International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-8, June 2020

Effective Revisable Data Hiding in Encrypted Image for Protection of Image Content

Anuja Bhondve, Shweta Koparde, Vaishali Latke

Abstract*: In this article, we propose a reversible method for hiding data, in which the original image and hidden data can be restored on the receiving side. The owner encrypts the original image using an encryption key to protect the privacy of the image content. Each block of encrypted image is added to the little secret by Hider data using the key data hiding. Data hiding process causes only a small change in each partial pixel flip block, which improves decoded image visual quality. The image can be easily decoded receiver using the key, data encryption key to hide the adaptive soft characteristic of the evaluation function along the direction of the isophote, the secret data can be extracted from a decoded image and original image recovery can be restored more successfully.

Keywords: Privacy protection, Reversible data hiding, encrypted images, decrypted images, image recovery.

I. INTRODUCTION

Image encryption and Data Hiding are two main means of securing data. First to convert a image into something like noise so that the contents of the image are not leaked [1, 2], while the latter imperceptibly encrypt the secret data in the cover image, the original image should be protected, and the secret data in the data hiding must be private information. Conventional methods of hiding data are usually irreversibly disrespectful, and the embedding process results in permanent distortions of the original medium, and in medical, military images, and forensic proof collection, where the original media can be restore without any loss. For restoring the original media without loss, reversible data hiding (RDH) was proposed. Most traditional RDH schemes in images were implemented using natural image compression ratios [3]. Tian proposed a differential extension (DE) mechanism for [4] RDH, and the secret data is embedded in the host Image by adjusting the parity by doubling the Pixel-to-pair difference. Ni et al. [5], proposed to check to change the Pixel value corresponding to the peak point of the image histogram, to achieve a reversible data embedding by shifting the other pixel values. Li et al. in [6], simply designing shift on embedded functions is proposed. Based on the DE, in [7] introduced a prediction error expansion (PEE) strategy, which successfully utilizing Inter-Pixel redundancy and achieving satisfactory speed distortion performance. In [8] introduced a new RDH scheme to improve wee through multipath PVO ideas. To instead of using only the prediction error, use the correlation more effectively. Ou et al, propose to consider jointly the two adjacent prediction errors [9] and obtain a higher embedding rate. Various histograms were proposed in [10] based on the RDH scheme, PEE. By calculating the histogram generation of the series, this scheme, the pixels of the complexity of the correction will be corrected with multiple histograms with secret data embedding. In this paper, [11], the reference Pixel is based on the local complexity distribution and inpinting the image to be selected adaptively depending on the centralized histogram of the prediction error of data embedding is obtained. All of the above RDH schemes are used for images with plain text. In recent years, due to the advent of cloud computing, more and more attention has been paid to reversible data hidden in an encrypted image (RDHEI) [12, 13, 14, 15, 16, 17, 18]. Under RDHEI, third parties, such as cloud servers, can effectively manage tagged and encrypted images in the cloud in this way some of the encrypted images disguised by the image content owners who want to protect their privacy, and the allowed recipients can extract the embedded data correctly. For example, in the hospital, CT image database network administrator can use certain marking data on the encrypted CT image, convenient management and Privacy Protection, therefore, in the RDHEI; deploy the data from the encrypted image and see empty space to ensure that the data extraction and image recovery are correct at the same time. One of typical types of their schemes is free rooms after encryption (VRAE). Puech et al. For the first time, the opposite method of hiding encrypted images was proposed [12]. They noted that the traditional plain text domain name, the RDHEI method was not available as long as the encrypted image, the entropy value was large. Retrieving secret data when performing an image description, a local standard deviation analysis was adopted for tagged encrypted images. By using a stream encryption the original image is encrypted and then image is dividing into two blocks [19]. Through a public key embed a secret bits presented in [20], the SVM classifier is used to classifies the images into encrypted and unencrypted image patches.

