

Encrypted image processing using compression and reversible data hiding

Yasmina Zine¹, Meriem Boumehed², Naima Hadj Said³

¹Coding and Information Security Laboratory-LACOSI, Department of Electronics, Faculty of Electrical Engineering, University of Science and Technology of Oran Mohamed Boudiaf, Oran, Algeria

²Department of Electrotechnics, Higher School of Electrical Engineering and Energetic of Oran, Oran, Algeria

³Department of Computer Science, University of Science and Technology Oran Mohamed Boudiaf, Oran, Algeria

Article Info

Article history:

Received Oct 17, 2023

Revised Jan 24, 2024

Accepted Feb 16, 2024

Keywords:

Compressed image

Encrypted image

Image bit-plane

Reversible data hiding

Scanning directions

ABSTRACT

Reversible data hiding within encrypted images reversible data hiding in encrypted images (RDH-EI) is a highly effective technique for image processing in the field of encryption. This paper, propose a RDH-EI technique, which utilizes bit-plane compression and various image scanning directions to generate vacant space for data embedding, referred to as vacating room. Initially, the prediction error of the pre-processed image is computed. Subsequently, each bit-plane image is converted into a bit-stream by following the pixel scan order employed before compression. The compressed image is then encrypted employing a stream cipher. Through the process of substitution, the secret data and additional information are incorporated into the acquired image without any knowledge of the original content or the encrypted key. Finally, the generated image is transmitted or archived. The experiments provide evidence that the proposed method surpasses the most advanced methods currently available.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Yasmina Zine

Coding and Information Security Laboratory-LACOSI, Department of Electronics, Faculty of Electrical Engineering, University of Science and Technology Oran Mohamed Boudiaf

El Mnaouar, BP 1505, Bir El Djir, 31000, Oran, Algérie

Email: yasmina.zine@univ-usto.dz

1. INTRODUCTION

In the field of communications, the exchange of images or their storage in cloud computing is an ever-evolving and promising area of research. This evolution has faced many challenges, the most common being image security, which has become an essential condition to ensure privacy and prevent unauthorized use of these data especially in sensitive fields, such as medical, civil, or military. For this reason, some processing techniques have been applied to the image.

In the past, the security of the image's content began with encryption which encodes the information from the clear domain to the encrypted domain and can be either symmetric (using the identical key for both encryption and decryption), asymmetric (using separate public and private keys), or chaotic (based on unpredictable algorithms) [1], [2]. With the development of technology, researchers showed a strong interest in the steganography [3], [4] and watermarking [5], [6] approaches, they both aim to hide information in multimedia support without encrypting it. In recent decades, a huge number of studies focused on reversible data hiding in the encrypted image. This method allows both hiding secret data within an encrypted image and later retrieving the original image and embedded data perfectly. Remarkably, it achieves this without having to know either the original image content or the encryption key to improve the security level of images during transmission or storage. Several reversible data hiding in encrypted images (RDH-EI)

approaches have been proposed to maximize the data hiding capacity and recover the original, those approaches are implemented in the spatial or the frequency domain [7], [8], this paper adopted a spatial domain approach. The RDH-EI schemes have been categorized into [9]: i) lossless compression [10], ii) differential expansion [11], iii) histogram shifting [12], iv) prediction error [13], and reserving room [14].

Puteaux and Puech [15] introduces the use of most significant bits (MSB) instead of the traditional least significant bits (LSB). This allows for direct embedding of secret data by replacing only the first MSB in each pixel. Error label map identifying non-embeddable pixels is generated via prediction error detection, this map itself becomes extra information embedded within the encrypted image.

Puyang *et al.* [16] have considerably increased the embedding capacity compared to P. Puteaux method by utilizing two-bit MSB. Yin *et al.* [17] have also proposed an improved approach of RDH-EI based on multi MSB prediction and Huffman coding. Firstly a label map has been generated using a predicted value of the original image, by a bitwise comparison until the first difference between the two bits. This difference varies between 0 and 8, this number corresponds to the MSB bits to be substituted. Then the label map is encoded with a predefined Huffman variable length coding labeling (HVLCL) rule and preserved as additional information with its length and HVLCL code. Finally, the additional information and the secret data are embedded into the encrypted image by Multi MSB substitution.

Yin *et al.* [18] presented a RDH-EI using two fundamental techniques: pixel prediction and reordered bitplanes. First, each prediction error bit plane of the original image is divided into blocks, and rearranged in a bit-stream to be compressed so as to vacate the room for hiding more data. After that, the resulting image is encrypted using stream cipher. Finally, the secret data is embedded by multi-LSB substitution.

In this paper, we improved RDH-EI by increasing the data hiding capacity based on reordered bit-planes and compressed bit-streams [19]. The major contributions are relied on: i) histogram equalization for the image pre-processing process which is a treatment that improves the bit-planes compression before encryption thanks to the distribution of the new pixels, or the similarity of the adjacent bits. The pre-processed image will be considered as the original image in the rest of the paper. ii) full bit-plane reordered instead of bloc bit-plane reordered for RDH-EI method and high embedded capacity obtained benefiting from special image correlation.

The current paper is structured as follows: In section 2, we present a schematic diagram of the proposed RDH-EI method and briefly describe the image transfer steps. The details of each step are described in section 3. Section 4 is reserved for the experimental results and discussion. A conclusion of this work is found in section 5.

