IJCTA, 9(10), 2016, pp. 4843-4857

Secure Binary Image Steganography Using F5 algorithm Based on Data Hiding and Diffusion techniques

***M.S. Antony Vigil **Sameer Singh Rathor ***Jitendra Singh**

Abstract : Steganography is the science of passing the information in the manner that the very existence of the message is unknown. The purpose of steganography is to avoid suspicion to the transmission of hidden message, such that it is highly secured. This paper represents a binary image steganographic scheme that aims to provide the Image quality and security as well as improving the capacity of data.

Statistical analysis : Steganography can be viewed as a kin to cryptography. Both have been used to provide the better security of information. Hiding a message using steganography technique minimizes the number of chances of a message being detected. The most commonly method used for steganography technique is a Least Significant Bit (LSB) algorithm. LSB algorithm performs the embedding operation of message along with the image file where each pixel has a size of 3 bytes. Each and every bit of the message is taken and this message bit is embedded along with the bytes of the image file such that, it doesn't make any perceivable change in the message embedded file.

Findings : That is the ease of cracking the message which is hidden in the image file due to the simplicity of the algorithm that provide the simple platform for the easy cracking and detectability of the hiding data. Thus the Security Extensibility algorithms for steganography used as a F5 algorithm and cryptography used as a AES algorithm which is used to provide the more security such that the cracking is made quite tough which is almost improbable. Steganography F5 algorithm is more secure than other existing algorithm such as LSB algorithm, RSA algorithm.

Conclusion : F5 algorithm performs the matrix encoding to improve the efficiency of embedding and extraction operation. Thus it minimizes the number of necessary changes. F5 algorithm employs per mutative straddling to scatter the message over the whole cover medium to reduce the chances of detectability and improving the security of data. This paper also introduces the quality of the image using various data hiding and diffusion techniques.

Keyword : Steganography, cryptography, AES Algorithm, F5 Algorithm, Data hiding and Diffusion technique.

1. INTRODUCTION

The Cryptography is used for protecting information from illegal activity by making message illegible. But In cryptography technique the data which is not hidden. Steganography perform the various technique of hiding of data in media (video, image, audio) in order to keep secret (hidden) the existence of the information. Steganography, analogous to the data hiding, aims to hide secret information inside the digital media such as image, audio and video

* Department of Computer Science and Engineering, Assistant Professor, SRM University Ramapuram Chennai-600089, Tamil Nadu, India, E-mail: antonyvigil@gmail.com

** Department of Computer Science and Engineering, Student, SRM University Ramapuram Chennai-600089, Tamil Nadu, India, E-mail: ssr.sameersingh@gmail.com

*** Department of Computer Science and Engineering, Student, SRM University Ramapuram Chennai-600089, Tamil Nadu, India E-mail: jitu95.singh@gmail.com

in such a way that no one can detect the existence of the information except of sender and receiver. Now a days various data hiding methods have been developed for binary images, which can be used as a digitally stored handwritings, CAD graphs, signatures, and so on. Stego images obtained by various schemes such as Data hiding and diffusion technique have also been reported to achieve extensive visual qualities. However, steganography methods ignore the security.

The most undetectability of the secret messages can diminish the suspicion from attackers and thus it effect on the enhance of the security of data .Steganography hide the existing data which is to be observed. The image which is used to store the information are called the stego image and cover image [1]. Images are digital media which is widely used by peoples and exchanged through the internet. Images are the best cover media to hide the secret information. The secret information bits are inserted in the area of the cover file of the image which is not observed by an human eyes. Steganography communication system consists of an algorithm for embedding and an extraction of the data. To provide the secret message, the original image is slightly changed or modified by the embedding algorithm. With the help of which, the stego image is obtained[2].Steganography application can be used for both illegal and legal purpose. Whereas ,Civilians may use it for protecting or maintaining privacy while terrorists may use it for spreading dangerous secret messages [3]. It is new technique for establishing a secure communication [4]. To this end, we focus on designing a secure binary image data hiding technology by improving the undetectability while maintaining the stego image quality and embedding capacity. The embedding process creates a stego images through the replacement of these redundant bits with data which is contained from the hidden messages.

In the spatial domain [5], message bits are commonly embedded by directly flipping pixel values as a bit of information in a binary image. But In grayscale images, pixels value of a binary images possess only two states: black represent[6] as 1 binary bit and white represent as a 0 bit. Such as flipping distortions on binary images are easily detected even by human visual systems. To deal with this problem, advanced steganographic technique [2]suggest constraining the embedding to the portions of images that are difficult to be detected[7]. Data Hiding methods traced the boundary to find most suitable pixels for embedding the message bits, whereas the other diffusion methods divided the cover image into overlapped and non-overlapped blocks and find the best flipping location for hiding the data in each block of the bytes of images [8]. Matrix embedding is usually employed to achieve a high embedding efficiency and they proposed a practical near optimal matrix embedding which is called as a syndrome-trellis code (STC) [9], to embed the capacity of distortion bound with respect to the specified flipping distortion measurement. Consequently, we employ this STC code to implement our steganographic scheme.

This Existing method perform the losslessly vacating room from the encrypted images which is relatively difficult and inefficient manner LSB techniques[10] can only achieve small payloads which generate the marked image with poor quality for large payload and all of them are focused to some low error rates on data extraction or image restoration. But in that method simple algorithm is used for the transfer of data. From an opponent's perspective, steganalysis is the art of detecting the stego images[11]. The first step is needed to determine whether a message is hidden within the image or not. Then find the type of Steganography algorithm which is used for developing the secret data in a image [12]. Then LSB method estimate the length of the message and then they try to estimate the message which is hidden behind the stego image. The most well known data hiding technique in a images[13] are least significant bit (LSB) substitution masking and filtering techniques. LSB is a simple method to embedding the information inside the image. But Image manipulation can diminish the hidden information which is inside in this image [16]. Using LSB technique each byte of a 24-bit image, and three bits of data can be encoded into each pixel of an image, such as each pixel[17] is represented by three bytes. Applying LSB technique to perform the each byte of an 8-bit image contained only one bit of data can be encoded into each pixel of the block images, whereas each pixel is represented by one byte.

