Bidirectional Circular Shift-Based Reversible Steganography Algorithm with Dual Stego-Images

Mujeebudheen Khan Immamuddinkhan
Department of Computer Engineering, Noorul Islam Centre for Higher Education, India
mujeebudheenkhan@outlook.com

Siva Sankar Kanahasabapathy
Department of Information Technology, Noorul Islam Centre for Higher Education, India
sivasankar.noorul@outlook.com

Abstract: *Reversible steganography with two stego-image deals with the embedding of secret data on another image that produce two stego-images in such a way that during extraction the initial cover image can be recovered. This research work proposes an algorithm capable of reversible hiding of data, which produce two stego-images based on bidirectional circular shifting i.e., shifting in clockwise and anti-clockwise direction. Initially, the approach involves categorizing the information into L bits and transforming it into a corresponding decimal representation. The two adjacent decimal number forms a pair such that the first decimal number is shifted in both clockwise and anticlockwise direction to produce clockwise value and anticlockwise value. From the clockwise value, anticlockwise value and secret decimal data a single minimum value is chosen with an index from which the minimum value is transformed on the basis of the minimum value's histogram. The modified transformed values and indices are incorporated into pairs of cover image pixels to generate the stego-images. During the extraction process, adjusted transformed value and index are extracted from which the hidden data is reconstructed by finding the number of clockwise and anticlockwise shifts. Experimental evaluation shows higher Peak Signal-To-Noise Ratio (PSNR) than the conventional methods*.

Keywords: *Data hiding, multiple stego-image, PSNR, bi-directional shift, stego-image*.

Received August 7, 2024; accepted March 3, 2025
https://doi.org/10.34028/iajit/22/4/6

1. Introduction

Data hiding technique provides a way to hide a confidential data within a media such as audio, video etc. This data hiding [6, 18, 23, 26] is essential in order to secure the data from illegal monitoring and illegal manipulation of media. Encryption [3, 27] provides one of the ways to preserve the media. This encryption makes the attackers intension to decrypt and view the data. Reversible steganography makes the attackers difficult to predict the presence of data inside the image, as the stego-image exhibits resemblance to cover image. The distinction between reversible and non-reversible steganography lies in the ability to reconstruct the original cover image from the stego-image. Different steganography methods [9, 10, 21] have been proposed which is discussed below. The steganography techniques can be characterized in 2 different varieties such as transformation-based data hiding and spatial domain-based data hiding. Transformation based data hiding does not embed the confidential data directly in the cover pixel while it embeds the data after transforming the cover pixel using a transformation method.

Transformation methods commonly used for data hidings are non-reversible. Reversibility can be easily achieved by using spatial domain methods such as prediction error expansion [8, 11, 24] and histogram shift [15]. The greater the disparity in intensity between adjacent pixels, the difference expansion method results in increased distortion [7]. The prediction error expansion method involves predicting a value from the surrounding pixel neighborhood. Histogram shifting techniques estimate the maximum histogram intensity and hide the information on that career intensity by shifting the intensities higher than the career intensity by one. This single stego-image based data hiding have less embedding capacity also the attackers can expose the hidden data. Since the entire hidden data is preserved within the single stego-image. For embedding capacity improvement, dual stego-images are proposed which are extremely secure. This dual steganographic techniques are highly secure because, even single bit cannot be extracted from one stego-images. 2 stego-images are required to extract the hidden data. Chang *et al*. [4] estimated the modulus for all the grey scale intensities from 0 to 255. Therefore, this modulus matrix has a size of 256×256. Chang *et al*. [5] enhanced the performance by embedding the data on the diagonal. Qin *et al*. [19] used the transformation technique and three rules to embed the data. Jung [14] introduced the selection of stego-image pairs using a gap function from the two pairs in horizontal and vertical direction. Lu *et al*. [16] introduced an Least Significant Bit (LSB) matching technique and it was improved by Wang *et al*. [25]. This method embeds a three-bit data on a cover pixel after replicating a copy of the cover pixel.

2. Related Works

2.1. EMD Technique on Single Stego Image

Zhang and Wang [28] introduced Exploiting Modification Technique (EMD) for data hiding. This technique embeds n number of pixels at a time using a modulus function. Consider a group of pixels P_1, $P_2 \ldots\ldots P_n$. From the group of pixels, this method estimates the extraction function using the Equation (1),

$$F(P_1, P_2 \ldots\ldots P_n) = \left[\sum_{k=1}^{n} (P_k . k)\right] mod(2n+1) \quad (1)$$

A group of m secret bits are converted to decimal data *D*. If the value of *D* is same as that of *F*, then the group of pixels remains unchanged. If the value of *D* differ from *F*, then the group of pixels are modified using the relation,

$$R = (D - F) mod(2n+1) \quad (2)$$

In this context, *R* denotes the position of the pixel that requires modification. Consider an image with cover pixels {54, 68, 35, 46}. From the pixel value calculate *F(*54, 68, 35, 46*)=(*54×1+68×2+35×3+46×4*) mod (*2×4+1=479 *mod* 9=2. Consider a data=110_2=6_9. Therefore=(6-2) *mod* (2×4+1)=4 *mod* 9=4. If *R* value is less than or equal to *n*, then increment P_R by 1, else, decrement P_{2n+1-R} by 1. Here *R*=4, therefore increment the value of P_4 by 1. Therefore, the stego pixels are {54, 68, 35, 47} as shown in Figure 1.

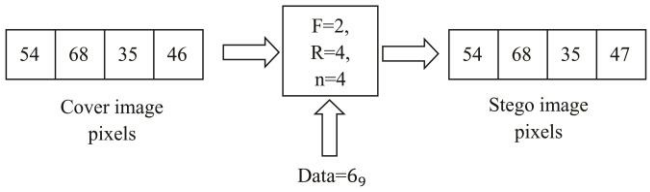


Figure 1. An example of data embedding by EMD in single stego image.

2.2. EMD Technique on Dual Stego Image

Chang *et al*. [4] developed dual stego-image scheme by modifying modulus function. Here, the modulus function is modified from Equation (3) as,

$$M(P_i, P_{i+1}) = (P_i + 2 \times P_{i+1}) mod\ 5 \quad (3)$$

Here modulus is calculated using possible intensity values from 0 to 255. A pixel pair is formed from the cover pixel and 5×5 block was selected by keeping the cover pixel as center element. Two bases 5 digits can be embedded on every pixel pair P_i, P_{i+1}. Consider a cover pixel pair {12, 43} and base 5 data {4, 2}. The modulus function for {12, 43} is *M*(12, 43)=(12+2×43) *mod* 5=98 *mod* 5=3. The stego pixel pair corresponding to the data *D*=4 is {11, 44} and the stego pixel corresponding to the data *D*=2 is {13, 42}. These stego pixel pair lies diagonal to the center modulus function as shown in Figure 2.

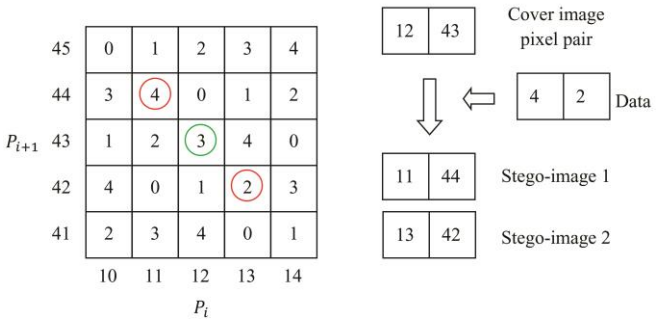


Figure 2. An example of steganography using EMD in dual stego-image.

Chang *et al.* [5] elevated the EMD technique by elevating the block size to 9×9. For a pixel P_i the modulus function can be represented as,

$$M(P_i, P_i) = (P_i + 3P_i) mod\ 9 \quad (4)$$

The stego pixel correspond to the pixel 17 are 16 and 18 as shown in Figure 3.

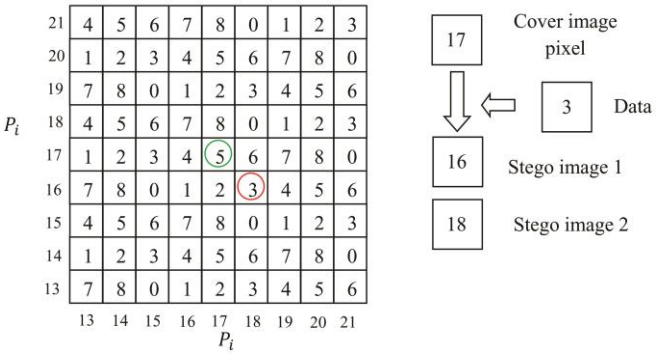


Figure 3. An example for data embedding using Chang *et al*. [5].

Ma *et al*. [17] proposed the properties of generative adversarial networks, a dual-adversarial steganography algorithm is developed to increase both the visual quality and the steganalysis resistance capabilities of the stego image.

AbdAl-Hameed *et al*. [1] reported an image steganography method that uses the Double Density Dual Tree Wavelet Transform (DDDT-DWT) to increase capacity while maintaining the best possible quality.

Qin *et al*. [19] developed a scheme on dual stego-image using EMD method. This method initially converts the secret bits into base 5 symbols. The symbols D_1 and D_2 is embedded on P_i and $P_{(i+1)}$ to get pixel pair p_i^1 and p_{i+1}^1. Stego-image was formed by utilizing the steps given below.

