Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

Secure Reversible Image Data Hiding Over Encrypted Domain Via MSB Prediction Method

Aswathy Lekshmi. S
P.G. Student, Department of Electronics and Communication Engineering,
Mount Zion College of Engineering, Kadammanitta, Pathanamthitta, India

Hari. S
Assistant Professor, Department of Electronics and Communication Engineering,
Mount Zion College of Engineering, Kadammanitta, Pathanamthitta, India

Netha Merin Mathew
Assistant Professor, Department of Electronics and Communication Engineering,
Mount Zion College of Engineering, Kadammanitta, Pathanamthitta, India

***Abstract*— Data Privacy has received considerable attention in all fields. In the last few years, data and visual privacy has become a major problem. Because of this, Reversible Data Hiding (RDH) in encrypted image has a lot of attention from the communities of privacy, security and protection. The previous RDH methods are not able to hide a large amount of information with very high embedding capacity as well as it does not provide a very good reconstructed image quality. This paper presents a high efficient reversible data hiding in encrypted image by MSB prediction method. In this method RDH is a process to embedded useful data into a cover media or encrypted image using MSB prediction. It is a two step data hiding procedure, first embed data inside a media file and second, the hidden data and the original image can be loss lessly extracted. Here present two approaches: (1) High Capacity Reversible Data Hiding on MSBs with Correction of Prediction Error (CPE – MHCRDH), (2) High Capacity Reversible Data Hiding on MSBs with Embedded Prediction Errors (EPE – MHCRDH). Before transmitting the data embedded image to the receiver we can use image compression techniques to reduce the amount of data required to represent an image. With these methods, the results are better than those obtained in the previous method. The results shows very good reconstructed image quality and the embedding capacity can be double compare to the previous method.**

Keywords*— Reversible Data Hiding (RDH); Embedding Capacity; Image compression; PSNR; SSIM; MSB Prediction.*

I. INTRODUCTION

Communication of digital information becomes frequent nowadays, because of its fast access capability. A wide range of technologies for end-to-end protection are needed to resist the security threats in modern communication. Data hiding and Cryptography are the two main techniques for secure communication. In cryptography, the plain data is changed into an unreadable form called cipher data. The limitation of cryptography is that the third party is always conscious about the communication. In data hiding, the data is hidden in a cover file and it will be transmitted over the network [2] - [6]. Hiding the existence of secret information is the main advantage of data hiding technique over cryptography. Therefore, hacker or eavesdropper, and other, does not access the original message or any other type of information transmitted through the public networks such as internet.

A. Reversible Data Hiding

Data hiding is a process to embed useful data (information) into a cover media. RDH is a data hiding techniques. Reversible Data Hiding can be defined as an approach where data is hidden in the cover media that may be an image. It is a two-step data hiding procedure: first, embed (hide) data inside a media file and second, the hidden data and the original image can be loss lessly extracted. A reversible data hiding is an approach, which can recover the original image loss lessly after the data have been extracted from the cover image.

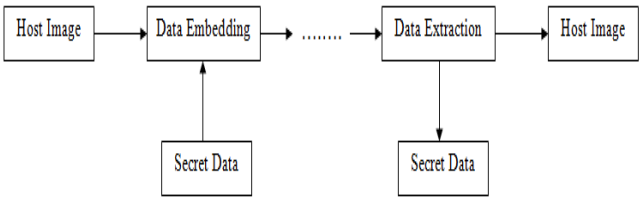


Fig 1: General Block Diagram of RDH

Reversible Data Embedding or hiding, which can also be called lossless data embedding or hiding, that embeds confidential data (which is called a payload) into a digital image in a reversible manner. As a basic requirement, the quality degradation on the cover image after data embedding should be low [7]. An interesting feature of reversible data

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

embedding is its reversibility, that is, one can remove the embedded data to restore the original image.

Reversible Data Hiding (RDH) in digital image that embeds data by altering the pixel values of the encrypted digital image for secret communication and the original image can be recovered after the extraction of the secret data. For data privacy, it is sometimes necessary to make an image unreadable. For this reason, a lot of encryption methods exist. The reason behind for hiding data is to avoid the misuse of data, hide traces of crime, military information, blackmail purpose, personal and private data etc...[11].

This paper introduces an effective approach for Data Hiding in Encrypted domain. We suggest embedding the secret message by two bit MSB substitution. As the values of the replaced MSB are lost during the data hiding phase, it is necessary to be able to predict them without errors during the decoding phase. In our design uses two methods for reversible data hiding, i.e., CPE – MHCRDH and EPE – MHCRDH.

The first CPE – MHCRDH (High Capacity Reversible Data Hiding on MSB by Correction of Prediction Errors) method consist of correcting all the prediction errors before encryption. At the decryption phase, the original image can be reconstructed perfectly but secret message has some distortions but where we are able to embed two bit per pixel. In the second method EPE – MHCRDH (High Capacity Reversible Data Hiding on MSB by Embedded Prediction Errors) consist of the original image is directly encrypted, but after the encryption step, the location of the prediction errors is embedded (EPE). During the data hiding phase, in both approaches, the MSB of each available pixel is substituted in the encrypted image by two bit of the secret message. After that the data hided image of both methods is compressed using loss lees LZ77 compression techniques. LZ77 compression works by finding sequence of data that are repeated. At the end of the process, the compressed image can be decompressed and the embedded data can be extracted without any errors and the clear image can be reconstructed loss lessly by using MSB prediction.

The proposed scheme provides a good security level and can be used to preserve the original image content confidentiality, while offering authenticity or integrity. It improves the reconstructed image quality as well as the embedding capacity.

