Reversible Data Hiding In Encrypted Images And Its Application To Secure Missile Launching

Radhika R. Patil *radhikap589@gmail.com*
Department of Electronics
DKTE Society's Textile & Engineering Institute, Ichalkaranji
Kolhapur-416115, Maharashtra, India

Deepali Y. Loni *deepaliloni@rediffmail.com*
Department of Electronics
DKTE Society's Textile & Engineering Institute, Ichalkaranji
Kolhapur-416115, Maharashtra, India

Abstract

This paper proposes reversible data hiding in encrypted images for secure missile launching. The work is presented in two stages: one involves encryption of cover image by block cipher algorithm and other is embedding secure data related to missile launching. For embedding data, vacant pixels are identified by Slepian-Wolf encoding method along with embedding key to hide the data. At the other end by using decryption algorithm the original cover image is recovered and the secret data is extracted. The performance analysis is presented by calculating parameters MSE, PSNR and SSIM.

Keywords: Image Encryption, Data Embedding, Reversible Data Hiding.

1. INTRODUCTION

In many applications, like law forensics, military imagery and medical imagery the information vendor requires to transmit data to a distant server for future processing. Now a day, internet is the prime medium to transfer information from one end to another across the world. The additional secret information can be hacked in a lot of different ways. This is the major problem with sending information over the internet. Therefore it becomes very important to take data security into consideration, during the procedure of data transferring. The intruder may also capture image, and view the significant contents and then alter the image before transferring it to receiver [1]. This is the way by which original image contents will be modified and receiver cannot have an idea about it. In general, a bit of content distortion is typically imperceptible to human imaginative. However, such distortion is not favored in some applications, like legitimate documentation, medical imaging, military observations, high-accuracy scientific investigation, since it might prompt risk of wrong decision making. Data security basically means given that safety to information from unauthorized users or hackers and imparting excessive level of protection to prevent information from modification. Data hiding is one kind of approach to secure data in cover media but there exists some distortion. In data hiding method the private and secret information is hidden into cover (host) image.

A large volume of data sent over internet is private and secret. Encryption is technique which transmits the secret data. The reversible data hiding is also treated as the new watermarking method which is used to validate an image by embedding some data on it as a watermark [2].

Most multimedia system data hiding method insert the extra information and modify original content [3], and thus distortion in cover image occurs. Data hiding activity insert information bits by changing the cover image, but enable the precise re-establishment of the original cover image after getting the embedded secret data. Within majority of applications, the little distortion

because of the data embedding is sometimes acceptable. For plaintext images many RDH methods have been proposed [4]-[7], these are not applicable to encrypted images since the redundancy in the original image cannot be used directly after image encryption.

Over the past few years, an excellent amount of schemes regarding to data hiding in encrypted images has been developed. Even though, inside these schemes, the cover image is lastingly distorted because of data embedding. In general, the cloud service supplier has no authority to add everlasting distortion. This means that, the original plaintext content i.e. cover image should be re-established without any mistake in image and data recovery for licensed receiver. To overcome this problem in encrypted images, the solution is the use of reversible data hiding (RDH).

The service supplier adds additional secret messages, e.g., notations, labels, verification information, or image data in encrypted images even not accessing the original cover image contents [8]. This is possible because of reversible data hiding technique in encrypted images. The original cover image is compulsorily recovered totally and also the hidden secret message is completely recovered at the receiving end. RDH in encrypted images is attractive. For example, in medical application, a patient will not give permission to expose his/her medical images to any outsider, whereas database manager may need to implant the medical records or patient's information in the encrypted image [9]. On the other end, the original cover image for diagnosis should re-established without any error after decryption and revival of the hidden secret information.