1. If $p_i^1 = P_i$ and $p_{i+1}^1 = P_{i+1}$ image was created using the EMD embedding procedure to obtain pixel pair p_i^2 *and* p_{i+1}^2.
2. If $p_i^1 \neq P_i$ and $p_{i+1}^1 = P_{i+1}$.

$$p_i^2 = P_i - L_1 \times sign(P_i^1 - P_i) \quad (5)$$

$$p_{i+1}^2 = P_{i+1} \quad (6)$$

$$\text{Where, } L_1 = \begin{cases} 5, if\ D_2 = F[P_1 - 5 \times sign(p_i^1 - P_i)P_{i+1}] \\ 4, if\ D_2 = F[P_1 - 4 \times sign(p_i^1 - P_i)P_{i+1}] \\ 3, if\ D_2 = F[P_1 - 3 \times sign(p_i^1 - P_i)P_{i+1}] \\ 2, if\ D_2 = F[P_1 - 2 \times sign(p_i^1 - P_i)P_{i+1}] \\ 1, if\ D_2 = F[P_1 - 1 \times sign(p_i^1 - P_i)P_{i+1}] \end{cases} \quad (7)$$

3. If $p_i^1 = P_i$ and $p_{i+1}^1 \neq P_{i+1}$ the second stego-image was created as,

$$p_i^2 = P_i \quad (8)$$

$$p_{i+1}^2 = P_{i+1} - L_2 \times sign(P_{i+1}^1 - P_{i+1}) \quad (9)$$

$$\text{Where, } L_2 = \begin{cases} 5, if\ D_2 = F[P_i, P_{i+1} - 5 \times sign(p_i^1 - P_{i+1})] \\ 4, if\ D_2 = F[P_i, P_{i+1} - 4 \times sign(p_i^1 - P_{i+1})] \\ 3, if\ D_2 = F[P_i, P_{i+1} - 3 \times sign(p_i^1 - P_{i+1})] \\ 2, if\ D_2 = F[P_i, P_{i+1} - 2 \times sign(p_i^1 - P_{i+1})] \\ 1, if\ D_2 = F[P_i, P_{i+1} - 1 \times sign(p_i^1 - P_{i+1})] \end{cases} \quad (10)$$

2.3. Authenticable Reversible Data Hiding

Jung [14] developed a scheme using modulus function. For every pixel pair a modulus function M is computed by,

$$M = M(x, y) = (x + 2 \times y) mod\ 5 \quad (11)$$

The data is initially transformed to base 5 values. The base 5 data is embedded on the pixel pair (x, y) to obtain dual pixel pairs (x_h^1, y_h^1) and (x_h^2, y_h^2) in horizontal direction and dual pixel pairs (x_v^1, y_v^1) and (x_v^2, y_v^2) in vertical directions. Consider the cover image pixel duo be (12, 43) and the data be d_1, d_2=31$_5$.

The vertical pixel pairs and horizontal pixel pairs can be calculated as shown in Figure 4. From this four-pixel pairs two-pixel pair is chosen (one pixel pair from two horizontal pixel pair and another pixel pair is chosen from two vertical pixel pair) using the gap function G. The horizontal gap function G is calculated as

$$G = |G_A(G_1, G_2) - G_B(G_1, G_2)| \quad (12)$$

$$G = \left| G_A\left[\frac{G_1 + G_2}{2}\right] - G_B\left[\frac{G_1 + G_2}{2}\right] \right| \quad (13)$$

$$\text{Where, } G_1 = \sum_{k=1}^{5} |A_k - u| \text{ , } G_2 = \sum_{k=1}^{5} |B_k - v| \quad (14)$$

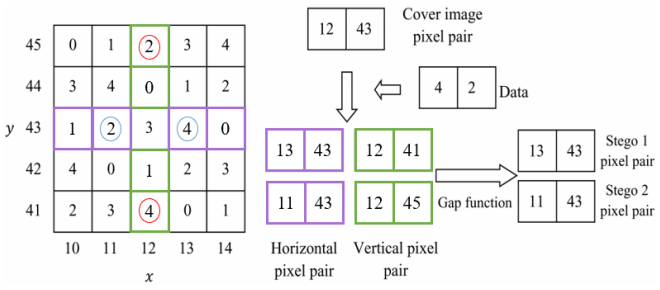


Figure 4. An example for data embedding using Chang *et al.* [5].

Here A_k is the neighborhood of u and B_k is the neighborhood of v as shown in Figure 5. For the function G_A(.) the value of (u, v) is (x_h^1, y_h^1) and for the function G_B(.) the value of (u, v) is (x_h^2, y_h^2). Similarly, the gap function G is calculated using the two vertical pairs.

The gap function is calculated both for two horizontal pixel pair as well as the two vertical pixel pair. The two-pixel pair is chosen such that the two pixels must provide a minimum gap function.

A_2	A_1	B_1	B_2
A_3	u	v	B_3
A_4	A_5	B_5	B_4

13	14	44	41
12	u	v	40
10	11	43	42

Figure 5. Estimating the gap function.

Consider a cover image pixel pair as {12, 43} and secret data as {4, 2}, the modulus function is estimated for x=12 and y=43 using Equation (11). The modulus function is calculated as M(12+2×43) *mod* 5=3. Using the modulus function as center value estimate the horizontal and vertical pixel pair. The horizontal pixel pair for the data {4, 2} can be calculated as {11, 43} and {13, 43} as shown in Figure 4. Similarly, the Vertical pixel pair for the data {4, 2} can be calculated as {12, 41} and {12, 45}.

Assume that the neighborhood pixels of u be A_k={14, 13, 12, 10, 11} and neighborhood pixels of v be B_k={44, 41, 40, 42, 43}. The gap function is estimated using two horizontal pixel pair {11, 43} and {13, 43}. For the pair {11, 43}, G_1=3+2+1+1+0=7 and G_2=1+2+3+1+0=7, and for the pair {13, 43}, G_1=1+0+1+3+2=7 and G_2=1+2+3+1+0=7. Therefore, the gap function for the horizontal pair is G=0. Similarly, the gap function is estimated using two horizontal pixel pair {12, 45} and {12, 41}. For the pair {12, 45}, G_1=2+1+0+2+1=6, G_2=1+4+5+3+2=15 and for the pair {12, 41}, G_1=2+1+0+2+1=6, G_2=3+0+1+1+2=7. Therefore, the gap function for the vertical pair is G=5. Since the horizontal gap function is less than the vertical gap function, the stego pixel pairs are {11, 43} and {13, 43}.

2.4. RDH Using LSB Matching

Wang *et al.* [25] introduced LSB matching scheme. This method initially creates the two copies of original cover image, and then convert the decimal value to binary. The data b_1b_0 are embedded in 1st stego-image and the bit b_2 is embedded in 2nd stego-image. Let the cover image copies be represented by $C_7C_6C_5C_4C_3C_2C_1C_0$. Inorder to embed b_1b_0 the first stego-image LSB's C_1C_0 are replaced by b_1b_0. Similarly, to embed b_2 the second stego-image, third LSB (C_2) is replaced by b_2. Therefore, 1st and 2nd stego-images are represented as $C_7C_6C_5C_4C_3C_2b_1b_0$ and $C_7C_6C_5C_4C_3C_2C_1C_0$. Consider the cover pixel as 45 and the data as 010. Initially the cover image copies are 45 and 45. Converting the cover pixels to decimal, the cover pixel copies become 00101101 and 00101101. Here the data is $b_2b_1b_0$=010.Therefore replacing the two LSB's of first copy by b_1b_0=10. We get the first stego-image as 00101110. By replacing the third LSB of second copy by b_2=0. We get the second stego-image as 00101001 as

illustrated in Figure 6. Converting new values to decimal we get the stego pixels as 46 and 41.

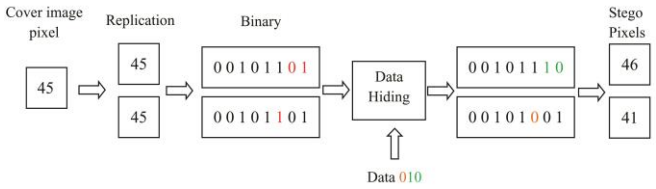


Figure 6. An example of data hiding using LSB matching.

During extraction the two stego pixels are transformed to binary to obtain the first stego bits $C_7C_6C_5C_4C_3C_2b_1b_0$ and second stego $C_7C_6C_5C_4C_3C_2C_1C_0$. Inorder to extract the data b_1b_0 the two LSB's from stego-image 1 is extracted and the bits b_1b_0 are replaced by C_1C_0 of the 2nd stego-image. Similarly, data is obtained from the 3rd LSB b_2 of 2nd stego-image. After extracting b_2 is replaced by C_2 of the 1st stego-image. Converting the stego pixel 46 and 41 to binary we obtain the stego binary sequence as 00101110. Extracting the bits b_1b_0=10 and replacing b_1b_0 by the C_1C_0=01 of the 2nd stego-image. We get 00101101, similarly the data b_2=0 is extracted from the third LSB of second stego-image.

3. Proposed Method

The proposed reversible dual steganographic technique has 2 stages. The initial phase involves embedding data, while the second phase encompasses data extraction and retrieval of cover image.

3.1. Data Embedding Phase

Figure 7, illustrates the process flow of the suggested data embedding phase.

