Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 12 (5): 428-439, 2012
ISSN 1812-5654 / DOI: 10.3923/jas.2012.428.439

Brownian Motion of Binary and Gray-Binary and Gray Bits in Image for Stego

Rengarajan Amirtharajan and John Bosco Balagurn Rayappan
Department of Electronics and Communication Engineering,
School of Electrical and Electronics Engineering, SASTRA University,
Thanjavur, Tamil Nadu, India

Abstract: Computers have invaded all premises of the human world, starting from a grocery store to a missile launching center. Because of the omnipresence of computers, it becomes more and more difficult everyday to secure the confidential information from misuse. The fairly common technique of cryptography has been proved inadequate in recent years. Steganography, a contemporary yet an age-old technique to hide secret data into an unsuspected cover media like an image, thereby preventing the recognition of the very presence of secret data, is an alternative. In this study, an improved image steganographic approach is proposed. This method reduces the mean square error (MSE) by localizing the error-reduction process to every row. The error reduction is performed by selective embedding of the actual secret, its binary complement, gray-coded version or inverted gray-coded version. Of the four versions, the version giving the least MSE is embedded on a row-by-row basis. This method reduces the MSE by a factor of 1.8 and boosts the peak signal to noise ratio (PSNR) by a 0.25 db and considerably increases the security.

Key words: Binary gray encoding, inverted pattern approach, optimum pixel adjustment process, steganography

INTRODUCTION

Power has taken several incarnations ever since its genesis. Right from Stone Age to the present day, power has kept its possessor at the summit of hegemony and thus it is the most sought after commodity. In the electronic epoch power has manifested itself in the form of classified and critical information. Since the human race has succumbed to enticing power to such an extent that iniquity today is skyrocketing, there is a need to protect information from falling into the wrong hands and to prevent clandestine and unscrupulous activities. Providentially, the advancements in technology have begot many techniques to maintain the veracity and variability of the crucial information giving rise to an entire discipline called information hiding. Information hiding is stratified into several subsets namely cryptography (Schneier, 2007), steganography and watermarking (Stefan and Fabin, 2000; Zaidan *et al*., 2010).

Cryptography (Schneier, 2007) is the art of writing esoteric information in an occult fashion thereby rendering it scrutable only to the authorized receiver. In contrast to cryptography which focuses on keeping the contents of a message secret, steganography (Stefan and Fabin, 2000; Zaidan *et al*., 2010) focuses on keeping the very existence of a message secret. Steganography is implemented in digital audio (Zhu *et al*., 2011), video (Al-Frajat *et al*., 2010) and images (Amirtharajan and Balaguru, 2009, 2010, 2011, Amirtharajan *et al*., 2012; Bender *et al*., 1996) of which image steganography has gained much appreciation and commendation in the recent past. In image steganography the vital information is dissembled in a cover image with assiduous efforts resulting in a stego image. The embedded secret information is imperceptible to the human eye thereby rendering the image impregnable (Yang, 2008).

In the available literature many researchers proposed an assortment of approaches to information hiding. These methodologies have different characteristics like capacity, imperceptibility and robustness (Amirtharajan and Balaguru, 2009, 2010, 2011; Kumar *et al*., 2011). These characteristic are inevitable for different applications, such as secret communication (Stefan and Fabin, 2000), copyright protection (Wang and Lin, 2004; Yen and Tsai, 2008) and tampering detection or integrity check (Lin *et al*., 2005).

Information hiding techniques could be categorized into two types: methods in the spatial domain and methods in the frequency domain. In the spatial domain approach, the secret messages are embedded by directly

Corresponding Author: Rengarajan Amirtharajan, Department of Electronics and Communication Engineering, School of Electrical and Electronics Engineering, SASTRA University, Thanjavur, Tamil Nadu, India

injecting secret data in the image pixels (Chan and Cheng, 2001, 2004; Wang *et al*., 2001; Chang *et al*., 2003; Yang, 2008; Thien and Lin, 2003). Whereas in the later case, the frequency domain approach the image is first transformed into its frequency domain (Amirtharajan and Rayappan, 2012a, b; Chang *et al*., 2002) then the secret messages are embedded in the transformed coefficients.

The major concern is about the objective of transmitting secret data, the stego method should possess high capacity, high quality and imperceptibility. More number of research papers have been intended for this theme and performs the embedding operations in the spatial domain either using raster scan or random scan (Amirtharajan and Balaguru, 2009, 2010; Amirtharajan *et al*., 2011, 2012b, 2012a; Yen and Lin, 2010). A detailed survey on Information hiding till 1999 is available by Petitcolas *et al*. (1999). A complete survey on image steganography could be found by Cheddad *et al*. (2010) and on random image steganography and steganalysis in Amirtharajan *et al*. (2012) three more survey on Field Programmable Gate Array (FPGA) for steganography, middle ware for cryptography/ steganography and Orthogonal Frequency Division Multiplexing (OFDM)+Code Division Multiple Access (CDMA)+stego for secure communication is available by Rajagopalan *et al*. (2012), Janakiraman *et al*. (2012a) and Thenmozhi *et al*. (2012), respectively. There are three kinds of approaches called LSB-based (Chan and Cheng, 2001, 2004; Wang *et al*., 2001; Chang *et al*., 2003; Yang, 2008; Thien and Lin, 2003; Amirtharajan and Balaguru, 2009, 2010), PVD-based (Wang *et al*., 2008; Amirtharajan *et al*., 2010) and mod-based (Chan and Cheng, 2004; Thien and Lin, 2003; Wang *et al*., 2008) are commonly available in literature and sometimes it could be combined to offer both capacity, imperceptibility and to improve the security (Chang *et al*., 2003; Hmood *et al*., 2010a, b; Xiang *et al*., 2011; Lin *et al*., 2005; Janakiraman *et al*., 2012b; Zaidan *et al*., 2010, 2011 and Zanganeh and Ibrahim, 2011). The counter attack on steganography called steganalysis are detailed (Xia *et al*., 2009; Qin *et al*., 2009). A detailed review on steganalysis is reported by Qin *et al*. (2010).

In LSB-based approaches, secret data are embedded by directly substituting the least-significant-bits (LSBs) with equal bits of the secret for each pixel. Furthermore, techniques based on pixel-value differencing (PVD) modify the difference value between a pair of pixels to fit the value of the embedded secret. Finally, mod based approaches which use the modular operation, are similar to k-bit LSB-based approaches if the modulus is 2^k.

Motivated by this study, a simple and effective stego method has been proposed to improve the stego image quality and to introduce cryptic effect while embedding.

