Scientific Journal Impact Factor (SJIF) : 5.278
International Journal of Modern Trends in Engineering and Research
IJMTER
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

IMPLEMENTATION OF ENCRYPTION BASED LOSSLESS AND REVERSIBLE DATA HIDING

Mr.Amol Narkhede[1] and Prof.Dinesh Patil[2]
[1,2]*Computer Science and Engineering, SSGBCOE&T*

Abstract— Image processing is used to improve the quality of image. Now days, more focus is on reversible data hiding (RDH) in encrypted images, so it maintains the excellent property that the original cover image can be easily recovered without any loss after embedded data. All previous techniques embed data by reversibly vacating room from the encrypted images, which may be leads to some errors on data extraction and image restoration. In this paper, we propose a method by reserving room before encryption with a RDH algorithm and LSB algorithm, and thus it is easy for the user to reversibly embed data in the encrypted image. The proposed system is reversible, that is, data extraction and image recovery is without any loss. A reversible data hiding algorithm and LSB algorithm, which can recover the original image without any loss from marked image after the hidden data have been extracted. Experiments show that this RDH method can embed more than 10times as large payloads for the same image quality as the previous methods, such as for PSNR dB.
Keywords— Reversible Data Hiding, Reserving Room, Vacating Room, Blowfish Algorithm Least Significant Bit Algorithm, Image/Data Embedding, Encryption/Decryption, Hide Key/Embed Key

I. INTRODUCTION

In the modern era of internet digital communication has been increased rapidly, therefore the protection of digital data is very important for many applications, such as confidential transmission, video surveillance, military and medical field application. Several methods have been proposed and investigated in the literature to provide privacy for communication. To decrease the transmission time, the data compression is necessary Data hiding technique conceals the secret message into cover image, where the image embedded with secret message is called stego-image. Then this stego-image is being transmitted to prevent the other party from modifying, intercepting, and tampering, thus protecting the data.The protection of multimedia data can be done with the compression, encryption and data hiding. Two main groups of technologies have been developed for this purpose. The first one is based on content protection through encryption. The second group bases the protection on data hiding, aimed at secretly embedding a message into the data. There are again two major research areas in data hiding techniques: irreversible data hiding and reversible data hiding. Irreversible hiding technique cannot recover images back to cover images even after the receiver retrieved the embedded secret message. Such technique holds an extremely high capacity but it destroys images. As for reversible data hiding technique, stego-images can be restored back to the original images after retrieving the embedded secret data with a lower capacity than irreversible method. In literature papers work is proposed to embed data in an encrypted image by using an reversible approach of data hiding. But the possibility of noise contained in the decrypted image. Now the challenge was to find an encryption method robust to noise. This problem can be resolved by the proposed method, reversible data hiding in encrypted by public key cryptosystems with probabilistic and homomorphism properties.We say a data hiding strategy is reversible if the first cover substance can be impeccably recouped from the spread form containing implanted information despite the fact that a slight twisting has been presented in information inserting technique. Various components, for

example, distinction extension, histogram shift and lossless pressure, have been utilized to build up the reversible information concealing procedures.so proposed methodology a combined data hiding schemes for cipher text images encrypted by public key cryptosystems with probabilistic and holomorphic properties. In the lossless scheme, the cipher text pixels are replaced with new values to embed the additional information into several LSB-planes of cipher text pixels by using 4LSB. Then, the embedded data can be directly extracted from the encrypted domain by using blowfish algorithm and the data embedding process does not affect the decryption of original plaintext image file. In the reversible procedure, a pre-handling is utilized to compress the image histogram before image encryption, so that the alteration on encoded image for information embedding does not create any pixel oversaturation in plaintext space. In spite of the fact that a little twisting is presented, the embedded information can be extracted and the first image record can be recover from the straightforwardly decrypted image. Because of the similarity between the lossless and reversible system, the information embedding forms in the two behaviour can be at the same time performed in a encrypted image. With the consolidated new system, a receiver might extricate a piece of embedded data before decoding, and extract other piece of embedded data of the record and recover the first plaintext image after decryption.