Revised Manuscript Received on June 30, 2020.
***** Correspondence Author
Anuja Bhondve*, Assistant Professor.Department of Computer Engineering, Pimpri Chinchwad College of Engineering & Research, Ravet, India. Email: bhondve.anuja@gmail.com
Shweta Koparde, Assistant Professor.Department of Computer Engineering, Pimpri Chinchwad College of Engineering & Research, Ravet, India. Email: Bhondve.anuja@gmail.com
Vaishali Latke, Assistant Professor.Department of Computer Engineering, Pimpri Chinchwad College of Engineering & Research, Ravet, India. Email: Bhondve.anuja@gmail.com

In paper [21] Qian, presented the RDG method based on distributed source encoding. To improve the embedded rate, first proposed an RRBE-based method in Ma et al. [22]. It embeds data in the room so that encryption can be obtained before embedding to achieve the growth and self-actualization of the traditional RDH method.

In [23] predicts some pixel values before encrypting them to embed data in predicted errors through the histogram shift, and special encryption methods are used. In [24], a new method was proposed by RDHEI, Nguyen et al. First, encrypt the original image, dividing the pixels areas based on four neighboring pixels. Secret data involved in the central plane of a smooth pixel of an encrypted image. In the paper [25], a new method of sparse representation at the patch level is proposed. How to hide data in encrypted images with the most significant bits (MSB) is presented in [26], which allows you to get a high hidden capacity. During pre-processing, the position map was generated by detecting a predictive error. Thanks to the exchange of the MSB, the secret data could be executed, and the built-in speed was close to 1BPP. Based on the methods [26] and [28], improved maximum embedding in feed between two MSB pixels.

Using the MSB with two methods the output will be not satisfying so the rate of embedding is still need to improve. This document proposes the RDHEI method, which is based on a multi-MSB embedding strategy. Using a multi-MSB embedding strategy, bits of secret can be embedded into encrypted images without oversaturated Pixel plain text areas. More importantly, the secret bit is encrypted from a multi-MSB domain of pixels without error or a very high quality reconstructed image is obtained only when the encryption key is obtained. Compared with other modern methods [29, 28], the proposed method allows achieving a maximum embedding speed.

II. METHODOLOGY

This section proposes an RDH method for encrypted images, including image encryption, encrypted image data hiding, data extraction, and image restoration. The proposed scheme (Fig 1), has three types of features, namely, content owners, Hider data, and receivers. The content owner is a cloud user. To prevent image information from leaking, we encrypted the image and then uploaded on cloud. Then cloud to generate data embedding services to encrypted images that are marked, without knowing the contents of the original image. So the data hider is cloud of proposed system. The receiver remarkably extracts the encrypted images and then decrypt. Number of operation performed by content owner such as repositioning the image, and encrypting the image. Some secret bits are inserted by data hider into the encrypted image using data hiding key, but the data hider do not have the any idea about the original image contents. On the receiver side, if the receiver has only a data hiding key, extract the secrets bits without any errors. If he has both key, he has extracted image errorless and restores the image without loss.

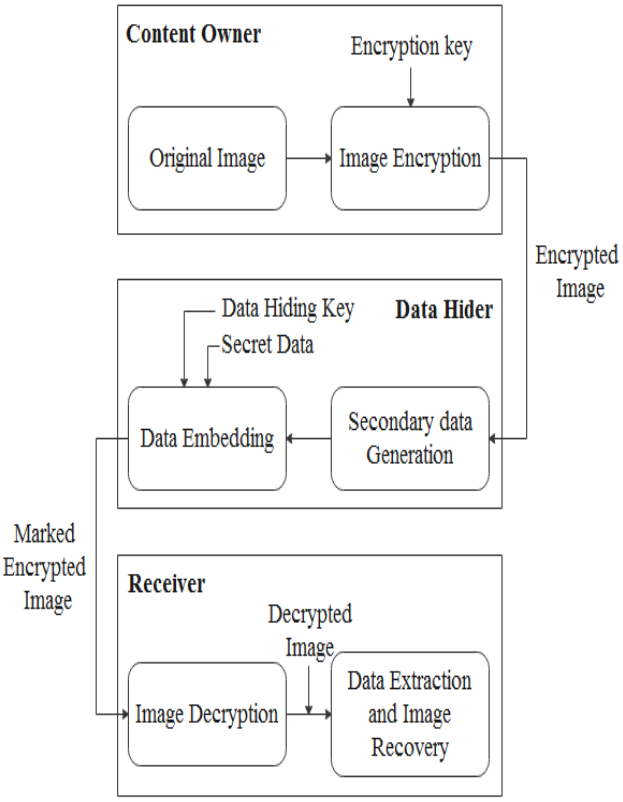


Fig. 1. Block diagram of proposed system.