2. PROPOSED METHOD

In this section, an efficient method of RDH-EI is proposed for the improvement of the data hiding capacity. As it can be seen in Figure 1, the original image is first pre-processed and then decomposed into bit-planes. A bit-stream is generated based on predicted error and reordering of each bit-plane and subsequently compressed for data hiding in the vacated room after encryption. The block schematic is divided into two main phases: a sending phase and a receiving phase. The sending phase is realized by two owners [20], as presented in Figure 1(a): i) image-owner: ensures the vacated room and encryption image, ii) data-hider: ensures the data embedding. The receiving phase consists of extracting the data and reconstructing the image, as presented in Figure 1(b).

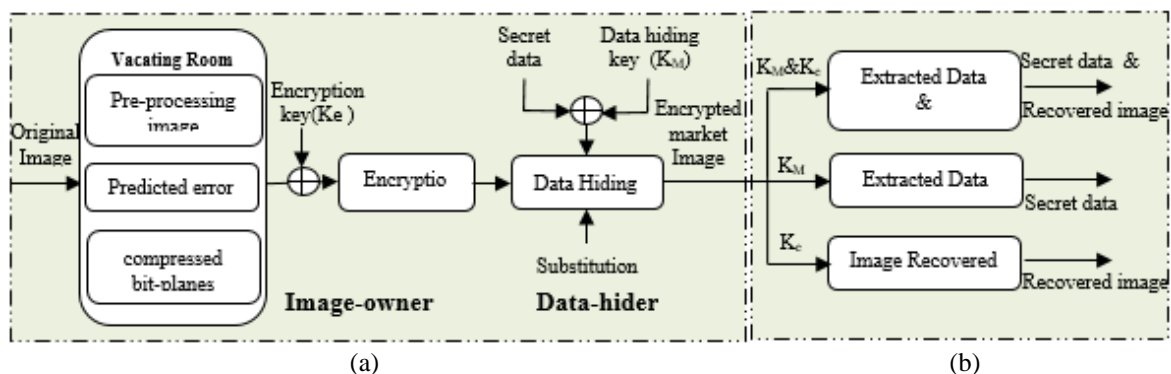


Figure 1. Block schematic of the method procedures (a) the sending phase and (b) the receiving phase

3. METHOD

This section explains a detailed description of each bloc in the proposed RDH-EI method. First, the image pre-processing procedure is described in subsection 3.1. The calculation of the predicted error is presented in subsection 3.2. A method to create space for data hiding is explained in subsection 3.3. After that, in subsection 3.4, the encryption image is treated. Subsection 3.5, presents data hiding procedures. Finally, subsection 3.6, describes the reversible way for extracting data and reconstructing images.

3.1. Image pre-processing

The pre-processing image step is performed by histogram equalization [21], [22], to display details not clearly visible in the original image. As presented in Figure 2, the resulting histogram is approximately uniform covering the range [0, 255], and the pre-processed image is visually comparable to the original one. Its pixel values correspond to the cumulative probability of its corresponding pixel values in the original image. This process increases the number of vacated rooms that will be demonstrated in section 4.

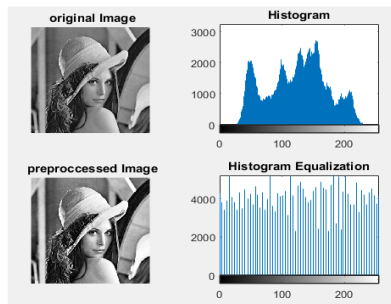


Figure 2. The histogram equalization of Lena's image

3.2. Calculation of the predicted error

The predicted value $predx(i, j)$ is given by (1), obtained by applying the median edge detection (MED) predictor [23] depending on a, b, and c, three neighboring pixels of $x(i, j)$ pixel of the original image I of size $M * N$, where $1 < i \leq M$ and $1 < j \leq N$, as demonstrated in Figure 3. The MED predictor is applied on all image pixels, except for row1 and column1 pixel values, these values cannot be predictable. So they maintain their original values.

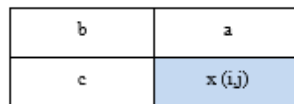


Figure 3. The MED predictor of the current pixel $x(i, j)$

$$predx(i, j) = \begin{cases} \max(a, b) & , \quad b \leq \min(a, c) \\ \min(a, b) & , \quad b \geq \max(a, c) \\ a + b - c & , \quad otherwise \end{cases} \quad (1)$$

The predicted error value $predE(i, j)$ is given by (2), then converted into binary.

$$predE(i, j) = x(i, j) - predx(i, j) \quad (2)$$

If $predE$ is over the range $[127, -127]$ the first MSB bit is presented by '1' for the negative value and '0' for the positive, and the seven remaining bits are presented by the binary form of $predE$ absolute value, given by (3). And if $predE$ is outside of the range, the values are considered as an overflow pixel, their coordinates are saved as additional information. In this case, $predE$ is presented by the original pixel value, and the binary form is given by (4). The predicted error value benefit is that the pixel values are more identical to increase the compression rate, thereby generating more free room for embedding data [24].

$$predE^k(i, j) = \left\lfloor \frac{|predE(i, j)|}{2^{k-1}} \right\rfloor \bmod 2 \quad k = 1, 2, \dots, 7 \tag{3}$$

$$predE^k(i, j) = \left\lfloor \frac{x(i, j) \bmod 2^{9-k}}{2^{8-k}} \right\rfloor \quad k = 1, 2, \dots, 8 \tag{4}$$

$predE^k(i, j)$ is the binary conversion of current $predE(i, j)$, and $\lfloor \cdot \rfloor$ is a floor function.