PRELIMINARY RELATED WORKS

Chan and Cheng (2004) proposed an LSB-based hiding scheme using an optimal pixel adjustment process (OPAP). Their method adjusts each pixel after the message is embedded to improve the quality of the stego object and their experimental results showed that their method yielded quicker results. Yang (2008) proposed new LSB-based approach, named as the Inverted pattern (IP) LSB substitution approach. Later this method combined with OPAP called IPLSB to improve the quality of the stego image. In this study, we have adapted a new LSB-based approach based on Yang (2008), named as the inverted pattern binary and gray (IPBG) LSB substitution approach, to further highlight the quality of the stego-image Before secret messages are embedded, some secret messages are transformed by inverting operation and some secret messages are not. A simple strategy is used to judge whether a section of messages is inverted and a bit string named as the IPKey is used to record these inverting actions. Also, we combine the concept of the OPAP with our approach to improve image quality further. The experimental results show that the proposed approach results in a better image quality than that of the optimal LSB substitution approach (Wang *et al*., 2001; Chan and Cheng, 2001; Thien and Lin, 2003), the OPAP LSB substitution approach (Chan and Cheng, 2004) and inverted pattern approach (Yang, 2008).

In a normal LSB substitution the RGB (red blue green) image is converted in to gray image and then last few least significant bits of gray image are selected according to key length k and the message which is to be embedded is converted to series of ASCII values of the characters in the message and then to binary. Message is then stored in the cover according to the method of embedding. The series of operations done in LSB substitution are as follows:

Let C be the original 8-bit grayscale cover-image of $M_c \times N_c$ pixels represented as:

$$C = \{x_{ij} | \ 0 \le I \le M_c, 0 \le j \le NC \\ x_{ij} \in :0, 1, 2,,,,, 225\}$$

D be the n-bit secret data represented as:

$$D = \{d_i \ | 0 \le I \le n, d_i \in \{0, 1\}\}$$

Suppose that the n-bit secret data D_d (decimal representation) is to be embedded into the k-rightmost LSBs of the cover-image C:

$$S = C\text{-}C \bmod 2^k + D_d$$

Here, S is the Stego object, C cover object and D_d is the decimal equivalent of the secret data.

In the extraction process, given the stego-image S, the embedded messages can be readily extracted without referring to the original cover-image. The k LSBs of the selected pixels are extracted and lined up to reconstruct the secret message bits. Mathematically, the embedded message bits D can be recovered by:

$$D_d = S \bmod 2^k$$

The OPAP simply improves the stego object after embedding the secret data, either by adding or subtracting 2^k without affecting the rightmost k secret data bits in the stego cover.

THE PROPOSED METHOD

Embedding: A schematic diagram of the proposed method is given in Fig. 1 and 2. Initially the secret data or message is encrypted using Data Encryption Standard DES (Schneier, 2007), is a symmetric key cryptography algorithm. The cover image is split into separate rows. The order of rows considered for embedding data is chosen using a Pseudo random number generator with a chosen seed. For each row, a try is made to embed data, inverse data, gray code of data and the inverted gray code of data. The encoded form of the confidential information on the selected row which offers minimum Mean Square Error (MSE) is chosen and fixed for the same. This binary/ inverted binary/gray and inverted gray data pattern is stored as IPKEY. Thus, on an average MSE is reduced to a greater extent. The Stego image, IPKEY and the seed are communicated.

Mathematical model for row wise, inverted pattern LSB embedding

- **General formulae:** 1's complement of a No.:

$$\bar{x} = (2^k - 1) - x \tag{1}$$

Where:

k = No. of bits
x = Number to be inverted in bits
$\bar{x}$ = 1's complement of the number

For example, take a 4 bit binary representation of a number '2' [0010] as x here:

k = 4 so, $\bar{x} = (2^4 - 1) - 2$ = (16-1)-2
$\bar{x}$ = 13[1101] complement of [0010] '2'

- **LSB embedding:**

$$S_i = C_i - C_i \bmod 2^k + m_i \tag{2}$$

Where:

k = No. of bits to be embedded
C_i = Cover pixel
S_i = Stego pixel
m_i = k – bit message block in decimal

For example, let k = 4. C_i = 16[0001000] and m_i is '2' [0010] as m_i:

S_i = 16- 16 mod 2^4 + 2 = 16-16mod16+2 = 18
S_i = 8[00010010]

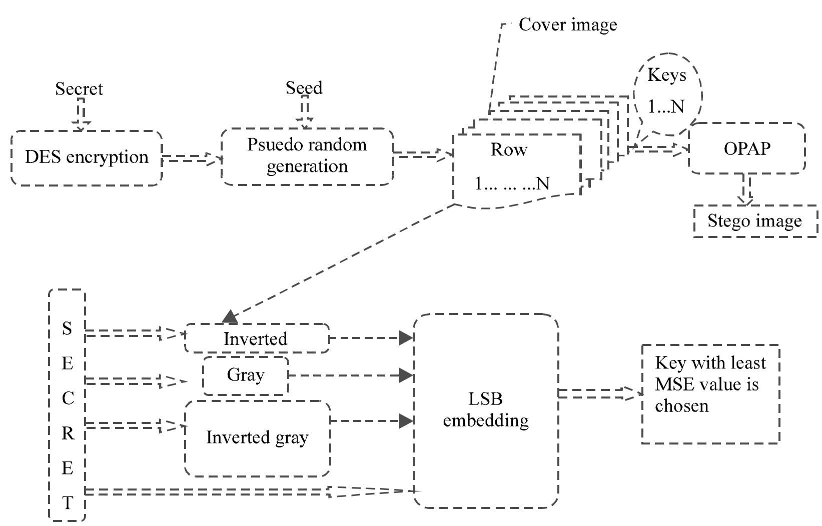

Fig. 1: Proposed schematic diagram for embedding

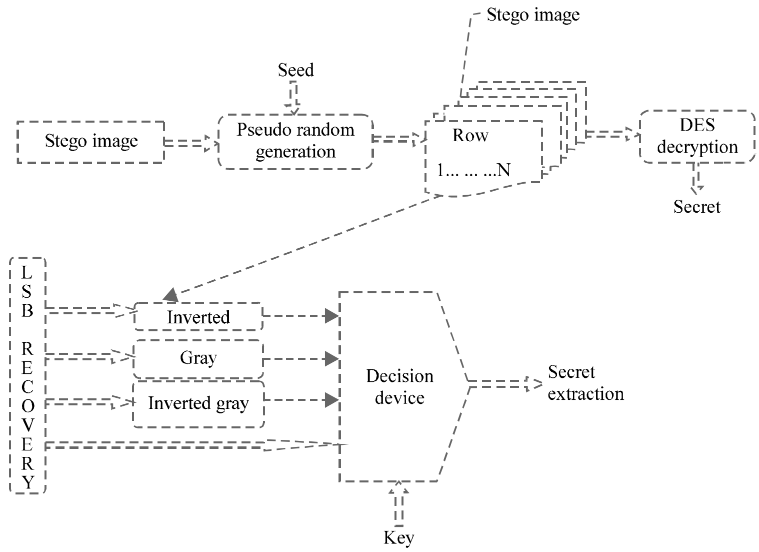

Fig. 2: Block diagram for extraction

- **LSB recovery:**

$$m_i = S_i \bmod 2^k \quad (3)$$

where, symbols are same as Eq. 2:

Let S_i = 18[00010010]
To Extract the last 4[since k = 4] bits, we have:

m_i = 18 mod
2^4 = 18 mod
16 = 2 [0010]

- **General IDEAS**
Four flavours of secret data:

- Plain data-m (i,j)
- Inverted data - $\bar{m}$ (i, j)
- Grey Coded data-g (i,j)
- Inverted Grey Coded data-g' (i,j)

- **R rows:** In each cover image, there are 'R' No. of rows, each of same length D, where:

$$R\times D = M_c\times N_c \quad (4)$$

where, the $M_c\times N_c$ are dimensions of the cover image. Each row is denoted as ri, where, i∈N and I≤R i.e., Set of rows = {ri, ∀ i∈N and i≤R}

- Each row r_i is in turn a matrix, denoted as:

$$r_i = [r_{i1}, r_{i2,\ldots\ldots}, r_{iD}] \text{ where } i \in N \text{ and } i \le R \quad (5)$$

In other words, each row has D pixels.

- **Message data (secret) to be embedded [k bit length]:** m (i, j), where:

i = Row identifier
j = Pixel inside a row

The complement of m (i, j) is denoted as $\bar{m}$ (i, j)

- **Embedding procedure:** Let the cover image be C with $M_c\times N_c$ pixels.
Let it be divided into R blocks named r_i, $r_{i,\ldots\ldots}$, r_{iR}, each having equal number of pixels D:

$$R \text{ x } D = M_c\times N_c \quad (6)$$

Also, $r_i = [r_{i1}, r_{i2,\ldots\ldots}, r_{iD}]$, where i ∈ N and i ≤ R
Let 'k' be the number of LSBs to be replaced in cover pixels.
Let the secret message be a matrix M, where each elements of M is made up of k bits. Then we can denote the message to be embedded in the i^{th} row, j^{th} pixel as m (i, j). Let s (i, j) denote the stego value of j^{th} pixel in the i^{th} row, when message m (i, j) is embedded in cover pixel r_{ij}. Alternatively $\bar{m}$ (i, j) is embedded instead of m (i, j) then the stego pixel is denoted as $\bar{s}$ (i, j):

$\therefore$ s (i, j) = r_{ij}-r_{ij}mod2^k+ m (i, j) (Applying Eq. 2) (7)
$\bar{s}$ (I, j) = r_{ij}-r_{ij}mod2^k+ $\bar{m}$ (i, j) (Eq. 1)
s_g (i,j) = r_{ij}-r_{ij}mod2^k+ g (i, j)
s_g' (i, j) = r_{ij}-r_{ij}mod2^k+ g' (i, j)

If we consider R blocks of stego image as $s_1, s_2, \ldots\ldots s_R$. Then, $s_i = s\ (i)$ or $\bar{s}\ (i)$ or $s_g\ (i)$ or $s_g'\ (i)$ where MSE is minimum and $s\ (i) = \{s\ (i, j), j \in N \text{ and } j \le D\}$

key matrix is denoted as:

$$K = [K_1, K_2, \ldots\ldots K_R]$$

where, K_i is chosen based on the following conditions 00-if m (i,j) is embedded:

01 = if $\bar{m}$ (i, j) is embedded
10 = if g (i,j) is embedded
11 = if g' (i,j) is embedded

- **Retrieval procedure:** Key matrix is denoted as:

$$K = [K_1, K_2, \ldots\ldots K_R]$$

The preliminary, unprocessed message $m_u\ (i,j)$ can be extracted from pixels in stego image as: $m_u\ (i,j) = s\ (i,j) \bmod 2^k$ from Eq. (3) the actual message m (i,j) can be extracted by processing $m_u\ (i,j)$ as follows:

- m (i,j) is chosen from the following conditions based on K_i value
- $m_u\ (i,j)$ - if corresponding $K_i = 00$
- $(2^k-1)-m_u\ (i,j)$, if $K_i = 01$
- $g^{-1}\ (i,j)$, if $K_i = 10$ (if g^{-1} denotes inverse of grey code function)
- $g'^{-1}\ (i,j) = (2^k-1)-g^{-1}\ (i,j)$ (if $K_i = 11$)
- (if g'^{-1} denotes inverse of inverted grey code function)

WORST CASE MSE

The worst case MSE for a block with D pixels is defined as:

$$\begin{aligned} MSE_w\ (i) &= \frac{1}{D}\sum_{j=1}^{D}(2k - 1)^2 \\ &= D^{-1}\ [D(2k-1)^2] \\ &= (2^k-1)^2 \end{aligned} \quad (8)$$

MSE for i^{th} row, when m (i) (actual data) is embedded, is given as:

$$MSE(i) = D^{-1}\sum_{j=1}^{D}(s(i,j) - c(i,j))^2$$

When inverted data $\bar{m}$ (i) is embedded, then, MSE for the same parameters is denoted as:

$$\overline{MSE}(i) = D^{-1}\sum_{j=1}^{D}(\bar{s}(i,j) - c(i,j))^2$$

When grey coded data g (i) is embedded, then MSE for the same parameters is denoted as:

$$MSE_g\ (i) = D^{-1}\sum_{j=1}^{D}(s_g\ (i,j) - c(i,j))^2$$

When inverse grey coded data g' (i) is embedded, then MSE for the same parameters is denoted as:

$$\overline{MSE}g(i) = D^{-1}\sum_{j=1}^{D}(\overline{s_g}\ (i,j) - c(i,j))^2$$

According to the embedding procedure, minimum MSE is chosen. The minimum MSE for a row is defined as:

$$MSE_{min}\ (i) = \min\ \{MSE\ (i), \overline{MSE}, MSE_g\ (i), \overline{MSE_g}\ \}$$
$$MSE\ (i) + \overline{MSE}\ (i) + MSE_g\ (i) + \overline{MSE_g}\ \text{(I):}$$

$$D\text{-}1\sum_{j=1}^{D}(s(i,j) - c(i,j))^2 + D\text{-}1\sum_{j=1}^{D}(\bar{s}(i,j) - c(i,j))^2 + D\text{-}1\sum_{j=1}^{D}(s_g\ (i,j) - c(i,j))^2 + D\text{-}1\sum_{j=1}^{D}(\overline{s_g}\ (i,j) - c(i,j))^2$$

