Research Journal of Information Technology

ISSN 1815-7432

Academic Journals Inc.

www.academicjournals.com

Research Journal of Information Technology 5 (3): 304-316, 2013
ISSN 1815-7432 / DOI: 10.3923/rjit.2013.304.316
© 2013 Academic Journals Inc.

Kubera Kolam: A Way for Random Image Steganography

Rengarajan Amirtharajan, Krishnamourthy Karthikeyan, Malligaraj Malleswaran and J.B.B. Rayappan
Department of Electronics and Communication Engineering, School of Electrical and Electronics Engineering, SASTRA University, Thanjavur, 613 401, India

Corresponding Author: Rengarajan Amirtharajan, Department of Electronics and Communication Engineering, School of Electrical and Electronics Engineering, SASTRA University, Thanjavur, 613 401, India

ABSTRACT

The developments in expertise and internet fruition have amplified dependence on systems and IT abided by the demand to secure the same. This intriguing effort in electronic world has unfolded a new boulevard called cyber defense. In this world of cyber hacking, information security plays a vital role. Primitive techniques though are old but are very helpful in giving a perfect outline of things away from human thoughts. One such technique is the Magic Square Method, wherein the brilliant orientation of the numbers leads to a perfect matrix useful for any mathematical developments. Block Segmentation in this study involves two kolams firstly the Kubera Kolam is the magic square that is employed and incorporated for introducing the randomization. Further is the Pulli Kolam, square for acting as the symmetric key for giving the precise bits to be hidden. Further on, modifiable pixel indicator gets slightly altered from the rudiments and is used to accomplish a much efficient and effective indicator liken the conventional one.

Key words: Cryptography, AES, steganography, pixel indicator, improved pixel indicator

INTRODUCTION

Communication being inevitable of daily routine has produced uprising right from the Stone period. Technology germinates for our own good, thus becomes better for our own best. Communicating technologies augments at each and every pace, say as of sheer contact to digital communication and to video tête-à-tête to even live discussion. This is all because of some brilliant brains. But this brilliant piece of work is illegally destroyed and killed by many computer professionals (as they call) but the hackers. These marvelous reforms in the technology are expanding, but not exponentially increasing for the same reason. However, this could be raised by only one means and that is information security (Cheddad *et al*., 2010; Salem *et al*., 2011; Schneier, 2007; Stefan and Fabin, 2000). Any form of life needs security so does the data. Myriad methods since the past decade are being discovered (Amirtharajan and Rayappan, 2012a-d; Amirtharajan *et al*., 2012; Bender *et al*., 1996; Cheddad *et al*., 2010; Janakiraman *et al*., 2012a, b; Rajagopalan *et al*., 2012; Thenmozhi *et al*., 2012) to save the data but in vain every method, hackers find better means to crack into the data (Schneier, 2007; Qin *et al*., 2010).

The need for a fulltime settlement of security arose and cryptography emerged as a powerful tool then, the algorithms created were almost the best, not de-cryptable and was providing the feature of protection of intermediate changing means (Salem *et al*., 2011; Schneier, 2007). Nevertheless, the only drawback to it was the image or content that was encrypted might had completely deformed, wherein one easily figures out that there is some manipulation done on that

piece of data (Schneier, 2007; Qin *et al.*, 2010; Zaidan *et al.*, 2010). Now, this was not advisable for the reason that any hacker could poach into the data owing to the fact that something has changed.

This led to the rise of the super power technology of hiding data, Steganography (Al-Azawi and Fadhil, 2010; Al-Frajat *et al.*, 2010; Xiang *et al.*, 2011; Zanganeh and Ibrahim, 2011; Zhao and Luo, 2012; Zhu *et al.*, 2011). Till date flawless, every feature incorporated proved to be the best and many more to be brought (Gutub, 2010; Luo *et al.*, 2011; Mohammad *et al.*, 2011; Padmaa *et al.*, 2011; Thanikaiselvan *et al.*, 2011; Zhao and Luo, 2012). The concept is of the usage of cover image to cover the secret data and create an illusion of nothing being in it. This piece of algorithm in the initial stage was done minimal number of randomizations in the image and data. But of late, several randomizations are made introduced in every fortnight, which for any hacker to actually know the piece of data after unveiling all the randomizations would at least a few years' time, by which the task would have been accomplished.

This addition to the security since the past few years has again brought back the zeal for inventions of better communication technologies along with information security, starting from Pranav mistry's sixth sense to thoughts of using 5G technology. Another classification in information security is watermarking (Abdulfetah *et al.*, 2010; Zeki *et al.*, 2011) its objective it to provide authorship through copyright protection. Reviewing the existing literature suggest to implement random image steganography with high capacity, good imperceptibility with additional complexity. Hence, this study proposes a method to accommodate all the necessary requirements of image steganography through kubera kolam.