3.3. Compressed bit-stream

This part focuses on image compression [25], which aims to maximize the available space for data hiding through a bit-plane reordering. This reordering is followed by a compression process that optimizes the vacated room for embedding secret data. This technique ensures a more efficient compression, thereby enlarging the capacity for data hiding. Furthermore, it enhances the security of the hidden data, as the bit-plane reordering introduces an additional layer of complexity, making unauthorized detection and extraction significantly more challenging.

3.3.1. Bit-plane reordering

Yin *et al.* [18] and Wu *et al.* [26] have adopted the rearrangement of the image bit-planes into bit-streams based on block. In this paper, full bit-plane reordering is proposed to improve embedding capacity. For this purpose, four scanning type directions [27] are introduced as illustrated in Figure 4. Figure 5 illustrates the bit-plane reordering types. From the bit-plane shown in Figure 5(a), four bit-streams of adjacent ‘0’ and ‘1’ are generated depending on the type direction which is identified by two bits, as shown in Figure 5(b), and presented by Figure 6.

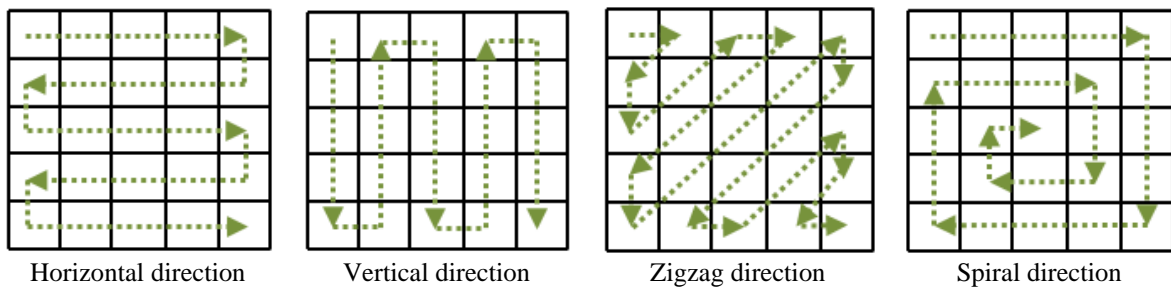


Figure 4. Different direction scanning of the bit-planes image

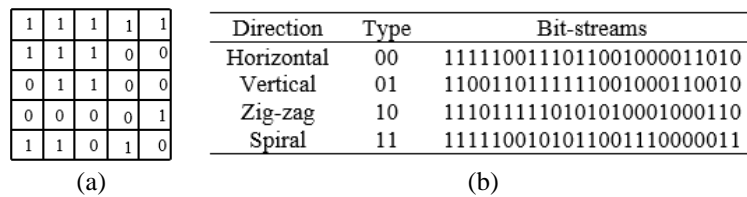


Figure 5. The bit-plane reordering types (a) the bit-plane and (b) the reordered bit-plane

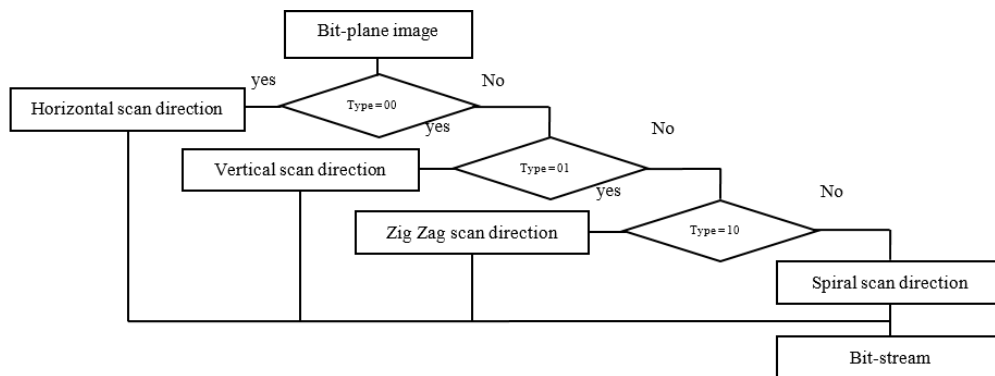


Figure 6. Reordered bit-stream

3.3.2. Compression concept

Once a bit-stream is obtained, the compression is required to vacate room for data hiding. Four compressed bit-streams CBS will be generated for each bit-plane, and the shortest of them is taken, as shown in Figure 7. The compressing step consists firstly of determining L , which is the number of the same consecutive bits. Then L is compared to the chosen default value, noted L_{fix} , in which two cases of compression are provided [18]:

Case1: if $L < L_{fix}$: $L_{pre} = 0$, and the value of L_{mid} is the identical bits of length less than L_{fix} .

Case2: if $L \geq L_{fix}$: L_{pre} composed of $l - 1$ consecutive 1 and terminates with 0, and $L_{tai} = 0$ or 1 depending on the identical bits.

