An Efficient MSB Prediction-Based Method for High-Capacity Reversible Data Hiding in Encrypted Images

Pauline Puteaux, William Puech

An Efficient MSB Prediction-Based Method for High-Capacity Reversible Data Hiding in Encrypted Images

Pauline Puteaux, *Student Member, IEEE* and William Puech, *Senior Member, IEEE*

***Abstract*—Reversible data hiding in encrypted images (RDHEI) is an effective technique to embed data in the encrypted domain. An original image is encrypted with a secret key and during or after its transmission, it is possible to embed additional information in the encrypted image, without knowing the encryption key or the original content of the image. During the decoding process, the secret message can be extracted and the original image can be reconstructed. In the last few years, RDHEI has started to draw research interest. Indeed, with the development of cloud computing, data privacy has become a real issue. However, none of the existing methods allows us to hide a large amount of information in a reversible manner. In this paper, we propose a new reversible method based on MSB (most significant bit) prediction with a very high capacity. We present two approaches, these are: high capacity reversible data hiding approach with correction of prediction errors (CPE-HCRDH) and high capacity reversible data hiding approach with embedded prediction errors (EPE-HCRDH). With this method, regardless of the approach used, our results are better than those obtained with current state of the art methods, both in terms of reconstructed image quality and embedding capacity.**

***Index Terms*—Image encryption, image security, reversible data hiding, MSB prediction.**

I. Introduction

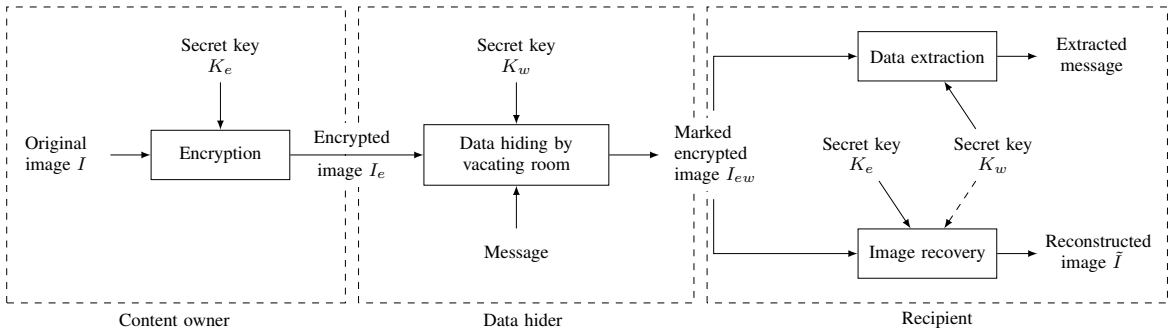
DIGITAL image security plays a significant role in all fields, especially in highly confidential areas like the military and medical worlds. With the development of cloud computing, the growth in information technology has led to serious security problems where confidentiality, authentication and integrity are constantly threatened, by illegal activities like hacking, copying or malicious use of information. The aim of encryption methods is to guarantee data privacy by fully or partially randomizing the content of original images [25]. During the transmission or the archiving of encrypted images, it is often necessary to analyze or to process them without knowing the original content, or the secret key used during the encryption phase [4].

In particular, methods of reversible data hiding in the encrypted domain (RDHEI) have been designed for data enrichment and authentication in the encrypted domain, when the encryption phase is necessarily done in the first place as, for example, in a cloud computing scenario. Without knowing the original content of the image or the secret key used to encrypt the image, it is then possible to embed a secret message in the encrypted image. During the decoding phase, the original image must be perfectly recoverable and the secret message must be extracted without error. Therefore, there exists a trade-off between the embedding capacity and the quality of the reconstructed image. In recent years, many methods have been designed. The space to embed the message may be vacated after or before the encryption phase and, during the decoding phase, image reconstruction and data extraction can be processed at the same time [17], [27] or separately [12], [27], [28].