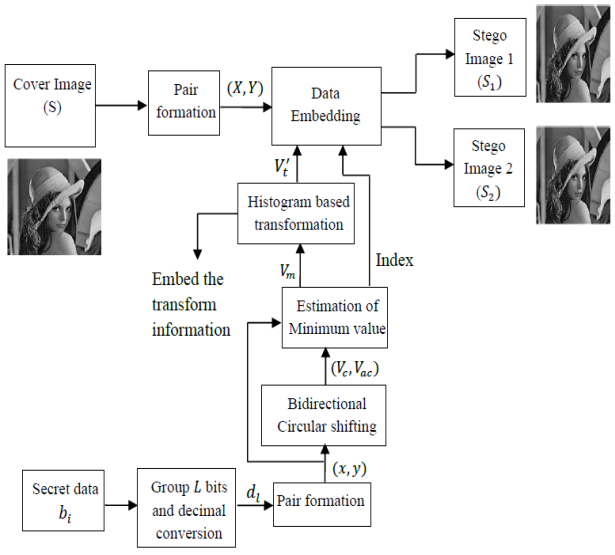


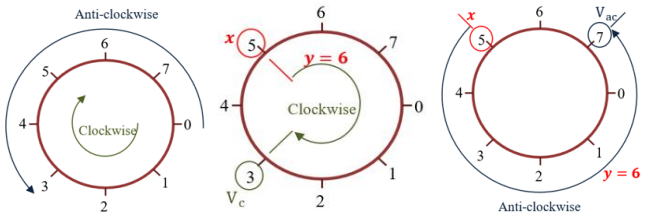
Figure 7. Block diagram of proposed data embedding phase.

Let 'S' represent cover image, where b_i represent binary secret data, and 'L' represent group size. Initially, b_i is grouped into segments of L bits and transformed into decimal data d_l.

$$d_l = \sum_{k=0}^{L-1} 2^k \times b_i \tag{15}$$

The minimum of d_l is '0' and the maximum of d_l is 2^L-1. Therefore $0 \le d_l \le 2^L-1$. The set of decimal numbers are grouped as decimal data $V_d=\{d_1, d_2, d_3, \ldots, d_n\}$. Where n is the count of V_d. Then V_d is grouped as overlapping pair. Let (x, y) be any data pair. The data 'x' is shifted 'y' times in both clockwise and anticlockwise direction to obtain the clockwise value (V_c) and anticlockwise (V_{ac}).

The clockwise shifting and anticlockwise shifting are performed as depicted in Figure 8. It shows an example where L=3, so that the decimal data d_l ranges to 0 to 7. This example shows a pixel pair data as (5, 6). The clockwise value (V_c) is estimated by shifting the value '5' six times in the clockwise direction to obtain the clockwise value V_c=3. Similarly anticlockwise value (V_{ac}) is estimated by shifting the value '5' six times in the anticlockwise direction to obtain the anticlockwise value V_{ac}=7.



a) Representation. b) Clockwise shifting. c) Anti-clockwise shifting.

Figure 8. Bidirectional circular shifting.

Without performing actual shifting operation, the clockwise value (V_c) can be calculated for shifting x in clockwise direction y times is represented by,

$$V_c = (x + y)\ mod\ 2^L \tag{16}$$

Similarly, the equation to shift x in counter-clockwise direction y times is given by

$$V_{ac} = \begin{cases} c & if\ c \ge 0 \\ 2^L + c & if\ c < 0 \end{cases} \tag{17}$$

$$\text{where, } c = x - y \tag{18}$$

By using the Equation (16), the clockwise shifted value V_c for shifting the value=5, for y=6 times can be calculated as V_c=(5+6) *mod* 2^3=11 *mod* 8=3. Similarly, by using the Equation (17) the anticlockwise shifted value V_{ac} for shifting x=5 value, y=6 times. The value of c is computed as c=5-6=-1. Since the value of c is less than 0, $V_{ac}=2^L+c=2^3+(-1)=7$, the clockwise and anticlockwise value are V_c=3 and V_{ac}=7 respectively.

The minimum value (V_m) is calculated from the minimum of secret decimal data (V_d=y), clockwise value (V_c), and anticlockwise value (V_{ac}).

$$V_m = \min(V_d, V_c, V_{ac}) \tag{19}$$

Where, V_d is equal to the second decimal data in the pair (x, y), i.e., V_d=y. The index value is calculated which may be 0,1 or -1 and it depends on the minimum value. The

index value (1) can be calculated using the relation,

$$I = \begin{cases} 0 & if\ V_m = V_d \\ 1 & if\ V_m = V_c \\ -1 & if\ V_m = V_{ac} \end{cases} \quad (20)$$

The histogram is calculated from the minimum value data V_m for different values of V_m. The histogram of minimum value V_m is represented by $n(V_m)$, where $n(V_m)$ is the number of minimum value data that contains the data V_m.

Tables 1 and 2 shows an example for histogram-based transformation. Based on the histogram values of the minimum value data (V_m) estimate the transformed value V_t. Table 2 shows the histogram-based transformation where the minimum value data V_m=0 is transformed to V_t=7 and V_m=1 is transformed to V_t=4 and so on.

Table 1. An example for histogram-based transformation with L=4.

Minimum value data V_m	**Count**	**Transformed value V_t**
0	6723	7
1	41,328	4
2	55,425	2
3	10,124	6
4	32,001	5
5	48,425	3
6	82,164	0
7	69,785	1

Table 2. Transformation information (I_n).

V_m	0	1	2	3	4	5	6	7
V_t	7	4	2	6	5	3	0	1

The transformed value (V_t), is adjusted to obtain the adjusted transformed value V_t' which can be calculated as,

$$V_t' = \begin{cases} -0.5(V_t + 1)\ if\ V_t\ is\ odd \\ 0.5V_t\ \ if\ V_t = 0\ or\ V_t\ is\ even \end{cases} \quad (21)$$

From the cover image *S*, form a pixel pair (*X*, *Y*). The adjusted transformed value V_t' and index (*I*) are embedded in *X* and *Y* respectively. The first stego image pixels for the pair (*X*, *Y*) is (X_1, Y_1) which is calculated using Equations (22) and (23).

$$X_1 = X - \left\lfloor \frac{V_t'}{2} \right\rfloor \quad (22)$$

$$Y_1 = Y - \left\lfloor \frac{I}{2} \right\rfloor \quad (23)$$

Similarly, the second stego image pixels for the pair (*X*, *Y*) is (X_2, Y_2) which is calculated using Equations (24) and (25).

$$X_2 = X + \left\lfloor \frac{V_t'}{2} \right\rfloor \quad (24)$$

$$Y_2 = Y + \left\lfloor \frac{I}{2} \right\rfloor \quad (25)$$

Histogram transformation info (I_n) is essential to extract secret data. Therefore, the transformation information is embedded in the first 2^{L-1} cover pixels using Equations (26) and (27) and remaining cover pixels forms the pair (*X*, *Y*). Note that first 2^{L-1} cover pixels are not used to embed the adjusted transformed value or Index value.

$$P_1 = P - \left\lceil \frac{I_n}{2} \right\rceil \quad (26)$$

$$P_2 = P + \left\lceil \frac{I_n}{2} \right\rceil \quad (27)$$

In the above transformation Table 2, the first 2^{L-1}=8 embeddable cover pixels *P* are embedded with the transformation information {7, 4, 2, 6, 5, 3, 0, 1} using Equations (26) and (27). (X_1, Y_1) forms the 1st stego image S_1, while (X_2, Y_2) forms the 2nd stego image S_2. The pixels eligible for embedding are those whose intensity falls within the interval [2^{L-1}, 256-2^{L-1}] and the remaining pixels are non-embeddable pixels. Pixels categorized as non-embeddable are excluded from data embedding to prevent overflow and underflow occurrences.

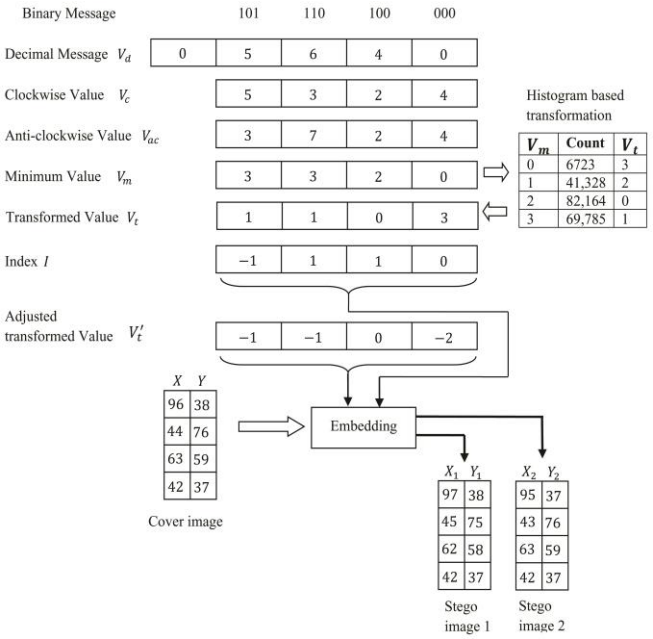


Figure 9. An example for data embedding by the proposed method.