Where L_{pre} , L_{mid} , and L_{tai} denote the prefix, middle, and tail of the partially compressed bit-streams, respectively. l and L_{mid} are calculated as follows:

$$l = \lfloor \log_2 L \rfloor \tag{5}$$

$$L_{mid} = (L - 2^l)_2 \tag{6}$$

Depending on the cases, the sub-sequences of the original bit-stream will be compressed, as illustrated in the following example, with $L_{fix}=5$:

Original bit-stream: $\underbrace{000000000}_{1^{st} \text{Sub-seq}} \underbrace{1100}_{2^{nd} \text{Sub-seq}} \underbrace{111111111111111}_{3^{rd} \text{Sub-seq}}$ (Length: 35)

Compressed bit-stream: $1100010:01100:01101101$ (length: 21)

The 35 original bits are compressed into 21 bits, which means that several vacated rooms will be generated for data hiding.

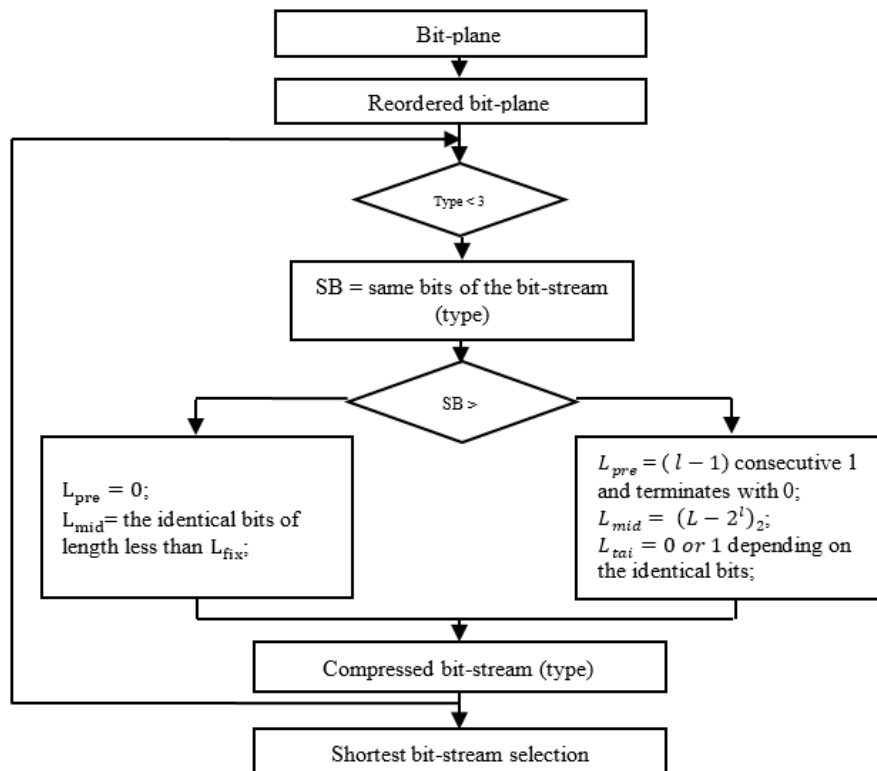


Figure 7. The shortest bit-stream

The compressed image is obtained as follows: i) the predicted error image is decomposed into eight bit-planes. ii) each bit-plane is reordered depending on the four different types, then compressed according to the two above cases. The length of the shortest CBS determines whether the bit-plane is compressible or not, and it will be marked by 1 or 0, the 1 mark indicates that the bit-plane is compressed. In this case, the

compressed bit-plane is represented by one bit of the mark followed by two bits of the type identification. Then the compressed bit-plane is followed by the shortest CBS bits. In the case that the bit-plane is not compressible, it is represented by one bit of the mark followed by the original bit-stream. iii) auxiliary information contains the parameter L_{fix} converted into three binary bits as well as the overflow number stored in eight binary bits, both are situated at the top bits of the MSB bit-plane. The length in binary of total compressed and uncompressed bit-planes is stored in eight binary bits and then located in the last bits of the LSB bit-plane. iv) finally, by connecting the compressed bit-planes followed by uncompressed bit-planes and then the vacated rooms completed with 0, the compressed image I_C is obtained, as shown in Figure 8.

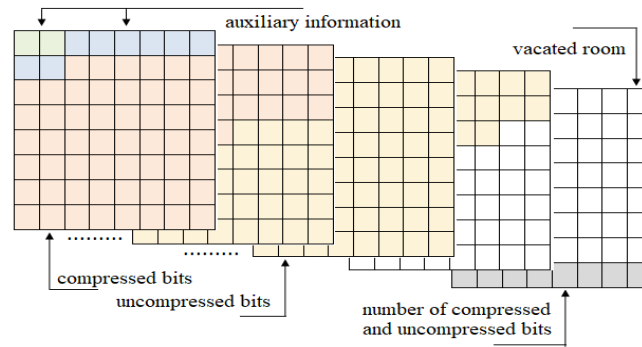


Figure 8. Compressed image bit-planes

3.4. Encryption image

There are several techniques for encrypting images [22], the encrypted image I_e is obtained using stream cipher by XORing bit by bit the compressed image $I_C(i, j)$ of size $M * N$ by the pseudo-random matrix $R(i, j)$ of the same size of I_C generated by the encryption key K_e , according to (7) [28].

$$I_e(i, j) = I_C^k(i, j) \oplus R^k(i, j) \quad (7)$$

$I_C^k(i, j)$ and $R^k(i, j)$ are the binary conversions of $I_C(i, j)$ and $R(i, j)$, respectively, as shown in (4).