The rest of the paper is organized as follows. Section II gives an overview of related work on reversible data hiding in encrypted images. Then, the proposed method is described in detail in Section III. Experimental results and analysis are provided in Section IV. Finally, the conclusion is drawn in Section V.

II. RELATED WORK

Reversible data hiding (RDH) is particularly suitable for authentication and data enrichment. It consists of embedding a hidden message into an image. After that, it is possible to extract the secret data and to losslessly recover the original image. The methods are based on lossless compression appending, difference expansion [16], [17], histogram shifting [18], [19], [15] or a combination of these schemes [11], [21]. Also, by altering the content of an original image, encryption provides a particular visual confidentiality. Cryptosystems can be divided into two groups

In [2], Pauline Puteaux and William Puech propose a reversible data hiding in encrypted images based on adaptive local entropy method. The original image is firstly encrypted with AES cryptosystem in ECB mode to ensure image content confidentiality. In fact, we suggest adapting the Shannon entropy measurement in order to be able to perform a significant local analysis. During the reconstruction phase, we use this statistical metric instead of standard deviation to loss lessly recover each block of 4 × 4 pixels from the original image. In this approach, the payload is quite small (0.0625 bpp). Weiming Zhang, Hui Wang, Dongdong Hou, Nenghai Yu in [3], propose a novel framework for RDH-EI by using reversible image transformation (RIT). RIT transfers the semantic (content) of the original image into the semantic of another image, and "reversibility" means that can be loss lessly restored from the transformed image.

In [4], this paper proposes a novel scheme of reversible data hiding (RDH) in encrypted images using distributed source coding (DSC). After the original image is encrypted using a stream cipher by the content owner, the data - hider compresses a series of bits selected from the encrypted image to make a room for the secret data. The bit selected from encrypted image is Slepian - Wolf encoded using low density parity check (LDPC) codes. On the receiver side, the secret bits can be extracted if the receiver has the embedding key only. In case were the receiver has only the encryption key, they can recover the original image using an image estimation algorithm. If the receiver has both the data hiding and encryption keys, they can recover both secret data and the original image using the distributed source decoding.

Zhang designed a separable method, where a part of the encrypted image was compressed to vacate room for the message embedding [6].

Wien Hong, Tung-Shou Chen, and Han-Yan Wu in [5], this letter proposes an improved version of Zhang's reversible data hiding method in encrypted images. The original work partitions the encrypted image into blocks, and each block carries one bit by replacing three LSBs of a set of pre-defined pixels. The data extraction and image recovery can be achieved by analyzing the block smoothness. Zhang's work is not fully utilize the pixels for calculating the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

smoothness of each block and does not consider the pixel correlations in the border of neighboring blocks. These two issues could reduce the correctness of data extraction. This paper gives a better scheme for measuring the smoothness of the blocks, and uses the side-match techniques to decrease the error rate of extracted-bits.

In [7], Cao et al. propose a sparse coding technique. By exploiting the local correlation between pixels, they could vacate a large space to hide information.

In [8], Zhang et al. encrypted the cover image by using public key cryptography with probabilistic and homomorphic properties. After the encryption phase, they embed data in the LSB planes of the encrypted pixels. During the decoding phase, as the introduced distortion was quite low, the embedded data is extracted and the original image was recovered loss lessly.

Chao-Yung Hsu, Chun-Shien Lu, and Soo-Chang Pei propose a privacy-preserving realization of the SIFT method based on homomorphic encryption. Security analysis is based on the discrete logarithm problem and RSA that PPSIFT is secure against cipher text attack only and known plaintext attack. Experimental results obtained from different case provides that proposed homomorphic encryption-based privacy-preserving SIFT performs comparably to the original SIFT and this method is useful in SIFT-based privacy-preserving applications [9].

Ma et al. were the first to describe a RRBE technique [10]. They proposed to release a part of the original image by applying a RDH method of histogram shifting. After that, they encrypted the image and then inserted information by substituting some LSB values in the encrypted image. With this method, the payload is higher than in previous methods (0.5 bpp) but the reconstructed image is altered when compared with the original (PSNR close to 40 dB).

In [11], prediction-error expansion (PEE) based reversible data hiding gives better exploiting image redundancy usually provides a superior performance. In this paper, an efficient RDH scheme based on pair wise PEE is proposed. The pair wise PEE is a novel reversible mapping that utilizes the correlations among prediction- errors. With the help of this type of correlations, the distortion can be controlled at a low level, and thereby the proposed scheme outperforms some state-of-the-art RDH algorithms.

III. PROPOSED SYSTEM

The existing methods are not succeeding in combining high embedding capacity and high visual quality. The previous method hide 1 bit per pixel in the MSB. It does not have high embedding capacity and visual quality. Also, the digital image usually requires large number of bits; cause critical problem for digital image data transmission and storage. For these reason,

here present a new high capacity reversible data hiding scheme for encrypted images based on MSB (two most significant bits) prediction with a very high capacity. The objective of the proposed algorithm is to simultaneously enhance the reconstructed image quality and embedding capacity. With this approach, in the encrypted domain, confidentiality is still the same during the decryption, the prediction of the MSB values is easier. Also we can use image compression techniques to reduce the amount of data required to represent an image and reducing the bandwidth required to transmit it. Compression is one of the most useful and a successful technology in the field of digital image processing.

We present two approaches, these are: high capacity reversible data hiding on MSBs with correction of prediction errors (CPE-MHCRDH) and high capacity reversible data hiding on MSBs with embedded prediction errors (EPE-MHCRDH).