The proposed method perform to achieve real reversibility, data extraction and image recovery[18] which is having no error in data .The proposed method are using RSA algorithm and F5 algorithm to provide better security of data. Combination of Cryptography and Steganography algorithm provides most efficient[19] algorithm for

embedding and extraction of data using various hiding and diffusion techniques. Based on data Hiding and Diffusion technique we also focus on the quality of stego Images[20] which is obtained by receiver without any obstrusion.Based on F5 algorithm we also improve the capacity and efficiency of the data embedding process using matrix encoding method.

1.1. Least Significant Bit Substitution Technique (LSB)

In this technique, the LSBs of pixel values of the cover image are modified according to bits of the message. The simplest steganography techniques is LSB replacement for all pixels of image. Since only LSB is changed, the difference between the cover or original image and the stego-image is hardly noticeable.

This is the simplest of the steganography methods based on use of Least Significant Bit, and therefore the most vulnerable. This embedding process consists of sequential substitution of the each Least Significant Bit (LSB-1) of the image pixel for the bit messages. For the simplicity, this method can camouflage a huge volume of information[4].

The following steps demonstrate how this method is used to hide the required secret data "A" in the cover image.

Step 1 : Convert the data from decimal to binary.

$$[\text{Message}] \xrightarrow{\text{Dec2Bin}} [1000001]$$

Step 2 : Read Cover Image.

Step 3 : Convert the Cover Original Image from decimal to binary bit.

Step 4 : Break the byte to be hidden into bits.

$$\text{Thus } [10100001] \xrightarrow{\text{is divided into 8 bits}} [10111011].$$

Step 5 : Take first 8 byte of the original data from Cover Image.

Step 6 : Replace the LSB *i.e*. least significant bit by one bit of the data to be hidden.

Advantages : The picture quality of cover image is merely affected. Hiding capacity is good. Very simple in implementation

Disadvantages: Robustness is slightly less : the hidden data is subjected to the alternation due image manipulation. Detection of the secret data is not a tough job because of easy algorithm. More information storage requires large image size thus requires high transmission rate[6] due to large size of the stego image.

1.2. RSA Algorithm

The RSA cryptography system is the most widespread used as public key cryptography algorithm in world. It can be used to encrypt the message without the need to exchange the secret key separately. The RSA algorithm can also be used for both public key encryption as well as for digital signatures[8]. The security is based on difficulty of factoring the large integers. Party X can send an encrypted message to party Y without any prior exchange of secret keys. X just uses Y's public key to encrypt the message and Y decrypts it using private key, which only he knows. RSA can also be used to sign a message, so X can sign a message using their private key[9] and Y can verify it using A's public key.

Key Generation Algorithm

(*a*) Generate two random primes, a and b, of approximately same size such that their product $c = ab$ is of the required bit length, *e.g*. 1024 bits.

(*b*) Compute $c = ab$ and (phi) $\varphi = (a\text{-}1)(b\text{-}1)$.

(*c*) Choose an integer g, $1 < g <$ phi, such that gcd(g, phi) = 1.

(*d*) Compute the secret exponent f, $1 < f <$ phi, such that $gf \equiv 1 \pmod{\text{phi}}$.

(*e*) The public key is (c, g) and the private key (f, a, b). Keep all the values f, a, b and phi secret. [We prefer sometimes to write the private key as (c, f) because you need the value of c when using f. Other times we might write the key pair as ((c, g), f).]

Where,

c is known as the modulus.

g is known as the public exponent or encryption exponent or just the exponent.

f is known as the private exponent or decryption exponent or extraction.

Encryption

Sender X does the following :

(a) Obtains the recipient Y's public key (m, n).

(b) Representation of the plaintext message as a positive integer w, $1 < w < c$.

(c) Computes the ciphertext $j = \text{wg} \bmod c$.

(d) Sends the ciphertext j to Y.

Decryption

Recipient Y does the following :

(a) Uses his private key (c, f) to compute $w = jf \bmod c$.

(b) Extracts the plaintext from the message representative w.

Signature verification

Recipient Y does the following :

(a) Uses sender X's public key (c, g) to compute integer $u = rt \bmod c$.

(b) Extracts the message coefficient from this integer.

(c) Independently computes the message coefficient of the information of data has been signed.

(d) Whenever both message coefficient are identical, the signature is valid.

2. SYSTEM MODELING

The proposed method provides better security when transferring the data or messages from sender to receiver. The main concept of this project is to hide the messages or a secret data into an image which further act as a carrier of a secret data and communicate to the receiver securely without any modification[2].Combination of Steganography and Cryptography is used to enhance the security and embedding capacity. In Cryptography we are using AES algorithm to encrypt a message, the message is hidden in DCT of an image and two secret keys are generated which make this system highly secured. F5 algorithm is used for embedding[4] and de- embedding process and provides better embedding capacity.

2.1. Module Description

2.1.1. Authentication Module

The Authentication module is used to enter into the application. The users will enter the valid information such as user id and password. If the user id and the password will be matched then it will take the user to the organization. This form contains the fields[6] such as user id and password. The user clicks the sign in button to login the application. If the user id and password doesn't match, then the login form will display a message as invalid user. The advantage of this form is that, if user forgets the password then he can find it.

2.1.2. Embed module

In Embed module, the first step is a selecting the input image file. The selection is made through opening a new dialog boxes and selected path has displayed through a textbox. The second step is selecting an output image file in which text data or a text file[8],[9] is embedded. The third step is selecting a text file or typing any text message for embedding process. Fourth step is a selecting a file of keys. In the fifth step the files that we have selected which are viewed and verification of the path of the embed module is done. In the sixth step the data is embedded into the image file using low bit encoding technique. After embedding the secret content into the image, both the image files are previewed and no difference is observed between both the images.

2.1.3. Extract module

In extract module, the first step is the process of selecting a file of encrypted image . This is a file from which a user has to extract secret information from the stego image. The second step is the process which is involved in selecting a new image file to display the embedded message. The encryption method used here is symmetric, so the key selected during the embedding process is used in decrypting[10] the message. All the process done till this step are displayed using a list of the box and finally embedded messages can be viewed with the help of a file .

2.1.4. Transformations

A more complex module is the way of hiding a secret data inside an image which comes under the uses and modifications of DCT (discrete cosine transformations). Discrete cosine transformations[11] (DCT) has been used by the JPEG compression F5 algorithm to transforms the successive 8 × 8 pixel blocks of an image, into 64 bit DCT coefficients values in each. Each DCT coefficient F(*u*, *v*) of an 8 × 8 blocks of image pixel in a matrix *f*(*a*, *b*) is given by:

Step 1 : F(*u*, *v*) = 16 C(*u*) C(*v*) "8 Xu = 08 Xy = 0 *f*(*a*, *b*) –cos (2*x* + 1)*u* 16 cos (2*y* + 1)*v*_16
where C(*x*) = 1/p2 when x equals 0 and C(*x*) = 1 otherwise.