PROPOSED METHODOLOGY

The indicator channel for the initial process is RED channel. For the embedding procedure, the algorithm is designed for a 6×6 block of an image, as in the image is sub-classified into blocks of 6×6 each and the algorithm is made to work on the each individual block of it. The bare part of Steganography, randomization is incorporated and inculcated with the magic square called Kubera Kolam.

The interesting part of Kubera Kolam is its wide usage in both puja room and may be in steganography algorithms. The magic square of Kubera Kolam is the sum of its any dimensional extension yields to 72 as shown in Fig. 1a. A subtraction of 19 from the Magic Square (Kubera Kolam) will interestingly lead to another magic square but with numbers from 1 extending to 9 and summing to 15 as shown in Fig. 1b.

Extending the 3×3 to 6×6 leads to the square as in Fig. 2.

Pixels of every 6×6 segment of the image is further rearranged and embedded in the same fashion with reference to the numbers formed in the above block, blinding the third party regarding the order of embedding.

(a)

23	28	21
22	24	26
27	20	25

(b)

4	9	2
3	5	7
8	1	6

Fig. 1(a-b): (a) The magic square of Kubera Kolam and (b) New magic square with a subtraction of 19 from 1a

6	32	3	34	35	1
7	11	27	28	8	30
19	14	16	15	23	24
18	20	22	21	17	13
25	29	10	9	26	12
36	5	33	4	2	31

Fig. 2: Extension of magic square from 3×3 to 6×6

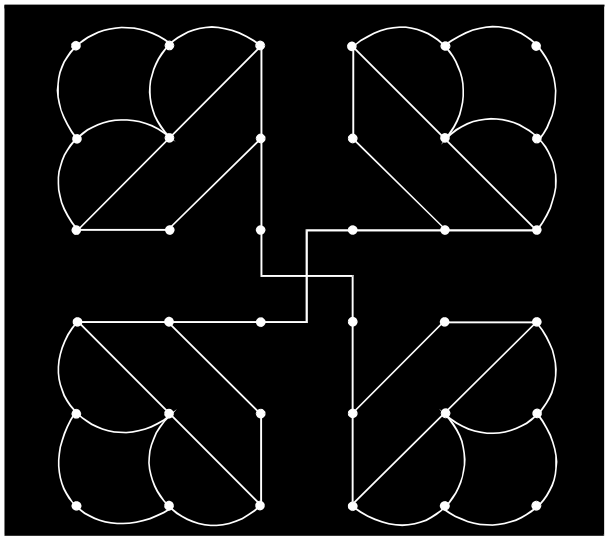

Fig. 3: Digital image of Pulli kolam

1	2	3	3	2	1
2	3	3	2	3	2
3	2	1	1	3	3
3	3	1	1	2	3
2	3	2	3	3	2
1	2	3	3	2	1

Fig. 4: Embedding capacity offered by the pixels in Pulli kolam

The Kolam design shown below in Fig. 3 is the Pulli Kolam which acts the symmetric key for the embedding of the user's message.

The number of curves or line on every dot is taken into account and a matrix is formed. For example, the top right dot has just one curve and the adjacent ones consist two each and so on.

Specifically, this dot in the Kolam gives the pixels for embedding as in Fig. 4 and gives the amount of bits for embedding in all pixels. After reading the sender's message, Kubera Kolam

matrix is now referred for the embedding order number and the equivalent position in the Pulli Kolam matrix is matched and the number in that position decides the number of bits to be read from the bit stream and embedded into appropriate place in the formed 6×6 block.

The further randomization is done with the conventional pixel indicator but with some slight modifications. Here, the 5th bit of the data channels (Blue and green) in that pixel is used to decide the reference bits in the indicator channel considered initially. As in, the 5th bits of both the channels turn out to be 0 and 0 then, last 2 bits of indicator channel will be acting as reference bits and if the bits are 0 and 1 then the 2nd and 3rd LSB's are taken as reference bits and if its 1 and 0, the 3rd and 4th LSB's are taken and finally if its 1 and 1 then 4th and 5th LSB's are taken as reference. Now based on the reference bits the mode of embedding is decided i.e., 0 0 implies no embedding and 0 1 means embedding in blue channel and 1 0 in green channel and 1 1 in both. Further methods have also been introduced for the dynamicity of the encryption.