In CPE-MHCRDH, it is possible to embed a secret message in the MSB of the encrypted image. At the decoding phase, the original image is reconstructed perfectly, but the secret message becomes distorted.

In EPE – MHCRDH method, it is possible to embed a secret message in the MSB of the encrypted image. At the decoding phase, the original image becomes distorted, but the secret message can be reconstructed perfectly.

A. CPE-MHCRDH

In the CPE-MHCRDH approach, as shown in Fig 2. In this method, we propose to embed the secret message (2 bit per pixel) by MSB substitution with very high embedding capacity and visual quality. The goal of our proposition is that an original image I, with m × n pixels, could be encrypted by using a secret key K_e and that another person could embed a message by using a data hiding key K_w, without knowing K_e [1].

The encoding phase consists of four steps: the prediction error detection, the image encryption, the 2 bit data hiding by MSB substitution and compression as shown in fig 4. For the decoding phase, there are three possible outcomes. If the recipient has just the encryption key, they can only obtain the original image, but not the embedded message. If the recipient has only data hiding key, the can obtain the secret message, but not the original image. Obviously, when they have both the encryption and the data hiding key, the recipient can extract the secret message and reconstruct the original image as shown in fig 5.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

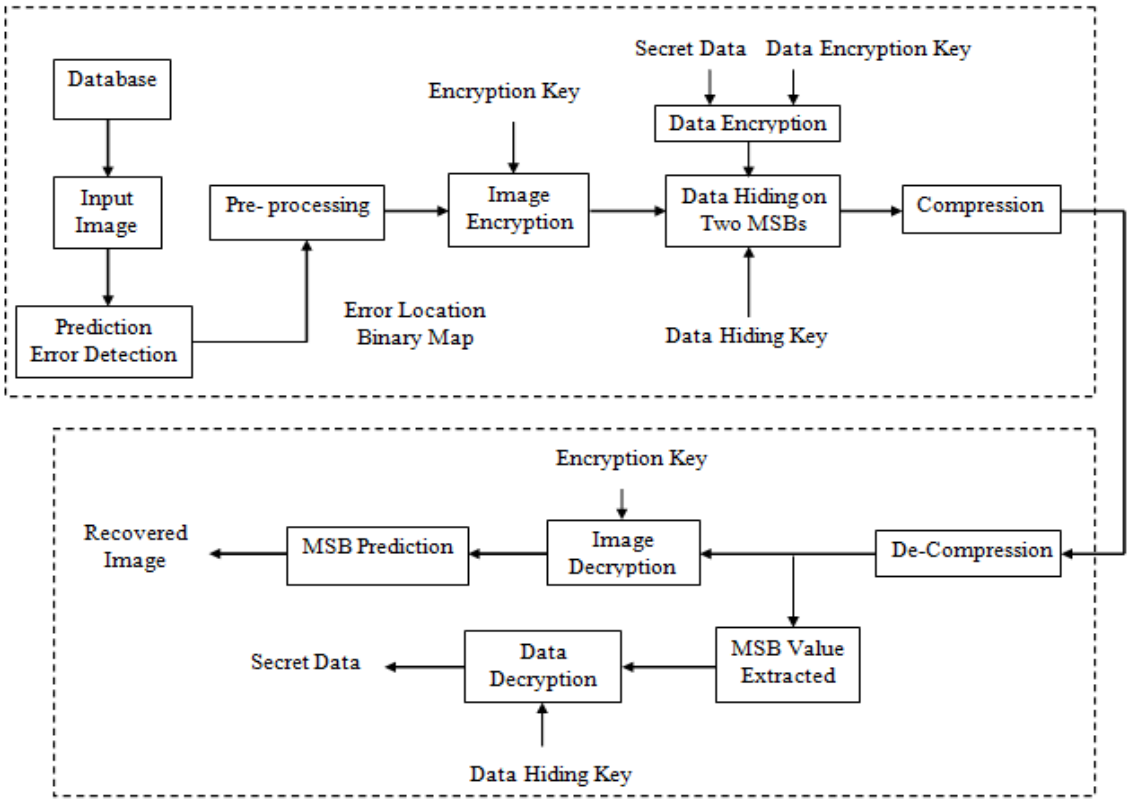


Fig 2: Block Diagram of CPE-MHCRDH Method

1. Prediction error detection:

In this method, the input image is taken from the data base. First predict all the prediction errors in the input image i.e., we can embed the secret message (2 bit per pixel) by MSB substitution, the original MSB values are lost after the data hiding step. It is important, during the decoding phase, to be able to predict them without any errors. Indeed, in order to reconstruct the original image, we propose to use the previous pixels to predict the current pixel value [1]. So, the first step consists of analyzing the original image content to detect all the possible prediction errors:

- Consider the current pixel P(i, j), with $0 \leq i < m$ and $0 \leq j < n$, and its inverse value, which is inv(i, j) = (P(i, j) + 128) mod 256.
- From the previously scanned neighbors of P(i, j), compute the value P_{error} (i, j) which is considered as a predictor during the decoding step.