II. LITERATURE REVIEW

In paper [1] they proposed data embedding over images has drawn tremendous interest, using either lossy or lossless techniques. Although lossy techniques can allow large hiding capacity, host image cannot be recovered with high fidelity. Some applications require exact recovery of the host image, i.e. in medicine patient data can be embedded without affecting the medical image. In general lossless data hiding techniques suffer from limited capacity as the host image should be kept intact. In this paper a lossless embedding technique is proposed. In this technique image histograms are analyzed to identify the embedding capacity of different image types. Histogram maxima and minima are used in embedding capacity estimation. The proposed technique gives hiding capacity that can reach up to 50% of the host image size for images with large homochromatic regions (cartoons-like)

In paper [2] Current difference-expansion (DE) embedding techniques perform one layer embedding in a difference image. They do not turn to the next difference image for another layer embedding unless the current difference image has no expandable differences left. The obvious disadvantage of these techniques is that image quality may have been severely degraded even before the later layer embedding begins because the previous layer embedding has used up all expandable differences, including those with large magnitude. Based on integer Haar wavelet transform, we propose a new DE embedding algorithm, which utilizes the horizontal as well as vertical difference images for data hiding. We introduce a dynamical expandable difference search and selection mechanism. This mechanism gives even chances to small differences in two difference images and effectively avoids the situation that the largest differences in the first difference image are used up while there is almost no chance to embed in small differences of the second difference image.

In paper [3] Digital watermarking, often referred to as data hiding, has recently been proposed as a promising technique for information assurance. Owing to data hiding, however, some permanent distortion may occur and hence the original cover medium may not be able to be reversed exactly even after the hidden data have been extracted out. Following the classification of data compression algorithms, this type of data hiding algorithms can be referred to as lossy data hiding. It can be shown that most of the data hiding algorithms reported in the literature are lossy. Here, let us examine three major classes of data hiding algorithm. With the most popularly utilized spread-spectrum water- marking techniques, either in DCT domain [1] or block 8x8 DCT domain [2], round- off error and/or truncation error may take place during data embedding. As a result, there is no way to reverse the stago-media back to the original without distortion.

paper [4] present a novel lossless (reversible) data-embedding technique, which enables the exact recovery of the original host signal upon extraction of the embedded information. A generalization of the well-known least significant bit (LSB) modification is proposed as the data-embedding method, which introduces additional operating points on the capacity-distortion curve. Lossless recovery of the original is achieved by compressing portions of the signal that are susceptible to embedding distortion and transmitting these compressed descriptions as a part of the embedded payload. A prediction-based conditional entropy coder which utilizes unaltered portions of the host signal as side-information improves the compression efficiency and, thus, the lossless data-embedding capacity.

III. PRAPOSED SYSTEM

This paper proposes a lossless, a reversible, and a combined data hiding schemes for cipher text images encrypted by public key cryptosystems with probabilistic and holomorphic properties. In the lossless scheme, the cipher text pixels are replaced with new values to embed the additional data into several LSB-planes of cipher text pixels by multi-layer wet paper coding. Then, the embedded data can be directly extracted from the encrypted domain, and the data embedding operation does not affect the decryption of original plaintext image.

The idea is to develop a system capable to transfer confidential data from sender to receiver such a way that no any loss of data in between sending and receiving is possible, If any the total recovery for that data is possible. The aim is to develop a system having sure reliability for sensitive data transmission. The main aim is to focus more on reversible data hiding (RDH) in encrypted images, so it maintains the excellent property that the original cover image can be easily recovered without any loss after embedded data. All previous techniques embed data by reversibly vacating room from the encrypted images, which may be leads to some errors on data extraction and image restoration. In this paper, we propose a method by reserving room before encryption with a RDH algorithm and LSB algorithm, and thus it is easy for the user to reversibly embed data in the encrypted image. The proposed system is reversible, that is, data extraction and image recovery is without any loss in quality and dimension of image. A reversible data hiding algorithm and LSB algorithm, which can recover the original image without any loss from marked image after the hidden data have been extracted. Experiments show that this RDH method can embed more than 10 times as large payloads for the same image quality as the previous methods, such as for PSNR dB.