Method 2: Deals with indicator channel being decided by the sender at run time by accepting a value (1, 2 and 3).

Method 3: Proves considerably more efficient than the previous two as the indicator channel for that block is decided by the Mod 3 of the ixj value of the present block under embedding. The MOD result of 0 implies RED, 1 Green and 2 Blue planes.

The algorithm for this scheme is clarified below and the flowchart is given in Fig. 5.

Method 1:

Embedding algorithm

Input: Secret data (D) and Cover image (C).

Output: Stego image(S) entrenched with secret information.

1. Change the secret data to binary stream.
2. Separate distinct RGB color planes of the cover/host image.
3. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:
4. Let pos=1
5. Go to the assumed 'pos' in the Kubera Kolam Matrix.
6. Assume index of pos to be (i, j) in Kubera Kolam.
7. Let k=value in indicated position (i, j) of the Pulli Kolam Matrix.
8. For each pixel in the cover image perform the following schema:

8.1. Say b [0] =current pixel's 5th bit of Bb.

8.2. Say b [1] =current pixel's 5th bit of Gb.

8.3. If b=00, then ref = the last two LSBs of current pixel in Rb should be taken.

Else if b=01, then ref = the 3rd and 2nd most LSBs of current pixel in Rb should be taken.

Else if b=10, then ref = the 4th and 3rd LSBs of current pixel in Rb should be taken.

Else if b=11, then ref = the 5th and 4th LSBs of current pixel in Rb should be taken.

9. If ref=00 then, move to the next pixel in the order of position.

Else if ref=01 then, secret data's k bits gets rooted in the Bb's current pixel.

Else if ref=10 then, embed the same in Gb's current pixel.

Else, embed the same in both Bb and Gb.

10. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.
11. If pos = 36, then go to the next 6×6 block in the cover image in the next cycle and similarly, repeat the steps from 4 to 10.
12. Perform the steps from 4 to 11 until all the data is embedded into the image and then finally stop the embedding procedure.

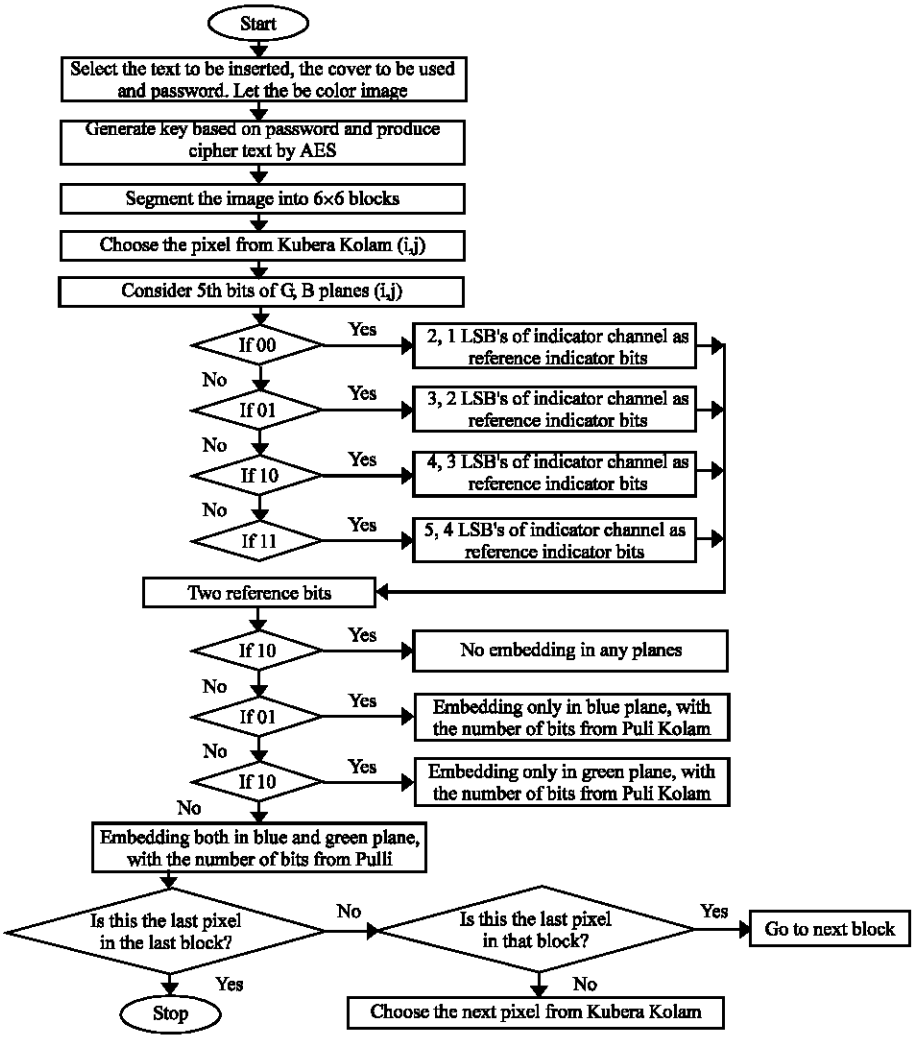

Fig. 5: Flow chart for proposed method

Recovery algorithm

Input: Stego image (S).