$= D^{-1}\sum_{j=1}^{D}\ (2^k - 1)^2$ (since sum of all s, $\bar{s}$, s_g and $\overline{s_g}$ components:

$= (2^k - 1)^2$) (as given in the IP paper)
$= (2^k - 1)^2 = MSE_w\ (i)$ (9)

We know that, if any n numbers $x_1, x_2, x_3, x_4, \ldots, x_n$ add up to produce a total T, then:

$$\text{Min}\{x_1, x_2, x_3, x_4, \ldots, x_n\} \le (T/n) \quad (10)$$

Thus, applying (10) in (9), we get:
$MSE_{min}\ (i) \le (1/2)\ MSE_w\ (i)$ for all $I \le R$

Thus, we get MSE for any block to be less than or equal to ½ of the worst case MSE.

Random k-bit Adaptive Embedding

Inputs:

- Sampled Cover Image C
- Secret data bit stream M
- Key E for Encryption

Outputs:

- Stego Image (S), containing embedded secret data
- KEY (Used for recovery)

Algorithm for embedding:

- **Step 1:** Encrypt the secret data (M) using DES (Data Encryption Standard) with key E
- **Step 2:** Let P = length of secret data stream M (in number of bits) got from Step-1
- **Step 3:** Split the cover image C into separate rows . Let N = Total number of rows
- **Step 4:** Generate a array PRN of N pseudo-random numbers in the range [0,N-1] where each No. occurs only once. Let the seed be stored in a text file
- **Step 5:** Invert the bit array M to give $\overline{M}$. Encode the bit array M using Grey Code to give G and invert G to give $\overline{G}$
- **Step 6:** Let i = 1 (Here, i is the row counter)
- **Step 7:** Select PRNG[i]th block and perform the following operations

Selective embedding

- {a. Let r = pixel index array (for traversal)
- b. For (j = 1 to length (r)) do (Here j is the pixel counter)
  {
Replace k LSBs of jth pixel of the selected block with k bits from M to give O[i,1]
  }
- c. Compute MSE
- d. For (j = 1 to length (r)) do (Here j is the pixel counter)
  {
Replace k LSBs of jth pixel of the selected block with k bits from $\overline{M}$ to give O[i,2]
  }
- e. Compute $\overline{MSE}$
- f. For (j = 1 to length (r)) do (Here j is the pixel counter)
  {
Replace k LSBs of jth pixel of the selected block with k bits from G to give O[i,3]
  }
- g. Compute MSEGray
- h. For (j = 1 to length (r)) do (Here j is the pixel counter)
  {
Replace k LSBs of jth pixel of the selected block with k bits from $\overline{G}$ to give O[i,4]
  }
- g. Compute MSE gray
- h. If MSE is greatest

KEY[i] = "00"
Else if $\overline{MSE}$ is greatest
KEY[i] = "01"
Else if MSEGray is greatest
KEY[i] = "10"
Else
KEY[i] = "11"
Assign MSE[i] = Minimum MSE
  }

- Choose STEG[i] as the value of O for which MSE is minimum
- P = P-k. (Reduce length as k bits have been embedded)
- If P>0 then assign i = i+1. Else, goto step-8 (that is check whether message is finished)
- If i>N then goto step-8 (check whether EOF is reached for cover image)
- **Step 7:** F. Goto
- **Step 8:** Save the array STEG as the stego image array S
- **Step 9:** Save S into a image file and KEY in a text file
- **Step 10:** Communicate S,KEY and seed used to generate PRN

Recovery process: The same Pseudo random number sequence is generated using the received seed. Using the KEY, the pattern is identified for different rows. Recovery modules are run to recover the secret. The result is then decrypted using DES to get the message back.

Random k-bit adaptive recovery

- **Inputs:**

- Stego Image (S), containing embedded secret data
- Key E for decryption. KEY in text file from embedding process
- Seed (to generate Pseudo Random Number Generator PRNG)

- **Output:**

- Secret data bit stream M

Algorithm for extraction

- **Step 1:** Split the stego image S into separate rows. Let N = Total number of rows
- **Step 2:** Generate a array PRNG of N pseudo-random numbers in the range [0,N-1] where each No. occurs only once
- **Step 3:** Let i = 1 (Here, i is the row counter)
- **Step 4:** Select PRNG[i]th row and perform the following operations:

- Get Message M using retrieval
- B. If KEY[i,1] = "01"

M[i] = $\overline{M}$ [I]
Else if KEY[i,1] = "10"
M[i] = MGray[i]

Else if KEY[i,1] = "11"
M[i] = MGray [I]
Else
M[i] = M[i]

- Assign i = i+1 (increment row count)
- If i>N goto step-5 else goto Step-4

- **Step 5:** Decrypt M using DES and write it to text file as output

RESULTS AND DISCUSSION

In this present implementation Lena, Baboon, Gandhi and Temple 256×256 pixel Images have been considered by varying k = 1, 2, 3 and 4 bit LSB embedding, then stego image quality has been improved with OPAP. The effectiveness of the proposed system has been estimated by computing the MSE and PSNR of the Stego object with cover object.

The MSE is calculated by using the equation:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(X_{i,j} - Y_{i,j}\right)^2 \tag{11}$$

where, M and N denote the total number of pixels in the horizontal and the vertical dimensions of the image Xi,j represents the pixels in the original image and Yi, j, represents the pixels of the stego-image.

The Peak Signal to Noise Ratio (PSNR) is calculated using the equation:

$$PSNR = 10\log_{10}\left(\frac{I_{max}^2}{MSE}\right)dB \tag{12}$$

where, I_{max} is the intensity value of each pixel which is equal to 255 for 8 bit gray scale images. Higher the values of PSNR better the image quality. The cover image is given in Fig. 3a and the corresponding stego images for k = 1 in Fig. 3 b, k = 2 in Fig. 3 c, k = 3 in Fig. 3 d, k = 4 in Fig. 3f and the proposed stego results in Fig. 3 g for k = 4, MSE, PSNR is given in Fig. 4 and 5, respectively.

In the case of simple LSB embedding for full embedding capacities 256×256 bits for k = 1, 256×256×2 bits for k = 2 and so on. While using secret data in binary format alone for k = 4 the MSE is 36.60, inverted binary is 36.90, gray is 41.17, inverted gray is 40.88 and the proposed is 34.84. The proposed method MSE by adapting quantum of 64 pixels further reduces it to 32.81. These values are shown in Table 1- 3.