Consider an example, where the cover pixels are {96, 38, 44, 76, 63, 59, 42, 37} and a part of the data as {101110100000} and *L*=3. The grouped data can be written as {(101), (110), (100), (000)}. Converting the binary data to decimal, the decimal data can be represented as V_d={5,6,4,0}. Initialize the initial value of V_d as 0, therefore V_d={0,5,6,4,0}. An overlapping pair is formed with elements of V_d, i.e., {(0,5), (5,6), (6,4), (4,0)}. The clockwise value and anticlockwise value are estimated with the pair using Equations (16) and (17) respectively. For example, for the pair (6, 4) treating 6 as *x* and 4 as *y*, V_c and V_{ac} are calculated as 2 and 2 respectively using Equations (16) and (17). i.e., V_c={5, 3, 2, 4} and V_{ac}={3, 7, 2, 4} Using Equation (19) the minimum value V_m is calculated as V_m={3, 3, 2, 0}. The index (I) is computed by utilizing Equation (20) as *I*={-1, 1, 1, 0} By calculating the histogram of the minimum values the minimum value V_m is transformed to transformed value V_t. In this example shown in Figure 9, minimum value data 0, 1, 2 and 3 are transformed to 3, 2, 0 and 1 respectively. Therefore, the transformed

values are V_t={1, 1, 0, 2}. Using Equation (21) the transformed values V_t are adjusted to V_t' as V_t'={-1, -1.0, -2}. Non overlapping pair is formed using the stego pixels {96, 38, 44, 76, 63, 59, 42, 37}, therefore the stego pixel pair can be formed as {(96, 38), (44, 76), (63, 59), (42, 37)}. The transformed values V_t'={-1, -1.0, -2} is embedded on the first value of the pair elements X={96, 44, 63, 42} using Equations (22) and (23) to obtain the first stego values of the pair X_1={97, 45, 62, 42} and X_2={95, 43, 63, 42}. Similarly, the index I={-1, 1, 1,0} is embedded on the second value of the pair elements Y= {38, 76, 59, 37} using Equations (24) and (25) to obtain the second stego values of the pair Y_1={38, 75, 58, 37} and Y_2={37, 76, 59, 37}as shown in Figure 9.

3.2. Data Extraction Phase

Figure 10, depicts the process flow of the proposed data extraction phase.

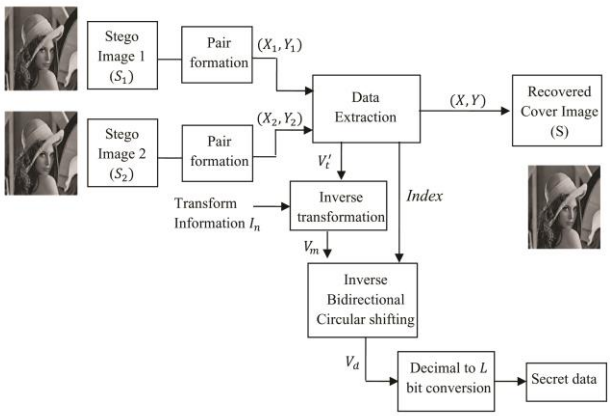


Figure 10. Process flow of data extraction and cover image recovery phase.

From S_1 and S_2 non-overlapping pair is formed which is represented as (X_1, Y_1) and (X_2, Y_2) respectively. From the stego pixel pair (X_1, Y_1) and (X_2, Y_2) the adjusted transformed values V_t' and the index can be calculated as,

$$V_t' = X_2 - X_1 \tag{28}$$

$$I = Y_2 - Y_1 \tag{29}$$

The transformed values V_t can be estimated from the adjusted transformed values V_t' using Equation (30)

$$V_t = \begin{cases} 2 \times |V_t'| - 1 & if\ V_t' < 0 \\ 2V_t' & if\ V_t' \geq 0 \end{cases} \tag{30}$$

Cover pixels' pair (X, Y) can be extracted from the stego pixels pairs (X_1, Y_1) and (X_2, Y_2) is computed using the Equation (31). Cover image is reconstructed from pixel pair (X, Y).

$$X = \left\lfloor \frac{X_1 + X_2}{2} \right\rfloor, Y = \left\lfloor \frac{Y_1 + Y_2}{2} \right\rfloor \tag{31}$$

The transform information I_n can be extracted from the first 2^{L-1} cover pixels of the two stego-images.

$$I_n = P_2 - P_1 \tag{32}$$

Based on the transform information I_n transform the transformed values V_t to minimum value V_m. The inverse bidirectional circular shifting block will estimate the decimal data V_d from the minimum value V_m by estimating the clockwise position (P_c) and anticlockwise position (P_{ac}) where the decimal data V_d is selected by the index value (I). Consider a pair of value (x, y), the clockwise position (P_c) can be estimated by calculating the number of clockwise shifts required to reach y from x in clockwise direction. Similarly, the anticlockwise position (P_{ac}) can be estimated by calculating the number of anti-clockwise shifts required to reach y from x in anti-clockwise direction as depicted in Figure 11.

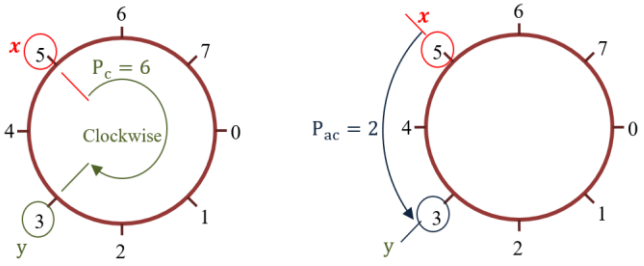


Figure 11. Estimation of clockwise and anti-clockwise position.

Consider an example where a pair of value is (5, 3) as shown in Figure 11. To obtain the clockwise position (P_c), the number of shifts needed to reach 3 from 5 in clockwise direction is 6, therefore P_c=6. Similarly, to obtain the anti-clockwise position (P_{ac}), the number of shifts needed to reach 3 from 5 in anti-clockwise direction is 2, therefore P_{ac}=2.Without performing actual count operation in clockwise and anticlockwise direction the clockwise position (P_c) and anti-clockwise position (P_{ac}) can be calculated from x and y as,

$$P_c = \begin{cases} c_1 & if\ y \geq x \\ 2^L + c_1 & if\ y < x \end{cases} \tag{33}$$

Where c_1=y-x

$$P_{ac} = \begin{cases} c_2 & if\ x \geq y \\ 2^L + c_2 & if\ x < y \end{cases} \tag{34}$$

Where c_2=y-x

The inverse bidirectional circular shifting is performed as follows.

Let the decimal data to be estimated be $V_d^{(i)} = \{V_d^{(1)}, V_d^{(2)}, \ldots\ldots\ldots V_d^{(n)}\}$. Here n represents the count of decimal data. Initialize the initial decimal data set as '0', i.e., $V_d^{(0)} = 0$. Therefore, the decimal data to be estimated becomes $V_d^{(i)} = \{V_d^{(0)}, V_d^{(1)}, V_d^{(2)}, \ldots\ldots\ldots V_d^{(n)}\}$. Let the minimum value V_m be represented as $V_m^{(i)} = \{V_m^{(1)}, V_m^{(2)}, \ldots\ldots\ldots V_m^{(n)}\}$. Consider index, $I^{(i)}$={$I^{(1)}$, $I^{(2)}$, …, $I^{(n)}$}. The decimal data $V_d^{(i)}$ can be calculated using following steps.

- *Step* 1: Initialize the value of i=1.
- *Step* 2: Find the clockwise position (P_c) and anticlockwise position between (P_{ac}) with the pair

($V_d^{(i-1)}$, $V_m^{(i)}$) using Equations (33) and (34) respectively.

- *Step* 3: Find $V_d^{(i)}$ using Equations (35)

$$V_d^{(i)} = \begin{cases} V_m^{(i)} & if\ I^{(i)} = 0 \\ P_c & if\ I^{(i)} = 1 \\ P_{ac} & if\ I^{(i)} = -1 \end{cases} \quad (35)$$

- *Step* 4: Increment *i* by 1 and repeat step 2 and step 3 till *i=n* to obtain the decimal data $V_d^{(i)} = \{V_d^{(0)}, V_d^{(1)}, V_d^{(2)}, \dots\dots\dots V_d^{(n)}\}$. Discard the initial decimal data $V_d^{(0)}$ to obtain the required decimal data $V_d^{(i)} = \{V_d^{(1)}, V_d^{(2)}, \dots\dots\dots V_d^{(n)}\}$.

Convert the decimal data to *L* bit binary and compute the secret data as depicted in Figure 12.

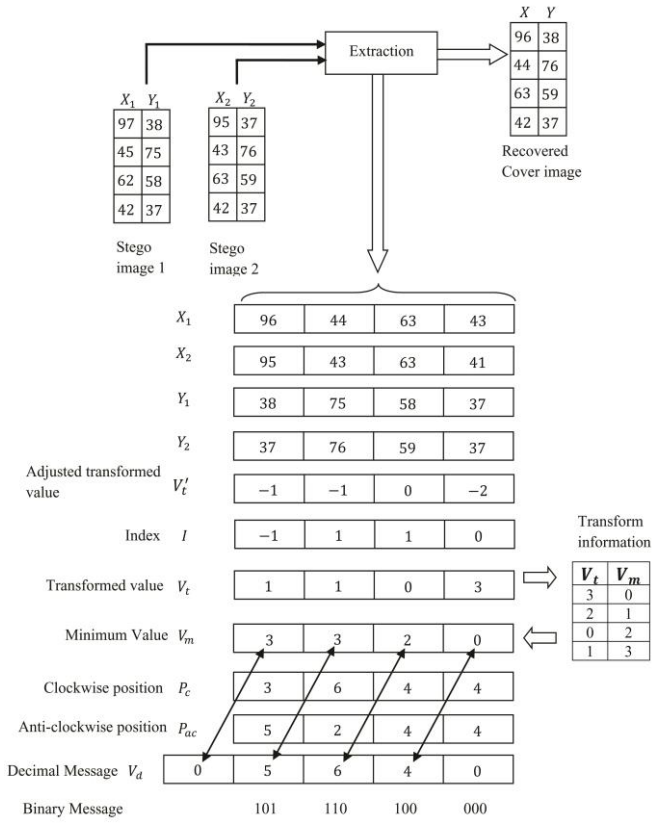


Figure 12. An example for data extraction by the proposed method.