Output: Secret data (D).

1 Divide distinct RGB color planes of the cover/host image.

2. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:

3. Let pos = 1

4. Go to the assumed 'pos' in Kubera Kolam Matrix.

5. Assume index of pos to be (i, j) in Kubera Kolam.

6. Let k = value in indicated position (i, j) of the Pulli Kolam Matrix.

7. For each pixel in the cover image perform the following schema:

7.1. Say b [0] = current pixel's 5th bit of Bb.

7.2. Say b [1] = current pixel's 5th bit of Gb.

7.3. If b = 00, then ref = the last two LSBs of current pixel in Rb should be taken.

Else if b = 01, then ref = the 3rd and 2nd most LSBs of current pixel in Rb should be taken.

Else if b = 10, then ref = the 4th and 3rd LSBs of current pixel in Rb should be taken.

Recovery algorithm continue

Else if b = 11, then ref = the 5th and 4th LSBs of current pixel in Rb should be taken.
8. If ref = 00 then, move to the next pixel in the order of position.
Else if ref = 01 then, read k bits from Bb's current pixel and concatenate to D.
Else if ref = 10 then, read the same in Gb's current pixel and concatenate to D.
Else, read k bits in Gb's current pixel as well as Bb color plane and concatenate to D.
9. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.
10.If pos = 36, then go to the next 6×6 block in the cover image in the next cycle and similarly, repeat the steps from 3 to 9.
11. Save the resultant extracted secret information D.

Method 2:

Embedding algorithm

Input: Secret data (D), Cover image and Indicator Plane Index (I).
Output: Stego image(S) having secret data.
1. Change secret data to binary stream.
2. Divide distinct RGB color planes of the cover/host image.
3. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:
4. Let I be the index of the plane to be chosen as indicator channel which abides by the definition from user.
If I = 1 then, P [1] = R, P [2] = G, P [3] = B
Else if I = 2, then P [1] = G, P [2] = B, P [3] = R
Else if I = 3, then P [1] = B, P [2] = R, P [3] = G
5. Let pos = 1
6. Go to the assumed 'pos' in Kubera Kolam Matrix.
7. Assume index of pos to be (i, j) in Kubera Kolam.
8. Let k = value in indicated position (i, j) of the Pulli Kolam Matrix.
9. For each pixel in the cover image perform the following schema:
9.1. Let b [0] = current pixel's 5th bit of the color plane block specified by P [3].
9.2. Let b [1] = current pixel's 5th bit of the color plane block specified by P [2].
9.3 If b = 00, then ref = the last two LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 01, then ref = the 3rd and 2nd most LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 10, then ref = the 4th and 3rd LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 11, then ref = the 5th and 4th LSBs of current pixel in color plane block specified by P [1] should be taken.
9. If ref = 00 then, move to the next pixel in the order of position.
Else if ref = 01 then, embed k secret bits in the present pixel in color plane block specified by P [3].
Else if ref = 10 then, embed the same in present pixel in color plane block specified by P [2].
Else, embed both in P [2] as well as P [3] color plane.
10. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.
11. If pos = 36, then go to the next 6×6 block in the cover image in the next cycle and similarly, repeat the steps from 4 to 10.
12. Perform the steps from 4 to 11 until all the data is embedded into the image and then finally stop the embedding procedure

Recovery algorithm

Inputs: Stego image (S).
Output: Secret data (D).
1 Divide distinct RGB color planes of the cover/host image.
2. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:
3. Let I be the index of the plane to be chosen as indicator channel which abides by the definition from user.
If I = 1 then, P [1] = R, P [2] = G, P [3] = B
Else if I = 2, then P [1] = G, P [2] = B, P [3] = R
Else if I = 3, then P [1] = B, P [2] = R, P [3] = G