3.5. Data hiding

At this stage, the image owner and the data-hider can be the same. Therefore, the data-hider has first to extract the auxiliary information stored in the LSB bit-plane of the encrypted image to deduce the net capacity embed N_C given by (8). After that, the additional data is encrypted by data hiding key K_D to increase security and will be then embedded in the vacated room by substitution. Finally, the resulting marked encrypted image I_{eM} is generated and transmitted to the recipient.

$$N_C = (M * N) - T_{CBS} - A_{inf} - Bl_{CBS} \quad (8)$$

Where $(M * N)$, is the size of the original image, T_{CBS} is the total bits of compressed and uncompressed bit- and it can be computed, A_{inf} is the auxiliary information's eleven bits length, and Bl_{CBS} , which equals eight bits storing the T_{CBS} length in binary.

3.6. Data extraction and image reconstruction

In this receiving phase, reversely the recipient has to extract the secret data without errors and reconstruct completely the original image. Knowing A_{inf} , Bl_{CBS} , and T_{CBS} and according to (8), the encrypted secret data is extracted from I_{eM} . To recover the encrypted image, the L_{fix} and the overflow pixels can be first extracted from the MSB bit-plane. Consequently, from the LSB bit-plane, the mark, the type identification, and the CBS length of each bit-plane are deduced. Finally, the encrypted image can be reconstructed. In conclusion, to extract the additional data and reconstruct the lossless original image depending on the key that the recipient contains, three scenarios are therefore possible: i) the receiver contains K_D the key of data hiding, ii) the receiver contains K_e the key of image encryption, and iii) the receiver contains K_D and K_e .

4. RESULTS AND DISCUSSION

The proposed method will be tested on the images in Figure 9. These are three common test images of standard 512×512 grayscale: Figures 9(a)-9(c). The performance will be evaluated using two metrics, namely peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM). This evaluation aims to verify the visual quality and similarity between the original image and the reconstructed image. The encrypted marked image is assessed using the embedded rate parameter (ER), which represents the amount of embedded data in the encrypted image per pixel. At the recipient's end, the embedded data can be extracted, and the original image can be restored without any loss. The ER is calculated by dividing the net embedding capacity N_C by the image size, expressed in bits per pixel (bpp).



Figure 9. The grayscale test images; (a) Lena, (b) Man, and (c) Baboon

4.1. Performance analysis of the proposed method

It should be stated that the experimental results shown in Figure 10, were obtained with the parameter $L_{fix} = 3$ and that their evaluation was realized by PSNR and SSIM metrics based on the test image Lena of 512x512 pixels shown in Figure 10(a). Figure 10(b) presents the pre-processed image with the histogram equalization, whereas Figure 10(c) shows its histogram. The encrypted image in Figure 10(d) is obtained after vacating room, by XORing the compressed image I' with the pseudo-random generated matrix R of the same size as I' with the encryption key Ke . Figure 10(e) is the encrypted marked image Ie_M by the secret data and auxiliary information. The attained embedding rate ER is 3.5822 bpp and the recovered image shown in Figure 10(f) is identical to the pre-processed image shown in Figure 10(b) with $PSNR = +\infty$, $MSE = 0$, and $SSIM = 1$. The same values of $PSNR$, MSE , and $SSIM$ are obtained for the other test images or $PSNR = +\infty$, $MSE = 0$, and $SSIM = 1$, that is to say that our method is completely reversible.

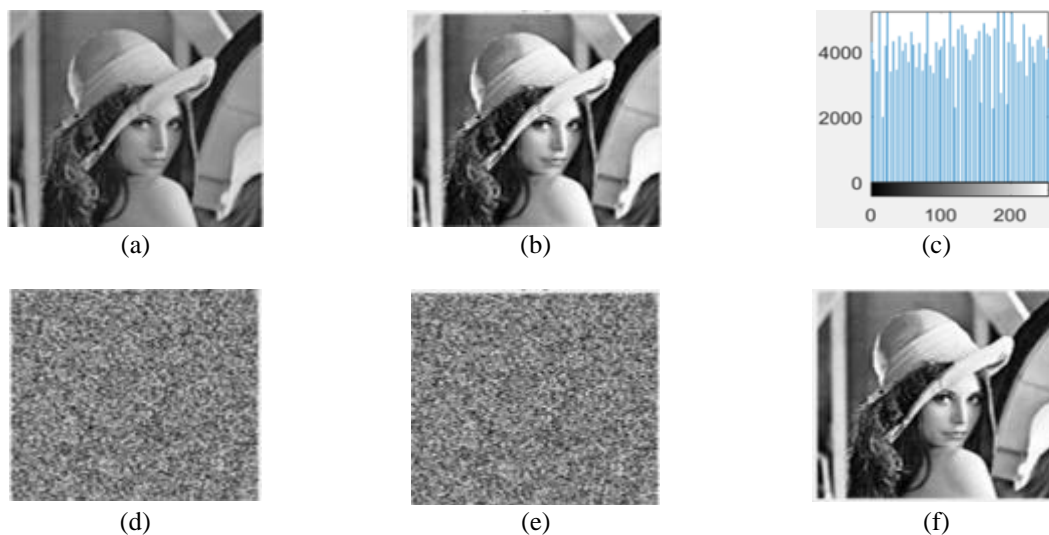


Figure 10. The experimental results with the test image Lena; (a) original image, (b) pre-processed image, (c) histogram of the preprocessed, (d) encrypted image, (e) marked encrypted image, and (f) recovered image with (PSNR→+ ∞dB, SSIM= 1.0)