	$p(i-1,j)$
$p(i,j-1)$	$p(i,j)$

Fig 3: Context of a pixel P(i, j)

$$p_{error}(i,j) = \frac{p(i-1,j) + p(i,j-1)}{2} \quad (1)$$

- Calculate the absolute difference between P_{error}(i, j) and P(i, j) and between P_{error}(i, j) and inv(i, j). Record the results as Δ and Δ*, so that:

$$\Delta = |p_{error}(i,j) - p(i,j)|$$
$$\Delta' = |p_{error}(i,j) - inv(i,j)| \quad (2)$$

- Compare the values of Δ and Δinv. If $\Delta < \Delta'$, there is no prediction error because the original value of P(i, j) is closer to its predictor than the inverse value. Otherwise, there is an error and we store this information into an error location binary map as shown in fig 2.
- After the prediction error detection phase, we propose to pre-process the original image I in order to obtain an image I' without any prediction errors. For each problematic pixel, we observe the amplitude of the error and we compute the value of the minimal pixel modification necessary to avoid this error. Eq. (4) shows the provision necessary to have no prediction errors during the decoding phase:

$$|p_{error}(i,j) - p(i,j)| < 64 \quad (3)$$

2. Image Encryption:

In order to make the original image I unreadable, we encrypt it by using an encryption key K_e, as shown in figure 3. The elements of this key are used as parameters of a chaotic generator, based on the Piecewise Linear

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

Chaotic Map (PWLCM) [20]. By using this chaotic generator, a sequence of pseudo-random bytes S(i, j) is obtained and the encrypted pixels P_e(i, j) can be calculated through exclusive-or (XOR) operation:

$$P_e(i,j) = s(i,j) \oplus p(i,j) \quad \textbf{(4)}$$

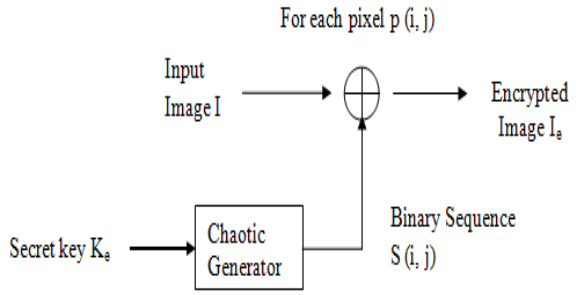


Fig 4: Encryption Step

3. Data embedding :

In the data embedding phase, it is possible to embed data in the encrypted image without knowing either the encryption key K_e used during the previous step or the original content of the image. By using the data hiding key K_w, the to-be-inserted message is first encrypted in order to prevent its detection after embedding in the marked encrypted image. Next, pixels of the encrypted image are scanned from left to right, then from top to bottom (scan line order) and the MSB of each available pixel is substituted by two bit b_T, with $0 \leq T < m \times n$, of the secret message:

$$p_{data}(i,j) = b_T \times 128 + (p_e(i,j) mod 128) \quad \textbf{(5)}$$

Note that only the first pixel cannot be marked because its value is not predictable, thus its value must not be changed.

4. Compression:

After this process, we can compress the data embedded image using image compression techniques. Image compression is reducing the size without degrading the quality of the image to an unacceptable level. Here we use LZ77 compression technique. LZ77 compression works by finding sequence of data that are repeated. The term "sliding window" is used; all of it really means that at any given point in the data, there is a record of what characters went before. A 32K of sliding window means that the compressor and decompressor have a record of what the last 32768 (32*1024) characters were. The next sequence of characters to be compressed is identical to the characters found within the sliding window, the sequence of characters is replaced by two numbers: a distance is representing how far back into the window the sequence starts, and a length, is representing the number of characters for which the sequence is identical.

5. Data extraction and image recovery

For the decoding phase, since our method is separable, we can extract the secret message and reconstruct the clear image I_R separately. Before that we can decompressed the data embedded image. I_R may be exactly like the original image I itself or a processed image I' very similar to the original image, depending upon which approach is used.

If the recipient has a key K_w, the pixels from the decompressed image are scanned in the scan line order and the MSB of each pixel are extracted in order to retrieve the encrypted secret message:

$$b_T = p_{data}(i,j)/128 \quad \textbf{(6)}$$

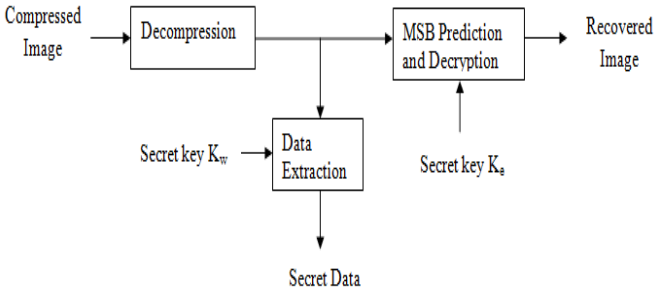


Fig 5: Decryption Step

If the recipient only has Ke, the image I_R can be reconstructed and the secret data can be extracted by using the following steps:

- The encryption key Ke is used to generate the sequence S(i, j), with m × n pseudo-random bytes.
- The pixels of the decrypted image are scanned in the scan line order, and for each pixel, the seven LSB are retrieved by XORing the decrypted value P_{data}(i, j) with the associated binary sequence S(i, j) in the pseudo-random stream:

$$\bar{p}(i,j) = s(i,j) \oplus p_{data}(i,j) \quad \textbf{(7)}$$

The MSB value is predicted:

- With the values of the previously decrypted adjacent pixels, the value of the predictor P_{error}(i, j) is computed.
- The pixel value is considered with MSB = 0 and with MSB = 1 and the differences between each of these two values and P_{error}(i, j) are calculated. These values are recorded as Δ^0 and Δ^1.

$$\Delta^0 = |p_{error}(i,j) - \bar{p}(i,j)^{MSB=0}|$$
$$\Delta^1 = |p_{error}(i,j) - \bar{p}(i,j)^{MSB=1}| \quad \textbf{(8)}$$

- The smallest value between Δ^0 and Δ^1 gives the searched pixel value:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

$$\bar{p}(i,j) = \begin{Bmatrix} \bar{p}(i,j)^{MSB=0}, if\ \Delta^0 < \Delta^1 \\ \bar{p}(i,j)^{MSB=1}, else \end{Bmatrix} \quad (9)$$

B. EPE – MHCRDH

In this method the original image is directly encrypted, but after the encryption step, the location of the prediction errors is embedded (EPE). During the data hiding phase, in both approaches, the MSB of each available pixel is substituted in the encrypted image by 2 bit of the secret message that embedded image can be loss lessly compressed. At the end of the process, the data embedded image can decompressed and the embedded data can be extracted without any errors and the clear image can be reconstructed loss lessly by using MSB prediction as shown in fig 6.