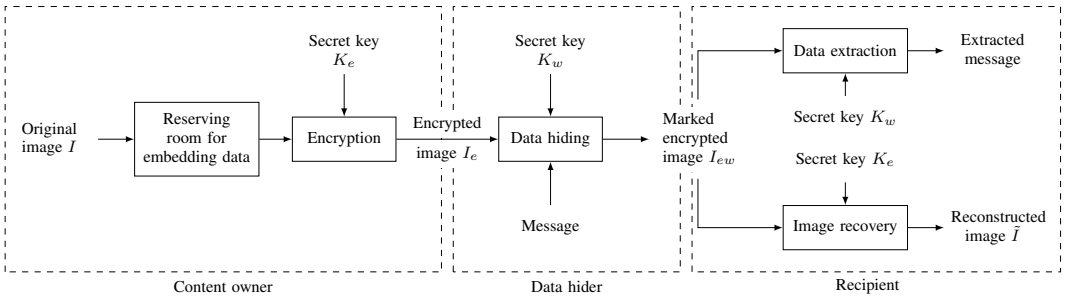
In all cases, the presented methods are not able to propose a high embedding rate together with a very good reconstructed image quality. In [12], the payload can be high (0.5 bpp), but the reconstructed image is altered when compared to the original (PSNR $\approx$ 40 dB). Moreover, other methods, such as Wu and Sun's version, propose a "high" embedding capacity, but it is only possible to embed approximately 0.1 bit per pixel at most [27]. Furthermore, in many of the existing methods, data hiding is made by LSB (least significant bit) substitution. However, in the encrypted domain, it is difficult to detect if an image contains a hidden message or not because pixels have pseudorandom values. For this reason, we propose to substitute the MSB (most significant bit) values instead of the LSB values. In fact, in the clear domain, MSB prediction is easier than LSB prediction and in the encrypted domain, confidentiality remains the same. Moreover, we do not need to preserve the high quality of the encrypted image compared to the clear domain.

In this paper, we present a new high capacity reversible data hiding scheme for encrypted images based on MSB prediction. Due to the local correlation between a pixel and its neighbors in a clear image, two adjacent pixel values are very close. For this reason, it seems natural to predict a pixel value by using already decrypted previous ones, as in many methods of image coding and compression. However, in some cases, there are some errors. So, the first step of our method consists of identifying all the prediction errors in the original image and to store this information in an error location binary map (note that using overhead such an additional map is not necessary for our proposed method). After that, we propose two different approaches: the CPE-HCRDH (high-capacity reversible data hiding with correction of prediction errors) and the EPE-HCRDH (high-capacity reversible data hiding with embedded prediction errors). The

P. Puteaux and W. Puech are with the Laboratoire d'Informatique, de Robotique et de Microlectronique de Montpellier, Centre National de la Recherche Scientifique, Univ. Montpellier, Montpellier 34095, France (e-mail: pauline.puteaux@lirmm.fr; william.puech@lirmm.fr).



(a) VRAE



(b) RRBE

Fig. 1: Two possible RDHEI schemes: vacating room after encryption (VRAE) and reserving room before encryption (RRBE).

CPE-HCRDH approach consists of correcting the prediction errors (CPE) before encryption. According to the error location map, the original image is pre-processed in order to avoid all the prediction errors and then, the pre-processed image is encrypted. In the EPE-HCRDH approach, 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 a bit of the secret message. At the end of the process, the embedded data can be extracted without any errors and the clear image can be reconstructed losslessly by using MSB prediction.

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 and future work is proposed 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. At the end of the process, it is possible to extract the secret message and to recover losslessly the original image. Methods are based on lossless compression appending, difference expansion [23], [24], histogram shifting [5], [15], [22] or a combination of these schemes [16], [21]. Also, by randomizing the content of an original image, encryption provides in particular visual confidentiality. Cryptosystems can be divided into two groups according to the method used: block cipher, or stream cipher [25]. Furthermore, encryption can be selective, when only certain details are hidden in the encrypted image [9], [18], [26], or fully when the global meaning of the image is kept entirely secret [13]. Sometimes, it is necessary to be able to analyze or process encrypted images without knowing the original content, or the secret key used during the encryption phase. Many applications exist, such as visual secret sharing (VSS) [3], [14], research and indexing in encrypted databases [7], [11] or recompression of crypto-compressed images [8].

For image notation or authentication purposes in the encrypted domain, reversible data hiding in encrypted images (RDHEI) methods have been proposed. They allow embed data in the encrypted domain without knowing the content of the clear image nor the encryption key. After the extraction of the message, it must be possible to reconstruct without distorting the original image. The challenge lies in finding the best trade-off between the embedding rate – also called payload – (in *bpp*), and the recovered image quality (in terms of PSNR or SSIM). These techniques can be classified into two categories, depending if the room is vacated after the encryption phase (VRAE) or reserved before image encryption (RRBE), as presented in Fig. 1. In addition, encryption and data hiding can be a joint process, when data extraction and original image reconstruction are completed at the same time, or separately.