As can be seen, the *ER* values of the pre-processed images presented in Table 1 are remarkably higher than the *ER* values of the images without pre-processing presented in Table 2. This demonstrates the impact of the histogram equalization step on the compression of images based on the similarity of the adjacent bits. The *ER* changes from one image to another according to the image grayscale, the longer the vacated room length is, the higher the *ER* is.

To prove that the suggested method has a superior *ER*, 200 random images from BOSSbase [29] and BOWS-2 [30] datasets are selected, where each one has 10,000 grayscale images of 512×512. A sample of the obtained results is recapitulated in Table 3. The metrics results range between a maximum and a minimum value of *ER* for both datasets. For the images of BOSSbase, the reached *ER* is 0.8694 bpp for the min value, 6.0892 bpp for the max value, and 3.885 bpp for the average value. For the BOWS-2 database, the *ER* for the min value and the max value are 0.713 bpp and 5.89 bpp, respectively, and the average *ER* is equal to 3.774 bpp. As we can see, for all the tested images the *PSNR* tends to $+\infty$, *MSE* = 0, and *SSIM* equal to 1, which means that the original image is losslessly recovered, and additional data are correctly extracted as well.

Table 1. The embedding rate with histogram equalization

Image	Vacated room (bit)	Embedding rate (bpp)
Lena	939059	3.5822
Man	804240	3.0679
Baboon	713039	2.7200

Table 2. The embedding rate without histogram equalization

Image	Vacated room (bit)	Embedding rate (bpp)
Lena	756389	3.038
Man	594013	2.266
Baboon	356441	1.359

Table 3. Results for the different databases

Database	Indicators	Max ER value	MinER value	Average value
BOSSbase	ER (bpp)	6.0892	0.869	3.885
	PSNR	$+\infty$	$+\infty$	$+\infty$
	SSIM	1	1	1
BOWS-2	ER (bpp)	5.891	0.713	3.774
	PSNR	$+\infty$	$+\infty$	$+\infty$
	SSIM	1	1	1

4.2. Comparison with the related work

This subsection presents the ranking of this method compared to the latest methods [9], [15], [17], [19], and [26] that use reversible data hiding and are realized simultaneously with data hiding extraction and image reconstruction. These methods focused as well on the embedding rate in the encrypted image. Figure 11 shows the *ER* of the common test images of these methods, which are Lena and Baboon. The bar graph demonstrates that the *ER* of the two test images of the suggested methods is greater than the *ER* of the other methods. Thus, we can deduce that the *ER* of the suggested method is the most effective.

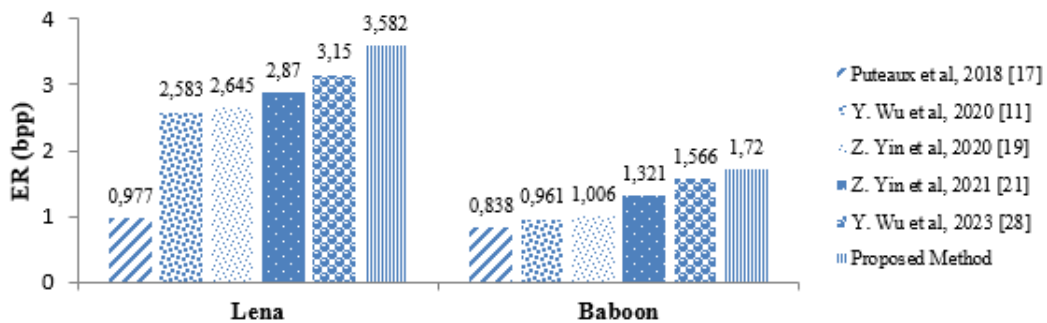


Figure 11. The ER of the common test images of the latest methods

Figure 12 illustrates an evaluation of both datasets [29] and [30]. As it can be seen, for BOSSbase dataset images, the average ER value reached by [15] is less than 1bpp, because this method is centered on one-bit MSB substitution, and the average ER of [17] proposed by Xiang et al is equal to 3.361 bpp, since this method is based on multi MSB substitution for each pixel, whereas the [19] method based on bit-plane compression reaches an average value of 3.763 bpp, which is improved compared to the previous methods. The suggested method stands out from the other methods by the highest average ER equal to 3.885 bpp due to multi-scanning directions. The BOWS-2dataset the same description is given for the BOSSbase dataset.