The corresponding PSNR value of the proposed method improved to 35.03826 dB which is for better than Chan and Cheng (2004) method PSNR of 34.8 dB. The corresponding MSE value of the proposed method reduced to 20.38 which is for better than Chan and Cheng (2004) method MSE of 21.6 and Yang (2008).

Image steganography is successfully implemented using a novel encoding method in which various bit

Table 1: Comparative MSE values for Full Embedding capacity on Lena and Baboon by splitting into 256 pixels as one block

	MSE for simple LSB substitution Lena					PSNR for simple LSB substitution Lena			
k bit embedding	1	2	3	4	k bit embedding	1	2	3	4
Binary	0.497101	2.210083	9.042145	36.60521	Binary	51.16636	44.68672	38.56809	32.49537
Inv. binary	0.502899	2.198792	8.976959	36.90590	Inv. binary	51.11599	44.70896	38.59951	32.45985
Gray	0.499008	2.509048	9.857178	41.16948	Gray	51.14973	44.13571	38.19328	31.98505
Inv. gray	0.500992	2.488876	9.86731	40.88161	Inv. gray	51.13250	44.17077	38.18882	32.01552
Best	0.465515	2.082016	8.578262	34.84331	Best	51.45147	44.94596	38.79681	32.70961
MSE after OPAP process Lena					**PSNR after OPAP process Lena**				
Binary	0.497101	1.495239	5.536530	21.59154	Binary	51.16636	46.38370	40.69843	34.78797
Inv. binary	0.502899	1.500183	5.463287	21.65492	Inv. binary	51.11599	46.36936	40.75626	34.77524
Gray	0.499008	1.505264	5.505615	21.61382	Gray	51.14973	46.35468	40.72275	34.78349
Inv. gray	0.500992	1.499008	5.498413	21.55495	Inv. gray	51.13250	46.37276	40.72843	34.79533
Best	0.465515	1.458481	5.368301	20.99858	Best	51.45147	46.4918	40.83243	34.9089
	MSE for simple LSB substitution Baboon					PSNR for simple LSB substitution Baboon			
k bit embedding	1	2	3	4	k bit embedding	1	2	3	4
Binary	0.502014	2.198547	9.108154	36.10493	Binary	51.12364	44.70945	38.5365	32.55514
Inv. binary	0.497986	2.213379	9.056458	36.02116	Inv. binary	51.15863	44.68025	38.56122	32.56523
Gray	0.500198	2.515854	9.962006	40.33107	Gray	51.13938	44.12395	38.14734	32.07441
Inv. gray	0.499802	2.485123	9.90799	40.33501	Inv. gray	51.14283	44.17733	38.17095	32.07398
Best	0.467468	2.090958	8.618286	34.28227	Best	51.43328	44.92735	38.77659	32.78011
MSE After OPAP Process Baboon					**PSNR after OPAP process Baboon**				
Binary	0.502014	1.497986	5.538086	21.48482	Binary	51.12364	46.37573	40.69721	34.80949
Inv. binary	0.497986	1.498291	5.4776	21.47478	Inv. binary	51.15863	46.37484	40.7449	34.81152
Gray	0.500198	1.509262	5.525238	21.52687	Gray	51.13938	46.34316	40.70729	34.80099
Inv. gray	0.499802	1.49379	5.480499	21.47319	Inv. gray	51.14283	46.38791	40.7426	34.81184
Best	0.467468	1.459976	5.374146	20.97417	Best	51.43328	46.48735	40.82771	34.91396

