Research Journal of Information Technology

ISSN 1815-7432

Academic Journals Inc.

www.academicjournals.com

Research Journal of Information Technology 4 (4): 124-139, 2012
ISSN 1815-7432 / DOI: 10.3923/rjit.2012.124.139

Pixel Authorized by Pixel to Trace with SFC on Image to Sabotage Data Mugger: A Comparative Study on PI Stego

Rengarajan Amirtharajan and John Bosco Balaguru Rayappan
Department of Electronics and Communication Engineering, School of Electrical and Electronics Engineering, SASTRA University, Tamil Nadu, 613401, India

Corresponding Author: Rengarajan Amirtharajan, Department of Electronics and Communication Engineering, School of Electrical and Electronics Engineering, SASTRA University, Tamil Nadu, 613401, India

ABSTRACT

There is a raging moral combat to secure information from nefarious attackers as infringement of data is alarmingly escalating with valuable secret information being sabotaged, manipulated or even sold. This has made it necessary for development of efficacious information hiding algorithm to prevent info-sabotage by undetectable secret sharing. Steganography has gained the limelight in the recent past and is formidable and belligerent as it involves embedding of secret data in either images, audio, video, etc., so unsuspectingly that even when intercepted cannot provide a hint to the hackers. However, to make it fool proof we cerebrate yet another algorithm in this paper which uses a combination of Hilbert and Moore Space Filling Curve (SFC) and the pixel indicator methodology, a steganographic tool to improve the randomness and cloak the scanning path such the adversary does not even spill on the clandestine information by an accident. Pixel Indicator (PI) technique uses the complexity of the color image which would be split into red, green and blue planes, respectively with one acting as an indicator, depending upon whose last two bits, data would be embedded in the other two planes. Thus instead of employing the usual raster scan, a random space filling curves (SFC) is used for embedding data into the pixels, because of which it would be impossible for eavesdropper to determine the path of embedding of data making it a highly robust system. The added advantages and enhancements provided by this technique can be observed by the readings of Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) that have been obtained on implementing this algorithm.

Key words: Information hiding, space filling curve, pixel indicator method, steganography

INTRODUCTION

Ever since life took birth on this planet, there has been an inexplicable need for human beings to communicate, from the daily conversations to the exchange of clandestine information. And as the need for communication escalated and got satisfied by an equated technological advancement in the field of communication, there has also been an identical scenario in the area of information extraction, espionage and infringement. Thus it arouses the need for normal banking sector to the powerful government to put their surreptitious information behind a barrage of efficacious security systems. These behoved the methods for information hiding to evolve from sheer primitive procedures to top class complex algorithms classified as data hiding (Bender *et al*., 1996; Amirtharajan and Balaguru, 2009, 2010, 2011, 2012a, b; Amirtharajan *et al*., 2012; Cheddad *et al*., 2010).

Data hiding can be stratified into cryptography (Schneier, 2007; Salem *et al*., 2011), steganography and digital water marking (Stefan and Fabin, 2000) over a large basis.

Cryptography involves the coding of the source message into a different form which the sender and receiver can understand. But it lacks the main purpose when it is transmitted as illegal data seekers may be attracted by the coded form and thus the secrecy get lost. Steganography involves the coding of the source message into a different form called cover object, most probably an image, audio or video (Amirtharajan and Balaguru, 2009, 2010, 2011, 2012a, b; Zhu *et al*., 2011; Al-Frajat *et al*., 2010; Amirtharajan *et al*., 2012) such that it can be decoded only by the appropriate receiver with the correct key. Also it overcomes the deficit of cryptography in that; it doesn't attract anyone on its transmission. This usually involves the replacement of the lower bit of each pixel of the cover image with the original message to form the stego image so as to reduce distortion (Thien and Lin, 2003; Chan and Cheng, 2004; Yang, 2008). Watermarking is mainly intended for copy right protection wherein the information to be sent is embedded in the digital signal in a way to verify its authenticity (Abdulfetah *et al*., 2009; Zhang *et al*., 2010; Abdulfetah *et al*., 2010; Zeki *et al*., 2011).

On going through the literature, it has been observed that steganography is one of the most effective methods for data hiding, with its strength being, it satisfies all three requirements of information hiding magic triangle of robustness, imperceptibility and high data payload:

- Robustness : Ability to withstand attacks and maintain integrity
- High payload : Maximum amount of data that can be hidden
- Imperceptibility : Ability to transmit data without attraction or suspicion of illegitimate party

Steganography is found to be the best as it satisfies all the above requirements fairly well than the other two. The usually employed technique in this method involves directly replacing the Least Significant Bit (LSB) of each pixel in the cover object thus causing very less distortion.

The steganographic methods may possibly be categorized into Spatial or Transform domain. In the former spatial domain of the cover objects are used to camouflage the secret data resulting in Stego object. Whereas the later employs the transformed domain of the cover objects are used to hide the clandestine information. The other classification is based on the types of cover object like Video, Audio, Image and Text (Shirali-Shahreza and Shirali-Shahreza, 2008; Al-Azawi and Fadhil, 2010). One more classification is based on the methods used on the chosen cover like Substitution (Amirtharajan and Balaguru, 2009, 2010, 2011, 2012a, b; Amirtharajan *et al*., 2012), Transform domain (Thanikaiselvan *et al*., 2011b) and Spread Spectrum (Kumar *et al*., 2011), Distortion, statistical and new cover generation (Xiang *et al*., 2011). The other side to break steganography called steganalysis are given by Xia *et al*. (2009) and Qin *et al*. (2009, 2010).