In this section, the encoding phase consists of four steps: the prediction error detection, the encryption, the embedding of the error location map and the reversible data hiding by MSB substitution, i.e., MSB of each available pixel is substituted by 2 bit of the secret message as same as the previous CPE–MHCRDH method. The difference between the two methods is that in CPE – MHCRDH method correcting the prediction errors (CPE) before encryption and in EPE – MHCRDH method the location of the prediction errors is embedded (EPE) [1]. The prediction error detection, compression and decompression are same as the CPE – MHCREDH method. Only difference in embedding error location binary map and data embedding step.

1. *Embedding of the error location binary map:*

During prediction error detection [1], the location of the prediction errors is stored in the error location binary map. Then, the original image I is encrypted by using Eq. (4). Before the embedding step, the encrypted image P_e is adapted to avoid prediction errors.

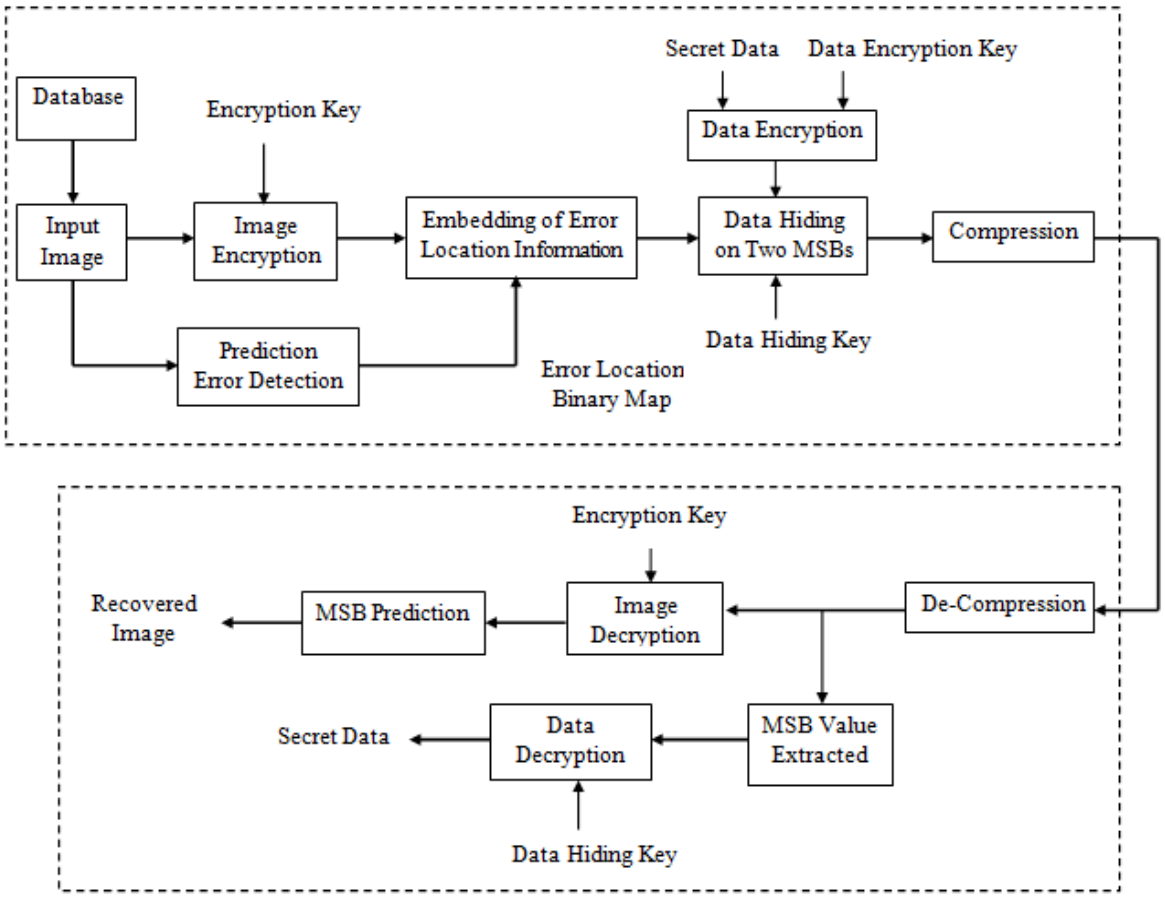


Fig 6: Block Diagram of EPE-MHCRDH Method

The encrypted image P_e is then divided into blocks of eight pixels and scanned, block by block, in the scan line order. If at least one prediction error is identified in a block according to the error location binary map, the current block is surrounded by two flags by replacing the MSB of each pixel in the previous and the following blocks by 1.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

