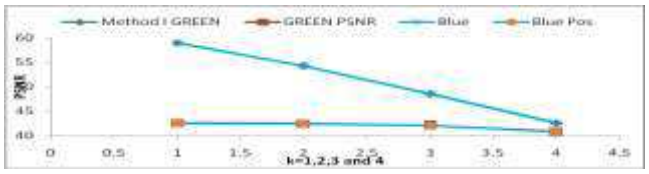

Fig 7 b. The performance PSNR of G,B channel for the method I, RED as Indicator

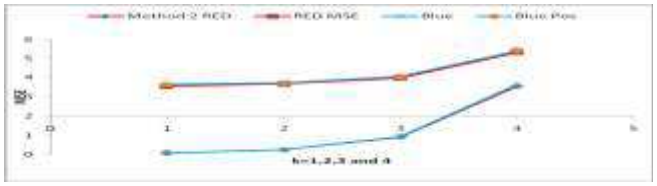

Fig 7 c. The performance MSE of R,B channel for the method II , Green as indicator

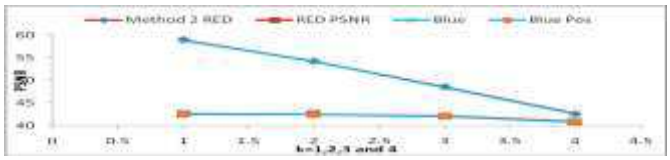

Fig 7 d. The performance PSNR of R,G,B channel for the method II, Green as indicator

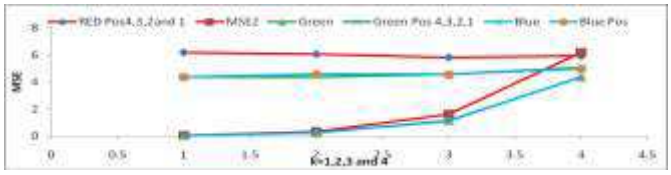

Fig 7 e. The performance MSE of R,G,B channel for the method III

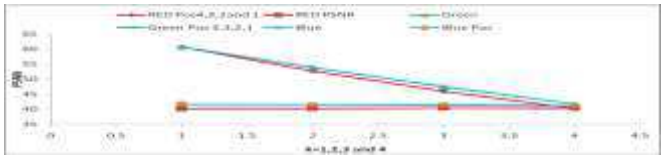

Fig 7 f. The performance PSNR of R,G,B channel for the method III

5. CONCLUSION

Thus it is observed that there are a plethora of techniques in which information can be hidden. And the basic algorithms and techniques followed are just for the primary denouement of obtaining Secrecy, Imperceptibility, Robustness and Randomization. While the so far described and well noted procedures encompass most of the above described attributes, robustness is normally compromised in the most algorithms as the loss of the last bits on compression. The proposed method covers the entire gamut of requirements and especially emphasizes on the flexibility and pliability of the image to prevent any loss of concealed data even on compression. Thus it is a very novel and adaptive method that encapsulates high security with imperceptibility and lithe. Also, the amount of the suppleness of the program is customer driven depending upon the value of the

position of the start of the embedding of information chosen. And better results are obtained so and when clever values of E are employed (above 2) , and thus when this is executed , it is observed that this conceived idea works better and has an edge over most other conventional methods. The combined implementation of cryptography and Steganography enhances the security level to another fold along with increased robustness.

6. ACKNOWLEDGEMENT

The authors wish to thank Sandeep Kumar Behera, Benita Bose, G.Vivek, and Sasidhar Imabathuni ECE Stego group Student /SEEE/ SASTRA University for their technical support.

7. REFERENCES

[1] Abbas Cheddad, Joan Condell, Kevin Curran, Paul Mc Kevitt, Digital image steganography: Survey and analysis of current methods Signal Processing 90 (2010) 727–752.

[2] Adnan Gutub, Mahmoud Ankeer, Muhammad Abu-Ghalioun, Abdulrahman Shaheen, and Aleem Alvi, "Pixel Indicator high capacity Technique for RGB image Based Steganography", WoSPA 2008 – 5th IEEE International Workshop on Signal Processing and its Applications, University of Sharjah, Sharjah, U.A.E. 18 – 20 March 2008

[3] R.Amirtharajan, Adharsh.D, Vignesh.V and R.John Bosco Balaguru, "PVD Blend with Pixel Indicator - OPAP Composite for High Fidelity Steganography" IJCA October edition 2010.

[4] R.Amirtharajan, Sandeep Kumar Behera, Motamarri Abhilash Swarup, Mohamed Ashfaaq K and John Bosco Balaguru Rayappan, " Colour Guided Colour Image Steganography" *Universal Journal of Computer Science and Engineering Technology 1 (1) (2010), 16-23.*

[5] R.Amirtharajan and Dr. R. John Bosco Balaguru, "*Tri-Layer Stego for Enhanced Security – A Keyless Random Approach*" - IEEE Xplore, DOI, 10.1109/IMSAA.2009.5439438.

[6] R.Amirtharajan, R. Akila, P.Deepikachowdavarapu, "A Comparative Analysis of Image Steganography". International Journal of Computer Applications 2(3)(2010):41–47.

[7] R.Amirtharajan, Krishnendra Nathella and J Harish, "Info Hide – A Cluster Cover Approach" International Journal of Computer Applications 3(5)(2010) 11–18.

[8] Bruice Schneier, Applied Cryptography Protocols, Algorithm and Source Code in C. Second edition. Wiley India edition 2007

[9] W. Bender, D. Gruhl, N. Morimoto, A. Lu, "Techniques for data hiding" IBM Syst. J. 35 (3&4) (1996) 313–336.

[10] C.K. Chan, L.M. Chen, Hiding data in images by simple LSB substitution, Pattern Recognition 37 (3) (2004) 469–474.

[11] Chang, C.C., Hsiao, J.Y., Chan, C.S., 2003. Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy. Pattern Recognition 36 (July), 1583–1595.

[12] Chang, C.C., Tseng, H.W., 2004. A steganographic method for digital images using side match. Pattern Recognition Letter 25 (September), 1431–1437.

[13] F.A.P. Petitcolas, R.J. Anderson, M.G. Kuhn, Information hiding—a survey, Proc. IEEE 87 (7) (1999) 1062–1078.

[14] S. Katzenbeisser, F.A.P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking, Artech House, Norwood, MA, 2000.

[15] Peter Wayner, "Disappearing cryptography: information hiding : steganography & watermarking" 2nd. ed. San Francisco: Morgan Kaufmann; 2002.

[16] N. Provos and P. Honeyman, "Hide and seek: An introduction to steganography," *IEEE Security Privacy Mag.*,1 (3) (2003) 32–44

[17] C.C. Thien, J.C. Lin, A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function, Pattern Recognition 36 (11) (2003) 2875–2881

[18] Po-Yueh Chen Hung-Ju Lin, "A DWT Based Approach for Image Steganography**",** International Journal of Applied Science and Engineering 4(3)(2006): 275-290

[19] R.Z. Wang, C.F. Lin, J.C. Lin, Image hiding by optimal LSB substitution and genetic algorithm, Pattern Recognition 34 (3) (2000) 671–683