Image steganography is extensively used now-a-days through internet (Hmood *et al*., 2010a, b). It is the most common and well known method for high capacity, imperceptibility (Thanikaiselvan *et al*., 2011a, b). It could be classified as Least Significant Bit (LSB) substitution and pixel value differencing. In LSB substitution the least significant k-bits of target pixel in cover image are embedded with message bits but these methods will considerably introduce distortion in Stego image. To improve this, many new optimized LSB approaches have been suggested. Chan and Cheng (2004) proposed a simple LSB substitution method with Optimal Pixel Adjustment Process (OPAP) to reduce the Mean square error. Wang *et al*. (2001, 2008) methods offers high embedding capacity with good imperceptibility using adaptive Least Significant Bit (LSB) substitution along with pixel-value differencing (PVD).

Abbas Cheddad discusses a detailed survey on digital image steganography methods and its Classification (Cheddad *et al*., 2010). It also describes the differences among steganography;

watermarking and encryption and few other reviews on steganography is available (Amirtharajan *et al.*, 2012; Rajagopalan *et al.*, 2012; Janakiraman *et al.*, 2012a,b; Thenmozhi *et al.*, 2012). Pixel indicator based random image stego system proposed by Gutub *et al.* (2008) and Gutub (2010) and exploited by Amirtharajan (Amirtharajan *et al.*, 2010, 2011; Padmaa *et al.*, 2011). The number of bits embedding decided by most significant values (MSBs) (Amirtharajan *et al.*, 2011), furthermore by calculating number of bits embedded through PVD (Amirtharajan *et al.*, 2010) and so on. However, the embedding here is carried out using a simple Raster scan so may be assailable by third parties and information may be easily extracted.

To compensate this glitch, Hilbert or Moore SFC (Amirtharajan and Balaguru, 2009, 2010, 2012a; Zhao and Luo, 2012) based embedding can be used instead of Raster Scan (Thien and Lin, 2003; Chan and Cheng, 2004; Yang, 2008). The specific scan used can be kept confidential between sender and recipient and if hacked by an eavesdropper, would be nearly impossible to track out any information or find a pattern of data embedding. The entire cover image here would first have to be segregated into smaller blocks and then data should be embedded as per requirement. Also, an added advantage would be the inclusion of the Pixel Indicator (PI) technique (Amirtharajan *et al.*, 2010, 2011; Padmaa *et al.*, 2011), where one channel would be determined as an indicator and the specified amount of bits by user (say k bits) are then embedded in the other two channels depending upon the last two bits of the indicator channel.

In the spatial domain it is seen that LSB substitution is done mostly by Raster scan (Thien and Lin, 2003; Chan and Cheng, 2004; Yang, 2008) and sometimes by random scans (Amirtharajan and Balaguru, 2009, 2010, 2012b; Amirtharajan *et al.*, 2012) to embed the information to help up the capacity, simplicity and time of implementation. In these random approaches all the pixels of the cover image have not been used for hiding secret data which in turn affect the payload besides good imperceptibility.

THE PROPOSED METHOD

Space filling curves (SFC) is a one dimensional curve which traverses through each and every point within a two dimensional space or image (Amirtharajan and Balaguru, 2009, 2010, 2012b; Zhao and Luo, 2012). SFC scans a pixel array which has a size of M×N pixels and while scanning, it will not retain the same direction but will turn around to embrace all the pixels at least and at most once. Hence the unpredictable traversing path of SFC through the image has been chosen to hide the secret message in the cover. In this scheme, both the sender and receiver can adapt a particular SFC so that there is no need to communicate the key and also providing a complicated traversing path for k-bit embedding which does not require any key.

Before considering the entire cover image for secret bit embedding, block of 4×4 pixels has been taken to implement Hilbert SFC and Moore SFC traversing path based stego technique by adapting a common traversing path for both the sender and receiver. After performing this process for a single 4×4 block, it has been extended to the full image by considering it as multiple of 4×4 blocks to cover up the entire $2^8 \times 2^8 \times 3$ pixels.

The traversing paths to embed secret data, based on Hilbert SFC and Moore SFC in gray scale are shown in Fig. 1 and 2 and Hilbert scan and Moore SFC in RGB in Fig. 3 and 4. Block diagram of the proposed embedding and extraction is shown in Fig. 5.

And the additional pixel indicator method used here involves selection of a channel as an indicator first and then depending upon the last two bits of the indicator, data is further embedded in the other two channels as given in Table 1.