Consider an example where the first stego image has X_1={97, 45, 62, 42} and Y_1={38, 75, 58, 37}. The second stego image has X_2={95, 43, 63, 42} and Y_2={37, 76, 59, 37}. Using Equations (28) and (29), the adjusted transformed value V_t' and Index *I* can be calculated as $V_t' = \{-1, -1, 0, -2\}$ and *I*={-1, 1, 1, 0}. The transformed value V_t is obtained from the adjusted transformed value V_t' using Equation (28) as V_t={1, 1, 0, 3}. Using the transformation information I_n perform inverse histogram transform to obtain V_m from V_t. i.e., 3, 2 0 and 1 are transformed to 0, 1, 2 and 3 respectively. Therefore the minimum value V_m can be calculated as V_m={3, 3, 2, 0}. Using the minimum value V_m={3,3,2,0} obtain the decimal values V_d using the above inverse Bidirectional circular shifting algorithm. The decimal value can be calculated as V_d={5, 6, 4, 0}. Converting the decimal values to L=3 bit binary numbers the secret data can be recovered as {101110100000}. (*X*, *Y*) is obtained by using Equation (31) as *X*={96, 44, 63, 42}, *Y*={38, 76, 59, 37} from which the stego image *S* can be recovered as {96, 38, 44, 76, 63, 59, 42, 37}.

The data embedding steps are summarized below:

- Input: Secret data, cover image
- Output: Stego-images
  - *Step* 1: Find the embeddable pixels
  - *Step* 2: Group the secret data into *L* bits and find the decimal. Let the decimal data be V_d. Also, initialize first secret decimal value as 0.
  - *Step* 3: Find the clockwise value (V_c) and anticlockwise value (V_{ac}) using Equation (16) and (17) respectively with two adjacent decimal secret data.
  - *Step* 4: Find the minimum value (V_m) from V_d , V_c and V_{ac} using Equation (19)
  - *Step* 5: The minimum value (V_m) is transformed to transformed value (V_t) using histogram-based transformation.
  - *Step* 6: Estimate the index *I* for every minimum value based on Equation (20)
  - *Step* 7: Find the adjusted transformed value V_t' from transformed value using Equation (21)
  - *Step* 8: Form the cover pixel pair (*X*, *Y*) from the cover image.
  - *Step* 9: Embed the adjusted transformed value V_t' in the first intensity *X* of the pair (*X*, *Y*) using Equation (22) and (23) to get the stego pixel X_1 and X_2 respectively.
  - *Step* 9: Embed the index *I* in the second intensity *Y* of the pair (*X*, *Y*) using Equation (24) and (25) to get the stego pixels Y_1 and Y_2 respectively.
  - *Step* 10: Also embed the histogram transformation information I_n on the first 2^{L-2} cover pixel using (26) and (27). The first 2^{L-2} cover pixel should not be used to embed the adjusted transformed value V_t' or index *I*.
  - *Step* 11: Construct the stego-image1 using the stego pixels X_1 and Y_1. Similarly construct the stego-image2 using the stego pixels X_2 and Y_2.

The data extraction procedure is explained below;

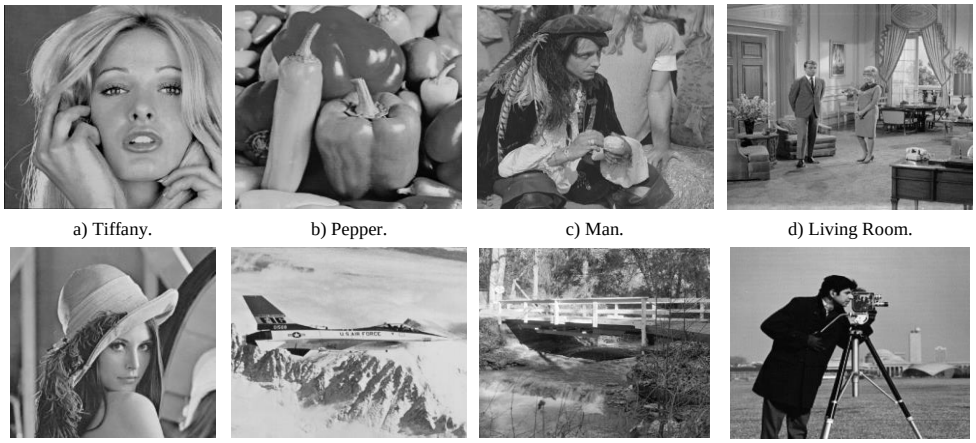
- Input: Stego-images
- Output: Secret data, cover image
  - *Step* 1: From (*X*, *Y*) compute the stego-image1 and the stego image 2. Let the pixel pair be (X_1, Y_1) and (X_2, Y_2) respectively.
  - *Step* 2: Calculate the adjusted transformed value V_t' from X_1 and X_2 using Equation (28).
  - *Step* 3: Calculate the index *I* from Y_1 and Y_2 using Equation (29).

- *Step* 4: Estimate the transformed value V_t from V_t' using Equation (30).
- *Step* 5: Using Equation (32) extract the histogram-based transform information (I_n) from first 2^{L-2} stego pixels of stego image 1 and 2
- *Step* 6: Based on the transform information (I_n) perform histogram based inverse transformation to obtain the minimum value V_m.
- *Step* 7: From the minimum value V_m and index I obtain the decimal data using inverse bidirectional circular shifting
- *Step* 8: Convert the decimal data to L bit binary data.
- *Step* 9: Using Equation (31) reconstruct the original cover image pixels (X, Y).

The next section discusses the experimental results of the paper.

4. Experimental Results

The implemented data hiding technique was realized through MATLAB, and experimental validation was conducted using eight standard grayscale cover test images, each with dimensions of 512×512. The test cover images [12] and test secret images [13] are illustrated in Figures 13 and 14, respectively. This feature enables active steganalysis of LSB-based steganography in grayscale photos with minimal complexity [22].



a) Tiffany. b) Pepper. c) Man. d) Living Room.

e) Lena. f) Jet Plane. g) Camera Man. h) Bridge.

Figure 13. Cover test images.

a) Baboon. b) Houses. c) Boats. d) Barbara. e) Aerial.

Figure 14. Secret images.

The effectiveness of the suggested data hiding method was assessed by measuring parameters such as Peak Signal-To-Noise Ratio (PSNR) and embedding capacity Bits Per Pixel (BPP).

$$PSNR_1 = 10\log_{10}\left[\frac{255^2}{\frac{1}{H\times W}\sum_{j=1}^{H}\sum_{k=1}^{W}(S-S_1)^2}\right]dB \quad (36)$$

$$PSNR_2 = 10\log_{10}\left[\frac{255^2}{\frac{1}{H\times W}\sum_{j=1}^{H}\sum_{k=1}^{W}(S-S_2)^2}\right]dB \quad (37)$$

Where, $H\times W$ is the dimensions of S, S_1 and S_2 are the two stego-images. The average PSNR is computed using Equation (38) and *bpp* is computed using Equation (39):

$$PSNR_{avg} = \frac{1}{2}(PSNR_1 + PSNR_2) \quad (38)$$

$$bpp = \frac{T}{2\times H\times W} \quad (39)$$

Where, T represents the total number of bits hidden in the two stego-images.

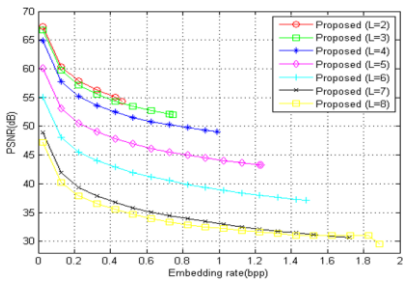
Table 3 presents the performance comparison for various sizes of L. When L is set to 2, the PSNR exceeds 54dB, and the total embedding capacity is approximately 262, 144 bits. For L equal to 4, the PSNR surpasses 49dB, and the total embedding capacity is approximately 524,286 bits. If L is chosen as 7, the PSNR is more than 30 dB, with a total embedding capacity of approximately 903,168 bits.

Figure 15, shows the graphical comparison between embedding rate bpp and PSNR for different group size L. The PSNR for group size L=2 provides a higher PSNR. The group size L=3 has a PSNR close to the PSNR of the group size L=2, but the maximum

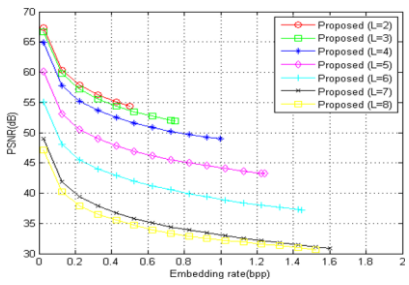
embedding capacity for group size L=3 is 0.25 higher than L=2 As L increases, the embedding rate shows an approximate increase of 0.25 bpp. For L=7 and L=8, the embedding capacity is approximately 1.8bpp. The difference between the PSNR for L=5 and L=6 is approximately 5dB around the embedding rate of 1bpp.

Table 3. Comparison of performance for various L size.