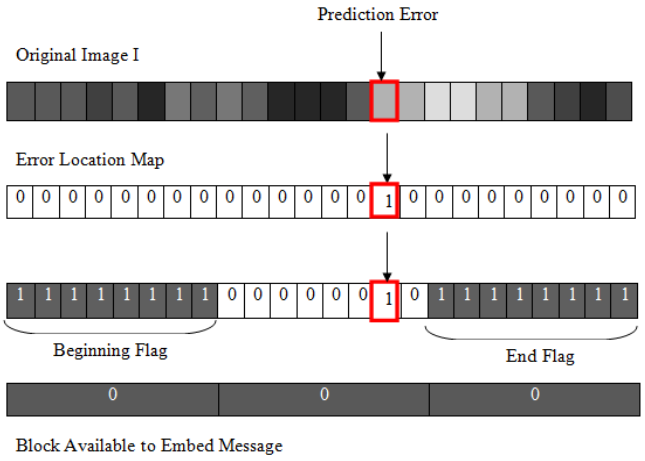


Fig 7: Finding available block for data hiding

If there are errors in two adjacent blocks, the flag which indicates the end of the error sequence is shifted until the next block without error. In the current block, the MSB value of a pixel is substituted by 1 if there is a prediction error and 0 if no error is detected, as indicated in Fig. 7. In the case where there is no error in the current block and if it does not serve as a flag, then the eight pixels of this block are used for data hiding i.e., the available block with 0 represent the block is not used for data hiding if there is 1 represent the block is used for data hiding.

2. Data extraction and image recovery:

In this phase, three cases are considered: (1) the recipient has only the data hiding key, (2) the recipient has only the encryption key and (3) the recipient has both the encryption and the watermarking keys.

In the first case, the recipient can extract the secret message by following these steps

- The pixels of the decompressed image are scanned in the scan line order and for each pixel; the MSB value is extracted, according to Eq. (6), and stored. Before the first sequence of eight MSB equal to 1, the extracted values are bits of the embedded message.
- When such a sequence is encountered, it indicates the beginning of an error sequence: the next pixels were not marked during the data hiding step. So, scan pixels until the next sequence of eight MSB equal to 1, which indicates the end of the error sequence.
- Repeat this process until the end of the image.
- Finally, use the data hiding key to obtain the clear text of the secret message

In the second case, the recipient can reconstruct the original image I by using MSB prediction:

- Use the encryption key to generate the pseudo-random chaotic sequence.
- Scan the pixels of the decompressed image in the Scan line order and for each pixel, retrieve the seven least significant bits (LSB) of P(i, j) by XORing the marked encrypted pixel value $P_{data}(i, j)$ with the associated binary sequence in the chaotic stream. Predict the MSB value
- Consider $P(i, j)^{MSB=0}$ and $P(i, j)^{MSB=1}$ as the pixel value with MSB = 0 and MSB = 1, respectively.
- Calculate the absolute difference between each of these two values with $P(i, j-1)$ and with $P(i-1, j)$ is $P_{error}(i, j)$. These values are recorded as Δ^0 and Δ^1 using Eq. 8.

The smaller value gives the original pixel value by using the Eq. 9. Finally, if the receiver has both the data hiding and encryption keys, they can extract the secret message and reconstruct the original image.

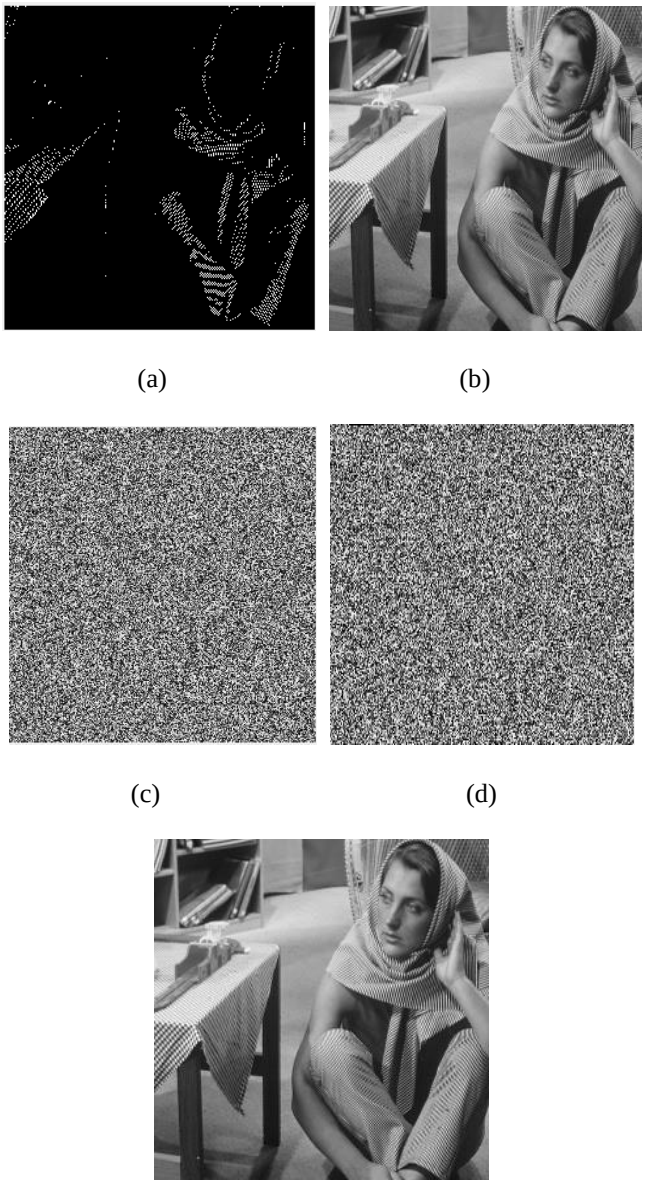
IV. RESULT

For data hiding methods in encrypted images, we have to measure different performances: embedding Capacity, PSNR, SSIM and Compression rate. It is necessary to find a trade-off between all of these parameters. We applied our method on different 512 x 512 images like Baboon, Barbara, Boat, Hill, House, Lena and BOWS-2 database [1].