Recovery algorithm continue

4. Let pos = 1 and move to the assumed 'pos' in Kubera Kolam Matrix.
5. Assume index of pos to be (i, j) in Kubera Kolam.
6. Let k = value in indicated position (i, j) of the Pulli Kolam Matrix.
7. For each pixel in the cover image perform the following schema:
7.1. Say b [0] = current pixel's 5th bit of the color plane block specified by P [3].
7.2. Say b [1] = current pixel's 5th bit of the color plane block specified by P [2].
7.3 If b = 00, then ref = the last two LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 01, then ref = the 3rd and 2nd most LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 10, then ref = the 4th and 3rd LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 11, then ref = the 5th and 4th LSBs of current pixel in color plane block specified by P [1] should be taken.
8. If ref = 00 then, move to the next pixel in the order of position.
Else if ref = 01 then, read k secret bits of the present pixel in color plane block specified by P[3] and concatenate to D.
Else if ref = 10 then, read k secret bits of the present pixel in color plane block specified by P[2] and concatenate to D.
Else, read the same of the current pixel in color plane block specified by P[2] as well as P[3] color plane and concatenate to D.
9. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.
10.If pos = 36, then go to the next 6×6 block in the cover image in the next cycle and similarly, repeat the steps from 5 to 9.
11. Save the resultant extracted secret information D.

Method 3:

Embedding algorithm

Inputs: Secret data (D), Cover image (C).
Output: Stego image(S) with secret buried in it.
1. Change secret data to binary stream.
2. Divide distinct RGB color planes of the cover/host image
3. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:
4. Let test be the index of the plane to be chosen as indicator channel and is given by test = mod((x1*x2), 3) where x1, x2 are the indices of the position of the block chosen in the cover image denoting row and column respectively.
If test = 1 then, P [1] = R, P [2] = G, P [3] = B
Else if test = 2, then P [1] = G, P [2] = B, P [3] = R
Else if test = 3, then P [1] = B, P [2] = R, P [3] = G
5. Let pos = 1
6. Go to the assumed 'pos' in Kubera Kolam Matrix.
7. Assume index of pos to be (i, j) in Kubera Kolam.
8. Let k = value in indicated position (i, j) of the Pulli Kolam Matrix.
9. For each pixel in the cover image perform the following schema:
9.1. Say b [0] = current pixel's 5th bit of the color plane block specified by P [3].
9.2. Say b [1] = current pixel's 5th bit of the color plane block specified by P [2].
9.3 If b = 00, then ref = the last two LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 01, then ref = the 3rd and 2nd most LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 10, then ref = the 4th and 3rd LSBs of current pixel in color plane block specified by P [1] should be taken.
Else if b = 11, then ref = the 5th and 4th LSBs of current pixel in color plane block specified by P [1] should be taken.
9. If ref = 00 then, move to the next pixel in the order of position.
Else if ref = 01 then, embed k secret bits of the present pixel in color plane block specified by P [3].
Else if ref = 10 then, embed k secret bits of the current pixel in color plane block specified by P [2].
Else, embed the same in both P [2] as well as P [3] color plane.
10. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.
11. If pos = 36, then go to the next 6×6 block in the cover image in the next cycle incrementing column wise and similarly, repeat the steps from 4 to 10 with corresponding changes in x1,x2.
12. Perform the steps from 4 to 11 until all the data is embedded into the image and then finally stop the embedding procedure.

Recovery algorithm

Input: Stego image (S)

Output: Secret data (D)

1 Divide distinct RGB color planes of the cover/host image.

2. Segment entirely the R, G and B planes into 6×6 sized blocks namely Rb, Gb and Bb and do the following procedure for each of the pixel:

3. Let test be the index of the plane to be chosen as indicator channel and is given by test = mod((x1*x2), 3) where x1, x2 are the indices of the position of the block chosen in the cover image denoting row and column respectively.

If test = 1 then, P [1] = R, P [2] = G, P [3] = B

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

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

4. Let pos = 1 and move to the assumed 'pos' in Kubera Kolam Matrix.

5. Assume index of pos to be (i, j) in Kubera Kolam.

6. Let k = value in indicated position (i, j) of the Pulli Kolam Matrix.

7. For each pixel in the cover image perform the following schema:

7.1. Let b [0] = current pixel's 5th bit of the color plane block specified by P [3].

7.2. Let b [1] = current pixel's 5th bit of the color plane block specified by P [2].

7.3 If b = 00, then ref = the last two LSBs of current pixel in color plane block specified by P [1] should be taken.

Else if b = 01, then ref = the 3rd and 2nd most LSBs of current pixel in color plane block specified by P [1] should be taken.

Else if b = 10, then ref = the 4th and 3rd LSBs of current pixel in color plane block specified by P [1] should be taken.

Else if b = 11, then ref = the 5th and 4th LSBs of current pixel in color plane block specified by P [1] should be taken.