A. Image Encryption

There are three steps to construct the encrypted image are as follows:

- Secondary data generation: The secondary data is important during the decoding phase for correctly extract the data and recover the original image. It includes threshold 8 bits *(t)*, M parameter 3 bits, the 16 bits number of pixels S in $I_W^{s'}$and LM. Pixels are denoted as $I_W^{s'}$. LM is the location map, and is expressed as L_{clm} and its size as l_{clm}. The secondary information is extracted in advance a N 16 bits parameter is record the value of l_{clm}. So secondary data total size is 43+l_{clm} bits.
- Image rearrangement: In image rearrangement the black pixels are arranged at the top of the quarter of the image and the grey pixels are arranged on the end of image. The original images are expressed as O.
- Image encryption: To avoid the leakage of content of original image, it is encrypted with the encryption key K. The encryption key is generated by using encryption algorithm because of its efficiency and security.

Let $O_l(p,q)$ be the l th ($l = 1,2,\ldots..8$) bit of $O(p,q)$;

$$O_l(p,q) = \left\lfloor \frac{O(p,q)}{2^{l-1}} \right\rfloor mod2 \qquad (1)$$

Where, floor function is |.|, and each bits of encrypted $E_l(p,q)$is calculated by Eq. (2):

$$E_l(p,q) = O_l(p,q) \oplus C_l(p,q) \quad (2)$$

Where, $\oplus$ denotes operation, $C_l(p,q)$ is cipher stream produced by using encryption key K. The pixel values $E(p,q)$ is calculated by using Eq. (3):

$$E(p,q) = \sum_{l=1}^{8} E_l(i,j) \times 2^{l-1} \quad (3)$$

Then you can get the encrypted image E.

B. Data Embedding

At the data embedding stage, the encrypted image E and secondary information is received, and then the secrets bits are embed by data hider into the image, without knowing the content. First of all, I_ED is an encrypted pixel-it is a small change plane with secondary information, while the bit plane and secret bits are connected to both embedded data, then is encrypted with an encrypted image.

S is the number of pixels in $O_W^{s\prime}$, that are rearranged behind the pixels I_b. Then the data hider select the pixels randomly in $I_W^{s\prime}$. System selects $(m+1)th \sim 8th$ MSB of placed the pixels on the data embedded to create the marked encrypted image, and the marked pixel value is expressed as Eq. (4):

$$E_h'(p,q) = \sum_{l=1}^{M} E_l(p,q) \times 2^{l-1} + \sum_{l=M+1}^{8} c_{l-M} \times 2^{l-1} \quad (4)$$

Where, c_{l-M} is a secret bits with a 0 or 1 values. Each pixel can hold 8 bit of data and the *t* and *M* determines the number of pixels. So the image embedded capacity under *t* and *M* is:

$$cap = (8-M) \times S \quad (5)$$

The maximum embedded image capacity depends on the values of *M* and *t*.

C. Data Extraction and Image Recovery

During the decoding stage, the recipient can perform various operations, depending on whether the encryption key and the data hiding key are present. Which includes three scenarios: First is, if the receiver has only the data hiding key K_h, second is, if the receiver has only the encryption key K and the third is, if the receiver has both the data hiding and encryption keys.

In first case, the receiver can use the data hiding key in encrypted domain to retrieve data. Extract the secondary information from pixels. Based on data hiding key take the position of marked pixels. Then the bits (8-m) of data are extracted from the marked pixels is expressed in Eq. (6):

$$c_{l-M} = \left| \frac{E_h'(p,q)}{2^{l-1}} \right| mod\ 2, l = M+1, \ldots 8. \quad (6)$$

The all process is performing in encrypted phase so the content of original image cannot be loss.

In second case, if the recipient only has the K encryption key, then the restored image $\breve{O}$ can be obtained, and the detailed process looks like:

1) First extracted the secondary information.
2) By using encryption key decrypt, the pixels of the encrypted image directly.

$$d_l(p,q) = E_l'(p,q) \oplus C_l(p,q), l = 1,2,3, \ldots .8 \quad (7)$$

Then the decrypted pixels values calculated by Eq. (8)

$$d(p,q) = \sum_{l=1}^{8} d_l(p,q) \times 2^{l-1} \quad (8)$$

Where, *l*th bit values are E_l' and $d_l(p,q)$ of the marked pixel and decrypted pixel.