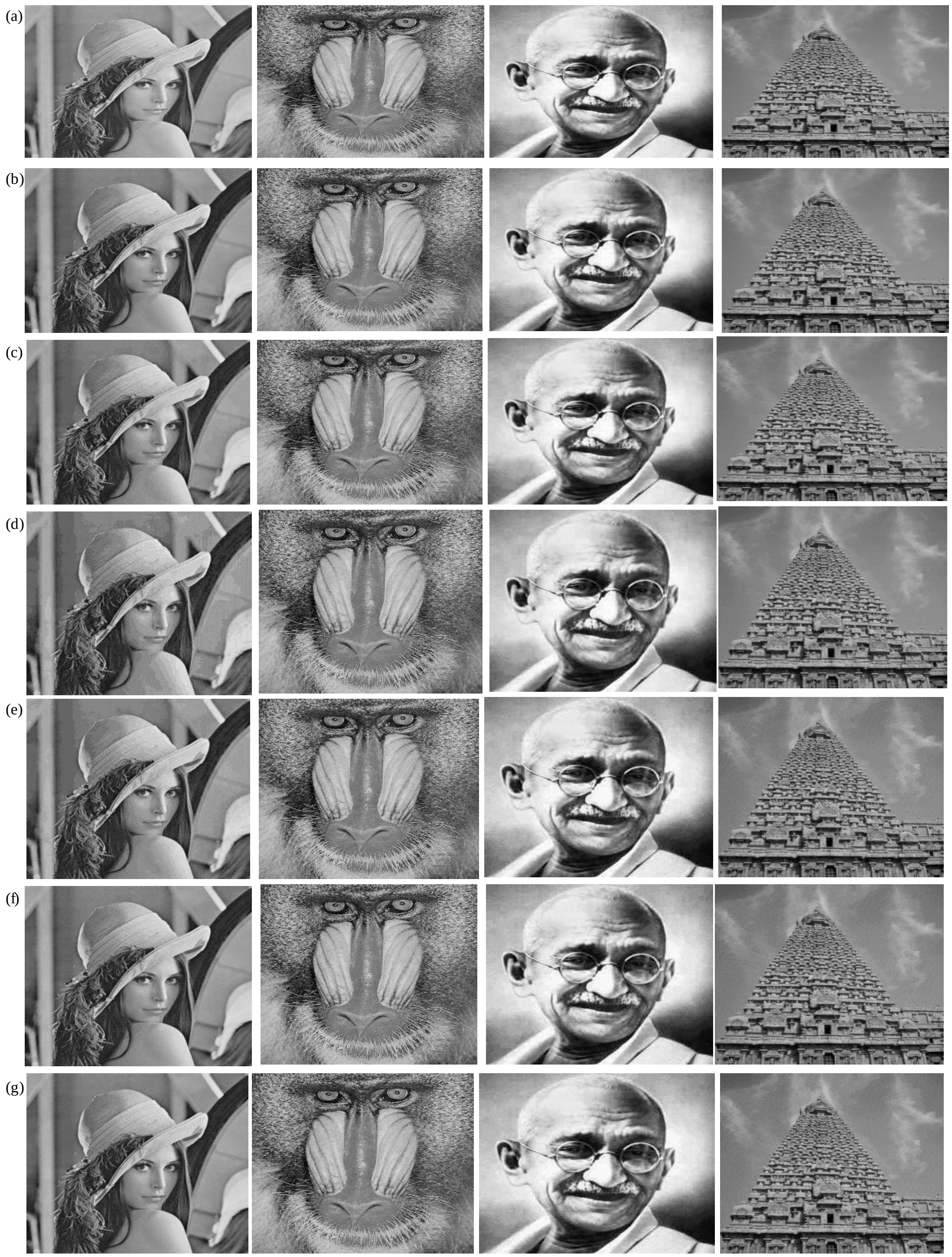

Fig. 3(a-g): (a) Cover Images Lena, Baboon, Gandhi and Temple, (b) k = 1, (c) k = 2, (d) k = 3, (e) k = 4, (f) k = 4 Proposed 256 Stego Images Lena, Baboon, Gandhi and Temple and (g) k = 4 Proposed 64 Stego Images Lena, Baboon, Gandhi and Temple

Table 2: Comparative MSE values for full embedding capacity on Lena and Baboon by splitting into 64 pixels as one block

	MSE for simple LSB substitution Lena image					PSNR for simple LSB substitution Lena image in dB			
k bit embedding	1 bit	2 bit	3 bit	4 bit	k bit embedding	1 bit	2 bit	3 bit	4 bit
Binary	0.4971	2.2100	9.0421	36.6052	Binary	51.1663	44.6867	38.5680	32.4953
Inv. binary	0.5028	2.1987	8.9769	36.9059	Inv. binary	51.1159	44.7089	38.5995	32.4598
Gray	0.4990	2.5090	9.8571	41.1694	Gray	51.1497	44.1357	38.1932	31.9850
Inv. gray	0.5009	2.4888	9.8673	40.8816	Inv. gray	51.1325	44.1707	38.1888	32.0155
Best	0.4331	1.9493	8.0310	32.8100	Best	51.7645	45.2319	39.0830	32.9707
MSE after OPAP process Lena image					**PSNR After OPAP Process in dB Lena Image**				
Binary	0.4971	1.4952	5.5365	21.5915	Binary	51.1663	46.3837	40.6984	34.7879
Inv. binary	0.5028	1.5001	5.4632	21.6549	Inv. binary	51.1159	46.3693	40.7562	34.7752
Gray	0.4990	1.5052	5.5056	21.6138	Gray	51.1497	46.3546	40.7227	34.7834
Inv. gray	0.5009	1.4990	5.4984	21.5549	Inv. gray	51.1325	46.3727	40.7284	34.7953
Best	0.4331	1.4091	5.2095	20.3823	Best	51.7645	46.6411	40.9628	35.0382
	MSE for simple LSB substitution Baboon image					PSNR for simple LSB substitution Baboon image in dB			
k bit embedding	1 bit	2 bit	3 bit	4 bit	k bit embedding	1 bit	2 bit	3 bit	4 bit
Binary	0.4997	2.1849	9.0795	36.0596	Binary	51.1440	44.7365	38.5502	32.5606
Inv. binary	0.5003	2.2141	9.0668	35.8387	Inv. binary	51.1382	44.6787	38.5562	32.5873
Gray	0.4999	2.4795	9.8695	40.5180	Gray	51.1418	44.1871	38.1878	32.0543
Inv. gray	0.5001	2.5086	9.9822	39.9203	Inv. gray	51.1404	44.1365	38.1385	32.1189
Best	0.4356	1.9426	8.0661	32.1796	Best	51.7403	45.2469	39.0642	33.0550
MSE after OPAP process Baboon image					**PSNR after OPAP process Baboon image**				
Binary	0.4997	1.4990	5.5185	21.5684	Binary	51.1440	46.3728	40.7126	34.7926
Inv. binary	0.5003	1.5061	5.4880	21.3119	Inv. binary	51.1382	46.3522	40.7367	34.8446
Gray	0.4999	1.4939	5.4679	21.5678	Gray	51.1418	46.3876	40.7526	34.7927
Inv. gray	0.5001	1.5041	5.5061	21.4189	Inv. gray	51.1404	46.3581	40.7223	34.8228
Best	0.4356	1.4162	5.1879	20.2704	Best	51.7403	46.6194	40.9809	35.0622

Table 3: Comparison of MSE values with other methods for full embedding capacity in Lena, Baboon, Gandhi and Temple

	MSE for Lena image					MSE for Baboon image			
K bit embedding	1	2	3	4	K bit embedding	1	2	3	4
Simple LSB	0.4971	2.2100	9.0421	36.6052	Simple LSB	0.4997	2.1849	9.1081	36.1049
Best	0.4655	1.9493	8.0310	32.8100	Best	0.4674	1.9426	8.0661	32.1796
Chan and Cheng (2004)	0.4971	1.4952	5.5365	21.5915	Chan and Cheng (2004)	0.4674	1.4979	5.5380	21.4731
Thien and Lin (2003)	0.5009	1.4990	5.4984	21.5549	Thien and Lin (2003)	0.4674	1.4989	5.5392	21.4849
Yang (2008)	0.4655	1.4942	5.4370	20.9467	Yang (2008)	0.4674	1.4952	5.4381	20.9991
Proposed [*256]	0.4655	1.4585	5.3683	20.9985	Proposed [*256]	0.4674	1.4599	5.3741	20.9741
Proposed [*64]	0.4331	1.4091	5.2095	20.3823	Proposed [*64]	0.4356	1.4162	5.1879	20.2704
	MSE for Gandhi image					MSE for Temple			
K bit embedding	1	2	3	4	K bit embedding	1	2	3	4
Simple LSB	0.5001	2.1925	9.0976	35.9473	Simple LSB	0.5005	2.2238	8.9121	36.4952
Best	0.4429	2.0774	8.6089	34.2189	Best	0.4649	2.0845	8.4842	34.4149
Chan and Cheng (2004)	0.4989	1.4957	5.5090	21.4870	Chan and Cheng (2004)	0.4649	1.5027	5.4848	21.5683
Thien and Lin (2003)	0.4999	1.4984	5.5134	21.66132	Thien and Lin (2003)	0.4649	1.5042	5.5152	21.6500
Yang (2008)	0.4989	1.4870	5.4663	21.4627	Yang (2008)	0.4649	1.5009	5.4839	21.4678
Proposed [*256]	0.4670	1.4671	5.3765	20.9625	Proposed [*256]	0.4649	1.4507	5.3470	21.0672
Proposed [*64]	0.4329	1.4140	5.2271	20.5736	Proposed [*64]	0.4354	1.4205	5.1832	20.4412