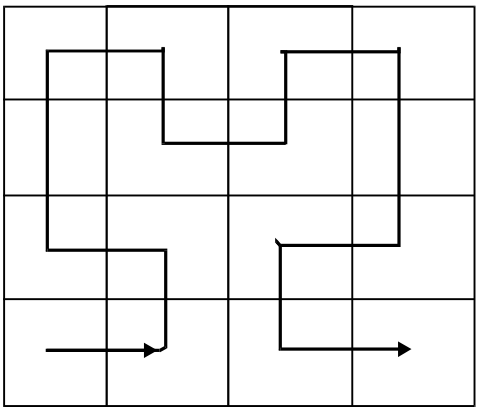

Fig. 1: Hilbert scan

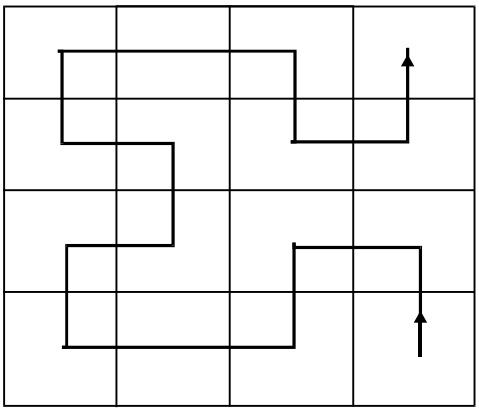

Fig. 2: Moore curve

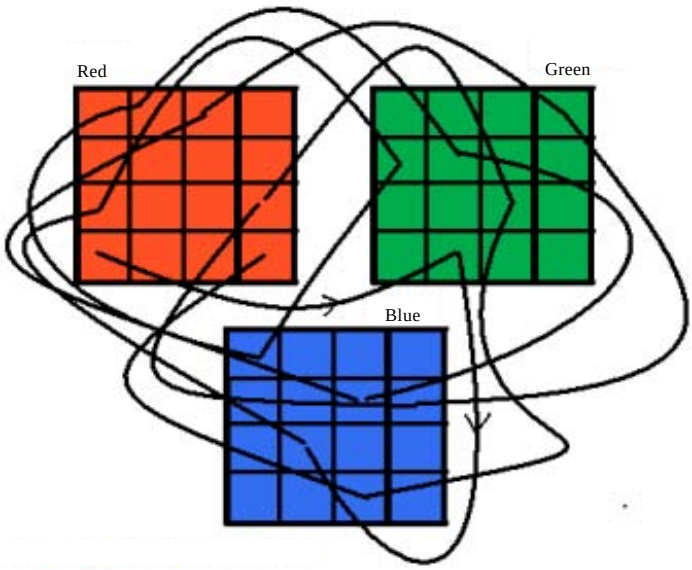

Fig. 3: Hilbert scan in RGB

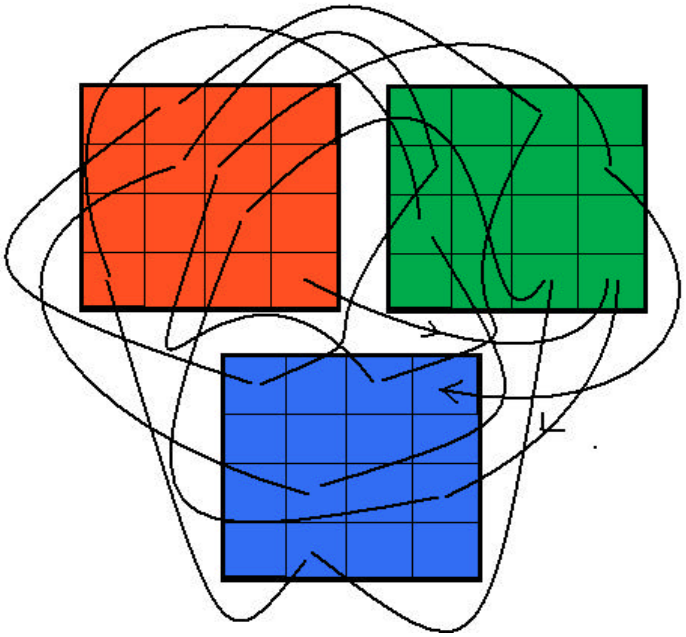

Fig. 4: Moore curve in RGB

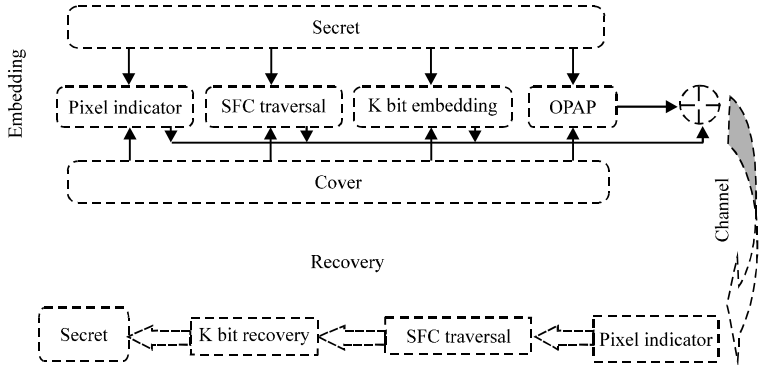

Fig. 5: Block diagram of the proposed embedding and extraction

Table 1: Meaning of indicator values

Indicator	Channel 1	Channel 2
00	No data embedded	No data embedded
01	No data embedded	k bits of data embedded
10	k bits of data embedded	No data embedded
11	k bits of data embedded	k bits of data embedded

Thus if 'red' plane is selected as an indicator and its last two bits be '11' then k bits (as defined by user) of data are embedded in the blue and the green channel, respectively. Furthermore, the PI technique can be employed in three flexible methods:

- Red being the standard indicator for the entire cover image
- User-defined indicator
- Cyclic indicator for the cover

Case 1: k bit LSB with tri-colour random image steganography

Embedding algorithm:

Inputs: Cover image (C), secret data (D)
Output: Stego image (S)

1. Conversion of secret data (D) into binary
2. Split the cover image C into red (R), green (G) and blue Planes(B)
3. For each pixel traversed by chosen SFC in R, do the following:
   3.1. Let b [0] = LSB of the current pixel in R
   3.2. Let b [1] = Next LSB of the current pixel in R
   3.3. If b = 00 then
        Go to next pixel.
   Else if b = 01 then
        Call k bit LSB Embedding to embed secret data in current pixel of G.
   Else if b =10 then
        Call k bit LSB Embedding to embed secret data in current pixel of B.
   Else
Call k bit LSB embedding to embed secret data in current pixel of both G and B.
   1.4. If all secret data is embedded, then
        Go to step-4
   2. Store the resulting image as stego image (S) after applying OPAP

Recovery algorithm:

Input: Stego image (S)
Output: Secret data (D)