3) In the third step, from secondary information, the encrypted image decrypted to take the pixel position. Placed the pixels in their original position. This way, all pixels can be placed in their original position.
4) The pixels in $I_W^{s\prime}$ are embedded with data, and some of the pixels in the I_{ED} have been replaced by secondary information, the original values cannot be restored by pixels of these two parts after the direct decryption. For the marked pixels in $I_{W'}^{s\prime}$, their surrounding pixels in I_b have been restored after direct decryption. To recover the original values surrounding pixels are used. The secondary information is replaced with pixels in I_{ED}. In this scenario without error original pixel values are not restored.

At last scenario the receiver of all keys cannot make errors. To extract data uses a data hiding key, including secret bits and the bit surface of the encrypted I_{ED}pixels. The encryption key is directly decrypted these encrypted bit planes. In second scenario mentioned that pixels are not only restore the original image but also the pixels bit planes on their edges are restored. This way, all pixels in the restored image $\breve{O}$ are restored without any loss.

III. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed method is tested with a standard image with a size of 512×512. In order to evaluate the effectiveness of the proposed method, the maximum embedding rate of the decoded image and the peak signal-to-noise ratio (PSNR) were used.

A. Maximum Embedding Rate

The system takes eight standard images that are publically available, such as Leena, baboon, Man, Aerial, Crowed, Bridge, Gold hill, and, Peppers to show the pure embedded rate (PER) of the proposed system. The PER is calculated as:

$$PER = \frac{Cap - 43 - l_{clm}}{G \times W} \quad (9)$$

Where, $Cap - 43 - l_{clm}$ is the secret bits pure embedding capacity.

PER denoted as *M* and *t*. Soft pixels are selected using the parameter *t*, and the soft pixels are only predicted to select the embeddable pixels. For large values of *t*, there are softer pixels. To get the maximum value of the image, select the optimal combination *(M, t)*. For each value of M (i.e., 3, 4, 5, 6, 7), the value of *t* increases incrementally, from a fairly small value of 1 to 50, in increments of 1, up to Eq. (10) be satisfied:

$$43 + l_{clm} \leq 16344 \qquad (10)$$

The Table 1 shows the using different values maximum pure embedding rates of eight images. Table 1 show that some images may have a very low embedding speed or 0 if the *M*-value is low. This is mainly small embedded pixels in a moderately smooth image when *M*-value is small. On the other hand, it produces more pixels to mark.

If the Pixel is not enough to store secondary information that we don't have built-in secrets for a bit in this scenario. Thus, the embedding speed is 0. In Table 1, images Crowed and, Bridge are very smooth than other, and when *M=3*were 3.4512 and 3.0245 the maximum pure embedding rates are obtained. The image is smother, more accurate values of pixels, so we can use small *M* value to get the maximum PER

Table 1: Calculation of maximum PER

Images	M=3	M=4	M=5	M=6	M=7	Maximum
Leena	0	0.4563	1.6543	1.4563	0.7646	1.6543
Baboon	0	1.0244	1.4567	1.2343	0.7234	1.4567
Man	0	0	1.1156	1.2367	0.7345	1.2367
Aerial	0	1.234	1.5438	1.2359	0.7645	1.5438
Crowed	3.4512	2.5647	2.2678	1.4865	0.7376	3.4512
Bridge	3.0245	2.3478	2.1970	1.2045	0.7424	3.0245
Gold Hill	0	0	0.6745	0.9456	0.6657	0.9456
Peppers	0	0	0.4563	0.3675	0.5567	0.5567

B. Peak signal-to-noise ratio (PSNR) of decrypted images

As shown in section II, image is decrypted using only the encryption key, some pixels did not recover their original value because their small change in plane used the secondary information. Hence, the decoded PSNR image is mostly dependent on these unrecoverable edged pixels. Thus, a given embedding rate (GER) requires a small amount of secondary information, fewer edge pixels to change, and the higher the PSNR. Therefore in Eq. (11), the optimal combination (M^*, t^*) is used to get highest PSNR.