Step 2 : After calculating the DCT coefficients, the following quantizing operation is performed:
FQ(*u*, *v*) = F(*u*, *v*)Q (*u*, *v*) where Q(*u*, *v*) is a 64 element quantization table.

Step 3: A simple pseudo code algorithm is used to hide the message inside a JPEG image could look like this:

(*a*) Input: message and the cover image
(*b*) Output: steganographic image containing the message
(*c*) while data left to embed do
(*d*) get next DCT coefficient from cover image
(*e*) if DCT 6= 0 and DCT 6= 1 then
(*f*) get next LSB from message
(*g*) replace DCT LSB with message bit
(*h*) end if
(*i*) insert DCT into steganographic image
(*j*) end while.

2.2. System architecture

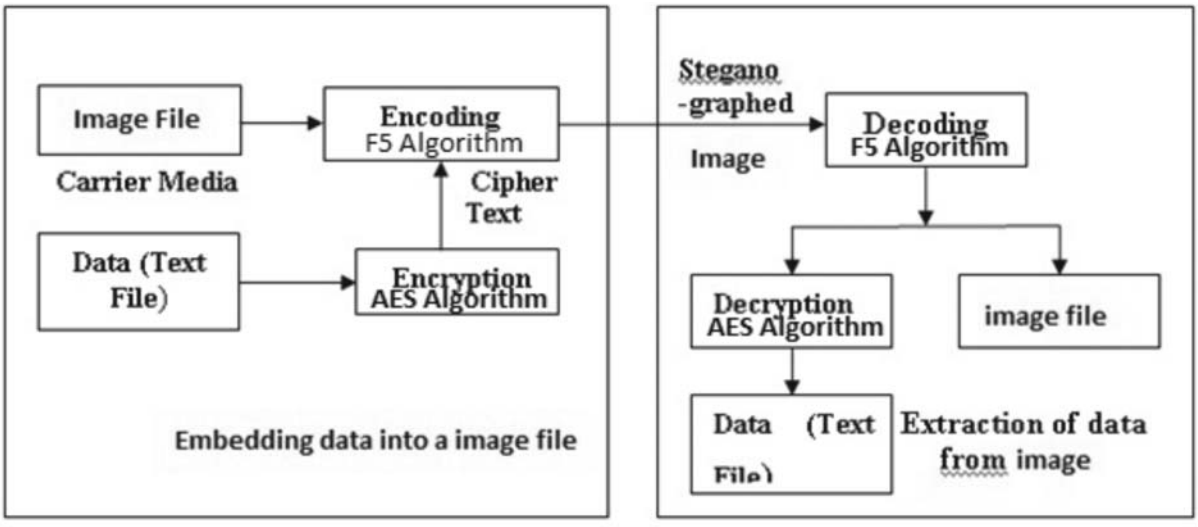

Fig. 1. System Architecture

In System Architecture diagram it consist two modules such as sender module and reciever module.In the sender module they are performing embedding process.In reciever module it consist of extraction process, it is also called as de-embedding procedure.Figure 2 represent the two algorithm used as F5 algorithm and AES algorithm in both embedding and extraction procedures.Finally we contain the original data which is contained from the stego image in Reciever module.

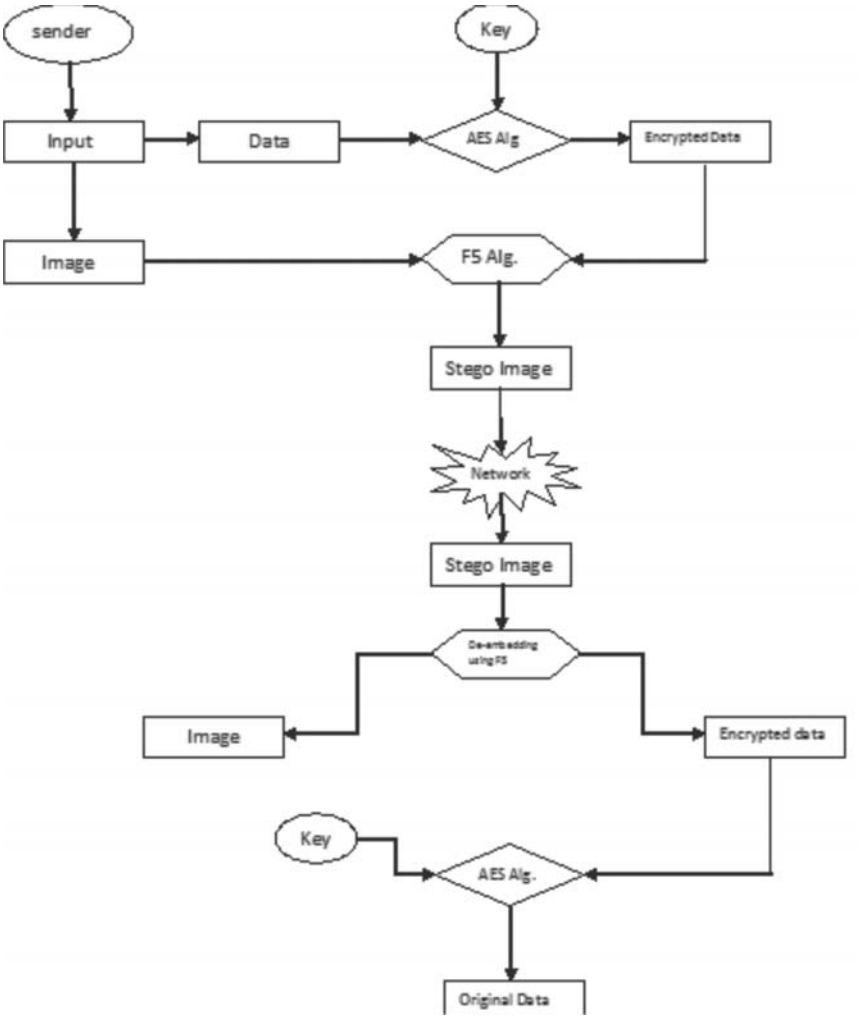

Fig. 2. Work Flow Diagram

In Workflow diagram it represent the working of system under the development of application and which component is necessary to perform the working flow of the software in embed and extraction modules.It start from the sender with necessary component and various algorithm is used to implement the advanced steganographic techniques.Using the secure network the user will pass the information with the reciever and hides the required information in images in a secure mannner.