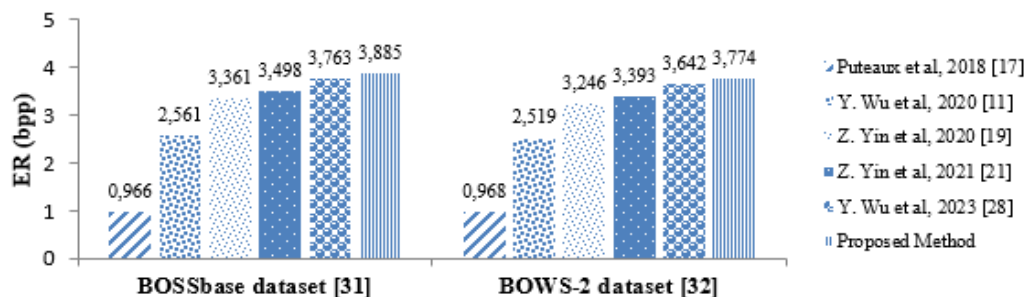


Figure 12. The average ER (bpp) ranking for BOSSbase and BOWS-2 datasets images

In this paper, we proposed an effective method for reversible data hiding in an encrypted image. We first applied pre-processing by histogram equalization to the original image, then, a predicted error value is generated by the pre-processed image predicted value that is compressed by bit-plane in order to generate a vacated room. To conclude, the secret data and the auxiliary information are embedded into the encrypted image by substituting the LSB. The practical results prove that the pre-processed image is perfectly reconstructed without error ($SSIM=1$ and $PSNR$ tends to $+\infty$ with $MSE=0$), which means that the original image and the reconstructed image are the same and the embedded data in average is equal to 3.885 bpp, where the data embedding capacity in the encrypted image is improved. In addition, our method which relies on bit-plane compression according to different directions and histogram equalization proved its effectiveness and accuracy compared to the latest related works.




REFERENCES

- [1] A. S. Sajitha and A. Shobha Rekh, "Review on various image encryption schemes," *Materials Today: Proceedings*, vol. 58, pp. 529–534, 2022, doi: 10.1016/j.matpr.2022.03.058.
- [2] M. Kaur and V. Kumar, "A comprehensive review on image encryption techniques," *Archives of Computational Methods in Engineering*, vol. 27, no. 1, pp. 15–43, Nov. 2020, doi: 10.1007/s11831-018-9298-8.
- [3] W. M. Abdulllah and A. M. S. Rahma, "A review on steganography techniques," *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)*, vol. 24, no. 1, pp. 131–150, 2016.
- [4] H. K. Tayyeh and A. S. A. Al-Jumaili, "A combination of least significant bit and deflate compression for image steganography," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 1, pp. 358–364, 2022, doi: 10.11591/ijece.v12i1.pp358-364.
- [5] M. Begum and M. S. Uddin, "Digital image watermarking techniques: A review," *Information (Switzerland)*, vol. 11, no. 2, p. 110, Feb. 2020, doi: 10.3390/info11020110.
- [6] N. E. Touati and A. M. Lakhdar, "Self embedding digital watermark using hybrid method against compression attack," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 2, pp. 864–870, 2021, doi: 10.11591/ijeecs.v24.i2.pp864-870.
- [7] P.-W. Huang, Y.-K. Chan, C.-Y. Chuang, and H.-C. Wang, "Reversible data hiding algorithm using dual domain embedding," in *Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation*, 2013, vol. 68, doi: 10.2991/3ca-13.2013.20.
- [8] F. Q. A. Alyousuf, R. Din, and A. J. Qasim, "Analysis review on spatial and transform domain technique in digital steganography," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, pp. 573–581, Apr. 2020, doi: 10.11591/eei.v9i2.2068.
- [9] Y. Wu, Y. Xiang, Y. Guo, J. Tang, and Z. Yin, "An improved reversible data hiding in encrypted images using parametric binary tree labeling," *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 1929–1938, Aug. 2020, doi: 10.1109/TMM.2019.2952979.
- [10] M. Bartwal and R. Bharti, "Lossless and reversible data hiding in encrypted images with public key cryptography," in *Proceedings of the Second International Conference on Research in Intelligent and Computing in Engineering*, Jun. 2017, vol. 10, pp. 127–134, doi: 10.15439/2017r88.
- [11] T. S. Nguyen, V. T. Huynh, and P. H. Vo, "A novel reversible data hiding algorithm based on enhanced reduced difference expansion," *Symmetry*, vol. 14, no. 8, p. 1726, Aug. 2022, doi: 10.3390/sym14081726.
- [12] W. Wang, J. Ye, T. Wang, and W. Wang, "A high capacity reversible data hiding scheme based on right-left shift," *Signal Processing*, vol. 150, pp. 102–115, Sep. 2018, doi: 10.1016/j.sigpro.2018.04.008.
- [13] X. Wu and W. Sun, "High-capacity reversible data hiding in encrypted images by prediction error," *Signal Processing*, vol. 104, pp. 387–400, Nov. 2014, doi: 10.1016/j.sigpro.2014.04.032.
- [14] Y. Qiu, Q. Ying, Y. Yang, H. Zeng, S. Li, and Z. Qian, "High-capacity framework for reversible data hiding in encrypted image using pixel prediction and entropy encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 9, pp. 5874–5887, Sep. 2022, doi: 10.1109/TCSVT.2022.3163905.