$$\begin{cases} (M^*, t^*) = \arg min\ (l_{lcm}) \\ s.t.\ PER \geq GER \end{cases} \qquad (11)$$

As described in the previous section, when both the data hidden and the encryption key are available, the receiver can extract the data without error, as well as the original data; therefore, in this case, the reconstructed image PSNR is approximated to $+\infty$. To get the best PSNR need to choose the optimal parameters (M^*, t^*). The Eq. (11), if the secondary information is small, you can see that the decoded image has a higher PSNR. In general, based on the fact that PER is higher than GER, the higher the value of M. Which means that secondary information is less, and the higher PSNR of image. In table 2, the optimal value of *M* decreases when the embedding rate increases.

Table II: Perform optimal parameters on image Leena with different GER

GRE	0.1	0.3	0.5	0.7	0.9	1.2	1.3	1.5	1.7
Optimal Parameter (M*,T*)	(7, 1)	(7,3)	(7, 5)	(7, 6)	(6, 8)	(6.1 7)	(6,1 4)	(5, 6)	(5, 8)

In terms of maximum PER and quality of reconstructed image with only encryption key, the proposed system is compared with the different methods in paper [28, 29, 30, 31, 32]. The paper [32], is based on VRBE framework and other papers [28, 29, 30, 31], are based on BBRE framework. The two prediction MSB method is proposed in [29] and, [28] are highly related to the proposed method. In General, all five methods are compared. The proposed methods and two another method proposed in [29] and, [28] are all based on MSB embedding strategies. We use publically available eight images for comparison. The Table 3, shows the comparison result.

Table III: Comparison between proposed systems with other methods using MER parameter

Images	Leena	Baboon	Man	Aerial	Crowed	Bridge	Gold Hill	Peppers	Average
Paper CPE [38]	1	1	1	1	1	1	1	1	1
Paper [39]	1.23 45	1.11 24	1.0 055	1.2 11 1	1.00 45	0.4 534	1.116 4	0.56 47	0.98 31
Our Method	1.75 67	1.86 57	1.6 754	1.5 64 7	1.34 56	0.4 897	1.456 7	1.28 56	1.42 48

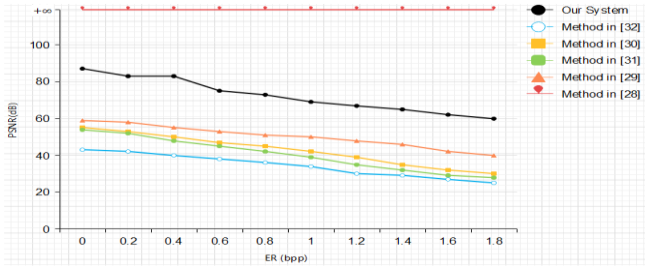
In [29], the original image is changed to avoid any prediction errors. Since all the pixels in the image fit in this approach a bit, embedded rate of image tested is 1bpp. However, it is reconstructed, which causes some damage to the image quality.

In paper [28], into one pixel two bits are embedded, that can improve the embedding rate. The methods proposed in [29] and [28] allow embedding no more than one or two data bits in pixels, which limits the embedding, rate. For images of various smooth levels, we can choose the most appropriate *M* value for embedding data that is more flexible than the methods [29] and [28].

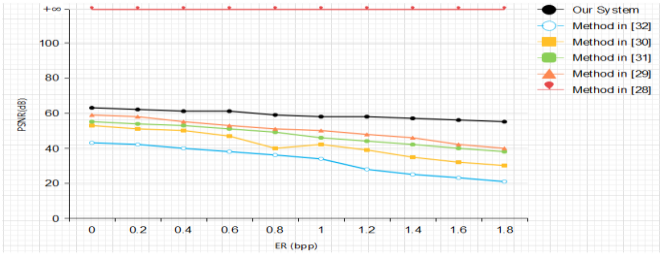
About the proposed method and paper [28] method, MER ability of these two methods has a great relationship with the smoothness of the image,

the difference in the MER of eight test images is relatively large. The complex image the embedded capacity is small. If the image bridge is more complex, the embedding speed will be smaller than other images. The proposed method was compared with the method [29]. The proposed method has a higher embedding speed for all test images except for the image bridge. Embedding speed increases by 0.4248 bpp on the average of eight images during the test. And the proposed method is compared with the method in paper [28], the proposed method has a higher embedding rate.