8. If ref = 00 then, move to the next pixel in the order of position.

Else if ref = 01 then, read k secret bits of the present pixel in color plane block specified by P[3] and concatenate to D.

Else if ref = 10 then, read k secret bits of the present pixel in color plane block specified by P[2] and concatenate to D.

Else, read k bits of the present pixel in color plane block specified by P[2] as well as P[3] color plane and concatenate to D.

9. Now, pos = pos+1 and go to step 5 until the pos becomes 36 in the specified matrix sized 6×6.

10. If pos = 36, then go to the next 6×6 block in the cover image in the next cycle incrementing column wise and similarly, repeat the steps from 4 to 9 with corresponding changes in x1,x2.

11. Store the resulting recovered secret data D.

ERROR METRICS

The quality of the stego image is calculated through two universal parameters viz., mean square error and the peak signal to noise ratio.

The MSE is calculated by the equation:

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

Where, M is pixels in horizontal dimension and N is pixels in the vertical dimension in X_i and Y_j where i, j constitutes pixels of original and stego images, respectively.

The peak signal to noise ratio is given by the equation:

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

where, I^2 max is every pixel's intensity value. High PSNR values indicate high visual quality.

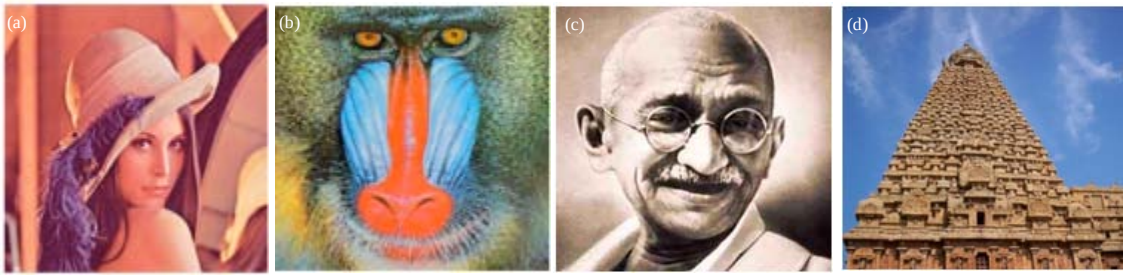

Fig. 6(a-d): Cover images (a) Lena, (b) Baboon, (c) Mahatma Gandhi and (d) Temple

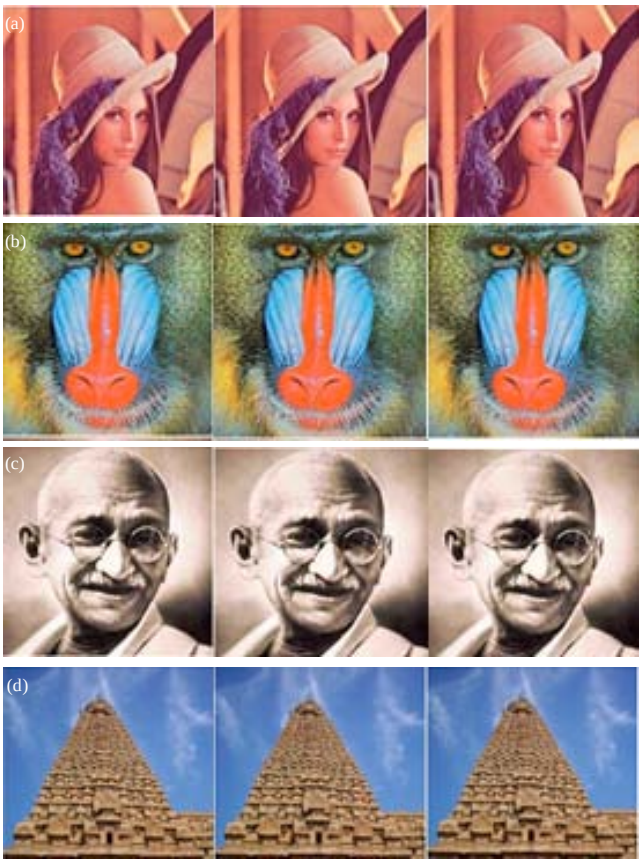

Fig. 7(a-d): Stego images for 3 methods, (a) Lena, (b) Baboon, (c) Mahatma Gandhi and (d) Temple

RESULTS AND DISCUSSION

In this script Lena, Gandhi, Baboon and Temple of 300×300 color images are taken the covers as given below in Fig. 6a-d and tested for full embedding capacity. The operation of the intended routine is rationalized via MSE and PSNR for the four covers in RGB planes through three different procedures and the results are given in the table.