1. Split the stego image S into red (R), green (G) and blue (B) Planes
2. For each pixel traversed by chosen SFC in R, do the following:
   2.1. Let b [0] = LSB of the current pixel in R
   2.2. Let b [1] = Next LSB of the current pixel in R
   2.3. If b = 00 then
        Go to next pixel.
   Else if b = 01 then
        Call k bit LSB recovery to recover secret data from current pixel of G.
   Else if b = 10 then
        Call k bit LSB recovery to recover secret data from current pixel of B.
   Else
        Call k bit LSB recovery to recovery secret data from current pixel of both G and B.
3. Store the resulting recovered data as secret data (D)

Case 2: k bit LSB with custom-indicator-plane tri-color random image steganography

Embedding algorithm:
Inputs: Cover image (C), indicator-plane index (I) and secret data (D)
Output: Stego image (S)

1. Convert the secret data (D) into binary format
2. Split the cover image C into red, green and blue planes
3. If I = 1 then

Case 2: Continued

P[1] = R, P[2] = G, P[3] = B
Else if I = 2, then
P[1] = G, P[2] = R, P[3] = B
Else if I = 3, then
P[1] = B, P[2] = R, P[3] = G

4. For each pixel traversed by chosen SFC in P[1], do the following:
   4.1. Let b[0] = LSB of the current pixel in P[1]
   4.2. Let b[1] = Next LSB of the current pixel in P[1]
   4.3. If b = 00 then
        Go to next pixel.
        Else if b = 01 then
        Call k bit LSB embedding to embed secret data in current pixel of P[2].
        Else if b = 10 then
        Call k bit LSB embedding to embed secret data in current pixel of P[3].
        Else
        Call k bit LSB embedding to embed secret data in current pixel of both P[2] and P[3].
   2.4. If all secret data is embedded, then Go to step-5
3. Store the resulting image as Stego Image (S) after applying OPAP

Recovery algorithm:

Input: Stego image (S), indicator-plane index (I)

Output: Secret data (D)

1. Split the stego image S into red, green and blue planes. (R, G and B, respectively)
2. If I = 1 then:
   P[1] = R, P[2] = G, P[3] = B
   Else if I = 2, then:
P[1] = G, P[2] = R, P[3] = B
   Else if I = 3, then:
P[1] = B, P[2] = R, P[3] = G
3. For each pixel traversed by chosen SFC in P[1], do the following:
   3.1. Let b[0] = LSB of the current pixel in P[1]
   3.2. Let b[1] = Next LSB of the current pixel in P[1]
   3.3. If b = 00 then
        Go to next pixel.
        Else if b = 01 then
        Call k bit LSB recovery to recover secret data from current pixel of P[2].
        Else if b = 10 then
        Call k bit LSB recovery to recover secret data from current pixel of P[3].
        Else
        Call k bit LSB recovery to recover secret data in current pixel of both P[2] and P[3].
4. Store the resulting data as secret data (D).

Case 3: k bit LSB with cyclic-indicator-plane tri-colour random image steganography

Embedding algorithm:

Inputs: Cover image (C), secret data (D)

Output: Stego image (S) with secret data embedded in it.

1. Conversion of secret data (D) into binary representation.
2. Split the cover image C into red, green and blue planes.
3. Let index I = 1.

Case 3: Continued

4. For each pixel in P[1], do the following:
   4.1. If (i mod 3) = 1 then,
        I[i] = 1
   Else if (i mod 3) = 2 then,
        I[i] = 2
   Else
        I[i] = 3
   4.2. Set i = i+1
5. Let index j = 0
6. For each pixel traversed by chosen SFC in P[1], do the following:
   6.1. If I[j] = 1 then,
        P[1] = R[i], P[2] = G[i], P[3] = B[i]
   Else if I[j] = 2, then

P[1] = G[i], P[2] = R[i], P[3] = B[i]
   Else if I[j] = 3, then

P[1] = B[i], P[2] = R[i], P[3] = G[i]

   6.2. Let b[0] = LSB of P[1]
   6.3. Let b[1] = Next LSB of P[1]
   6.4. If b = 00 then
        Go to next pixel
   Else if b = 01 then
        Call k bit LSB Embedding to embed secret data in P[2].
   Else if b =10 then
        Call k bit LSB Embedding to embed secret data in P[3].
   Else
        Call k bit LSB Embedding to embed secret data in both P[2] and P[3].
   6.5. If all secret data is embedded, then
        Go to step-7
   Else
        j = j+1
7. Store the resulting image as Stego Image (S) after applying OPAP.

Recovery algorithm:

Input: Stego image (S)
Output: Secret data (D)

1. Split the stego image S into red, green and blue planes. (R, G and B, respectively)
2. Let index I = 1.
3. For each pixel in P[1], do the following:
   4.1. If (i mod 3) = 1 then,
        I[i] = 1
   Else if (i mod 3) = 2 then,
        I[i] = 2
   Else
        I[i] = 3
   4.2. Set i = i+1
4. Let index j = 0

Case 3: Continued

5. For each pixel traversed by chosen SFC in P[1], do the following:
   5.1. If I[j] = 1 then,
        P[1] = R[i], P[2] = G[i], P[3] = B[i]
   Else if I[j] = 2, then

P[1] = G[i], P[2] = R[i], P[3] = B[i]
   Else if I[j] = 3, then

P[1] = B[i], P[2] = R[i], P[3] = G[i]

   5.2. Let b[0] = LSB of P[1]
   5.3. Let b[1] = Next LSB of P[1]
   5.4. If b = 00 then
        Go to next pixel
   Else if b = 01 then
        Call k bit LSB Recovery to embed secret data from P[2].
   Else if b = 10 then
        Call k bit LSB Recovery to embed secret data from P[3].
   Else
        Call k bit LSB Recovery to embed secret data from both P[2] and P[3].
6. Store the resulting data as Secret Data (D)

RESULTS AND DISCUSSION

To evaluate the performance of our proposed method several experiments are performed. Figure 6 shows embedding and extraction flowchart. Four colour images are taken with size 256×256 as cover images which are shown in Fig. 7. Initially for varying k = 1, 2, 3 and 4 bit embedding performed and the values for K= 4 bit embedding is given in Table 2, 3 and 4 for method 1, 2 and 3 and the corresponding Stego covers for method 4 are given in Fig. 8.