The Fig 2, shows the performance comparison of our system with other methods related in paper [28, 29, 30, 31, 32]. For this comparison we used Leena and Airplane images. From Fig 2, the PSNR of the encrypted image clearly decreases with increasing volume of embedded data. The PSNR of our reconstructed image is not reaching to $+\infty$, all they are very high. The PSNR of the Leena and Airplane reconstructed images are between 60db to 87 db and 55 db to 63 db, that are higher than the method in [29, 30, 31, 32]. In our proposed system some pixels are changed and most pixels are recovered without any loss. The proposed method achieves a higher embedding rate than other comparable methods.



(a) Leena



(b) Airplane

Fig 2: Performance comparison of proposed our system with other methods related in paper [28, 29, 30, 31, 32].

The proposed method allows you to restore the original image with embedded data without any errors and only with the help of an encryption key to achieve an excellent compromise between the embedded rate and image quality of the restored image.

IV. CONCLUSION

Proposes effective RDHEI using in multi-MSB embedding strategy with very high embedding speed, significantly higher than the corresponding method [28, 29, 30, 31, 32]. The *M*-value varied for different images at maximum embedding speed. Generally, smoothing the image will lower the *M* value chosen to get the maximum embedded rate. At a certain embedded speed, you can simply use the encryption key to choose the optimal value *(M, t)* to get the highest PSNR of the decrypted image. Restored picture using only the encryption key, only one of the edges is damaged and the restored image, and the remaining pixels all original values are restored. In other words, no matter to how much data maximum volume is embedded, the visual quality is significantly degraded and the pixel values will be restored without loss. The experimental results show that the proposed method archiving excellent embedded performance.

REFERENCES

1. Z. Tang, x. Zhang and f. Wang, “image encryption based on random projection partition and chaotic system”, multimed. Tools appl. 2016.
2. U. Hayat, and n. Azam, “a novel image encryption scheme based on an elliptic curve”, signal process. 2019.
3. Goljan. M and fridrich. J, “lossless data is embedding for all image formats”, in: spie proc. Photonics west, electronic imaging, 2002.
4. Tian. J, “reversible data embedding using a difference expansion”, ieee trans, 2003.
5. Ansari n, y. Shi, ni. Z, and su. W, “reversible data hiding”, ieee trans. Circuits- 2006.
6. Li. B, yang. B, li. X, zeng. T.y, “general framework to histogram-shifting-based reversible data hiding”, ieee trans-2013.
7. Rodriguez. J.j and thodi. M, “expansion embedding techniques for reversible watermarking”, ieee trans-2007.
8. Xiong. G, cai. Z.c and weng. S.w, “reversible data hiding using multi-pass pixel-value-ordering and pair wise prediction-error expansion”, inf. Sci.-2018.
9. Li. L, ni. R, zhao. Y and ou. B, “pair wise prediction-error expansion for efficient reversible data hiding”, ieee trans-2013.
10. Yang. B, zhang. M, gui. X, and li. X, “efficient reversible data hiding based on multiple histograms modification”, ieee trans-2015.
11. chang. C, liao. L, huang. Y, qin. C, “an inpainting-assisted reversible steganographic scheme using a histogram shifting mechanism”, ieee trans-2013.
12. Chaumont. M, puech. W, strauss. O, “a reversible data hiding method for encrypted images”, in: proceedings of spie- 2008.
13. Zhang. X.p, “reversible data hiding in encrypted image”, ieee signal process-2011.
14. Hong. W, chen. T and wu. H, “an improved reversible data hiding in encrypted images using side match”, ieee-2012.
15. Huang. J and shi. Y, “new framework for reversible data hiding in encrypted domain”, ieee trans-2016.
16. Xiong. G, cai. Z, weng. S, and, y.m. Wang, “reversible data hiding using multi-pass pixel-value-ordering and pair wise prediction-error expansion”, inf. Sci-2018.
17. Li. B, yang. B, li. X, and zeng. X, “general framework to histogram-shifting-based reversible data hiding”, ieee trans-2013.
18. Pun. C and liu. Z, “reversible data-hiding in encrypted images by redundant space transfer”, inf. Sci-2018.
19. X. Zhang, “reversible data hiding in encrypted image”. Ieee signal process- 2011.
20. W. Sun, o. Au, l. Dong, j. Zhou, x. Liu, “secure reversible image data hiding over encrypted domain via key modulation”. Ieee trans- 2015.
21. X. Zhang, and qian. Z, “reversible data hiding in encrypted image with distributed source encoding”, ieee trans- 2015.
22. W. Zhang, zhao, k. Ma, f. Li, n. Yu, “reversible data hiding in encrypted images by reserving room before encryption”, ieee trans-2013.
23. K. Ma, w. Zhang, and n. Yu, “reversibility improved data hiding in encrypted images”, signal process-2014.
24. C. Chang, s. Nguyen, and w. Chang, “high capacity reversible data hiding scheme for encrypted images”, signal process-2016.
25. Du. L, meng. D, wei. X, cao. X, and guo. X, “high capacity reversible data hiding in encrypted images by patch-level sparse representation”, ieee trans-2016.