The stego images that were obtained in the various methods are given in Fig. 7 and histograms for Lena image for all three methods are depicted in Fig. 8.

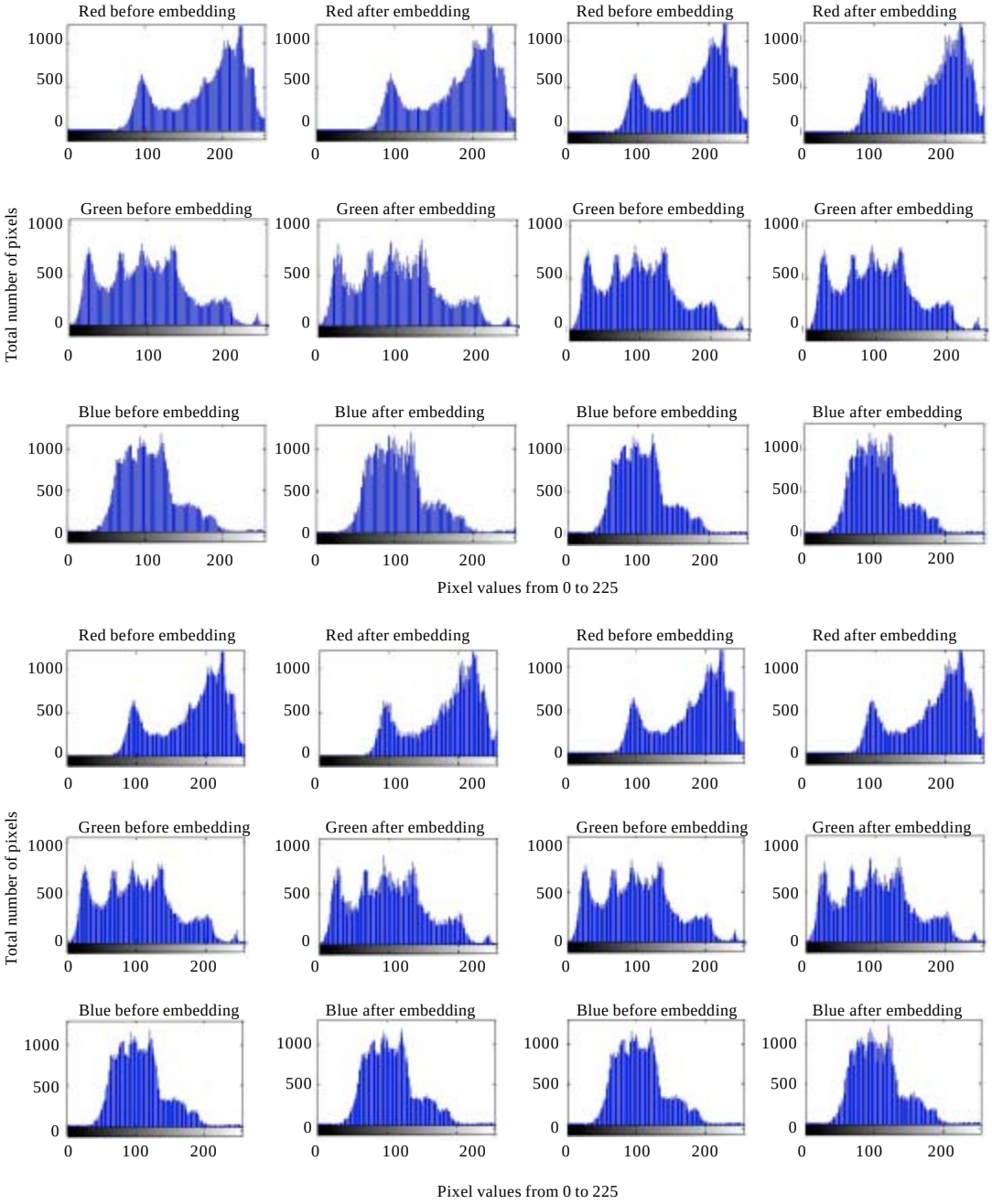

Fig. 8: Histograms for Lena image, Red: Method 1, Green: Method 2 and Blue: Method 3

Method 1 uses Red as indicator channel and results are given in Table 1.The drawback of the first method is overcome in the second method by giving the user the liberalization to choose the indicator channel, whose results are given in Table 2. The third method changes the indicator channel for each pixel which gives no clue to the intruder about the indicator channel. The results of the third method are given in the Table 3. The corresponding tables are as follows.