Strategies proposed in [10] - [12] makes use of the reversible data hiding is accomplished by using LSB modification. First the original cover image is encrypted using special encryption algorithm and then some of them embeds one bit of data into each block by a way of just flipping the last three LSBs. The spatial relationship exists in natural images and the interfered block, interfered block must be less smooth than the original block. Thus, original cover is recovered along with secret information. If selection of block falls in inappropriate block size, during data extraction and image recovery errors may occur. Thus block size is a factor which decides embedding rate of this method. Some RDH methods use histogram modification [9]. A histogram modification and *n*-nary data hiding scheme used to embed secret information into encrypted image. At the receiver end, original cover image can be totally recovered and the additional information can be extracted with the aid of the embedding key and the encryption key.

Another approach proposed in [8] uses the Slepian-Wolf encoding for data hiding. This idea is inspired by distributed source coding (DSC) [13]-[14]. In this first the image encryption is done by stream cipher algorithm then by using low density parity check codes the spare room is generated to add secret data in that vacated room. The information extraction and image recovery is with the aid of using distributed source coding technique. Along with RDH in encrypted images algorithm the reserving room before encryption technique [2] is used. And consequently it is straightforward for the data hider to reversibly hide data within the encrypted image. This approach has been given an amazing amount of reversibility, that is, data extraction and image recovery are not containing any error. The difference expansion method [4] calculates the neighboring pixel values differences, and for the difference expansion (DE) selects some difference values. This is applicable for audio and video as well. The information about original content restoration, additional data, and message authentication code will be implanted into the difference values.

This work proposes reversible data hiding in encrypted image for secure missile launching. The original cover is entirely encrypted by using block cipher, and the secret data is embedded by modifying a part of encrypted image. At receiver side, with the help of embedding key and encryption algorithm, the embedded data are successfully extracted while the original image is perfectly recovered.

2. SYSTEM DESCRIPTION

The general structure of the proposed method is presented in Figure 1, which consists of 4 phases, namely image encryption, data hiding, image recovery, data extraction, and the last unit of missile launching.

The original cover image is first transformed into encrypted image using encryption algorithm. The data hider embeds the additional secret information into encrypted image using an embedding key to generate embedded image.

At the receiving end, receiver extracts the inserted secret information independently only if it has an embedding key. If receiver has knowledge about encryption algorithm and embedding key both, the inserted secret bits can be extracted and original cover image can be recovered.

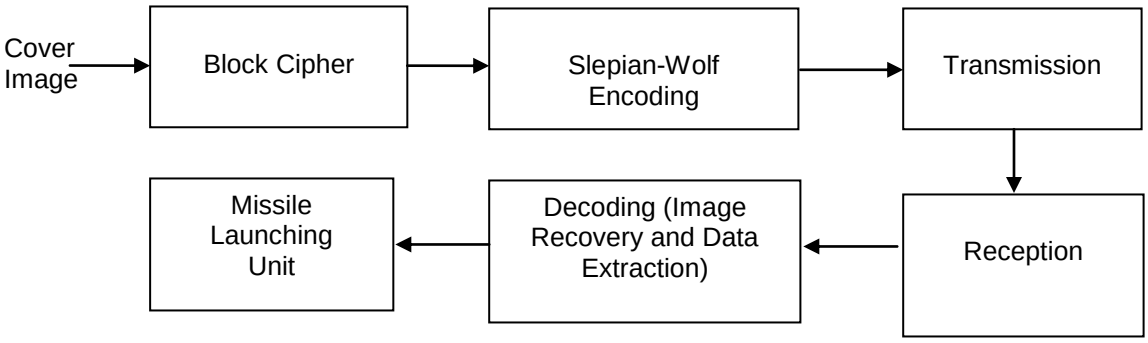
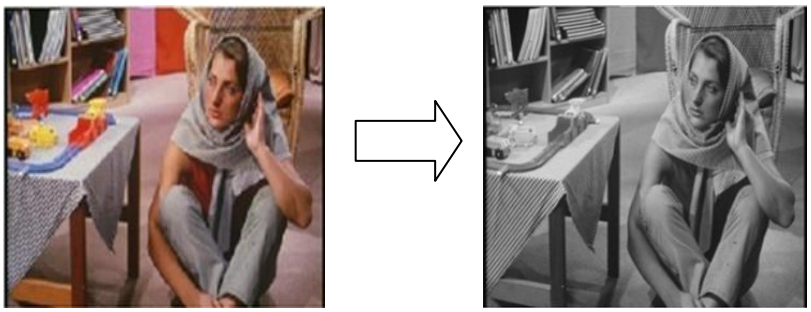


FIGURE 1: The Proposed System.