- [15] P. Puteaux and W. Puech, "An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1670–1681, Jul. 2018, doi: 10.1109/TIFS.2018.2799381.
- [16] Y. Puyang, Z. Yin, and Z. Qian, "Reversible data hiding in encrypted images with Two-MSB prediction," Dec. 2018, doi: 10.1109/WIFS.2018.8630785.
- [17] Z. Yin, Y. Xiang, and X. Zhang, "Reversible data hiding in encrypted images based on Multi-MSB prediction and Huffman coding," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 874–884, Apr. 2020, doi: 10.1109/TMM.2019.2936314.
- [18] Z. Yin, Y. Peng, and Y. Xiang, "Reversible data hiding in encrypted images based on pixel prediction and bit-plane compression," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 992–1002, 2022, doi: 10.1109/TDSC.2020.3019490.
- [19] Z. Yin, X. She, J. Tang, and B. Luo, "Reversible data hiding in encrypted images based on pixel prediction and multi-MSB planes rearrangement," *Signal Processing*, vol. 187, p. 108146, Oct. 2021, doi: 10.1016/j.sigpro.2021.108146.
- [20] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 4, pp. 255–258, Apr. 2011, doi: 10.1109/LSP.2011.2114651.
- [21] R. P. Singh and M. Dixit, "Histogram equalization: A strong technique for image enhancement," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, no. 8, pp. 345–352, Aug. 2015, doi: 10.14257/ijsp.2015.8.8.35.
- [22] S. R. Maniyath and V. Thanikaiselvan, "A novel efficient multiple encryption algorithm for real time images," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 2, pp. 1327–1336, Apr. 2020, doi: 10.11591/ijece.v10i2.pp1327-1336.
- [23] L. Liu, A. Wang, and C. C. Chang, "Separable reversible data hiding in encrypted images with high capacity based on median-edge detector prediction," *IEEE Access*, vol. 8, pp. 29639–29647, 2020, doi: 10.1109/ACCESS.2020.2972736.
- [24] R. Wang, G. Wu, Q. Wang, L. Yuan, Z. Zhang, and G. Miao, "Reversible data hiding in encrypted images using median edge detector and two's complement," *Symmetry*, vol. 13, no. 6, p. 921, May 2021, doi: 10.3390/sym13060921.
- [25] A. J. Qasim, R. Din, and F. Q. A. Alyousuf, "Review on techniques and file formats of image compression," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, pp. 602–610, Apr. 2020, doi: 10.11591/eei.v9i2.2085.
- [26] Y. Wu, W. Ma, Y. Peng, R. Zhang, and Z. Yin, "Reversible data hiding in encrypted images based on bit-plane compression of prediction error," *arXiv preprint arXiv:2007.04057*.
- [27] F. Lin, B. Wang, and Y. Li, "Differential direction adaptive based reversible information hiding," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10603 LNCS, Springer International Publishing, 2017, pp. 346–356.
- [28] A. Malik, H. X. Wang, Y. Chen, and A. N. Khan, "A reversible data hiding in encrypted image based on prediction-error estimation and location map," *Multimedia Tools and Applications*, vol. 79, no. 17–18, pp. 11591–11614, Jan. 2020, doi: 10.1007/s11042-019-08460-w.
- [29] P. Bas, T. Filler, and T. Pevný, "'Break our steganographic system': The ins and outs of organizing BOSS," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6958 LNCS, Springer Berlin Heidelberg, 2011, pp. 59–70.
- [30] P. Bas and T. Furon, "Image database of BOWS-2." <http://bows2.ec-lille.fr/> (accessed Jun. 20, 2017).

BIOGRAPHIES OF AUTHORS






Yasmina Zine    was born in Oran in Algeria. She received the engineering, the Master from the University of Sciences and Technology of Oran USTO-MB, Algeria in 2002 and 2016 respectively all in electronics, PHD student since 2020, Option: Cryptography and data security at the same university. Her areas of research are cryptography, data security and telecommunication. In addition, she works at Higher School of Electrical Engineering and Energetics of Oran, ESGEEO. She is member of the local commission for the promotion of visibility and ranking in ESGEEO. She can be contacted at email: yasmina.zine@univ-usto.dz.



Dr. Meriem Boumehed    was born in Algeria, completed her graduate and postgraduate studies at the University of Sciences and Technology, Mohammed Boudiaf (Oran, Algeria) where she successfully received the engineer (2004), magister diplomas (2007) and doctorate of science degree in Electronics (2013). Her research interests include computer vision, motion analysis in monocular and stereoscopic image sequences (detection, estimation, and segmentation). She is currently a senior lecturer at the Higher School of Electrical Engineering and Energetics of Oran, ESGEEO, Algeria and member of LDREI Intelligent Electrical Networks Laboratory and LSI Signal and Image Laboratory at USTO-MB University. She can be contacted at email: m_boumehed@yahoo.fr.



Prof. Dr. Naima Hadj Said    was born in Algeria. She received the engineering degree in telecommunications from the Institute of Telecommunication of Oran–Algeria (ITO) in 1986, and the magister degree from ITO in (1992) and a Ph.D. from the University of Sciences and Technology of Oran–Algeria (USTO) in 2005. Now, she is a Professor (teacher/researcher) at the computer sciences Department of University of Sciences and Technology of Oran (USTO). Her interest researches are in the area of digital communications, and cryptography. She can be contacted at email: naima.hadjsaid@univ-usto.dz.