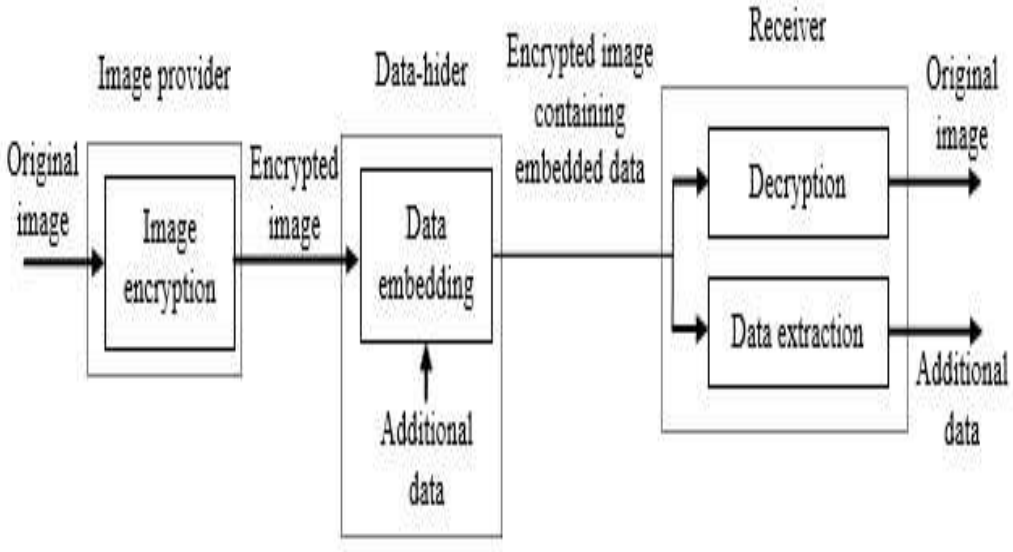

Figure. 3.1 System Architecture

The Proposed system work in a three modules, in first module lossless data hiding scheme will be focused in second module reversible data hiding scheme will be focused and in last and third module the combination of reversible and lossless data hiding is implemented.

3.1 Lossless Data Hiding Scheme

- A lossless data hiding scheme for public-key-encrypted images is proposed. There are three parties in the scheme: an image provider, a data-hider, and a receiver.
- With a cryptosystem possessing probabilistic property, the image provider encrypts each pixel of the original plaintext image using the public key of the receiver, and a data-hider who does not know the original image can modify the cipher text pixel-values to embed some additional data into the encrypted image by multi-layer wet paper coding under a condition that the decrypted values of new and original cipher-text pixel values must be same.
- When having the encrypted image containing the additional data, a receiver knowing the data hiding key may extract the embedded data, while a receiver with the private key of the cryptosystem may perform decryption to retrieve the original plaintext image.
- The embedded data can be extracted in the encrypted domain, and cannot be extracted after decryption since the decrypted image would be same as the original plaintext image due to the probabilistic property

3. 2 Reversible Data Hiding Scheme

- This section proposes a reversible data hiding scheme for public-key-encrypted images. In the reversible scheme, a pre-processing is employed to shrink the image histogram, and then each pixel is encrypted with additive homomorphic cryptosystem by the image provider.
- When having the encrypted image, the data-hider modifies the cipher text pixel values to embed a bit-sequence generated from the additional data and error-correction codes.
- Due to the homomorphic property, the modification in encrypted domain will result in slight increase/decrease on plaintext pixel values, implying that a decryption can be implemented to obtain an image similar to the original plaintext image on receiver side.
- Because of the histogram shrink before encryption, the data embedding operation does not cause any overflow/underflow in the directly decrypted image. Then, the original plaintext image can be recovered and the embedded additional data can be extracted from the directly decrypted image.

3.3 Combined Data Hiding Scheme

- A lossless and a reversible data hiding schemes for public-key-encrypted images are proposed. In both of the two schemes, the data embedding operations are performed in encrypted domain.
- On the other hand, the data extraction procedures of the two schemes are very different. With the lossless scheme, data embedding does not affect the plaintext content and data extraction is also performed in encrypted domain.
- With the reversible scheme, there is slight distortion in directly decrypted image caused by data embedding, and data extraction and image recovery must be performed in plaintext domain.
- That implies, on receiver side, the additional data embedded by the lossless scheme cannot be extracted after decryption, while the additional data embedded by the reversible scheme cannot extracted before decryption.
- In this section, we combine the lossless and reversible schemes to construct a new scheme, in which data extraction in either of the two domains is feasible