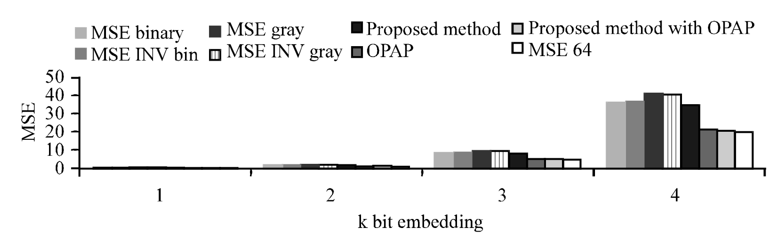

Fig. 4: Comparative MSE values for full embedding capacity on Lena

representations namely binary, inverted binary, gray and inverted gray are employed. Here the secret data, encoded in all the four representations is embedded in a row of the cover image and the MSE is calculated exclusively for each of the four encoding bit representations. Of the representations the one that yields the least MSE is adopted for the respective row. In this way all the four forms of representation are used in each row and the form resulting in the least MSE and PSNR is espoused and the results are given in Fig 4 and 5.

Finally a key is formulated using a code to depict the bit representation format employed in each row which

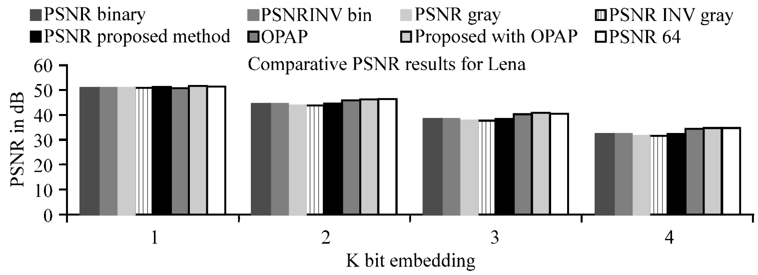

Fig. 5: Comparative PSNR values for full embedding capacity on Lena

again is arcane thereby protecting the stego image from malicious aggressors.

Considering an image of dimensions 256×256, key bits per row is 2. Therefore, in order to account for 256 rows, we get 256×2 = 512 bits. These 512 bits of data form a secret key array. Thus, we can define key-to-data ratio as 512/(256×256×8) = 512/ (65536×8) = 0.00098 = 0.098%. Furthermore one more experiment has been carried out to improve the quality of the stego image by splitting each row into 4 quantum units of 64 pixels. The results are encouraging with slight increase in the key length.

Since the embedding depends upon the Least Mean Square Error which is dynamically determined by the combination of cover image pixels and secret data bits, any attack to recover the data without using the secure key becomes impossible.

COMPLEXITY ANALYSIS

- The DES cryptography system introduces a complexity of 2^{64}
- For 256*256 pixel image, total number of rows will be 256
- These 256 rows can be selected in a random manner in 256! Ways
- In each row one embedding technique out of four is chosen
- If we are embedding k bits in each pixel then
- The total complexity = 2^{64} *256!* 4*8/k
- So total complexity in this case will be 2^{64} *256!* 4 *8/k

For proposed 64 method:

- The DES cryptography system introduces a complexity of 2^{64}
- For 256*256 pixel image, total number of rows will be 256
- These 256 rows can be selected in a random manner in 256! Ways
- Each row is grouped into 4 blocks of 64 pixels.
- For each block a particular technique is selected out of 4
- If we are embedding k bits in each pixel then
- Total complexity is 2^{64} *256!* 4 *8/k
- If we select the blocks in each row in a random manner, there will be 4 blocks and we can select it in 4! Ways and if we select the pixels in a block in a random manner then
- Total complexity is 2^{64} *256!* 4 *4!*8/k
- This security level estimation reveals the of the proposed stego against hackers

CONCLUSION

By simultaneously serving two ultimate requirements of security, i.e., greater imperceptibility (least MSE) and high complexity (cryptic effect created by the choice of row-wise embedding), the proposed technique promises un-tampered transmission and authorized use of secret data. Usage of nominal key length reduces the cost associated with the transport of key over a secure channel. To summarize the key points in this paper:

- An improved image steganographic method has been proposed, implemented and tested called Brownian motion of Binary and Gray-Binary and Gray Bits in Image for stego
- It is a variation of Yang (2008) method with additional choices of Gray code and inverted Gray code along with binary and inverted binary. This provides four choices for the data to be embedded. Thus, it further reduces the effective Mean Square Error to make the stego image more imperceptible and also gives cryptic effect
- A mathematical model has been developed to justify the work
- The worst case Mean square error is derived $MSE_{Proposed} \leq$ (1/2) MSE_{wLSB} and the results are discussed in detail
- This method reduces the MSE by a factor of 1.8, without compromising the data embedding capacity

and marginal improvement in imperceptibility. (In Information hiding with respect to magic triangle capacity, imperceptibility and Robustness). The proposed method will not consider robustness, because robustness will come for watermarking definitely not for spatial domain steganography

- Security analysis has been made to highlight its firmness against hackers
- Total complexity is 2^{64} *256!* 4 *4!*8/k
- The work tested for 10 cover images, due to large data values, only four frequently used cover images are given in the result & discussion. Table 3 highlights the superiority of the proposed method with available literature
- Usage of nominal key length reduces the cost associated with the transport of key over a secure channel about 0.098% of the embedded text

REFERENCES

Al-Frajat, A.K., H.A. Jalab, Z.M. Kasirun, A.A. Zaidan and B.B. Zaidan, 2010. Hiding data in video file: An overview. J. Applied Sci., 10: 1644-1649.

Amirtharajan, R. and R.J.B. Balaguru, 2009. Tri-layer stego for enhanced security-a keyless random approach. Proceedings of the IEEE International Conference on Internet Multimedia Services Architecture and Applications, December 9-11, 2009, Bangalore, India, pp: 1-6.

Amirtharajan, R and R.J.B. Balaguru, 2010. Constructive role of SFC and RGB fusion versus destructive intrusion. Int. J. Comput. Appl., 1: 30-36.

Amirtharajan, R., D. Adharsh, V. Vignesh and R.J.B. Balaguru, 2010. PVD blend with pixel indicator-OPAP composite for high fidelity steganography. Int. J. Comput. Appl., 7: 31-37.