26. Puteaux. P, and puech. W, “an efficient msb prediction-based method for high-capacity reversible data hiding in encrypted images”, ieee trans-2018.
27. D. Xiao, z. Peng, m. Li, and h. Nan, “a modified reversible data hiding in encrypted images using random diusion and accurate prediction”,. Etri j- 2014.
28. Z. Yin, y. Puyang and z. Qian, “reversible data hiding in encrypted images with two-msb prediction”, ieee december 2018.
29. W. Puech, and p. Puteaux, “an eient msb prediction-based method for high-capacity reversible data hiding in encrypted images”. Ieee trans-2018.
30. L. Du, x. Cao, w. Guo, x. Wei, “high capacity reversible data hiding in encrypted images by patch-level sparse representation”, ieee trans-2016.
31. J. Long, z. Wang, and h. Cheng, “lossless and reversible data hiding in encrypted images with public-key cryptography”, ieee trans-2016.
32. X. Zhang, z. Tang, c. Yu, c. Xie, and x. Xie, “separable and error-free reversible data hiding in encrypted image based on two-layer pixel errors”. Ieee access-2018.
33. W. Stalling, “cryptography and network security: principles and practice”, 3rd ed.; prentice-hall: upper saddle river, nj, usa, 2003.
34. T. Furon, and p. Bas, “image database of bows-2”, available online: http://bows2.ec-lille.fr/ (accessed on 20june 2017).
35. Sun. W, and wu. X, “high-capacity reversible data hiding in encrypted images by prediction error”, signal process-2014.
36. Shen. J, wang. Y and hwang. M, “a novel dual image-based high payload reversible hiding technique using lsb matching”, int. J. Network secur-2018.
37. Xiang. Y, zheng. H, xiao. D and wang. Y, “separable reversible data hiding in encrypted image based on pixel value ordering and additive homomorphism”, j. Visual commun. Image represent-2017.
38. Zhang. X, “reversible data hiding in encrypted image”, ieee signal process lett-2011.
39. Zhang. X, “separable reversible data hiding in encrypted image”, ieee trans-2012.
40. Ma. K, zhang. W and yu. H, “reversibility improved data hiding in encrypted images”, signal process-2014.
41. Au. O, sun. W, dong. L, zhou. J, liu. X, and tang. Y, “secure reversible image data hiding over encrypted domain via key modulation”, ieee trans-2016.

AUTHORS PROFILE

Anuja Bhondve She is currently an Assistant Professor of Computer Engineering Department at PCCOE&R. she has 3 years of experience in Microsoft CRM development and 1-year experience in Teaching. She has expertise in Machine Learning, and Image processing

Shweta Koparde She is currently an Assistant Professor of Computer Engineering Department at PCCOE&R. She has more than 11 year of Teaching Experience. Her field of interest is Artificial Intelligent and Data Mining.

Vaishali Latke She is currently an Assistant Professor of Computer Engineering Department at PCCOE&R. She has more than 9 year of Teaching Experience. Her field of interest is Artificial Intelligent and Machine Learning.