We first applied our two approaches on the Barbara 512 × 512 pixels, illustrated in Fig. 8. Fig. 9 shows the results obtained with the CPE-MHCRDH approach and Fig. 10, with the EPE-MHCRDH approach. In Fig. 9.a and Fig. 10.a, in white, we can see the location of all the pixels with prediction errors.

Fig 8: Original image I

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019



(a) (b)

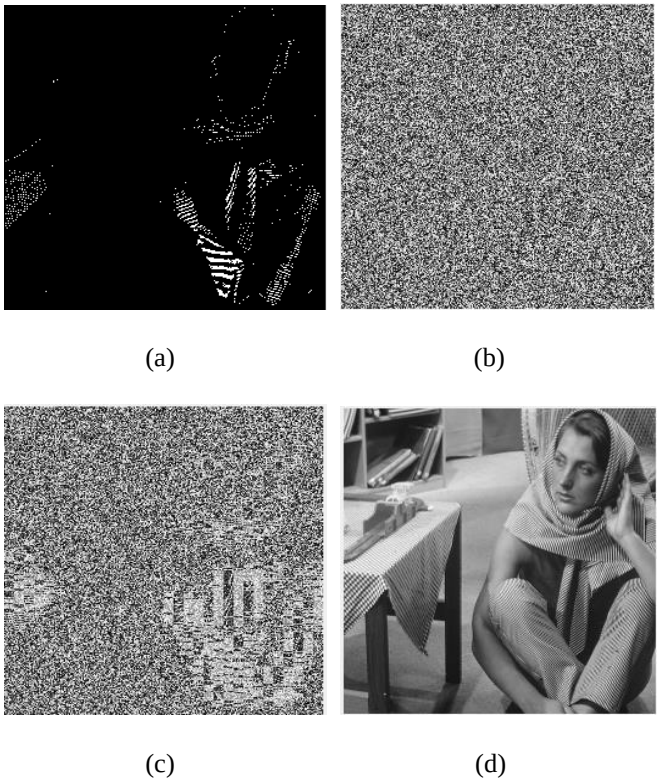
(c) (d)

(e)

Fig 9: Illustration of our proposed CPE – MHCRDH method on the test image Barbara (512 x 512), (a) Binary error location map with predicted error, (b) Pre-processed image $I^{'}$ (Error corrected image), (c) Encrypted image $P_e(i, j)$, (d) Data embedded image (2 bit per pixel) with 1.999985 capacity, (e) Recovered image, PSNR=44.699724 dB, SSIM= 0.996905

In the CPE-HCRDH approach (Fig. 9.a), they are pixels of the original image whose the MSB would be badly predicted if we do not adapt their values during the pre-processing phase. After this pre-processing step, the adapted images are obtained. This is similar to original image. The compression reduces the number of bits required to represent an image. The size before compression is 524288 and the size after compression is 89570. Therefore the percentage of compression is 17.0841. At the decoding phase the compressed image can be decompressed then extract the data and the image.

Finally, after data extraction, reconstructed images are exactly the same as pre-processed images and very similar to original images, even when some pixel values were changed during the preprocessing step. In these cases, PSNR is high and SSIM is close to 1: PSNR = 44.699 dB, SSIM = 0.9969 and PSNR and SSIM for other image are similar with this result.



(a) (b)

(c) (d)

Fig 10: Illustration of our proposed EPE – MHCRDH method on the test image Barbara (512 x 512), (a) Binary error location map with predicted error, (b) Encrypted image $P_e(i, j)$, (d) Data embedded image (2 bit per pixel) with 1.999985 capacity, (e) Recovered image, PSNR=43.999652 dB, SSIM= 0.995837

In the EPE-HCRDH approach (Fig. 10.a); they indicate all the pixels which will not be marked. Indeed, in addition, in grey we show the pixels which are not used to embed bits of the secret message because they serve as flags or are part of an error sequence. Note that the prediction errors are often on the edges and there is sometimes more than one error in the same block and, in these cases, the loss in terms of embedding capacity decreases.

Note that the prediction errors are often on the contours and there is sometimes more than one error in the same block. Moreover, if there are errors in two adjacent blocks, the flag which indicates the end of the error sequence is shifted. In these two cases, the loss of embedding capacity is then less important. All other pixels, in black, are used to embed bits of the secret message. Fig. 9.c and Fig. 10.b are the encrypted images. Fig. 9.a and Fig. 10.a are the highlighted prediction errors: the initial information and the errors location are not visible anymore. Fig. 9.d and Fig. 10.c present the data embedded encrypted images obtained in the final step of the encoding. Then data embedded image is compressed as same in the CPE – MHCRDH method.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

TABLE I. COMPARISON BETWEEN 1 BIT AND 2 BIT DATA HIDING

		Embedding Capacity	PSNR	SSIM	Compression Size	
					Before	After
CPE – MHCRDH Method	Hiding 1 bit per pixel	0.999985	41.017507	0.996883	524288	89578
	Hiding 2 bit per pixel	1.999985	44.513260	0.998328		
EPE – MHCRDH Method	Hiding 1 bit per pixel	0.999390	46.786415	0.995311	524288	89580
	Hiding 2 bit per pixel	1.998779	70.952962	0.995837		

In Fig. 9.e and Fig. 10.d, note that the reconstructed images are exactly the same as the original ones: all pixels are correctly reconstructed (PSNR = ∞, SSIM = 1). To conclude, our method proposes a very good tradeoff between the embedding capacity and the reconstructed image quality: it is possible to hide a large amount of data in an encrypted image and to recover perfectly the original image after data extraction.