To evaluate the performance of the proposed system MSE and PSNR have been computed for all the three methods. Peak Signal to Noise Ratio(PSNR) and Mean Square Error (MSE).

The PSNR is calculated using the equation:

$$PSNR = 10\log_{10}\left(\frac{I_{max}^2}{MSE}\right)dB \quad (1)$$

where, I_{max} is the intensity value of each pixel which is equal to 255 for 8 bit gray scale images.

The MSE is calculated by using the Eq. 2 given below:

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

where, M and N denote the total number of pixels in the horizontal and the vertical dimensions of the image $X_{i,j}$ represents the pixels in the original image and $Y_{i,j}$ represents the pixels of the stego-image.

From Table 2 it's observed that Mahatma Gandhi cover has the maximum embedding capacity and minimum PSNR values. There is no change in the red plane histogram Fig. 9.

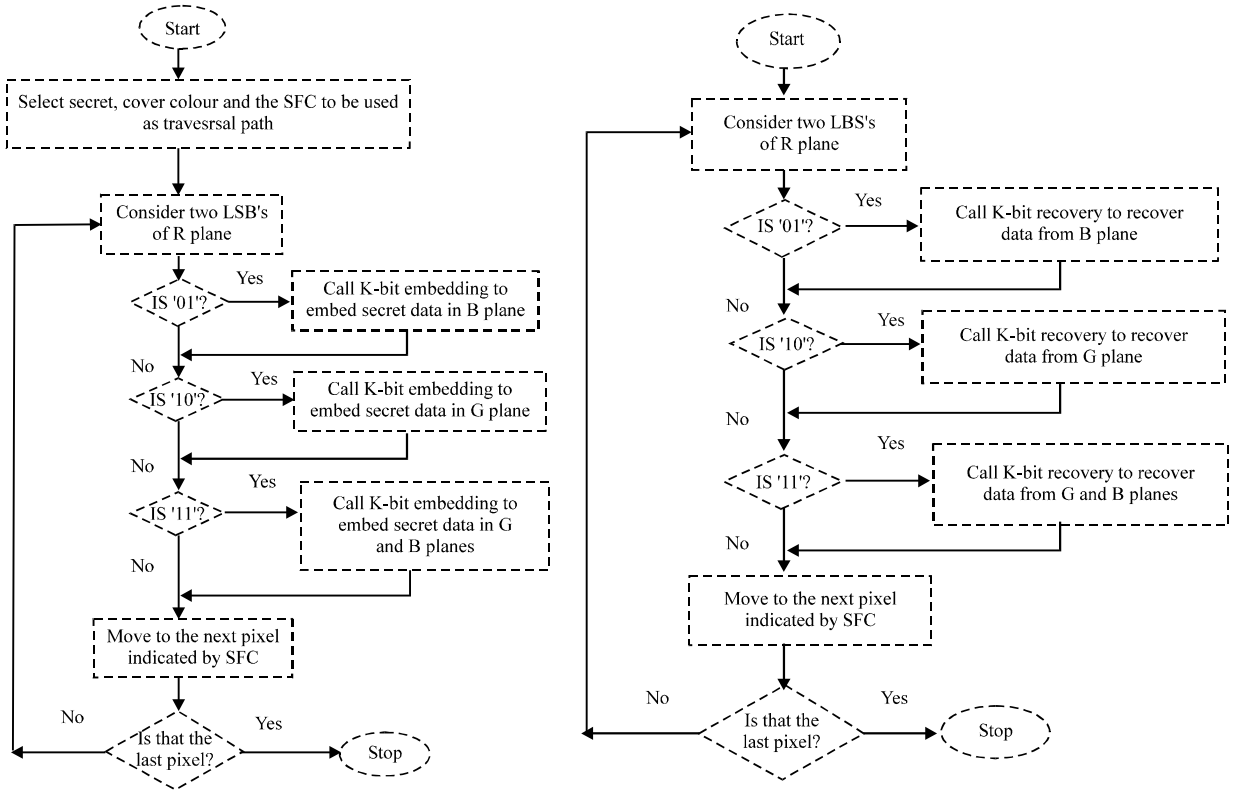

Fig. 6: Embedding and extraction flowchart

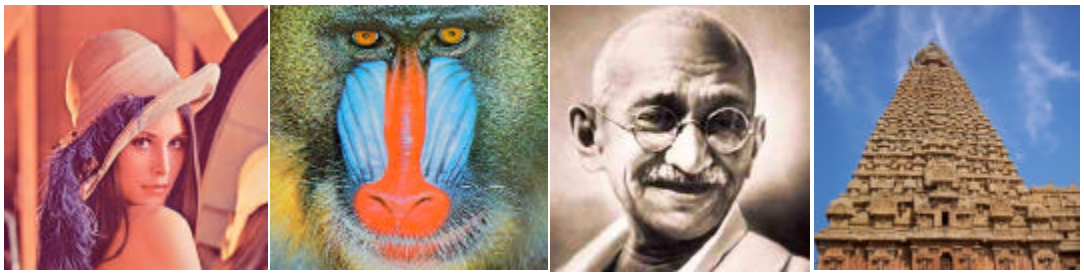

Fig. 7(a-d): Cover images; (a) Lena, (b) Baboon, (c) Gandhi and (d) Temple

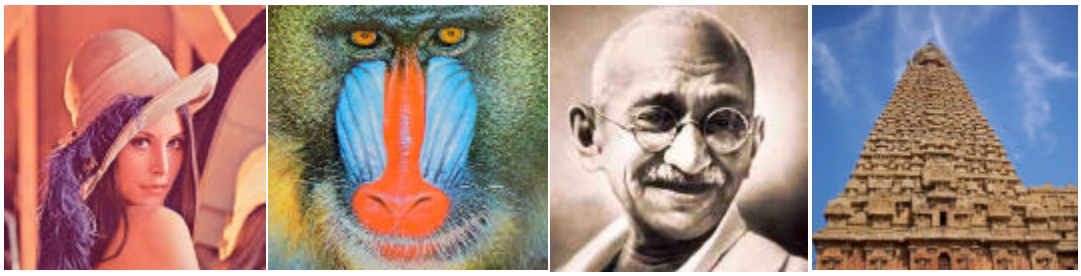

Fig. 8(a-d): Stego output k = 4 bit for maximum embedding capacity; (a) Lena, (b) Baboon, (c) Gandhi and (d) Temple

Table 2: MSE PSNR values for method 1 k = 4 bit for maximum embedding capacity

Cover images	Lena	Baboon	Gandhi	Temple
MSE red plane	0	0	0	0
MSE green plane	3.5752	3.5317	3.6697	3.5102
MSE blue plane	3.5516	3.5539	3.6543	3.548
PSNR red plane	8	8	8	8
PSNR green plane	42.5978	42.651	42.4845	42.6775
PSNR blue plane	42.6266	42.6237	42.5027	42.6309
No. of bits in green plane	132296	131576	132836	129768