2.1 Image Encryption

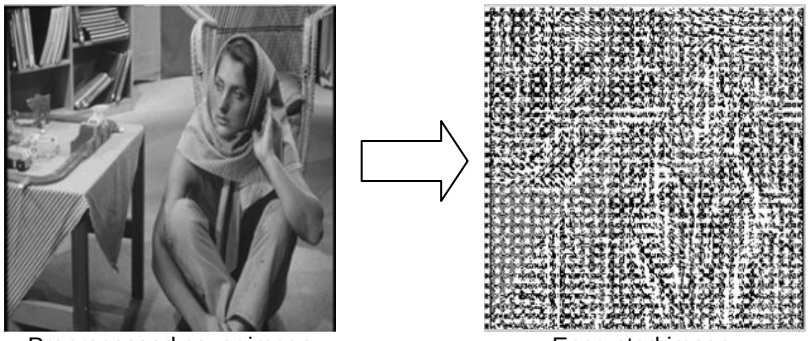
Original image must be grayscale; if input image is color then we first convert it into grayscale (0 to 255). The image is preprocessed such as image resizing and converted into particular intensity range. The mathematical operations on the image may results into negative value or may exceed the upper boundary. At recovery stage this may result in receiving the random symbols like $, #, etc. To avoid such circumstance at receiving end we set intensity range of image to 15 and 240.

Encryption is not directly applied on whole image; we select non-overlapping blocks from cover image. We divided selected block into two sub regions and then calculate pixel difference value and integrative component. Again we further divided the result obtained into two sub regions resulting into total four sub regions. For these four sub regions we calculated the difference and integrative components labeled as c1, c2, c3, and c4. We shuffled all the four components before combining them. We have not used any shuffling key, shuffling is done by simply rearranging the four components in different order (like c2, c1, c4, c3). The result of preprocessing and encryption are shown in Figure 2 and 3 respectively.



Original cover image Preprocessed cover image

FIGURE 2: Preprocessing of Cover Image.



Preprocessed cover image Encrypted image

FIGURE 3: Encryption of Preprocessed Image.

This decomposition and combing process is applied for each and every non overlapping block of cover image. Combined results (c2, c1, c4, c3) of each block, result into total encrypted image.

2.2 Data Hiding

The Slepian-Wolf encoding method is used for data hiding. To compress the selected bits from encrypted image the Slepian-Wolf codes are useful. Defined low density parity check code (LDPC) matrix **H,** it can be constructed in various different forms. The data-hider arbitrarily chooses a parity-check matrix **H** corresponding to a regular or irregular LDPC code by setting the numbers of variable nodes and the check nodes. The different algorithms have been proposed for the matrix construction, for example, matrices used in Gallager codes, MacKey codes, and finite geometry codes. For example by using MacKey method the matrix is constructed by following steps

1. The H matrix is created by initially creating all zero matrix and then randomly flipping bits in the matrix H. The flipped bits must not be necessarily distinct.
2. The matrix H is generated by randomly creating weight j columns.
3. The matrix H is generated with weight j per columns and uniform weigh per row and no two columns are connected to the same row more than ones (avoiding four cycles)
4. Matrix H is generated as in step 3 with the girth condition further constrained so that the girth is larger than six.

The MacKey's algorithms were used to find good performing codes with the variety of length and rates. The more details of LDPC matrix **H** can be found in [14].