In 2008, Puech *et al.* proposed one of the first joint methods [17]. They encrypted the original image by using AES and, after that, they embedded a bit of the secret message at a randomly selected position in each block of 4×4 pixels. In order to reconstruct the cover image, they performed an

analysis of the local standard deviation. In this approach, the payload is quite small (0.0625 bpp). Zhang, in [30], suggests encrypting the original image with a simple XOR operation. Then, the encrypted image is divided into blocks and each of them was partitioned into two sets. In one set, the three LSB of each pixel were compressed to vacate room for additional data. During the decoding phase, the block smoothness was observed to recover the original information and to extract the message. Hong *et al.* improved this approach by using a side match technique and an advanced formula to smoothness evaluation [6]. However, the reconstructed image quality remains of poor quality (with globally a PSNR less than 30 dB) when the payload is high. Zhou *et al.* designed a joint method where the image encryption was partial [33]. After the encryption phase, they used a public key modulation mechanism to embed additional data, without any access to the encryption key. To reconstruct the original image, they have to know which blocks of the image have been encrypted by using a SVM classifier.

Ma *et al.* were the first to describe a RRBE technique [12]. 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). Zhang *et al.* analyzed the prediction errors (PE) of some pixels and made space to hide data by using PE-histogram shifting before image encryption [29]. Zhang designed a separable method, where a part of the encrypted image was compressed to vacate room for the message embedding [31]. In this case, data extraction can be done before or after image decryption. In [28], Xu and Wang propose a new method based on histogram shifting and difference expansion. They used a stream cipher during the encryption phase and designed a specific encryption mode in order to encrypt the interpolation-error. In [2], Cao *et al.* propose a sparse coding technique. By exploiting the local correlation between pixels, they could vacate a large space to hide information. Qian and Zhang, in [20], described a method based on distributed source coding (DSC). They first encrypted the original image with a stream cipher and, after that, they compress some bits of the MSB planes to make room for the secret data. In [32], 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 losslessly.

In [27], Wu and Sun propose an advanced method, developed in two ways. The first approach is joint. They encrypted the original image in the same way as Zhang in [31] and, according to a data hiding key, selected some pixels to conceal data by histogram shifting. The second approach is separable: they hid bits of the secret message by MSB substitution. During the decoding phase, a median filter is applied on the marked image. Although the embedding capacity of this scheme was described as high, it is only possible to embed 0.1563 bpp at most.

III. Proposed RDHEI method with high capacity

None of the existing methods succeed in combining high embedding capacity (near 1 bpp) and high visual quality (greater than 50 dB). In most cases, the methods based on prediction error analysis (PE) or using a histogram shifting technique, the LSB values of some pixels are replaced to hide bits of the secret message. However, if an image is encrypted, it is difficult to detect if it contains a hidden message or not. In fact, the pixel values of an encrypted image are pseudo-randomly generated. So, there is no correlation between a pixel and its adjacent neighbors. For this reason, we propose to use the MSB values instead of the LSB values to embed the hidden message. With this approach, in the encrypted domain, confidentiality is still the same and during the decryption, the prediction of the MSB values is easier to obtain than those of the LSB.

In this section, we first introduce the global scheme of our proposed method of separable reversible data hiding in the encrypted domain. We suggest embedding the secret message by 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 the second part of this section, we present two possible approaches in detail taking into account the most important constraint which can be the fully reversibility (PSNR $\rightarrow +\infty$) or the maximum capacity (1 bpp). The first approach, which is not fully reversible, but where we are able to embed one bit per pixel, is called high-capacity reversible data hiding approach with correction of prediction errors (CPE-HCRDH). Compared to the approach proposed in [19], in this paper we develop more the three main steps and give more explanations and justifications. Moreover, we present more results by using a larger database for the experiments and provide a statistical analysis to evaluate the security level of the scheme. The second approach, where the original image is perfectly reconstructed, but where we have to adapt the to-be-inserted message, is called high-capacity reversible data hiding approach with embedded prediction errors (EPE-HCRDH).

A. Overview of the proposed method

The encoding phase includes three main steps, these are: the MSB prediction error detection, the joint MSB error consideration and encryption, and the data hiding by MSB substitution. This process is shown in Fig. 2. The goal of our proposition is that an original image I, with $m \times n$ 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. After this process, we obtain a marked encrypted image I_{ew}, which has exactly the same size as the original image.