IV. TECHNIQUES AND ALGORITHMS USED

A new separable reversible data hiding technique is proposed based on the concept of LSB and Blowfish algorithm. The proposed scheme is also a separable reversible data hiding scheme and is also made up of three phases as we see above. These phases are image encryption, data embedding and data extraction / image recovery. Firstly the content owner encrypts the original uncompressed image using an encryption key to produce an encrypted image. In the proposed scheme we use the blowfish algorithm for encryption and decryption of images, because it provides a stronger security

as compared to other existing encryption algorithm. Then in second phase, the data-hider compresses the least significant bits (LSB) of the encrypted image using a data-hiding key to create a sparse space to accommodate the additional data [9]. For this we use LSB method for data embedding. At the receiver side, the data embedded in the created space can be easily retrieved from the encrypted image containing additional data according to the data-hiding key. Since the data embedding only affects the LSB, a decryption with the encryption key can result in an image similar to the original version. When using both of the encryption and data-hiding keys, the embedded additional data can be successfully extracted and the original image can be perfectly recovered.

In this new proposed scheme we can use both kind of images i.e. grayscale as well as color images as the cover image which was not in the previous methods. The second new thing in this scheme is that we can hide the data i.e. text as well as an image as a data inside the cover image. F

Firstly the sender takes an original uncompressed image as a cover medium in which the secret data is to be transmitted over the network to the receiver. After this, the sender can encrypt the original image using a standard blowfish encryption algorithm. Then sender enters the data to hide into the encrypted image using a LSB method that makes some room inside the image for this secured (additional) data by calculating and compressing the least significant bits of the encrypted pixel. This process is known as data embedding process.

There are three possibilities in the data hiding/embedding process of proposed scheme. The first one is we can embed only an image as a data in encrypted image and send it to the receiver. Second possibility is we can embed a data i.e. only text in an encrypted image and then send it to the receiver. The third possibility is we can embed both data i.e. text as well as image inside encrypted an image. At the receiver side the receiver can extract embedded additional data i.e. text or image or both according to the data hiding key and recover original image according to an encryption key. Now with all this knowledge the complete work of the proposed system is as follows.

A sender who wants to transmit the image called as original uncompressed image first encrypts the image using an encryption key. Then inside this encrypted image, message image or additional data or both can be embedded using a data hiding key to produce a stego image. This stego image consists of three parts, first is the original image, second is the encrypted image and the third is embedded data that may be an image, an additional data or both. After embedding data, at the receiver side, the receiver can perform the reverse operation i. e. decryption and decompression using data hiding and an encryption key to obtain the image similar to the original image without any loss. If the receiver has only the data hiding key then he is able to extract the additional data though he don't know the image content. If the receiver has only the encryption key then he can decrypt the received data to obtain an image similar to the original one, but could not extract the additional data. If the receiver has both the keys i.e. data hiding key and an encryption key then he can extract the embedded additional data as well as can recover the original image without any error.

As stated earlier, the proposed work is consists of three phases i.e. Image Encryption, Data Embedding and Data Extraction and/or Image Recovery. The detail working is as follows:

A. Image Encryption: The first step in the proposed scheme is the image encryption. For that, take one image to which we can call it as an original image. Now this original image is used to hide the message (Text/Image). For this operation, we need to perform an encryption operation on original image. This encryption operation is performed using blowfish algorithm Blowfish Algorithm: Blowfish algorithm is a 64 bit block cipher that contains a variable length key from 32 bit to 448 bits. This algorithm is used in an application where key does not changed often such as an automatic file encryption. It is observed that when this algorithm is implemented on 32 bit microprocessors having large data caches, the performance is faster than other existing encryption algorithm.The blowfish algorithm contains two different parts: one is the key expansion part and the other is data encryption part. It is noted that the key expansion converts a key of at most 448 bits into several subkey arrays around 4168 bytes. The data encryption occurs through 16 round Feistel network and each round consists of a key dependent

permutation and a key, and a data dependent substitution. All the operations are performed by XORs and additions on 32 bit words. Blowfish uses large number of subkeys and can be
precomputed before any data encryption or decryption. The P-array consists of 18 32-bit sub keys from P1 to P18. There are 4 32-bit S-boxes having 256 entries each as follows.
S1,0, S1,1,..., S1,255; S2,0, S2,1,.., S2,255; S3,0, S3,1,..., S3,255; S4,0, S4,1,.., S4,255.
The blowfish has 16 round. Let the input is a 64 bit data element, x. Now the actual algorithm is as follows:
Divide X into two 32-bit halves XL and XR
For i=1 to 16:
XL = XL XOR Pi
XR = F (XL) XOR XR
Swap XL and XR
End for
Swap XL and XR
XR = XR XOR P17
XL = XL XOR P18
Finally, Recombine XL and XR to get the cipher text. Output X (64-bit data block: cipher text)