We have selected a non-overlapping blocks of encrypted image for data embedding. We have selected an embedding key and checked for embedding key bit equals to one. And where the embedding key bit is one the value at that position in the selected block of encrypted image is considered as coefficient. We performed matrix multiplication between selected coefficient and H matrix, and the spare room is generated for data embedding. In the vacated room the data is embedded and checked for if we have completed with all secret data to be embedded. Otherwise data embedding process is continued. Along with this we have checked for one more condition, whether we completed with selected block of encrypted image, if done then next block is selected. Finally all non overlapping blocks are combined to form an embedded image which is encrypted image containing secret data. Generated embedded image is transmitted and is shown in Figure 4. The whole embedding process is shown in Figure 5.

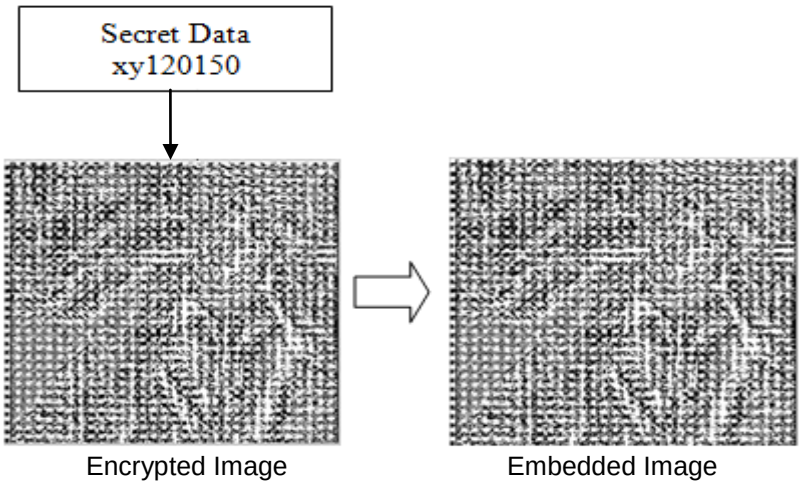


FIGURE 4: Embedded Image.

2.3 Image Decryption

At the receiver side, using the received embedded image, the original cover image can be recovered by decryption algorithm. First we divided the received embedded image into blocks and those blocks are decomposed into four sub regions and applied decryption algorithm on it. We performed exact reverse of the process applied at the transmission side. We have calculated the pixel differences and integrative components for all four sub regions. We combined the obtained results into two sub regions. Again by using these two sub regions we have calculated pixel difference and integrative component and combined obtained results into single block. The process was repeated for each block and all resulted blocks were combined to form decrypted/recovered image. The recovered cover image is shown in Figure 6.

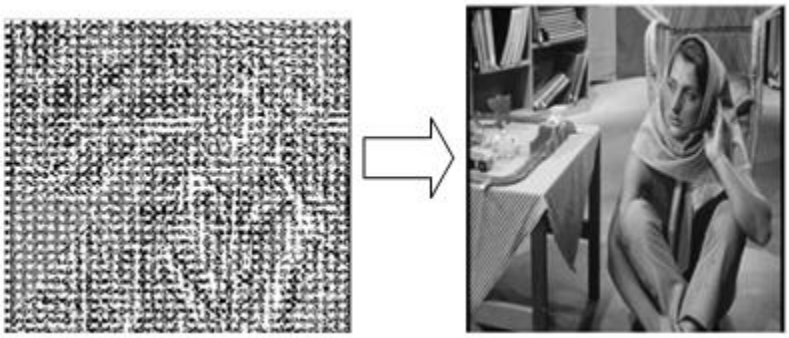


FIGURE 5: Image Extraction.

2.4 Data Extraction

In data extraction, we select a non overlapping block of embedded image. We used an embedding key to extracts the embedded secret data. We checked for embedding key bit, if it is one it means data is embedded at that position in selected block of embedded image. The value at that position is considered as coefficient and LDPC matrix is applied on it. From this we get to know the position where data is embedded and those bits are extracted. The extracted data is shown in Figure 7.

2.5 Missile Launching Unit

The decrypted secret data are the coordinates required for missile launching. The launching of missile involves coordinates corresponding to azimuth and elevation angle made by missile.