2.3. Proposed Algorithms

2.3.1. F5 Algorithms

The F5 algorithm is used to develop a embedding method for JPEG images which provide high steganographic capacity without compromising the security. With the guidance of $\lambda 2$ attack,the paradigm of replacing bits of information is challenged in the cover image with secret message while proposing a slightly different paradigm of incrementing the image components to embed message bits.The F5 algorithm embeds the message bits into the randomly chosen DCT coefficients and uses matrix embedding to embed the certain length message. According to the F5 algorithm, the program accepts five inputs:

(*a*) Quality factor of the stego-image K;
(*b*) Input file (BMP, JPEG, or GIF).
(*c*) Name of output file.
(*d*) File containing the secret message.
(*e*) User password to be used as a seed for PRNG.
(*f*) Comment to be inserted in the header.

In embedding process, the length of message and the number of non-zero as well as non-DC coefficients[8],[9] are used to determine the most appropriate matrix embedding that minimizes the number of modifications in the cover-image. Matrix embedding has three parameters (*k*, *c*, *n*), where *k* is the number of changes per group of *c* coefficients, and *n* is the number of embedded bits. In other paper [16], the authors describe a simple matrix embedding (1, 2n–1, n) using a "hash" function that outputs n bits when applied to 2n–1 coefficients.

Fig. 3. F5 Encoding process.

The process of embedding starts with deriving a seed for a PRNG from the password of the user and by generating the random value using the DCT co-efficient bit. The PRNG is used to encrypt the value 'c' using a stream cipher[12],[14] and also embeds it in a particular manner together with the message length. The body of the message is embedded using the matrix embedding process, inserting n message bits into one group of 2n–1 coefficients by decreasing the absolute value of at most 1 coefficient from each group by 1 (one).

The complete embedding process consists of the following 6 steps :

(*a*) Get the RGB representation of the input image.

(*b*) Calculate quantization table corresponding to quality factor F and while storing the quantized DCT coefficients compress the image.

(*c*) Compute the approximate capacity with no matrix embedding X = *v*DCT– *v*DCT/64 –*v*(0) – *v*(1) + 0.49 *v*(1), where *v*DCT is the number of all DCT coefficients, *v*(0) is the number of DCT coefficients equal to zero, *v*(1) is the number of DCT coefficients with absolute value 1, *v*DCT/64 is the number of DCT coefficients, and –*v*(1) + 0.49*v*(1) = –0.51 *v*(1) is the loss estimation due to shrinkage. The parameter X and the message length together determine the matrix embedding.

(*d*) The user password is used to create a new seed for the PRNG that determines the random walk for embedding message bits. The PRNG is also used to create a pseudo random bit stream that is XORed with the message to make it a randomized bit stream.

(*e*) The message is divided into segments of *n* bits that are embedded into the group of 2*n*–1 coefficients along the random walk. If the hash value of the group does not match the message bits, the absolute value of 1 of the coefficients in the group is decreased. If the coefficient becomes 0(zero), the event is called *shrinkage*, and the same *n* message bits are re-embedded in the next group of discrete cosine transform coefficients (we note that LSB(*d*) = *d* mod 2, for *d* > 0, and LSB (*d*) = 1– *d* mod 2, for *d* < 0).

(*f*) If the message size fits into the approximate capacity, the embedding proceed further, otherwise the maximum possible length error message is shown.

There are very rare cases when the capacity estimation is wrong due to bigger than anticipated shrinkage. In those cases, the program embeds data as much as possible and displays the warning. While the F5 algorithm does modifies the histogram of discrete cosine transform coefficients, the authors show that some important characteristics of the histogram are preserved, such as monotonicity.

The F5 algorithm will not be detected using the chi-square attack because the embedding of data is not based on the bit replacement or swap any fixed pairs of values. In the upcoming section, we discuss an attack on F5. It is based on idea [15],[16] that one can accurately determine the histogram of the cover image from the stego image. Because F5 modifies the histogram in well defined manner, we can calculate the number of all the modified coefficients by comparing all the estimated histogram with the histogram of stego image.

Steganography is an art of embedding secret messages in a carrier medium such that other than the sender and intended recipient knows about the hidden message. Mediums can include images, video, and audio.F5 will scatter the entire message throughout the carrier.

F5 Implementation Steps:

(*a*) Start JPEG compression. Stop immediately after the quantization of coefficients.
(*b*) Initialize a cryptographically stable and strong random number generator with the derived key from the password.
(*c*) Initiate a permutation (two parameters: random number generator and number of coefficient).
(*d*) Embed the secret message with matrix encoding (Hash based embedding).
(*e*) Implement the inverse permutation.
(*f*) Continue JPEG compression (Huffman coding).

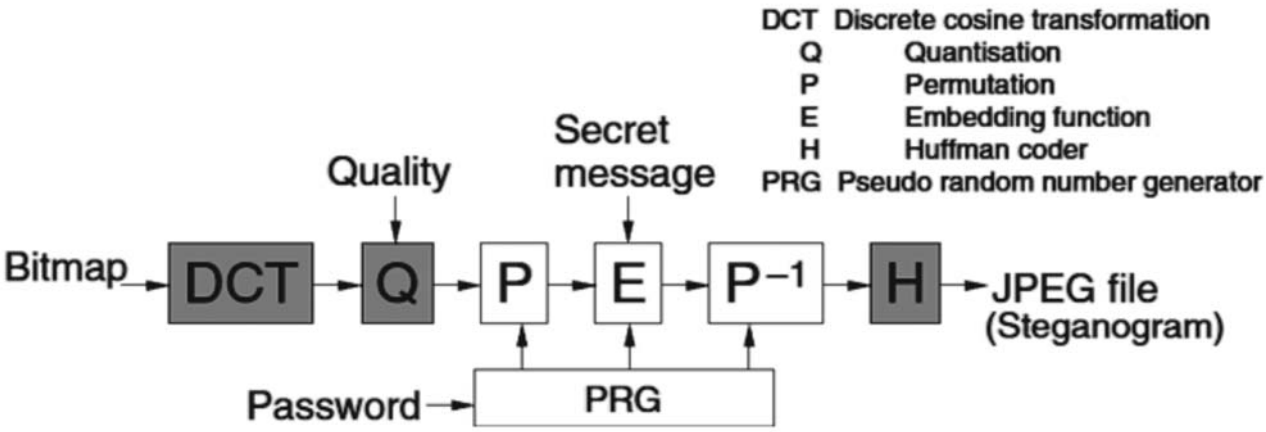

Fig. 4. F5 implementation process.