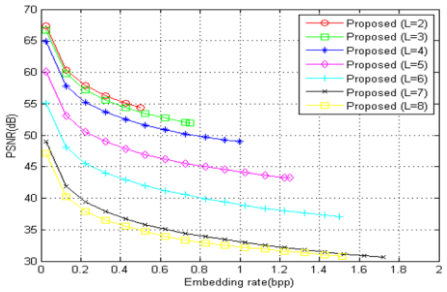
L	Metrics	Tiffany	Pepper	Man	Living room	Lena	Jet Plane	Cameraman	Bridge
2	$PSNR_1$	55.7767	55.7458	55.7453	55.7715	55.7453	55.7453	55.9038	55.7454
	$PSNR_2$	52.9821	52.951	52.9506	52.9753	52.9506	52.9506	53.1171	52.9507
	$PSNR_{avg}$	54.3794	54.3484	54.3479	54.3734	54.3479	54.3479	54.5105	54.348
	Capacity	260229	262126	262144	260597	262144	262144	252196	262139
3	$PSNR_1$	53.1515	53.1153	53.1147	53.1431	53.1147	53.1147	53.3069	53.1188
	$PSNR_2$	50.8335	50.7967	50.796	50.8244	50.796	50.796	50.984	50.8005
	$PSNR_{avg}$	51.9925	51.956	51.9553	51.9838	51.9553	51.9553	52.1454	51.9596
	Capacity	389583	393159	393215	390338	393215	393216	374772	392814
4	$PSNR_1$	49.1303	49.0924	49.0894	49.1212	49.0894	49.0894	49.3216	49.0966
	$PSNR_2$	49.0231	48.9729	48.9699	49.0122	48.9699	48.9699	49.2675	48.9789
	$PSNR_{avg}$	49.0767	49.0326	49.0297	49.0667	49.0297	49.0296	49.2946	49.0378
	Capacity	518506	523972	524286	519656	524286	524288	494596	523284
5	$PSNR_1$	43.4313	43.4161	43.3793	43.4217	43.3793	43.3793	43.6483	43.3008
	$PSNR_2$	43.1649	43.15	43.114	43.1553	43.114	43.114	43.3823	43.1353
	$PSNR_{avg}$	43.2981	43.283	43.2467	43.2885	43.2467	43.2467	43.5153	43.268
	Capacity	647718	650108	655358	649248	655358	655360	616228	652210
6	$PSNR_1$	37.1203	37.2347	37.0632	37.1372	37.0638	37.0632	37.5052	37.1438
	$PSNR_2$	37.1249	37.2396	37.0674	37.1414	37.0681	37.0674	37.5708	37.1484
	$PSNR_{avg}$	37.1226	37.2371	37.0653	37.1393	37.0659	37.0653	37.508	37.1461
	Capacity	776346	757413	786417	773418	786309	786405	711984	772314
7	$PSNR_1$	30.6049	30.9222	30.5985	30.7227	30.5669	30.5579	31.5745	30.8846
	$PSNR_2$	30.5804	30.8973	30.574	30.6967	30.5423	30.5332	31.5509	30.8595
	$PSNR_{avg}$	30.5927	30.9097	30.5862	30.7097	30.5546	30.5455	31.5627	30.872
	Capacity	902069	840648	903168	879214	909265	910567	730674	847648
8	$PSNR_1$	29.5813	30.6935	30.7291	30.147	30.6252	33.2527	30.7955	30.9726
	$PSNR_2$	29.5727	30.6828	30.7216	30.1343	30.6144	33.2348	30.7852	30.9643
	$PSNR_{avg}$	29.577	30.6882	30.7254	30.1406	30.6198	33.2438	30.7904	30.9685
	Capacity	988520	798960	793440	888236	809332	385472	783108	753256



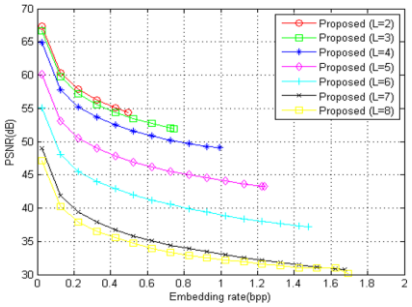
a) PSNR for cover image Tiffany.



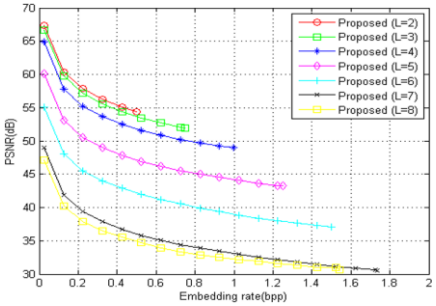
b) PSNR for cover image Pepper.



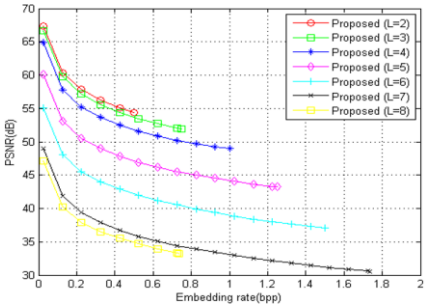
c) PSNR for cover image Man.



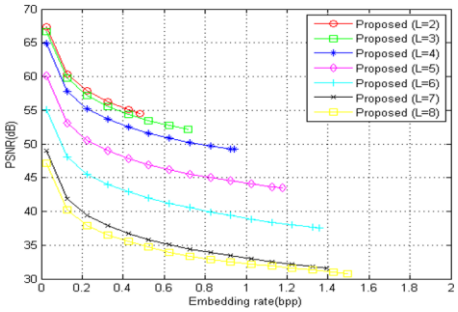
d) PSNR for cover image Living Room.



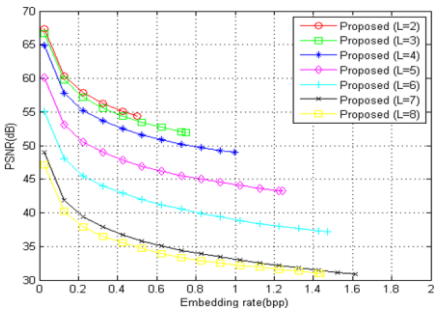
e) PSNR for cover image Lena.



f) PSNR for cover image Jet Plane.



g) PSNR for cover image Camera Man.



h) PSNR for cover image Bridge.

Figure 15. Performance comparison of PSNR for various embedding rate and group size L.

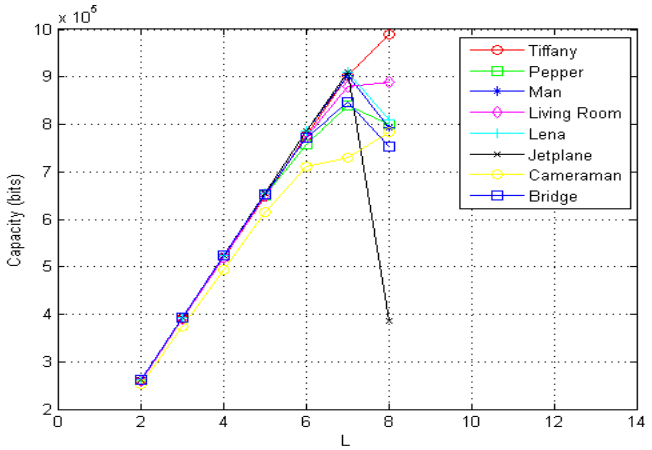


Figure 16. Embedding capacity of various cover images for different group size L.

Figure 16, illustrates the comparison of embedding capacity for different group sizes (L). The embedding capacity demonstrates an increase as the value of L rises from 2 to 7. Notably, the embedding capacity increases almost linearly as L escalates from 2 to 6. The embedding capacity decreases as the group size is changed from L=7 to *L*=8 for the image Pepper, Man, Lena, Jet Plane and Bridge. As the group size is increased from L=8 to *L*=9, bpp reduces due to the decrease in the number of embeddable pixels.

For the comparison of performance, the group size L was set to 4 as depicted in Table 4. When L=4, the PSNR of the 1st and 2nd stego-images are greater than 49 dB which is higher than all the traditional methods. The value of bpp for L=4 is almost equal to the Chang *et al.* [4] and Jung [14] but the PSNR of these two methods was 4dB less than the proposed method. The bpp is less than Chang *et al.* [5], Qin *et al.* [19] and Wang *et al.* [25]. However, the PSNR of these methods are significantly lower compared to proposed scheme.

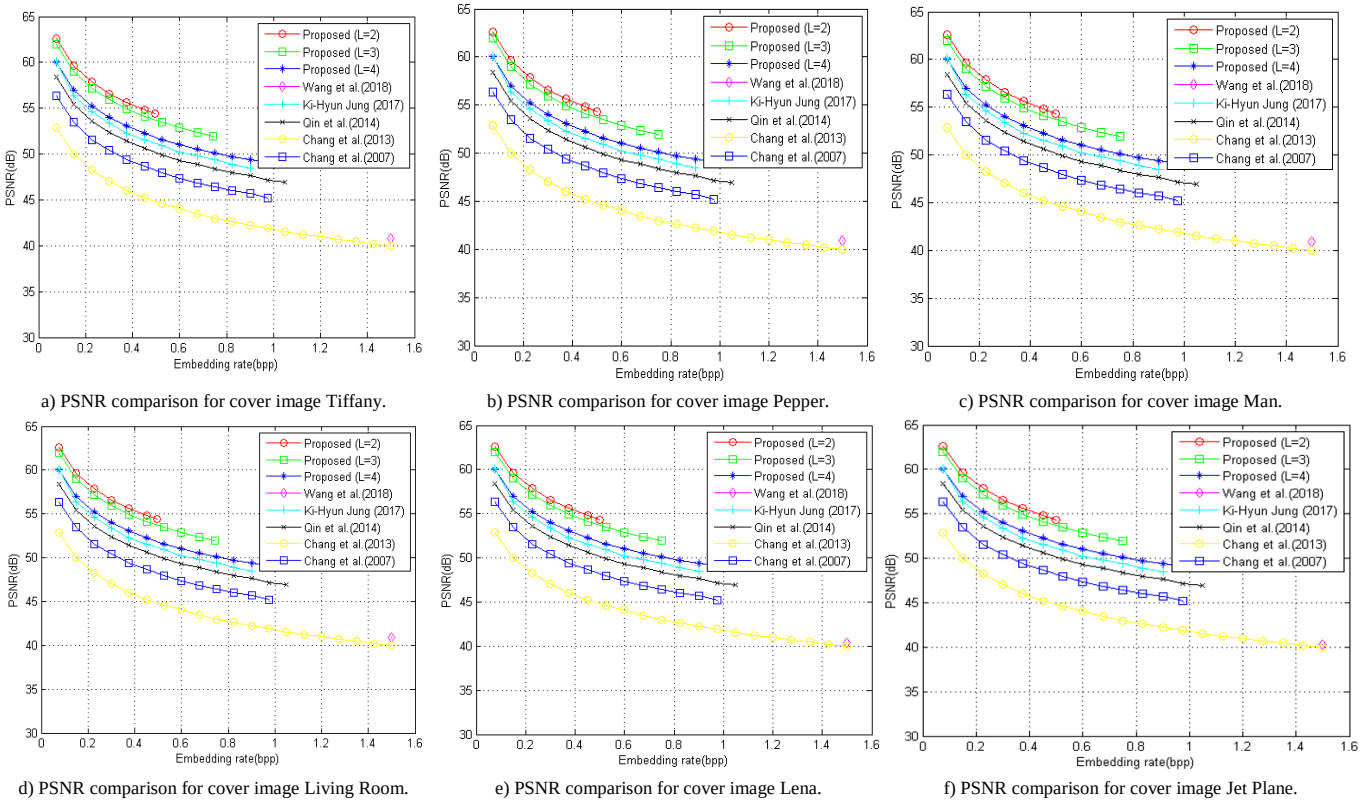
Table 4. Comparison of performance.