B. Data Embedding : After image encryption phase, the next phase is data embedding phase in which the secured data (Text/Image) is embedded into an encrypted image by implementing a well known LSB method. In this proposed work, both grayscale and color images are used. If we take grayscale image to embed then the LSB algorithm is working as mentioned in [9]. If we take color image then the working is slightly different. Each pixel in color images will have three planes i.e. Red(R), Green(G) and Blue(B). The pixel values of these color components will be in the range of [0 255]. The message bits are embedded in all these three planes and can be recombined to form the original color image. Here the message bits are embedded in every Red component in the RGB plane [10]. The rest of the work is same as in [9]. All the calculation regarding embedded rate are same as in [3], [9] and [11]. If we take text to embed in an image then working of LSB is same as in [11]. C. Data Extraction/Image Recovery In this last phase, we have to consider the three cases as we discuss earlier i.e. the receiver has only the data hiding key, only an encryption key, and both data hiding as well as encryption keys. If we consider case one i.e. only data hiding key, then the receiver can extract additional data though he don't know the image content. For data extraction the LSB is used. If the text is to be extracted then the logic is same as in [11]. If an image is to be extracted then the logic is same as in [3], [9]. If we consider case two i.e. only an encryption key, then the receiver can decrypt the received data to obtain an image similar to the original image, but cannot extract the additional embedded information. For decryption of image again we use the blowfish algorithm. Here decryption is same as an encryption with only difference is using the Parray. The P1 to P18 are used in reverse order. If we consider case three, then the receiver can extract the additional data and recover the original image without any error. To extract data the method is same as in case one above and to recover the image the method is same as in case two.

V. EXPERIEMENTAL RESULT

In our experiment, we use various images of standard size and different format (.jpg, .bmp, .png) for testing as cover image and hidden image. Here in the experiment, we are interesting in calculating the PSNR (Peak Signal to Noise Ration) of images. The PSNR, in simple language, is nothing but the difference between the two images for example, the difference between Original Image and the decrypted image after extracting the hidden data. It gives the ratio of corrupting noise produced in the original image after extraction of hidden data. From the study of various existing algorithm, it is found that if the difference is very close to zero or near about 1% then the PSNR will be around 38-39 dB. In our experiment, if the images are grayscale images then obtained resulting

PSNR will be nearly same as in [12]. If the images are color images then the resulting PSNR that obtained by our experiment is near around 43 dB. That means PSNR value is improved in our system. Our proposed system is very close to the zero difference according to the PSNR value. The PSNR value is calculated by using formula:

PSNR (dB) = 10 * log10 (2562 / MSE) (1)

Where MSE is Mean Square Error and is calculated by,

$$MSE = \sum_{i=1}^{x} \sum_{j=1}^{y} \frac{|A_{ij} - B_{ij}|^2}{x * y} \quad (2)$$

Where x: width of image
y: height of image
By multiplying x & y, we get the number of pixels.

For testing the result, we take a standard lena image of size 512X512 as original cover image shown in fig. 4(a). After that, we encrypt the original cover image, as fig. 4(b). Then we take another image to hide into the encrypted cover image in fig. 4(c). Then embed this image into encrypted cover image to produce an image to which we call it as a stego image, as in fig. 4(d). Now, this stego image contains an original cover image, encrypted image and an embedded message (image). We can also embed the text as well into the stego image along with another hidden image as shown in fig. 5.

(a) (b) (c) (d)

(e) (f) (g)

Figure. 4: (a) Original Lena Image (b) Its Encrypted version (c) Message Image to be hide (d) Stego Image containing Original, Encrypted and Message Image (e) Extracted Encrypted Image from Stego Image (f) Extracted Cover Image (g) Extracted Message Imag

From the stego image, we then decrypt the encrypted image and an original image, according to the key, shown in fig. 4(e) & fig. 4(f), respectively. Then at last we extract an additional embedded data (text and hidden image) from the decrypted image, as shown in fig. 4(g) and fig. 5

Figure. 5: Data Embedding and Extraction

From the experimental result, it is clear that, if we had a data hiding key, then we could extract the additional data from an encrypted image containing embedded data. If we had an encryption key then we can directly decrypt an encrypted image containing embedded data. If we had both keys, then we could successfully extract an embedded data and perfectly restored the original image from the encrypted image containing embedded data.