No. of bits in blue plane	131332	131364	134032	131440
Total No. of bits embed	263628	262940	266868	261208

Table 3: Method 2 MSE PSNR values for k = 4 bit embedding and green plane as indicator (User choice)

Cover images	Lena	Baboon	Gandhi	Temple
MSE red plane	3.1329	3.5176	3.6542	3.9821
MSE green plane	0	0	0	0
MSE blue plane	3.9139	3.8921	3.9653	3.8215
PSNR red plane	42.1349	42.928	42.5484	42.8971
PSNR green plane	8	8	8	8
PSNR blue plane	42.9873	42.8151	42.6754	42.9271
No. of bits in green plane	131044	130628	131520	130664
No. of bits in blue plane	131684	130984	129136	132092
Total No. of bits embed	262728	261612	260656	262756

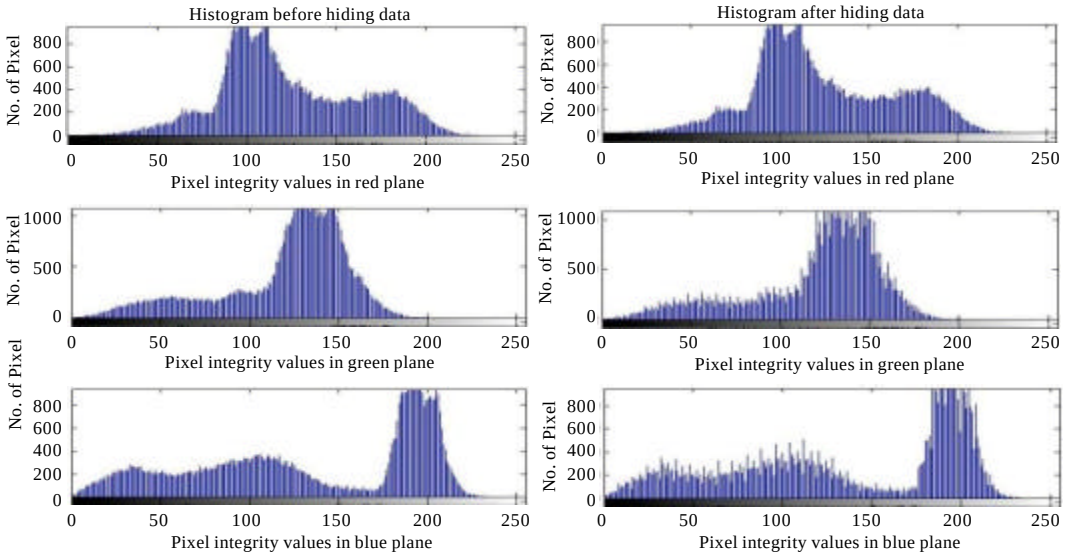

Fig. 9: Method 1 RGB histogram for temple, no change in red plane

From Table 3 it's observed that Thanjavur Big Temple cover has the maximum embedding capacity and minimum PSNR values. There is no change in the green plane histogram (Fig. 10).

From Table 5 it's observed that Baboon cover has the maximum embedding capacity and minimum PSNR values and the error are evenly distributed in all the planes (Fig. 11).

Table 4: Method 3 MSE PSNR values for k = 4 bit embedding and cyclic indicator

Cover images	Lena	Baboon	Gandhi	Temple
MSE red plane	2.4387	2.3702	2.5728	2.3143
MSE green plane	2.3066	2.3255	2.4212	2.3095
MSE blue plane	2.3389	2.3619	2.3595	2.3764
PSNR red plane	44.259	44.3829	44.0267	44.4866
PSNR green plane	44.501	44.4657	44.2904	44.4957
PSNR blue plane	44.441	44.3981	44.4026	44.3716
No. of bits in red plane	85460	85652	85112	85768
No. of bits in green plane	85596	85804	86552	84956
No. of bits in blue plane	85720	85652	85132	86436
Total No. of bits embed	256776	257108	256796	257160