Method	Metric	Tiffany	Pepper	Man	Living room	Lena	Jet Plane	Cameraman	Bridge
Chang *et al.* [4]	$PSNR_1$	45.13	45.14	45.13	45.13	45.12	45.11	45.58	45.12
	$PSNR_2$	45.11	45.15	45.14	45.11	45.13	45.13	45.13	45.13
	PSNRavg	45.12	45.15	45.14	45.12	45.13	45.12	45.36	45.13
	Capacity	524,196	523,356	524,288	524,288	524,288	524,148	524,288	524,284
Chang *et al.* [5]	$PSNR_1$	39.89	39.94	39.9	39.89	39.89	39.91	39.88	39.89
	$PSNR_2$	39.89	39.94	39.9	39.89	39.89	39.91	39.88	39.89
	PSNRavg	39.89	39.94	39.9	39.89	39.89	39.91	39.88	39.89
	Capacity	802,535	799,684	802,698	802,888	802,895	802,524	802,789	802,716
Qin *et al.* [19]	$PSNR_1$	52.06	51.25	52.12	52.12	52.11	52.04	51.72	52.11
	$PSNR_2$	41.57	41.52	41.58	41.58	41.58	41.56	41.75	41.57
	PSNRavg	46.82	46.39	46.85	46.85	46.85	46.8	46.74	46.84
	Capacity	557,129	557,245	557,194	557,339	557,052	557,096	557,264	557,194
Jung [14]	$PSNR_1$	48.03	48.12	48.18	44.64	48.15	48.18	48.18	48.18
	$PSNR_2$	47.04	47.12	47.18	43.79	47.26	47.2	47.19	47.2
	PSNRavg	47.535	47.62	47.68	44.215	47.705	47.69	47.685	47.69
	Capacity	519180	519180	519180	519180	519180	519180	519180	519180
Wang *et al.* [25]	$PSNR_1$	42.69	42.72	42.71	42.69	42.7	42.63	42.71	42.53
	$PSNR_2$	38.99	39.15	39.01	39.12	38.09	38.01	39.01	38.01
	PSNRavg	40.84	40.935	40.86	40.905	40.395	40.32	40.86	40.27
	Capacity	786432	786432	786432	786432	786432	786432	786432	786432
Proposed (L=4)	$PSNR_1$	49.1303	49.0924	49.0894	49.1212	49.0894	49.0894	49.3216	49.0966
	$PSNR_2$	49.0231	48.9729	48.9699	49.0122	48.9699	48.9699	49.2675	48.9789
	PSNRavg	49.0767	49.0326	49.0297	49.0667	49.0297	49.0296	49.2946	49.0378
	Capacity	518506	523972	524286	519656	524286	524288	494596	523284

Figure 17, displays experimental results with group sizes set at L=2, L=3, and L=4. The PSNR for the proposed method is observed to be higher than that of the conventional methods. While the PSNR for L=4 is close to the Jung's method [14], proposed scheme still attains a PSNR 2dB superior to Jung's scheme [14]. The graphical comparison emphasizes the superiority of proposed scheme in contrast with conventional schemes.

Figure 18, illustrates the histogram of stego images for L=4 and L=5, using the cover image "Tiffany." The histogram plots for the stego images exhibit a similarity to that of the original cover image. Abd-El-Atty [2] an average PSNR of 44.1 and a payload capacity of 2 bits per 1 byte, the proposed mechanism is large enough for the human eye to not distinguish between the carrier image and its stego one. Shaji and Sam [20] the dual stego images are obtained by embedding the encoded indices on the cover image, which encodes the message intensities. This observation indicates that the secret data introduces minimal changes to the intensity of cover image pixels. The proposed algorithm demonstrates versatility beyond experimental test images and performs well with practical images as well.

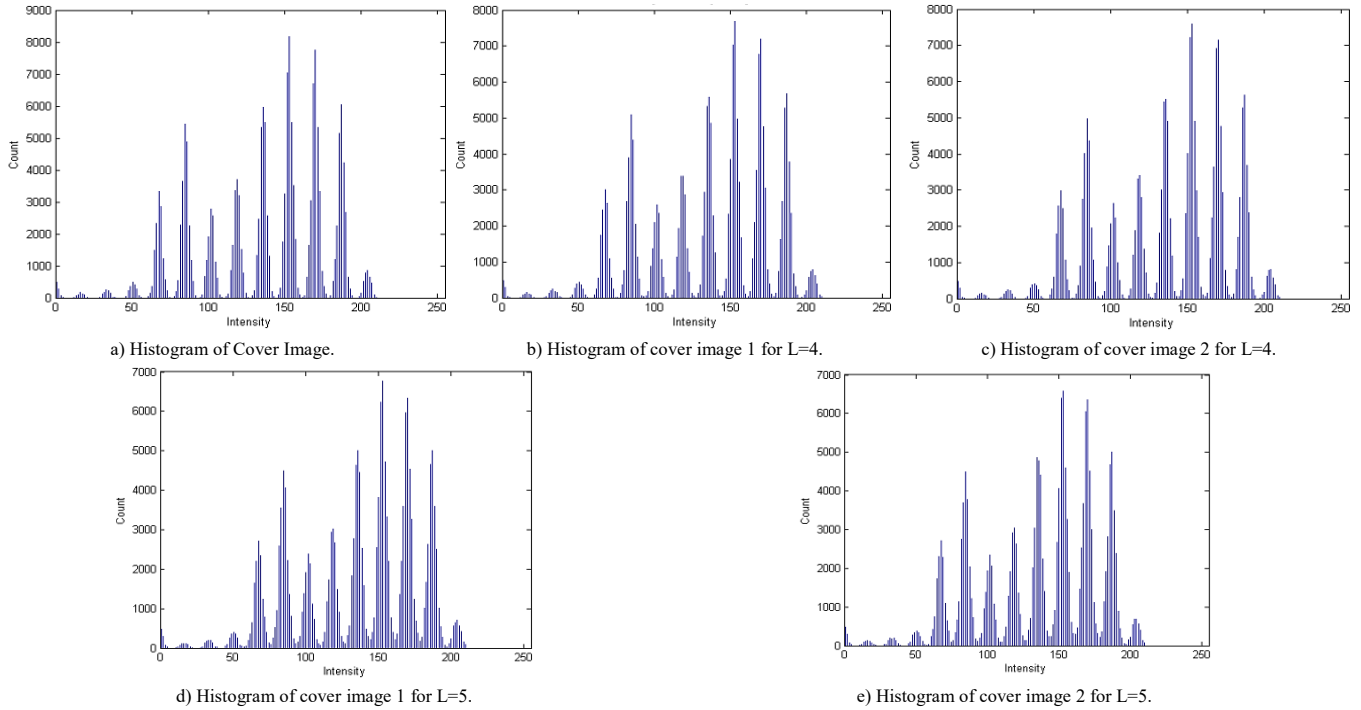
For the comparison of performance, the group size L was set to 4 as depicted in Table 4. When L=4, the PSNR of the 1st and 2nd stego-images are greater than 49 dB which is higher than all the traditional methods. The value of bpp for L=4 is almost equal to the Chang *et al.* [4] and Jung [14] but the PSNR of these two methods was 4dB less than the proposed method. The bpp is less than Chang *et al.* [5], Qin *et al.* [19] and Wang *et al.* [25]. However, the PSNR of these methods are significantly lower compared to proposed scheme.



a) PSNR comparison for cover image Tiffany. b) PSNR comparison for cover image Pepper. c) PSNR comparison for cover image Man.

d) PSNR comparison for cover image Living Room. e) PSNR comparison for cover image Lena. f) PSNR comparison for cover image Jet Plane.

Figure 17. Performance comparison of Proposed and conventional methods.



a) Histogram of Cover Image. b) Histogram of cover image 1 for L=4. c) Histogram of cover image 2 for L=4.

d) Histogram of cover image 1 for L=5. e) Histogram of cover image 2 for L=5.