From the tested color image, the PSNR value of the decrypted image that we obtain is 43.6 dB, which shows that the improved PSNR value, and improved result of the algorithm. The image recovered using blowfish algorithm is same as the original image.

Table 5.1 Compare Existing Vs Proposed w.r.t Performance

Methodology	4LSB	PSNR	Accuracy
Enhanced proposed System	80%	90%	90.6%
Proposed System	70%	76%	65%
Existing System	60.5%	52.5%	35%

VI. CONCLUSION AND FUTURE SCOPE

This work proposes a lossless, a reversible, and a combined information hiding plans for figure content pictures scrambled by open key cryptography with probabilistic and homomorphic properties. In the lossless plan, the ciphertext pixel qualities are supplanted with new values for installing the extra information into the LSB-planes of ciphertext pixels. Thusly, the installed information can be straightforwardly removed from the scrambled area, and the information implanting operation does not influence the unscrambling of unique plaintext picture. In the reversible plan, a preprocessing of histogram therapist is made before encryption, and a half of ciphertext pixel qualities are altered for information inserting. On beneficiary side, the extra information can be separated from the plaintext space, and, in spite of the fact that a slight twisting is

presented in unscrambled picture, the first plaintext picture can be recuperated with no mistake. Because of the two's similarity plots, the information implanting operations of the lossless and the reversible plans can be all the while performed in a scrambled picture. In this way, the collector may remove a piece of installed information in the scrambled space, and concentrate another piece of inserted information and recoup the first plaintext picture in the plaintext area.

VII. ACKNOWLEDGMENTS

It is my great pleasure to express my deep sense of gratitude to my project guide Prof. Dinesh D. Patil, Head of Computer Department for his valuable guidance, inspiration and wholehearted involvement during every stage of this paper presentation. I am very much grateful to the entire SSGBCOE&T Bhusawal for giving me all facilities and work environment which enable me to complete my task.

REFERENCES

[1] N. A. Saleh, H. N. Boghdad, S. I. Shaheen, A. M. Darwish, "High Capacity Lossless Data Embedding Technique for Palette Images Based on Histogram Analysis," Digital Signal Processing, 20, pp. 1629−1636, 2010.
[2] J. Tian, "Reversible Data Embedding Using a Difference Expansion," IEEE Trans. on Circuits and Systems for Video Technology, 13(8), pp. 890−896, 2003.
[3] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible Data Hiding," IEEE Trans. on Circuits and Systems for Video Technology, 16(3), pp. 354−362, 2006.
[4] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless Generalized-LSB Data Embedding," IEEE Trans. on Image Processing, 14(2), pp. 253–266, 2005
[5] X. Zhang, "Reversible Data Hiding with Optimal Value Transfer," IEEE Trans. on Multimedia, 15(2), 316−325, 2013.
[6] W. Zhang, X. Hu, X. Li, and N. Yu, "Optimal Transition Probability of Reversible Data Hiding for General Distortion Metrics and Its Applications," IEEE Trans. on Image Processing, 24(1), pp. 294-304, 2015.
[7] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Commutative Encryption and Watermarking in Video Compression," IEEE Trans. on Circuits and Systems for Video Technology, 17(6), pp. 774−778, 2007.
[8] V. Suresh, C. Saraswathy, "Separable Reversible Data Hiding Using Rc4 Algorithm", proceedings of the 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME), pp. 164-168, Feb 2013.
[9] X. Zhang, "Separable reversible data hiding in encrypted image," IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 826–832, Apr. 2012.
[10] Shilpa Sreekumar, Vincy Salam, "Advanced Reversible Data Hiding with Encrypted Data", International Journal of Engineering Trends and Technology (IJETT), Vol. 13, No. 7, pp. 310-313, July 2014.
[11] Vinit Agham, Tareek Pattewar, "A Novel Approach Towards Separable Reversible Data Hiding Technique", IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques, pp. 776-780, 2014.
[12] X. Zhang, "Reversible data hiding with optimal value transfer," IEEE Trans. Multimedia, vol. 15, no. 2, pp. 316–325, 2013.