Table 5: Comparative estimation parameters of the proposed embedding scheme-III

		Channel I red		Channel II green		Channel III blue			
Cover image	Methods	MSE	PSNR	MSE	PSNR	MSE	PSNR	BPP (bits per pixel)	Maximum embedding capacity
Lena	Proposed	2.4387	44.259	2.3066	44.501	2.3389	44.441	3.9181	256776
	Padma[10]	1.227	47.24	1.3641	46.782	1.02	48.045	2.114	138549
	Amir[11]	1.68	45.89	1.57	46.18	1.52	46.31	2.13	139592
	Amir[12]	1.2906	47.0230	1.2374	47.2059	1.2049	47.3213	2.3139	151645
Baboon	Proposed	2.3702	44.3829	2.3255	44.4657	2.3619	44.3981	3.9232	257108
	Padma[10]	4.065	42.04	4.002	42.108	4.2847	41.812	3.657	239262
	Amir[11]	2.61	43.96	2.65	43.88	2.72	43.77	2.51	164496
	Amir[12]	1.5540	46.2162	1.5544	46.2151	1.5904	46.1157	2.3975	157121
Gandhi	Proposed	2.5728	44.0267	0.5798	44.2904	2.3595	44.4026	3.9184	256796
	Padma[10]	1.348	46.83	2.4212	47.025	1.2478	47.169	2.028	132945
	Amir[11]	1.34	46.83	1.30	47.07	1.24	47.15	2.07	135660
	Amir[12]	3.2721	42.9825	3.2944	42.9530	3.1355	43.1677	3.0880	202377
Temple	Proposed	2.3143	44.4866	2.3095	44.4957	2.3764	44.3716	3.9240	257160
	Padma[10]	1.853	45.45	1.766	45.662	1.632	46.003	2.352	154409
	Amir[11]	1.85	45.45	1.76	45.66	1.63	46.00	2.30	150732
	Amir[12]	1.1159	47.6547	1.1062	47.6924	1.1240	47.6232	2.4659	161604

COMPARATIVE RESULTS

Complexity analysis: The complexity of the proposed system is also good. This method divides the cover image into 4×4 blocks. So there will be 4096 such blocks. so by taking one red, green, blue 4×4 block and use Hilbert or Moore SFC for traversing and if they are selected randomly then there are 4096!*4096!*4096! ways to select one red, green, blue block. Pixel indicator additional increase the randomness assuming 25% on each case like 00, 01, 10 and 11 which decides the embedding capacity with secret data encrypted with AES. Then the complexity would be 4096!^3*.5*2^128 Furthermore, first the indicator can be selected in three ways. If the LSBs of the first indicator is not zero then embedding is done.

- So the probability of embedding is 3/4
- The cover image is divided into 1024×4 blocks of 4×4 pixels
- Two different scan paths for random embedding process can be adopted Hilbert or Moore SFC 2 ways