Matrix Encoding

Embedding efficiency is improved from 1.5 bit to 3.8 bit per change.

Consider we want to embed x_1 and x_2 in LSB locations a_1, a_2, and a_3.

$$x_1 = a_1 \oplus a_3, x_2 = a_2 \oplus a_3 \Rightarrow \text{change nothing}$$
$$x_1 \neq a_1 \oplus a_3, x_2 = a_2 \oplus a_3 \Rightarrow \text{change } a_1$$
$$x_1 = a_1 \oplus a_3, x_2 \neq a_2 \oplus a_3 \Rightarrow \text{change } a_2$$
$$x_1 \neq a_1 \oplus a_3, x_2 \neq a_2 \oplus a_3 \Rightarrow \text{change } a_3$$

Permutation is generated using user-defined password.

Discrete Cosine Transform

One-dimensional DCT $$C_{ij} = a_1 \cos\frac{i(2j+1)\pi}{2n} C_{ij} \cos\frac{i(2j+1)\pi}{2n}$$

Let *n* be a positive integer. The one dimensional Discrete Cosine Transformation of order *n* is defined by an $n \times n$ matrix

The Advantage of Orthogonality

C is orthogonal if : $$C^T C = I$$

Implies $$C^{-1} = C^T$$

To provide the easiest way of solving the matrix

Solve $$Y = CXC^T \text{ for X:}$$

$$C^T Y = C^T C \, X \, C^Y = X \, C^T$$

$$C^T Y C = X C^T C = X$$

One-dimensional DCT

The discrete cosine transform, C, has one most basic characteristic that it is an real orthogonal matrix.

$$C = \sqrt{\frac{2}{n}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}} \\ \cos\frac{\pi}{2n} & \cos\frac{3\pi}{2n} & \cdots & \cos\frac{(2n-1)\pi}{2n} \\ \vdots & \vdots & \cdots & \vdots \\ \cos\frac{(n-1)\pi}{2n} & \cos\frac{(n-1)2\pi}{2n} & \cdots & \cos\frac{(n-1)\pi}{2n} \end{bmatrix}$$

$$C = C^T = \sqrt{\frac{2}{n}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{\pi}{2n} & \cdots & \cos\frac{(n-1)\pi}{2n} \\ \frac{1}{\sqrt{2}} & \cos\frac{3\pi}{2n} & \cdots & \cos\frac{(2-1)3\pi}{2n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{1}{\sqrt{2}} & \cos\frac{(2n-1)\pi}{2n} & \cdots & \cos\frac{(n-1)(2n-1)\pi}{2n} \end{bmatrix}$$

DCT Interpolation Theorem

$$P_n(t) = \frac{1}{\sqrt{n}} y_0 + \frac{\sqrt{2}}{\sqrt{n}} \sum_{k=1}^{n-1} y_k \cos\frac{k(2t+1)\pi}{2n}$$

satisfies $$P_n(j) = x_j \text{ for } j = 0,\ldots, n-1$$

C transforms all the n data points into all the n interpolation coefficients. The discrete cosine transform provides the trigonometric interpolation function coefficients using only cosine terms

Suppose we are given a vector

The Discrete Cosine Transform is defined as :

$$x = [x_0,\ldots,x_{n-1}]^T$$

$$y = [y_0,\ldots y_{n-1}]^T$$

Where C is defined as $$y = Cx$$

DCT interpolation gives terms that are already arranged in terms of importance to the HVS *i.e.* human visual system.

First terms are the most significant and important, while the last terms are least important.

DCT Least Squares Approximation Theorem can be defined as:

$$x = [x_0,\ldots,x_{n-1}]^T$$
$$y = [y_0,\ldots,y_{n-1}]^T = Cx$$
$$P_m(t) = \frac{1}{\sqrt{n}} y_0 + \frac{\sqrt{2}}{\sqrt{n}} \sum_{k=1}^{m-1} y_k \cos\frac{k(2t+1)\pi}{2n}$$

Image compression is a method that minimizes the amount of memory it uses or takes to store in image.

We will exploit the fact that the Discrete Cosine Transform matrix is based on the visual system used for image compression.

This means we can delete the least or minimum significant values without our eyes noticing the difference.

Now we have found the matrix $Y = C(CX^T)^T$

Using the DCT, the entries in Y will be organized based on the HVS *i.e.* human visual system.

The most important values are placed in upper left corner of the matrix. And the least important values will be mostly placed in the lower right corner of the matrix.

2.3.2. AES Algorithm

AES is a block cipher with the block length of 128 bits. AES allows for three different key lengths: 128 bits, 192 bits, or 256 bits. In most of the cases we will assume that the key length is 128 bits. With regard to using the key length other than 128 bits, the factor that changes in AES[16],[18],[20] is how you generate the key schedule from the key. AES Encryption consists of 10 rounds of processing in total for 128-bit keys, 12 rounds in total for 192-bit keys, and 14 rounds in total for 256-bit keys. Except the last round, all the other rounds similar. Each round of processing includes a single byte based substitution step, a row wise permutation step, a column wise mixing step, and addition of the round key. The order in which all four steps are executed is different for encryption [17],[18] and decryption.AES requires the block size of 128 bits, the original Rijndael cipher works with any block size and any key. The state array for the different block sizes has only four rows in the Rijndael cipher. The number of columns depends on the size of the block. For example, when the block size is 192 bits, the Rijndael cipher requires a state array that consist of 4 rows and 6 columns.