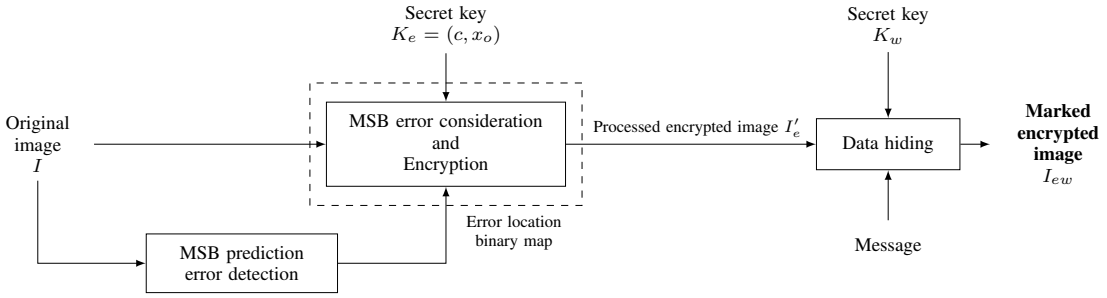


Fig. 2: Overview of the general encoding method.

1) Prediction error detection: In this method, since we propose to embed the secret message 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. 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 $inv(i,j) = (p(i,j) + 128) \mod 256$. Note that since there is a difference equal to 128 between these two values, then the inverse value must correspond to the original value of $p(i,j)$, but with the wrong MSB value.

- From the previously scanned neighbors of $p(i,j)$, compute the value $pred(i,j)$ which is considered as a predictor during the decoding step.

- Calculate the absolute difference between $pred(i,j)$ and $p(i,j)$ and between $pred(i,j)$ and $inv(i,j)$. Record the results as Δ and Δ^{inv}, so that:

$$\begin{cases} \Delta = |pred(i,j) - p(i,j)| \\ \Delta^{inv} = |pred(i,j) - inv(i,j)| \,. \end{cases} \tag{1}$$

- Compare the values of Δ and Δ^{inv}. If $\Delta < \Delta^{inv}$, 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 (note that using overhead such an additional map is not necessary for our proposed method), as illustrated in Fig. 2.

2) Image encryption: In order to make the original image I unreadable, we encrypt it by using an encryption key $K_e = (c, x_0)$, as shown in Fig. 3. The elements of this key are used as parameters of a chaotic generator, based on the Piecewise Linear Chaotic Map [10]. 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:

$$p_e(i,j) = s(i,j) \oplus p(i,j). \tag{2}$$

Note that since the encryption phase is fully reversible without overflow, it is then possible to recover the clear image without any alteration. Moreover, we also observe that even if we use a chaotic generator in our method, it is quite possible to generate a pseudo-random sequence with a cryptographically secure pseudo-random number generator (CSPRNG), or for example to use the AES algorithm in OFB mode. The only requirement is to use a stream cipher during the encryption phase.

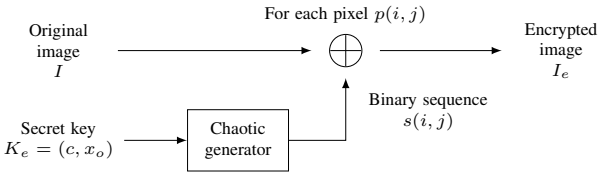


Fig. 3: Encryption step.

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 one bit b_k, with $0 \leq k < m \times n$, of the secret message:

$$p_{ew}(i,j) = b_k \times 128 + (p_e(i,j) \mod 128). \tag{3}$$

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

4) 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 $\tilde{I}$ separately. $\tilde{I}$ 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. There are three possible outcomes:

1) the recipient has only the data hiding key K_w,
2) the recipient has only the encryption key K_e,
3) the recipient has both keys.

An overview of the decoding method is presented in Fig. 4.

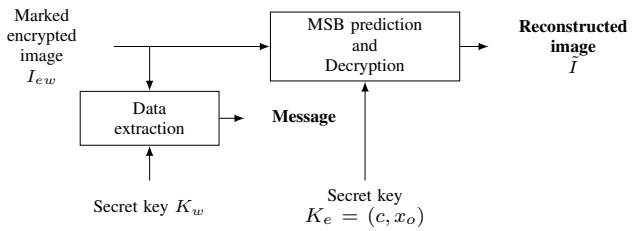


Fig. 4: Overview of the decoding method.

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

$$b_k = p_{ew}(i,j)/128, \tag{4}$$

where $0 \leq k < m \times n$ and refers to the index of the extracted bit in the message.