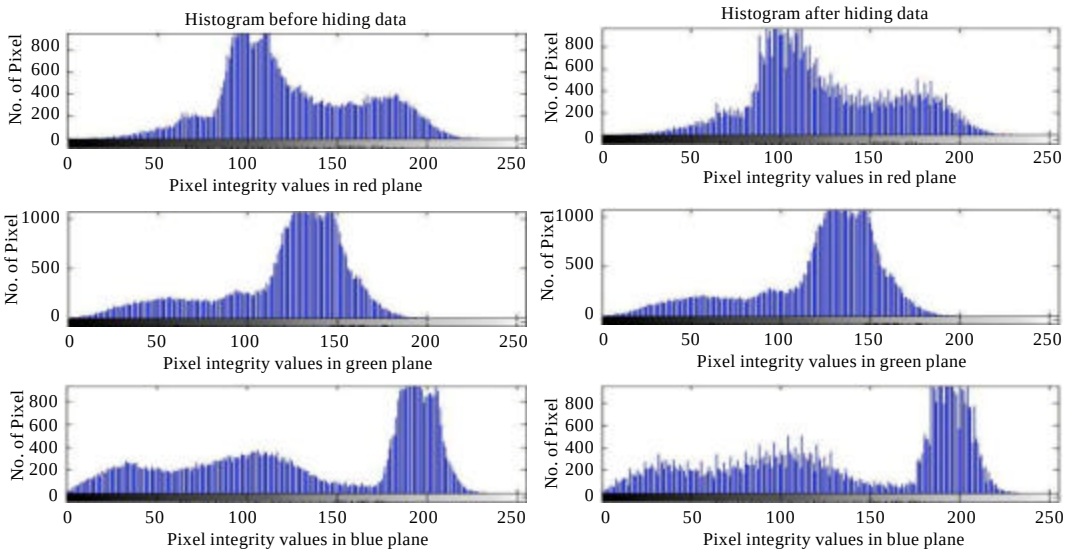

Fig. 10: Method 2 RGB histogram for temple, no change in green plane

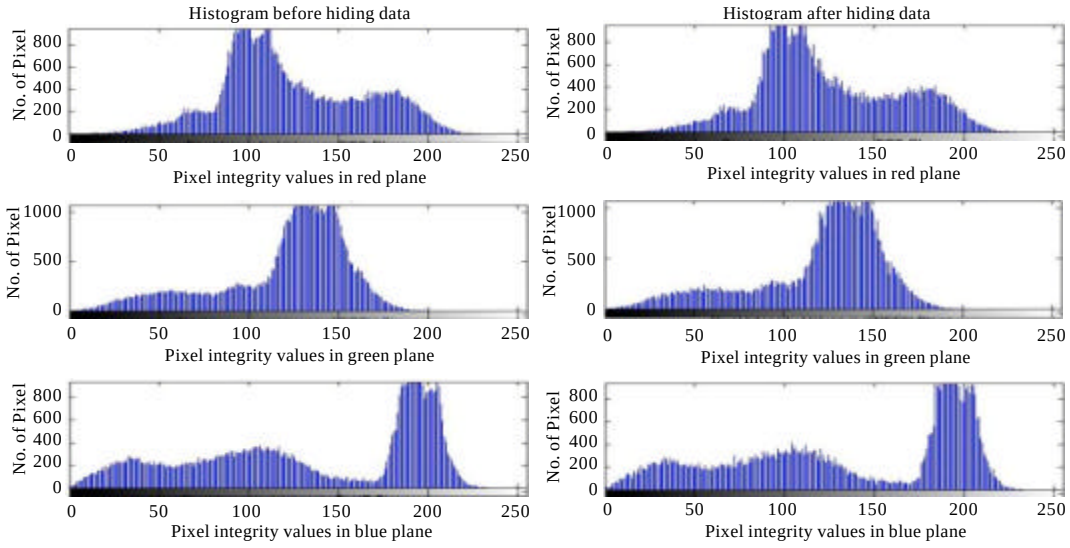

Fig. 11: Method 3 histogram for temple, errors are evenly distributed

- The PRN scheme has been adopted to select the block of 1024*4. = NpR = N!/(N-r)! = 1024*4p1024*4 = 4096!
- The starting point of each 4×4 SFC has been changed. 16 different paths can be selected by having 16 different starting points in the selected SFC
- So total complexity for embedding/ or hacking at least one bit is 2^128*3*4/3*2*4096!^3*16
- Which is far better than Padmaa *et al*. (2011) 2^64*7*0.5*2*2 and Amirtharajan *et al*. (2010) is only 2^64*3/4*7 and Amirtharajan *et al*. (2011) is 2^128*3*2*4/3*(8+8/3+8/3+8)

CONCLUSION

Thus it is seen there are myriad of methodologies for incorporating data security and impregnability in our day-to-day life. While there may be a plethora of techniques to incur the basic objective of information hiding involving imperceptibility, robustness, high payload and security, it is seen that not all of them may obtain it simultaneously. Especially the robustness factor seems to be compromised with the routine techniques being employed, with the hackers too being continually educated with them. Thus this proposed methodology brings about a fantabulous shift from the mundane methods, by involving Hilbert or Moore space filling curves for traversing through the pixels with more integrity obtained by employing pixel indicator (PI) concurrently. The added advantage in the proposed method is the ability to determine the indicator; either by the user or a cyclic one or a fixed one. To re-enforce the entire objective of security, the data is scrambled using DES before embedding. With 30 dB being fixed as the threshold PSNR value for human visual system, it is seen that this technique on embedding causes very little degradation and provides high imperceptibility 42 dB and above for all the cases. Thus it is observed that this algorithm involving random curves such as Hilbert or Moore SFC for traversing along with PI, is a boon for transfer of information, because when it is intercepted by hackers too, is an absolute nightmare to decode as they wouldn't have a clue as where as to start from. Overall, this method caters to all the requirements of information hiding in a single package.

REFERENCES

Abdulfetah, A., X. Sun and H. Yang, 2009. Quantization based robust image watermarking in DCT-SVD domain. Res. J. Inform. Technol., 1: 107-114.

Abdulfetah, A.A., X. Sun, H. Yang and N. Mohammad, 2010. Robust adaptive image watermarking using visual models in DWT and DCT domain. Inform. Technol. J., 9: 460-466.

Al-Azawi, A.F. and M.A. Fadhil, 2010. Arabic text steganography using Kashida extensions with Huffman code. J. Applied Sci., 10: 436-439.

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

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.

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, 2004. Hiding data in images by simple LSB substitution. J. Pattern Recognit. Soc., 37: 469-474.

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.

Gutub, A., M. Ankeer, M. Abu-Ghalioun, A. Shaheen and A. Alvi, 2008. Pixel indicator high capacity technique for RGB image based steganography. Proceedings of the 5th IEEE International Workshop on Signal Processing and its Applications, March 18-20, 2008, Sharjah, UAE.

Gutub, A.A.A., 2010. Pixel indicator technique for RGB image steganography. J. Emerging Technol. Web Intel., 2: 56-64.

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.

Padmaa, M., Y. Venkataramani and R. Amirtharajan, 2011. Stego on 2^n:1 platform for users and embedding. Inform. Technol. J., 10: 1896-1907.

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.

Salem, Y., M. Abomhara, O.O. Khalifa, A.A. Zaidan and B.B. Zaidan, 2011. A review on multimedia communications cryptography. Res. J. Inform. Technol., 3: 146-152.

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

Shirali-Shahreza, M. and S. Shirali-Shahreza, 2008. High capacity Persian/Arabic text steganography. J. Applied Sci., 8: 4173-4179.

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

Thanikaiselvan, V., P. Arulmozhivarman, R. Amirtharajan and J.B.B. Rayappan, 2011a. Wave (let) decide choosy pixel embedding for stego. Proceedings of the International Conference on Computer, Communication and Electrical Technology, March 18-19, India, pp: 157-162.

Thanikaiselvan, V., S. Kumar, N. Neelima and R. Amirtharajan, 2011b. Data battle on the digital field between horse cavalry and interlopers. J. Theor. Applied Inform. Technol., 29: 85-91.

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.

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.

Zeki, A.M., A.A. Manaf, A.A. Ibrahim and M. Zamani, 2011. A robust watermark embedding in smooth areas. Res. J. Inform. Technol., 3: 123-131.

Zhang, Y., Z.M. Lu and D.N. Zhao, 2010. A blind image watermarking scheme using fast Hadamard transform. Inform. Technol. J., 9: 1369-1375.

Zhao, Z. and H. Luo, 2012. Reversible data hiding based on Hilbert curve scan and histogram modification. Inform. Technol. J., 11: 209-216.

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.