Accordingly during embedding process we embedded these coordinates. For example xy120150; where xy is used as an identifier which shows the start of valid bit frame, the next three digits corresponds to azimuth angle and last three digits corresponds to elevation angle.

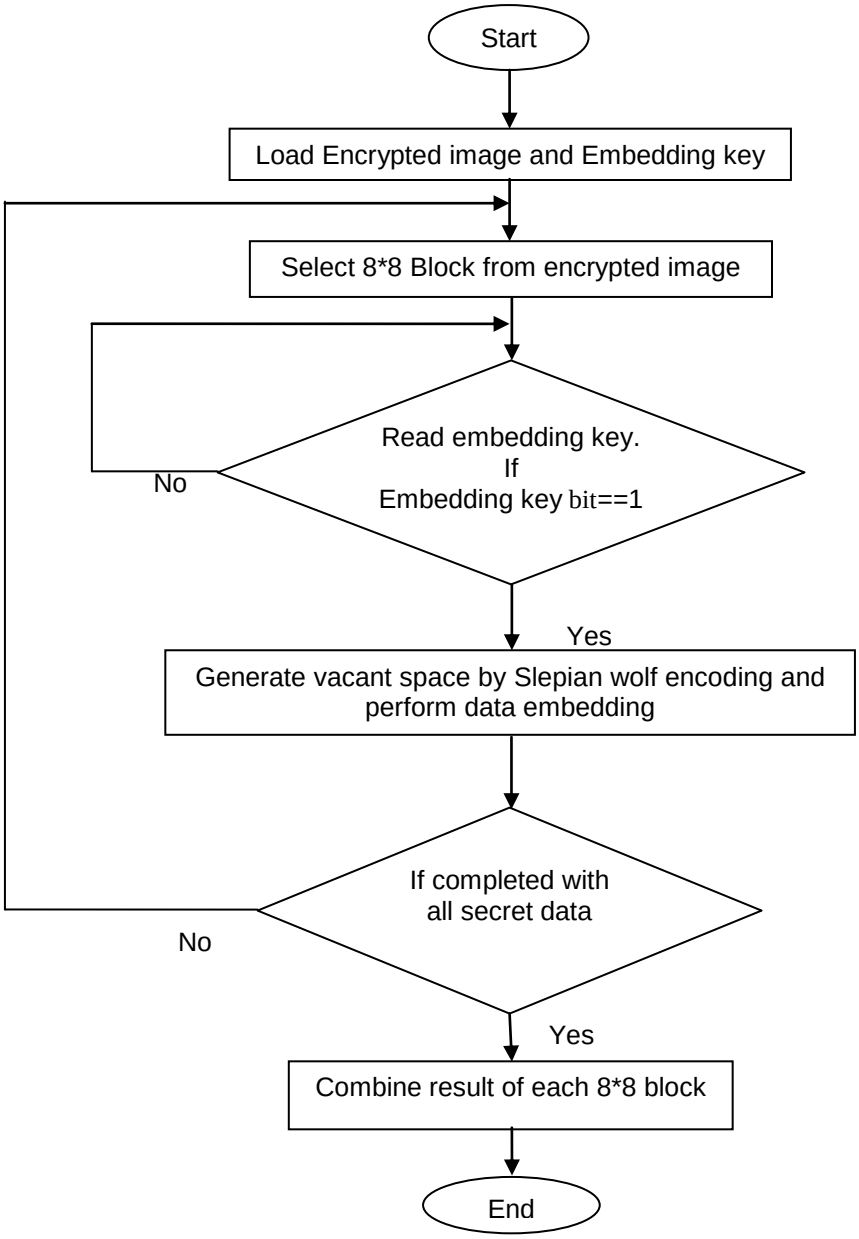
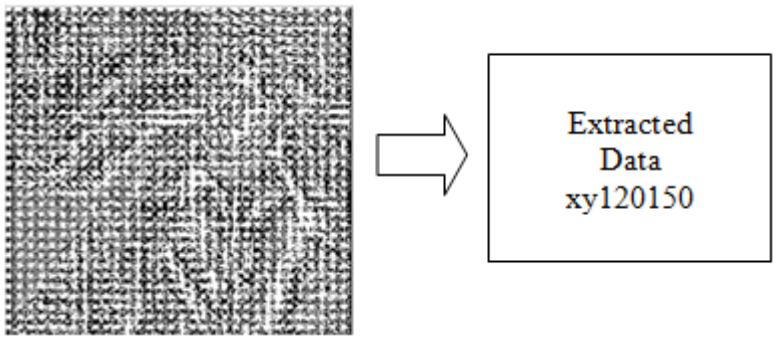