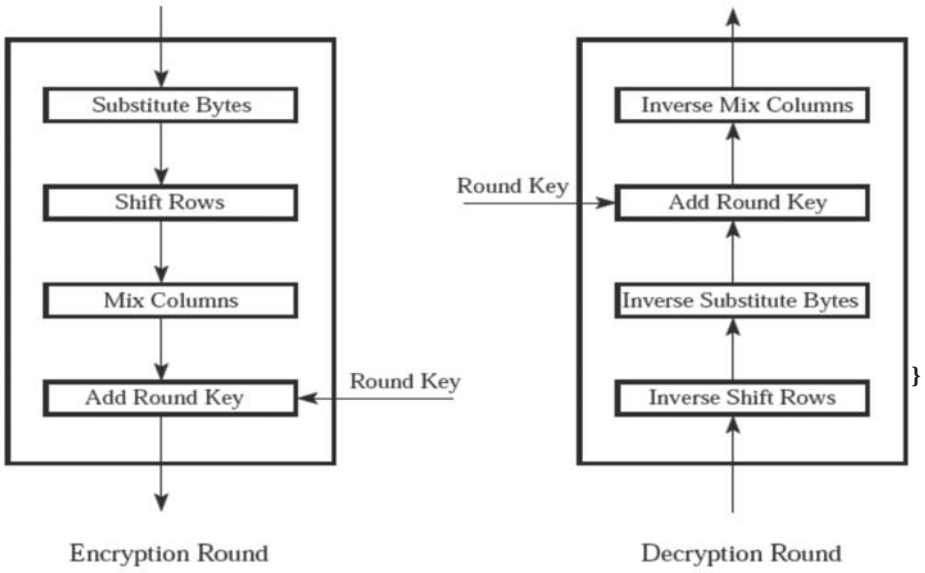

Fig. 5. AES Algorithm

3. EXPERIMENTAL RESULTS

The first part of this section contains experimental results and discussion for grayscale image and second part is for the colour images.

Grayscale Images : Initially we will compare the perceptual impact of the image after hiding the secret information. Then, impact on the image by statistical measures is compared. Initially experiments are performed with the grayscale images. In the next example, we extend our results for colored images. Fig 6(*a*) shows the original image (grayscale) and Fig 6(*b*) is the histogram of grayscale image[1], cover image that shows the minimum pixel value and maximum pixel value respectively, in cover image and pixel count having same color and (*c*-*j*) are the bit plans of the Cover image.

Fig 7 shows the bit plans of the 0th (most significant bit MSB) to the 7th (least significant bit LSB) bit stego images. We obtain the stego image with good quality and original data with the best distortion measurement and embedding capacity is also improved in a efficient manner and perform the most secure data using various data hiding schemes.

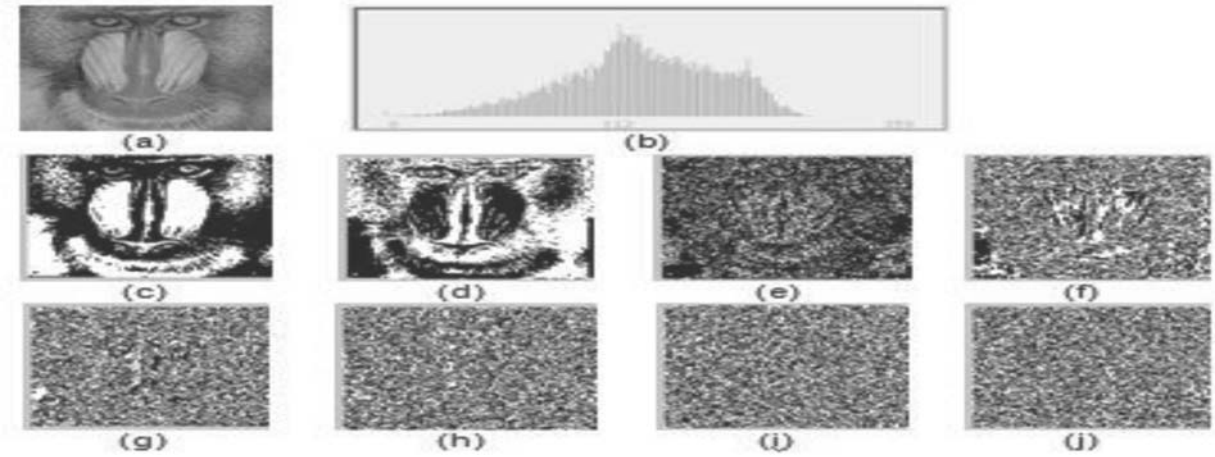

Fig.6. (*a*) Cover Image (*b*) Histogram of Cover Image (*c*-*j*) Bit plan of Cover Image

Colour Image : Color image section is expansion of experimental results[18] and discussion to colour images. Fig 7 (*a*) is the original RGB cover image having a total of 34,645 pixels in pixel data (*b*) is the histogram of the RGB cover image that shows minimum pixel value, maximum pixel value present[6] in cover image and maximum number of pixels having same color.(*c*),(*d*) and (*e*) are the channel based histograms of RGB cover image. All the experimental results for RGB images are displayed or shown after replacing the 960 bits of RGB cover image.Fig.8 (*a*-*c*) shows the RGB bit plan of colour image of Fig 7 (*a*). Fig. 9. (*a*-*h*) are the 0th to 7th bit stego images. Fig 10 is the 0th to 7th bit RGB Bit plans of stego image.

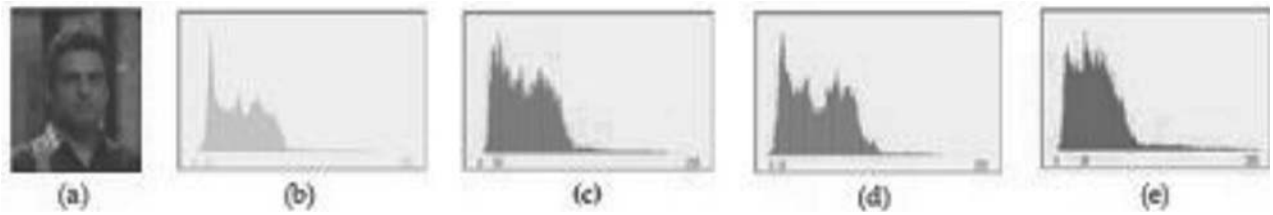

Fig. 7. (*a*) Cover image (*b*) Histogram of the Cover image (*c*-*e*) Channel Based Histogram of the Cover image.

We use as a support of communication (cover-object) digital grayscale image whose pixels are integers between 0 and 255, and each pixel value is assigned to 8 bits shown in Fig.8(*a*-*c*). Our steganographic scheme applied into the images proceeds in two steps: the first, is the choice of places that could be modified[18] to hide the secret message. The second, the choice of the protocol used to hide the message as shown in Fig 9(*a*-*h*) are the 0th to 7th bit stego images. Fig 10 is the 0th to 7th bit RGB Bit plans of stego image.