Figure 18. Histogram of cover image and stego images.

5. Conclusions

This paper introduced a novel reversible data hiding approach utilizing dual stego-images and employing bidirectional circular shifting. The method intricately involves the estimation of clockwise and anticlockwise values derived from the secret data, with subsequent determination of the minimum value. A cover pixel pair is then constructed from the cover image, wherein the index of the minimum value is embedded in the second component of the pair. The minimum value undergoes a transformation based on the histogram of the secret data, resulting in a transformed value. This transformed value is carefully adjusted and embedded into the first component of the cover pixel pair. During the extraction process, the adjusted transformed value and index are precisely estimated from the stego-pixel pair, facilitating the recovery of the secret data. Experimental results demonstrate impressive performance in terms of image

quality. This research contributes a robust and reliable reversible data hiding technique that proves to be particularly advantageous for preserving image fidelity while concealing sensitive information.

Acknowledgment

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

References

[1] AbdAl-Hameed S., Abdullah H., Khalf N., and Alghazo J., "An Enhanced Steganography Approach for Concealing Audio in Images Using Double Density-Dual Tree Wavelet Transform," *Revue d'Intelligence Artificielle*, vol. 37, no. 5, pp. 1237-1244, 2023. https://doi.org/10.18280/ria.370516

[2] Abd-El-Atty B., "A Robust Medical Image Steganography Approach Based on Particle Swarm Optimization Algorithm and Quantum Walks," *Neural Computing and Applications*, vol. 35, no. 1, pp. 773-785, 2023. https://doi.org/10.1007/s00521-022-07830-0

[3] Cao X., Du L., Wei X., Meng D., and Guo X., "High-Capacity Reversible Data Hiding in Encrypted Images by Patch-Level Sparse Representation," *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1132-1143, 2015. DOI:10.1109/TCYB.2015.2423678

[4] Chang C., Kieu T., and Chou Y., "Reversible Data Hiding Scheme Using Two Steganographic Images," *in Proceedings of the 10th IEEE Region Conference*, Taipei, pp. 1-4, 2007. https://ieeexplore.ieee.org/document/4483783

[5] Chang C., Lu T., Horng G., Huang Y., and Hsu Y., "A High Payload Data Embedding Scheme Using Dual Stego-Images with Reversibility," *in Proceedings of the 9th International Conference on Information, Communications and Signal Processing*, Tainan, pp. 1-5, 2013. DOI:10.1109/ICICS.2013.6782790

[6] Chhajed G., Deshmukh K., and Kulkarni T., "Review on Binary Image Steganography and Watermarking," *International Journal on Computer Science and Engineering*, vol. 3, no. 11, pp. 3645, 2011. https://www.researchgate.net/publication/285812082

[7] Dong L., Zhou J., Tang Y., and Liu X., "Estimation of Capacity Parameters for Dynamic Histogram Shifting (DHS)-based Reversible Image Watermarking," *in Proceedings of the IEEE International Conference on Multimedia and Expo*, Chengdu, pp. 1-6, 2014. DOI:10.1109/ICME.2014.6890276

[8] Dragoi I. and Coltuc D., "Local-Prediction-based Difference Expansion Reversible Watermarking," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1779-1790, 2014. DOI:10.1109/TIP.2014.2307482

[9] Fridrich J. and Kodovsky J., "Rich Models for Steganalysis of Digital Images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868-882, 2012. DOI:10.1109/TIFS.2012.2190402

[10] Goljan M., Fridrich J., and Cogranne R., "Rich Model for Steganalysis of Color Images," *in Proceedings of the IEEE International Workshop on Information Forensics and Security*, Atlanta, pp. 185-190, 2014. DOI:10.1109/WIFS.2014.7084325

[11] Hong W., Chen T., and Shiu C., "Reversible Data Hiding for High Quality Images Using Modification of Prediction Errors," *Journal of Systems and Software*, vol. 82, no. 11, pp. 1833-1842, 2009. https://doi.org/10.1016/j.jss.2009.05.051

[12] Image Processing Place, Image Databases, Test Cover Images, https://www.imageprocessingplace.com/root_files_V3/image_databases.htm, http://www.imageprocessingplace.com/downloads_V3/root_downloads/image_databases/standard_test_images.zip, Last Visited, 2024.

[13] Image Repository-University of Waterloo, Test Secret Images, http://links.uwaterloo.ca/Repository.html, Last Visited, 2024.

[14] Jung K., "Authenticable Reversible Data Hiding Scheme with Less Distortion in Dual Stego-Images," *Multimedia Tools and Applications*, vol. 77, pp. 6225-6241, 2018. https://doi.org/10.1007/s11042-017-4533-0

[15] Li X., Zhang W., Gui X., and Yang B., "Efficient Reversible Data Hiding based on Multiple Histograms Modification," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 2016-2027, 2015. DOI:10.1109/TIFS.2015.2444354

[16] Lu T., Tseng C., and Wu J., "Dual Imaging-based Reversible Hiding Technique Using LSB Matching," *Signal Processing*, vol. 108, pp. 77-89, 2015. https://doi.org/10.1016/j.sigpro.2014.08.022

[17] Ma B., Li K., Xu J., Wang C., and Li X., "A High-Performance Image Steganography Scheme Based on Dual-Adversarial Networks," *IEEE Signal Processing Letters*, vol. 31, pp. 2655-2659, 2024. DOI:10.1109/LSP.2024.3440176

[18] Mei Q., Wong E., and Memon N., "Data Hiding in Binary Text Documents," *in Proceedings of the International Society for Optical* Engineering, San

Jose, pp. 369-375, 2001. DOI:10.1117/12.435420
[19] Qin C., Chang C., and Hsu T., "Reversible Data Hiding Scheme Based on Exploiting Modification Direction with Two Steganographic Images," *Multimedia Tools and Applications*, vol. 74, pp. 5861-5872, 2015. https://doi.org/10.1007/s11042-014-1894-5
[20] Shaji C. and Sam I., "Dual Encoding Approach with Sequence Folding for Reversible Data Hiding in Dual Stego Images," *Multimedia Tools and Applications*, vol. 80, no. 9, pp. 13595-13614, 2021. https://doi.org/10.1007/s00521-022-07830-0
[21] Shi Y., Sutthiwan P., and Chen L., "Textural Features for Steganalysis," *in Proceedings of the 14th International Conference on Information Hiding*, Berkeley, pp. 63-77, 2012. https://doi.org/10.1007/978-3-642-36373-3_5
[22] Thanasekaran V. and Selvaraj A., "Low Dimensional Multi Class Steganalysis of Spatial LSB Based Stego Images Using Textural Features," *The International Arab Journal of Information Technology*, vol. 21, no. 2, pp. 233-242, 2024. https://doi.org/10.34028/iajit/21/2/6
[23] Tseng Y., Chen Y., and Pan H., "A Secure Data Hiding Scheme for Binary Images," *IEEE Transactions on Communications*, vol. 50, no. 8, pp. 1227-1231, 2002. DOI:10.1109/TCOMM.2002.801488
[24] Wang C., Li X., and Yang B., "Efficient Reversible Image Watermarking by Using Dynamical Prediction-Error Expansion," *in IEEE International Conference on Image Processing*, Hong Kong, pp. 3673-3676, 2010. DOI:10.1109/ICIP.2010.5652508
[25] Wang Y., Shen J., and Hwang M., "A Novel Dual Image-based High Payload Reversible Hiding Technique Using LSB Matching," *International Journal of Network Security*, vol. 20, no. 4, pp. 801-804, 2018. http://ijns.jalaxy.com.tw/contents/ijns-v20-n4/ijns-2018-v20-n4-p801-804.pdf
[26] Wu M. and Liu B., "Data Hiding in Binary Image for Authentication and Annotation," *IEEE Transactions on Multimedia*, vol. 6, no. 4, pp. 528-538, 2004. DOI:10.1109/TMM.2004.830814
[27] Wu X. and Sun W., "High-Capacity Reversible Data Hiding in Encrypted Images by Prediction Error," *Signal Processing*, vol. 104, pp. 387-400, 2014. https://doi.org/10.1016/j.sigpro.2014.04.032
[28] Zhang X. and Wang S., "Efficient Steganographic Embedding by Exploiting Modification Direction," *IEEE Communications Letters*, vol. 10, no. 11, pp. 781-783, 2006. DOI:10.1109/LCOMM.2006.060863

Mujeebudheen Khan Immamuddinkhan is currently pursuing Ph.D. at Noorul Islam Centre for Higher Education in Kumarakovil, Tamil Nadu, and holds an M.E. from Anna University. He has 14 years of teaching experience and 3 years of research experience, specializing in steganography, MANETs, and blockchain. He holds 1 patent and has published research papers in Scopus Indexed Journals and reviewed journals. Mujeebudheen Khan A.I. has delivered expert talks and addresses on diverse topics in various Short-Term Training Programmes/Faculty Development Programmes.

Siva Sankar Kanahasabapathy has completed his Ph.D. from M.S. University and his M.E. from Annamalai University. He has 18 years of teaching experience and 9 years of research experience, specializing in System Software. He has published 2 books/book chapters and holds 2 patents. He has published 54 research papers in Web of Science/Scopus Indexed Journals and 17 papers in UGC CARE listed, refereed, and reviewed journals. Dr. Siva Sankar K has guided 9 Ph.D. scholars and is currently guiding 6 Ph.D. scholars. He has delivered 5 expert talks and addresses on diverse topics in various AICTE/other sponsored short-term training programmes/Faculty Development Programmes. He has visited numerous places in connection with research/technological collaboration.