FIGURE 6: Data Embedding Process.

By using the serial communication the extracted data is send to missile launching hardware unit. The hardware unit consists of two DC motors, motor driver circuitry and controller. If identifier is received properly then the received data is treated as the valid frame and the DC motor rotation is made accordingly. The missile is then positioned at the target location. If received identifier does not match then it indicated invalid data received.



` Embedded Image Extracted Secret Data

FIGURE 7: Data Recovery.

3. PERFORMANCE PARAMETER

To evaluate the system performance we used mean square error (MSE), structural similarity index matrices (SSIM), and peak signal to noise ratio (PSNR). These parameters give an idea about how efficient the system is and from SSIM parameter we come to know how closer the original cover image to the recovered image. The original cover image and recovered cover image are used to calculate these performance parameters.

a. Mean Square Error

Let us consider two digital images X and Y. The MSE between these two images is expressed mathematically as:

$$\mathrm{MSE(X,Y)} = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)\text{^}2 \qquad (1)$$

Where N is the number of pixels in digital image

The value of MSE ranges in between 0 and 1. The lower value of MSE indicates less error, as MSE goes on increasing error also increases.

b. Peak Signal To Noise Ratio

The PSNR between two images is expressed mathematically as:

$$\mathrm{PSNR}\ (X, Y) = 20 log_{10}\left(\frac{MPP\text{^}2}{MSE(x,y)}\right) \qquad (2)$$

Where, MPP is Maximum Possible Pixel in an image, i.e. if the image of 8 bit then the MPP=2^8–1 = 255 pixels.

As the lower PSNR the lower relative image quality. Higher PSNR indicates a good quality image.

c. Structural Similarity Index Metric

Structural Similarity Index quality is based on the computation of three terms namely luminance, contrast, and structural term. The overall index is multiplicative combination of these three terms.

$$\mathrm{SSIM(x,y)} = [l(x,y)]^{\alpha}.[c(x,y)]^{\beta}.[s(x,y)]^{\gamma} \qquad (3)$$

Where,

$$l(x,y) = \frac{2\mu_x\mu_y + C1}{\mu_x{}^2 + \mu_y{}^2 + C1}, \qquad c(x,y) = \frac{2\sigma_x\sigma_y + C2}{\sigma_x{}^2 + \sigma_y{}^2 + C2}, \qquad \text{And } s(x,y) = \frac{2\sigma_{xy} + C3}{\sigma_x\sigma_y + C3}$$

Where, s
μ_x, μ_y, σ_x, σ_y and σ_{xy} are local mean, standard deviation, and cross covariance for images x, y.
If $\alpha = \beta = \gamma = 1$ then

$$SSIM(x, y) = \frac{[(2\mu_x\mu_y + C1)(2\sigma_x\sigma_y + C2)]}{[(\mu_x^2 + \mu_y^2 + C1)(\sigma_x^2 + \sigma_y^2 + C2)]} \quad (4)$$

The range of SSIM is 0 to 1. The value 1 indicates two images are exactly same. As SSIM value goes on decreasing, similarity between original and recovered image is less.

The obtained parameter MSE, PSNR, and SSIM for different images are listed in Table 1.