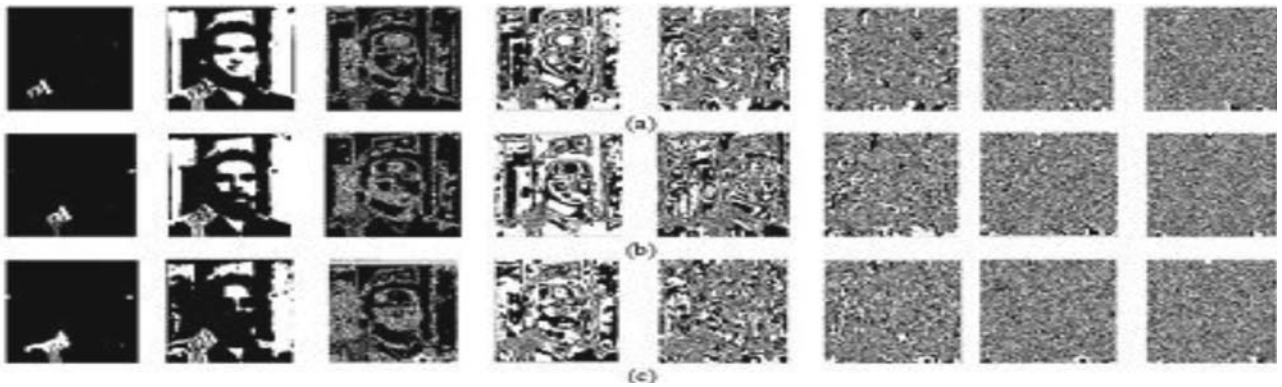

Fig. 8. (*a-c*) Bit plans of RGB channel of Cover Image.

Fig. 9. (*a-h*) 0th to 7th Bit Stego-images

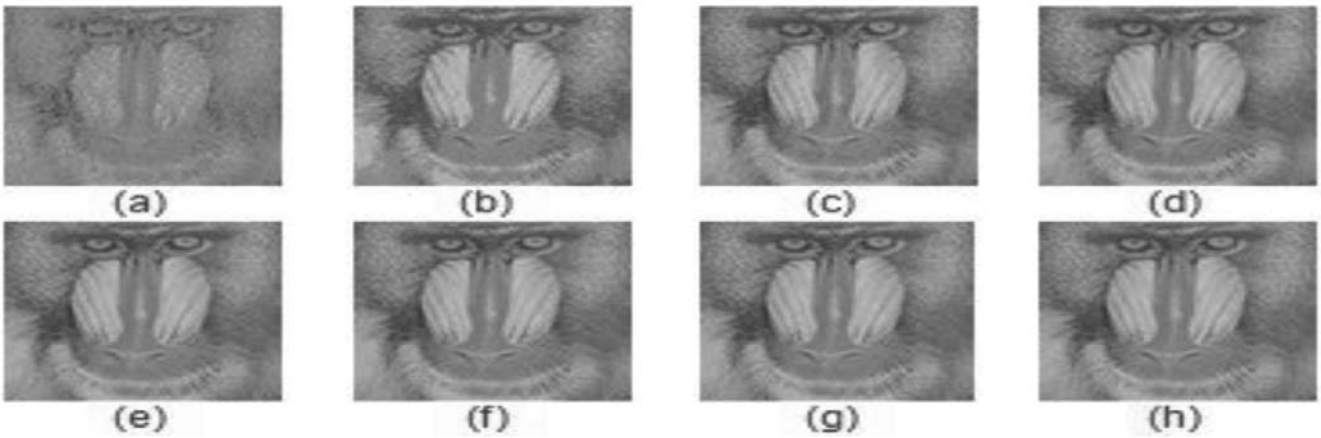

Fig. 10. (*a-h*) 0th (MSB) to 7th (LSB) Bit Stego-Image

Fig.11. (*a*) Original host image

Fig.11. (*b*) After embedding

5914 bytes

Fig. 11. (*c*) After Embedding

2957 bytes

We used in our experiences the LSB steganographic algorithm to practically demonstrate our results, and to facilitate the implementation of the proposed method. This technique makes use of the fact that the least significant bits(LSB) in an image could be the random noise and changes would not have any effect on the image[20].

In our example for $m = 3$, let G3 be a Z4-linear Goethals code of length23 and covering image pixel radius and switch has the parity-check matrix (PCM) as described below:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 2 & 3 & 1 \\ 0 & 0 & 1 & 0 & 3 & 3 & 3 & 2 \\ 0 & 0 & 0 & 1 & 2 & 3 & 1 & 1 \\ 0 & 2 & 2 & 2 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \\ 0 & 0 & 2 & 2 & 2 & 2 & 0 & 2 \end{pmatrix}$$

We take as inputs to our scheme:

1. The hosted image stated as A, can be modifed and embed the data. A is partitioned into blocks of size $2m + 1$ pixels.
2. r the number to be embedded in block of $2m + 1$ pixels of A.
3. The message M is partitioned into sequences of r bits to be embedded.

We embed the message $(M1,\cdots, Mr)$ in the LSBs of $2m + 1$ pixels gray values $(p1,\cdots, p2m + 1)$ by at most 6 changes in the following manner $(M1,\cdots, Mr) = (x1,\cdots, x2m + 1)$.HT.where xi denotes the LSB of pi.

3.1. Design Details

The design component is used to visualize the effect of our proposed scheme on the images, an implementation under Matlab is made by using as cover two grayscale[15],[16] images of different sizes:

1. Read the host gray-scaling image A.
2. The host image is partitioned into groups of 16 pixels.
3. The size of the image and the number of bits to be embedded in each group of 16 pixels together determine the capacity of embedding.
4. If the message size is to the estimated capacity, the embedding proceeds .Else an error message showing the maximal possible length is displayed.
5. The text message to be embedded is divided into segments of 14 bits that are embedded into a groups of 16 pixels along the embedding process.
6. For each group of 16 pixels, do the following:
   (*a*) Extract the cover-data $x = (x1,\cdots, x16)$ of 16 bits from the group by concatenation the LSB of each pixel value;
   (*b*) Hide the secret message $M = (M1,\cdots, M14)$ of 14 bits into the cover-data such that
   $$\varphi - 1(M) = (\varphi - 1(M1M2),\cdots, \varphi - 1(M13M14))$$
   has the form $(t, A + 2B, 2C)$. If the last 6 bits of the message do not verify
   $(\varphi - 1(M9M10),\cdots, \varphi - 1(M13M14)) = 2C$.
   they are replaced by zero, and the same 6 message bits are re-embedded in the next group of pixels,then go to step 7.
7. Store the resulting image as Stego Image (S).