Amirtharajan, R. and R.J.B. Balaguru, 2011. Covered CDMA multi-user writing on spatially divided image. Proceedings of the Wireless ViTAE Conference, February 28-March 3, 2011, IEEE, Chennai, India, pp: 1-5.

Amirtharajan, R., R.R. Subrahmanyam, P.J.S. Prabhakar, R. Kavitha and J.B.B. Rayappan, 2011. MSB over hides LSB: A dark communication with integrity. Proceedings of the 2011 IEEE 5th International Conference on Internet Multimedia Systems Architecture and Application (IMSAA), December 12-14, 2011, Bangalore, Karnataka, India.

Amirtharajan, R. and J.B.B. Rayappan, 2012a. An intelligent chaotic embedding approach to enhance stego-image quality. Inform. Sci., 193: 115-124.

Amirtharajan, R. and J.B.B. Rayappan, 2012b. Inverted pattern in inverted time domain for icon steganography. Inform. Technol. J., 11: 587-595.

Amirtharajan, R., J. Qin and J.B.B. Rayappan, 2012. Random Image Steganography and Steganalysis: Present Status and Future Directions. Inform. Technol. J., 11: 566-576.

Bender, W., D. Gruhl, N. Morimoto and A. Lu, 1996. Techniques for data hiding. IBM Syst. J., 35: 313-336.

Chan, C.K. and L.M. Cheng, 2001. Improved hiding data in images by optimal moderately significant-bit replacement. Electron. Lett., 37: 1017-1018.

Chan, C.K. and L.M. Cheng, 2004. Hiding data in images by simple LSB substitution. J. Pattern Recognit. Soc., 37: 469-474.

Chang, C.C., J.Y. Hsiao and C.S. Chan, 2003. Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy. Pattern Recognit., 36: 1583-1595.

Chang, C.C., T.S. Chen and L.Z. Chung, 2002. A steganographic method based upon JPEG and quantization table modification. Inform. Sci., 141: 123-138.

Cheddad, A., J. Condell, K. Curran and P. Mc Kevitt, 2010. Digital image steganography: Survey and analysis of current methods. Signal Process., 90: 727-752.

Hmood, A.K., B.B. Zaidan, A.A. Zaidan and H.A. Jalab, 2010a. An overview on hiding information technique in images. J. Applied Sci., 10: 2094-2100.

Hmood, A.K., H.A. Jalab, Z.M. Kasirun, B.B. Zaidan and A.A. Zaidan, 2010b. On the Capacity and security of steganography approaches: An overview. J. Applied Sci., 10: 1825-1833.

Janakiraman, S., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012a. Firmware for Data Security: A Review. Res. J. Inform. Technol., 4: 61-72.

Janakiraman, S., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012b. Pixel forefinger for gray in color: A layer by layer stego. Inform. Technol. J., 11: 9-19.

Kumar, P.P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2011. Steg-OFDM blend for highly secure multi-user communication. Proceedings of the 2nd International Conference on Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology, February 28-March 3, 2011, IEEE, Chennai, India, pp: 1-5.

Lin, P.L., C.K. Hsieh and P.W. Huang, 2005. A hierarchical digital watermarking method for image tamper detection and recovery. Pattern Recognit., 38: 2519-2529.

Petitcolas, F.A.P., R.J. Anderson and M.G. Kuhn, 1999. Information hiding-a survey. Proc. IEEE, 87: 1062-1078.

Qin, J., X. Sun, X. Xiang and Z. Xia, 2009. Steganalysis based on difference statistics for LSB matching steganography. Inform. Technol. J., 8: 1281-1286.

Qin, J., X. Xiang and M.X. Wang, 2010. A review on detection of LSB matching steganography. Inf. Technol. J., 9: 1725-1738.

Rajagopalan, S., R. Amirtharajan, H.N. Upadhyay and J.B.B. Rayappan, 2012. Survey and analysis of Hardware Cryptographic and steganographic systems on FPGA. J. Applied Sci., 12: 201-210.

Schneier, B., 2007. Applied Cryptography: Protocols, Algorithm and Source Code in C. 2nd Edn., Wiley, India.

Stefan, K. and A. Fabin, 2000. Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, London, UK.

Thenmozhi, K., P. Praveenkumar, R. Amirtharajan, V. Prithiviraj, R. Varadarajan and J.B.B. Rayappan, 2012. OFDM+CDMA+Stego = Secure Communication: A Review. Res. J. Inform. Technol., 4: 31-46.

Thien, C.C. and J.C. Lin, 2003. A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function. Pattern Recog., 36: 2875-2881.

Wang, C.M., N.I. Wu, C.S. Tsai and M.S. Hwang, 2008. A high quality steganographic method with pixel-value differencing and modulus function. J. Syst. Software, 81: 150-158.

Wang, R., C. Lin and J.C. Lin, 2001. Image hiding by optimal LSB substitution and genetic algorithm. Pattern Recognit., 34: 671-683.

Wang, S.H. and Y.P. Lin, 2004. Wavelet tree quantization for copyright protection watermarking. IEEE Trans. Image Proc., 13: 154-165.

Xia, Z., X. Sun, J. Qin and C. Niu, 2009. Feature selection for image steganalysis using hybrid genetic algorithm. Inform. Technol. J., 8: 811-820.

Xiang, L., X. Sun, Y. Liu and H. Yang, 2011. A secure steganographic method via multiple choice questions. Inform. Technol. J., 10: 992-1000.

Yang, C.H., 2008. Inverted pattern approach to improve image quality of information hiding by LSB substitution. J. Patt. Recog. Soc., 41: 2674-2683.

Yen, E. and K. Tsai, 2008. HDWT-based grayscale watermark for copyright protection. Expert Syst. Appl., 35: 301-306.

Yen, E. and L.H. Lin, 2010. Rubik`s cube watermark technology for grayscale images. Expert Syst. Appl., 37: 4033-4039.

Zaidan, B.B., A.A. Zaidan and M.L.M. Kiah, 2011. Impact of data privacy and confidentiality on developing telemedicine applications: A review participates opinion and expert concerns. Int. J . Pharmacol., 7: 382-387.

Zaidan, B.B., A.A. Zaidan, A.K. Al-Frajat and H.A. Jalab, 2010. On the differences between hiding information and cryptography techniques: An overview. J. Applied Sci., 10: 1650-1655.

Zanganeh, O. and S. Ibrahim, 2011. Adaptive image steganography based on optimal embedding and robust against chi-square attack. Inform. Technol. J., 10: 1285-1294.

Zhu, J., R.D. Wang, J. Li and D.Q. Yan, 2011. A Huffman coding section-based steganography for AAC audio. Inform. Technol. J., 10: 1983-1988.