Table 1: Experimental values of image metrics obtained for method 1

		Channel 1 (red)		Channel 2 (green)		Channel 3 (blue)		
Cover image	Indicator channel	MSE	PSNR	MSE	PSNR	MSE	PSNR	Bits per pixel
Lena	Red	0	8	0.8373	48.9020	0.8619	48.7764	2.2461
Baboon	Red	0	8	0.8416	48.8795	0.8534	48.8193	2.2192
Gandhi	Red	0	8	0.8363	48.9073	0.8473	48.8502	2.2672
Temple	Red	0	8	0.9122	48.5299	0.8561	48.8055	2.3154

Table 2: Experimental values of image metrics obtained for method 2

		Channel 1 (red)		Channel 2 (green)		Channel 3 (blue)		
Cover image	Indicator	MSE	PSNR	MSE	PSNR	MSE	PSNR	Bits per pixel
Lena	Red	0	8	0.8373	48.9020	0.8619	48.7764	2.2461
	Green	0.8487	48.8432	0	8	0.8333	48.9228	2.2167
	Blue	0.8643	48.7640	0.8550	48.8111	0	8	2.2188
Baboon	Red	0	8	0.8416	48.8795	0.8534	48.8193	2.2192
	Green	0.8421	48.8771	0	8	0.8343	48.9178	2.2107
	Blue	0.8360	48.9088	0.8556	48.8083	0	8	2.2089
Gandhi	Red	0	8	0.8363	48.9073	0.8473	48.8502	2.2672
	Green	0.7718	49.2555	0	8	0.8304	48.9378	2.1342
	Blue	0.8613	8.7795	0.8672	48.7498	0	8	2.2463
Temple	Red	0	8	0.9122	48.5299	0.8561	48.8055	2.3154
	Green	0.8485	48.8442	0	8	0.9784	48.2259	2.3680
	Blue	0.8705	48.7334	0.8447	48.8638	0	8	2.2750

Table 3: Experimental values of image metrics obtained for method 3

	Channel 1 (red)		Channel 2 (green)		Channel 3 (blue)		
Cover image	MSE	PSNR	MSE	PSNR	MSE	PSNR	Bits per pixel
Lena	0.3876	52.2466	0.6503	49.9993	0.6569	49.9555	2.2308
Baboon	0,3839	52.2888	0.6469	50.0045	0.6505	49.9983	2.2074
Gandhi	0.3780	52.3555	0.6552	49.9671	0.6551	49.9679	2.2280
Temple	0.3976	52.1361	0.6843	49.7784	0.6921	49.7292	2.3249

CONCLUSION

Kolam is a mincing floor painting symbolizing exquisiteness as well as welcoming environs. To put it technically, it is a unique form of image incorporating diverse patterns and moreover they have various names based on such patterns. Each mention has its own distinct feature and devout values. Using Kolas as gizmos in image steganography is veritably thought provoking. Undoubtedly, it is an unparalleled choice to render this paper as a unique work. In this paper the Kubera kolam and the Pulli kolam are utilized for randomizing the pixel and finding the volume of bits for embedding in each and every pixel. Pixel indicator method is adopted for embedding the secret bits which is a universally agreed efficient mode for Steganography. This paper is a blend of cryptography and steganography thus assuring security and complexity in its own right. MSE and PSNR values confirm this wrap up which also stands for enhanced imperceptibility. Thus the proposed methods increase the complexity of the secret data embedding and are determined to be beneficial.

REFERENCES

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. Brownian motion of binary and gray-binary and gray bits in image for stego. J. Applied Sci., 12: 428-439.

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

Amirtharajan, R. and J.B.B. Rayappan, 2012d. Pixel authorized by pixel to trace with SFC on image to sabotage data mugger: A comparative study on PI stego. Res. J. Inform. Technol., 4: 124-139.

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.

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

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

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.

Luo, H., Z. Zhao and Z.M. Lu, 2011. Joint secret sharing and data hiding for block truncation coding compressed image transmission. Inform. Technol. J., 10: 681-685.

Mohammad, N., X. Sun and H. Yang, 2011. An excellent Image data hiding algorithm based on BTC. Inform. Technol. J., 10: 1415-1420.

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. Xiang and M.X. Wang, 2010. A review on detection of LSB matching steganography. Inform. 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.

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

Thanikaiselvan, V., S. Kumar, N. Neelima and R. Amirtharajan, 2011. 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.

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

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.

Zeki, A.M., A.A. Manaf and S.S. Mahmod, 2011. High watermarking capacity based on spatial domain technique. Inform. Technol. J., 10: 1367-1373.

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.