3.2. Embedding Capacity of the proposed scheme

In this present implementation Lena gray-scale image of 512*512 pixels and Baboon grayscale image of 298*298 pixels, has been taken as cover images as shown in Figures 11(*a*). For each image, we applied our method and we display a comparative study in Figures 11 (b) and Fig 11 (*c*) of the proposed method with the optimal F 5 algorithm[21].

Applying the proposed method much amount of data could be embedded in the image: we can hide 14 bits in a sequence of 16 bits by changing at most 6 bits.

The amount of data embedded using the proposed scheme leads

indeed to a good results compared to the F 5 algorithm for embedding the secret data into a cover image[22].

4. CONCLUSION

The work is to secure the image and data using steganography and cryptography approach has been done and the existence technology has been tested and improved using different algorithm such as F5 and AES algorithm. A secure stegano algorithm F5 which is based on the Data hiding and Diffusion method is proposed in this paper. Beneûted from the effective optimization, good balance between the security of data and quality of image is achieved. The combination of steganography and cryptography algorithm which is used to maintain the quality of image as well as security of the data

The work can be extended for transmitting videos and audios as secret messages. For the further enhancement can be carried out by designing sophisticated software based on combination of steganography and cryptography techniques which will targeted to use in highly secure multimedia data transmission applications

5. REFERENCES

1. LiA ,Junhui He, Jiwu Huang, Yun Qing Shi "A Survey on Image Steganography and Steganalysis"Journal of Information Hiding and Multimedia Signal Processing Volume 2, Number 2, April 2011.
2. Hatimaboalsamh, hassanmathkour, samidokheekh, monamursi, ghazyas- sassa "An Improved Steganalysis Approach forBreaking the F5 Algorithm "WSEAS TRANSACTIONS onCOMPUTERS , Issue 9, Volume 7, September 2008,pp1447-1456.
3. .Katzenbeisser, S. and Petitcolas, F.A.P.: On Defining Security in Steganographic Systems. Proceedings of SPIE: ElectronicImaging 2002, Security and Watermarking of MultimediaContents, Vol. 4675. San Jose, California (2002).
4. H. Yang and A. C. Kot, "Pattern-based data hiding for binary image authentication by connectivity-preserving," *IEEE Transactions on Multimedia*,vol. 9, no. 3, pp. 475–486, 2007.
5. H. Yang, A. C. Kot, and S. Rahardja, "Orthogonal data embedding for binary images in morphological transform domain-a high-capacity approach," *IEEE Transactions on Multimedia*, vol. 10, no. 3, pp. 339–351, Apr. 2008.
6. T. Filler, J. Judas, and J. J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 920–935, Sept.2011.
7. V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *IEEE International Workshop on Information Forensics and Security*. IEEE, 2012, pp. 234–239.
8. N.F.Johnson, SushilJojadia George Mason University, "Exploring Steganography: Seeing theUnseen", (0018-916/98/ $10.00©) 1998 IEEE.
9. R.Poornima, R.J.Iswarya, "An Overview of Digital Image Steganography", International Journal of Computer Science & Engineering Survey (Vol.4, No 1),February 2013.
10. T.Morkel, T.H.P.Eloff, M.S.Olivier, "An Overview of Image Steganography", ICSA Research Group,Department of Computer Science.
11. Ashok, Y.Raju, S.Munishankaralak, K.Srinivas, Jammi Ashok, "Steganography: AnOverview", et.01./International Journal of Engineering Science and Technology, (Vol.2(10)), 2010,5985-5992.
12. Shikha Sharda, Sumit Budhiraja , "Image Steganography:A Review", International Journal ofEmerging Technology and Advance Engineering (volume 3, Issue 1), January 2013.

13. V. Asha, P. Nagabhushan, N. U. Bhajantri, “Similarity Measures for Automatic Defect Detection onPatterned Textures”, International Journal of Image Processing and Vision Sciences (IJIPVS)Volume-1 Issue-1, 2012.

14. RajkumarYadav, “Analysis of Various Image Steganography Techniques Based Upon PSNRMetric”, International Journal of P2P Network Trends and Technology- (Volume1, Issue2)- 2011,ISSN: 2249-2615.

15. Anderson, R.J. and Petitcolas, F.A.P.: On the Limits of Steganography. IEEE Journal of Selected Areas in Communications: Special Issue on Copyright and Privacy Protection),Vol. 16(4) (1998) 474"481.

16. Cachin, C.: An Information-Theoretic Model for Steganography. In: Aucsmith D. (eds.):Information Hiding: 2nd International Workshop. Lecture Notes in Computer Science, Vol.1525. Springer-Verlag, Berlin Heidelberg New York (1998) 306–318.

17. Katzenbeisser, S. and Petitcolas, F.A.P.: On Defining Security in SteganographicSystems.Proceedings of SPIE: Electronic Imaging 2002, Security and Watermarking of MultimediaContents, Vol. 4675. San Jose, California (2002).

18. Chandramouli, R. and Memon, N.: Analysis of LSB Based Image SteganographyTechniques. Proceedings of ICIP 2001 (CD version). Thessaloniki, Greece (2001) .

19. Fridrich, J., Goljan, M., and Du, R.: Reliable Detection of LSB Steganography in Grayscaleand Color Images. Proc. of ACM: Special Session on Multimedia Security and Watermarking. Ottawa, Canada (2001) 27–30.

20. Fridrich, J., Goljan, M., and Du, R.: Detecting LSB Steganography in Color and GrayscaleImages. Magazine of IEEE Multimedia: Special Issue on Security, Vol. Oct-Dec (2001) 22–28.

21. H. Lu, A. C. Kot, and Y. Q. Shi, “Distance-reciprocal distortion measurefor binary document images,” *Signal Processing Letters, IEEE*, vol. 11,no. 2, pp. 228–231, Feb. 2004.

22. J. Cheng and A. C. Kot, “Objective distortion measure for binary textimage based on edge line segment similarity,” *IEEE Transactions onImage Processing*, vol. 16, no. 6, pp. 1691–1695, June 2007.