Image	MSE	PSNR	SSIM
barb Color	0.140772	56.645653	0.999983
Cameraman	0.006736	69.846711	0.999979
Lena	0.003264	72.993450	0.999991
Peppers	0.004462	71.635697	0.999990
office_6	0.187708	55.395968	0.999966

TABLE 1: Performance Parameter Calculated for Different Images.

We have got MSE value approximately 0.1 which indicates less error and the PSNR value nearly about 60 to 70 which indicates good quality of the recovered image. The last parameter we calculated SSIM equals to 0.99 which indicates that original cover image and recovered cover image are exactly same. From the Table 1 the MSE and PSNR value we achieved are good for gray images than color. Also SSIM we attained is approximately same for color as well as gray images.

The comparison the proposed work with other research approaches for the PSNR performance parameter for Lena image is shown in Table 2.

Method used	Proposed method	Reserving room before encryption [2]	Distributed source coding [8]	LSB modification [10]
PSNR	72.993450	67.16	37.9	37.9

TABLE 2: Comparison of Performance Parameter for Lena Image.

From the Table 2 it is observed that the PSNR we achieved (72.99) is greater than those obtained in [2], [8] and [10]. The higher value of PSNR indicates better system performance.

4. CONCLUSION

Here we propose a technique of reversible data hiding in encrypted images and its application to secure missile launching. Initially encryption is applied on original image. Depending upon embedding key, bits of encrypted image are selected and Slepian-Wolf encoding is applied to

make spare room for the secret data. At the receiver end, all hidden secret data is extracted using embedding key, also original image is approximately recovered with good quality with the help of decryption algorithm. To generate corresponding syndromes LDPC parity-check matrix is used. On encrypted image we performed data embedding operation, so the data-hider cannot access the contents of the original image. That ensures security of the contents in data hiding. As the embedding and recovery is protected by the encryption and embedding keys, an adversary is unable to break into the system without these keys. The future direction is to improve system performance parameters by considering noisy channel.

5. REFERENCES

[1] Shilpy Mukherjee, A. Mahajan, "*Review on Algorithms and Techniques of Reversible Data Hiding*" International Journal of Research in Computer and Communication Technology, Vol 3, Issue 3, March- 2014

[2] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, "*Reversible data hiding in encrypted images by reserving room before encryption*," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 3, pp. 553–562, Mar. 2013.

[3] Mehmet U. Celik, Gaurav Sharma, A. Murat Tekalp and Eli Saber, "*Reversible DATA Hiding*" IEEE ICIP pp. 157-160. 2002.

[4] J. Tian, "*Reversible data embedding using a difference expansion*," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, Aug. 2003.

[5] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 354–362, Mar. 2006.

[6] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 721–730, Mar. 2007.

[7] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[8] Zhenxing Qian, and Xinpeng Zhang, "*Reversible data hiding in encrypted images with distributed source encoding*," IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 4,pp. 636-646 April 2016.

[9] Z. Qian, X. Han, and X. Zhang, "*Separable reversible data hiding in encrypted images by n-nary histogram modification*," in Proc. 3rdInt. Conf. Multimedia Technol. (ICMT), Guangzhou, China, 2013, pp. 869–876.

[10] X. Zhang, "*Reversible data hiding in encrypted image*," IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.

[11] X. Zhang, "*Separable reversible data hiding in encrypted image*," IEEE Trans. Inf. Forensics Security, Vol. 7, No. 2, pp. 826–832, Apr. 2012.

[12] W. Hong, T.-S. Chen, and H.-Y. Wu, "*An improved reversible data hiding in encrypted images using side match*," IEEE Signal Process. Lett., vol. 19, no. 4, pp. 199–202, Apr. 2012.

[13] W. Liu, W. Zeng, L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1097–1102, Apr. 2010.

[14] S. S. Pradhan and K. Ramchandran, "*Distributed source coding using syndromes (DISCUS): Design and construction,*" IEEE Trans. Inf. Theory, vol. 49, no. 3, pp. 626–643, Mar. 2003.