V. CONCLUSION

In this paper, we propose a high capacity reversible data hiding method in encrypted images based on the MSB prediction. Indeed, by replacing all the MSB in the image, it is possible to hide two bit per pixel. In addition to this excellent embedding capacity, the reconstructed image quality is high. By analyzing the content of the original image, all the prediction errors are highlighted and the encrypted image is modified accordingly. After that, by substituting most of the MSB values in the image, it is possible to hide a large message and compression reduces the number of bit used to represent an image. There for we can reduce the bandwidth required to transmit and receive an image. Finally, in the extraction phase, the compressed image is decompressed and the original image is reconstructed without any errors and the secret message is perfectly extracted.

ACKNOWLEDGMENT

We thank the anonymous reviewers and the editors for the thorough reviews and comments. Also thanks the Head of Electronics and communication Department Professor Rangit Varghese and the M.Tech co-coordinator Asst. Prof. Shahana Habeeb Mohammed. Express our sincere gratitude internal supervisor, Asst Prof. Hari S for rendering us all the facilities for the completion of the project. Also like to express our thanks to all other faculty members of Electronics and Communication Department at Mount Zion College of Engineering, Kadammanitta for the valuable help provided by them.

REFERENCES

[1] Pauline Puteaux, and William Puech, “An Efficient MSB Prediction-Based Method for High-Capacity Reversible Data Hiding in Encrypted Images” IEEE Transactions on Information Forensics and Security, 2018.

[2] Pauline Puteaux and William Puech “ Reversible Data Hiding in Encrypted Images based on Adaptive Local Entropy Analysis” IEEE, 2017.

[3] Weiming Zhang, Hui Wang, Dongdong Hou, Nenghai Yu “Reversible Data Hiding in Encrypted Images by Reversible Image Transformation” Computing for Sustainable Global Development (INDIACom), 2016.

[4] Z. Qian and X. 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, 2016.

[5] W. Hong, T.-S. Chen, and H.-Y. Wu, “An improved reversible data hiding in encrypted images using side match,” IEEE Signal Processing Letters, vol. 19, no. 4, pp. 199–202, 2012.

[6] Xinpeng Zhang “Separable reversible data hiding in encrypted image,” IEEE Transactions on Information Forensics and Security, 2012.

[7] X. Cao, L. Du, X. Wei, D. Meng, and X. Guo, “High capacity reversible data hiding in encrypted images by patch-level sparse representation,” IEEE Transactions on Cybernetics, vol. 46, no. 5, pp. 1132–1143, 2016.

[8] X. Zhang, J. Long, Z. Wang, and H. Cheng, “Lossless and reversible data hiding in encrypted images with public-key cryptography,” IEEE Transactions on Circuits and Systems for Video Technology, 2016.

[9] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, “Image feature extraction in encrypted domain with privacy-preserving SIFT,” IEEE Transactions on Image Processing, vol. 21, no. 11, pp. 4593–4607, 2012.

[10] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, “Reversible data hiding in encrypted images by reserving room before encryption,” IEEE Transactions on Information Forensics and Security, 2013.

[11] B. Ou, X. Li, Y. Zhao, R. Ni, and Y.-Q. Shi, “Pairwise prediction-error expansion for efficient reversible data hiding,” IEEE Transactions on Image Processing, vol. 22, no. 12, pp. 5010–5021, 2013.

[12] P. Korshunov and T. Ebrahimi, “Scrambling-based tool for secure protection of JPEG images,” in Image Processing (ICIP), 2014 21th IEEE International Conference on, 2014, pp. 3423–3425.

[13] T.-H. Chen and K.-H. Tsao, “User-friendly random-grid-based visual secret sharing,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 21, no. 11, pp. 1693–1703, 2011.

[14] S. Li, G. Chen, and X. Mou, “On the dynamical degradation of digital piecewise linear chaotic maps,” International Journal of Bifurcation and Chaos, vol. 15, no. 10, pp. 3119–3151, 2005.

[15] W.-L. Tai, C.-M. Yeh, and C.-C. Chang, “Reversible data hiding based on histogram modification of pixel differences,” IEEE Transactions on Circuits and Systems for Video Technologyy, vol. 19, no. 6, pp. 906–910, 2009.

[16] J. Tian, “Reversible watermarking by difference expansion,” in Proceedings of Workshop on Multimedia and Security, vol.19,2002

[17] ——, “Reversible data embedding using a difference expansion,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 8, pp. 890–896, 2003.

[18] X. Gao, L. An, Y. Yuan, D. Tao, and X. Li, “Lossless data embedding using generalized statistical quantity histogram,” IEEE Transactions on Circuits and Systems for Video Technology, 2011.

[19] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 16, no. 3, pp. 354–362, 2006.

[20] S. Li, G. Chen, and X. Mou, “On the dynamical degradation of digital piecewise linear chaotic maps,” International Journal of Bifurcation and Chaos, vol. 15, no. 10, pp. 3119–3151,2005.

[21] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, “Reversible watermarking algorithm using sorting and prediction,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 19, no. 7,pp.989–999,2009.