Then, by using the data hiding key K_w, the corresponding plaintext can be obtained.

In the second scenario, if the recipient only has K_e, the image $\tilde{I}$ can be reconstructed, before the data hiding and the encryption steps, by proceeding as follows:

1) The encryption key K_e is used to generate the sequence $s(i,j)$, with $m \times n$ pseudo-random bytes.
2) The pixels of the marked encrypted image are scanned in the scan line order, and for each pixel, the seven LSB are retrieved by XORing the marked encrypted value $p_{ew}(i,j)$ with the associated binary sequence $s(i,j)$ in the pseudo-random stream:

$$\tilde{p}(i,j) = s(i,j) \oplus p_{ew}(i,j), \tag{5}$$

   where $\oplus$ represents the XOR operation.
3) The MSB value is predicted:
   - With the values of the previously decrypted adjacent pixels, the value of the predictor $pred(i,j)$ is computed.
   - The pixel value is considered with MSB $= 0$ and with MSB $= 1$ and the differences between each of these two values and $pred(i,j)$ are calculated. These values are recorded as Δ^0 and Δ^1:

$$\begin{cases} \Delta^0 = \left| pred(i,j) - \tilde{p}(i,j)^{MSB=0} \right|, \\ \Delta^1 = \left| pred(i,j) - \tilde{p}(i,j)^{MSB=1} \right|. \end{cases} \tag{6}$$

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

$$\tilde{p}(i,j) = \begin{cases} \tilde{p}(i,j)^{\text{MSB}=0}, \text{ if } \Delta^0 < \Delta^1, \\ \tilde{p}(i,j)^{\text{MSB}=1}, \text{ else.} \end{cases} \tag{7}$$

B. CPE-HCRDH approach

In the CPE-HCRDH approach (high-capacity reversible data hiding approach with correction of prediction errors), as shown in Fig. 5, we first pre-process the original image to avoid all the prediction errors in order to be able to reconstruct the image during the decoding step. After this process, we can encrypt the pre-processed image without any problems. During the embedding phase, all the pixels of the encrypted image are marked with one bit of the message. Using this approach, we have a maximal payload, equal to 1 *bpp*.

1) Used predictor: As explained in the Section III-A1, we proposed to use the previous pixels to predict the value of the current pixel. For this approach (except for the first row and the first column) we consider the average of the left and the top pixels as a predictor $pred(i,j)$ for example:

$$pred(i,j) = \frac{p(i-1,j) + p(i,j-1)}{2}. \tag{8}$$

Indeed, using the average value as a predictor mitigates the to-be-performed pixel modification when there is an error, especially when there is a high difference between the current pixel value and one of its neighboring values.

2) Image pre-processing: 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. (9) shows the provision necessary to have no prediction errors during the decoding phase:

$$|pred(i,j) - p(i,j)| < 64. \tag{9}$$

The detailed pre-processing algorithm to correct all the prediction errors is presented in Algorithm 1.

For example, if we have $p(i,j) = 50$, $p(i-1,j) = 78$ and $p(i,j-1) = 154$, then:

$$inv(i,j) = (50 + 128) \mod 256 = 178,$$

$$pred(i,j) = \frac{78 + 154}{2} = 116.$$

We compute Δ and Δ^{inv}:

$$\Delta = |116 - 50| = 66, \ \Delta^{inv} = |116 - 178| = 62.$$

As $\Delta \geq \Delta^{inv}$, there is an error and we have to modify the value of the current pixel $p(i,j)$. We would like to have:

$$pred(i,j) - p(i,j) < p(i,j) + 128 - pred(i,j).$$

By developing this expression, we obtain:

$$p(i,j) > pred(i,j) - 64.$$

The modification of $p(i,j)$ which minimizes distortion is also:

$$p'(i,j) = pred(i,j) - 63 = 116 - 63 = 53.$$

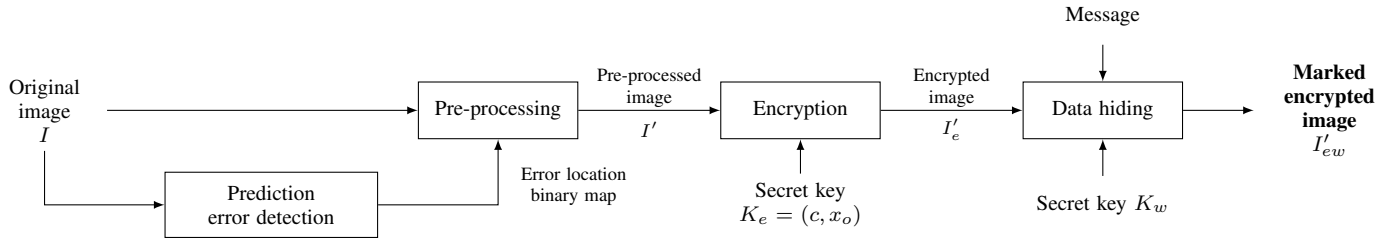


Fig. 5: CPE-HCRDH approach encoding phase.

Algorithm 1 Pre-processing algorithm.

Require: Original $m \times n$ image I
Ensure: Pre-processed $m \times n$ image I'
  for $i \leftarrow 0$ **to** m **do**
    for $j \leftarrow 0$ **to** n **do**
      $inv(i,j) \leftarrow (p(i,j) + 128) \bmod 256$;
      if $i = 0$ **or** $j = 0$ **then**
        special processing;
      else
        $pred(i,j) \leftarrow \frac{p(i-1,j)+p(i,j-1)}{2}$;
      end if
      $\Delta \leftarrow |pred(i,j) - p(i,j)|$;
      $\Delta^{inv} \leftarrow |pred(i,j) - inv(i,j)|$;
      if $\Delta \geq \Delta^{inv}$ **then**
        if $p(i,j) < 128$ **then**
          $p'(i,j) \leftarrow pred(i,j) - 63$;
        else
          $p'(i,j) \leftarrow pred(i,j) + 63$;
        end if
      else
        $p'(i,j) \leftarrow p(i,j)$;
      end if
    end for
  end for

After this phase, the pre-processing image I' is encrypted according to Eq. (2). Then, we perform the data hiding by embedding one bit of the secret message in each pixel of the encrypted image I'_e by MSB substitution, by following Eq. (3). We then obtain the marked encrypted image I'_{ew} with a maximum payload of 1 *bpp*.

3) Data extraction and image recovery: During the decoding phase, to extract the secret message, the marked encrypted image I'_{ew} is scanned and the MSB of each pixel is simply extracted by using Eq. (4). On the other hand, the pre-processed image I' can be reconstructed without any alteration. We first decrypt the marked encrypted image I'_{ew} to obtain the seven less significant bits (Eq. (5)) and, then, we predict the MSB value, according to Eq. (6) and Eq. (7). The reconstructed image is very similar to the original one.

C. EPE-HCRDH approach

In the EPE-HCRDH approach (high-capacity reversible data hiding approach with embedded prediction errors), the main goal is to exactly reconstruct the original image. In this case, the payload could decrease a little because of the storage of the error location information. In order to highlight the prediction errors, we adapt the to-be-inserted information according to the error location binary map, built during the prediction error detection phase. Then, the original image is encrypted and immediately after, the error location information is embedded in the encrypted image. During the data hiding step, we can only hide bits of the secret message in the available pixels. At the end of the decoding step, with the help of the location error information, the original image can be reconstructed without any visible alteration, which is indicated by a PSNR which tends to $+\infty$. A global scheme of this approach is presented in Fig. 6.

1) Used predictor: In this scheme, for each pixel, we have two possible predictors: the left pixel $p(i, j-1)$ and the top pixel $p(i-1, j)$. To determine which of these values is considered as a predictor, the absolute difference with the current pixel $p(i,j)$ is calculated and the closest value is chosen:

$$\begin{aligned} &\text{If} \quad |p(i-1,j) - p(i,j)| < |p(i,j-1) - p(i,j)|\,, \\ &\text{then,} \quad pred(i,j) = p(i-1,j), \\ &\text{else,} \quad pred(i,j) = p(i,j-1). \end{aligned} \tag{10}$$

In some cases, the other value can be chosen as a predictor for the inverse pixel value $inv(i,j)$ during the prediction error detection phase, but the result will remain the same. Note that it is also possible to use the average value of the left and the top pixels as a predictor, like in the CPE-HCRDH approach, but experimentally, we note that results are slightly less good.

2) Embedding of the error location information: During prediction error detection, the location of the prediction errors is stored in the error location binary map, as explained in Section III-A1. Then, the original image I is encrypted by using Eq. (2). Before the embedding step, the encrypted image I_e is adapted to avoid prediction errors. The encrypted image I_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. 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

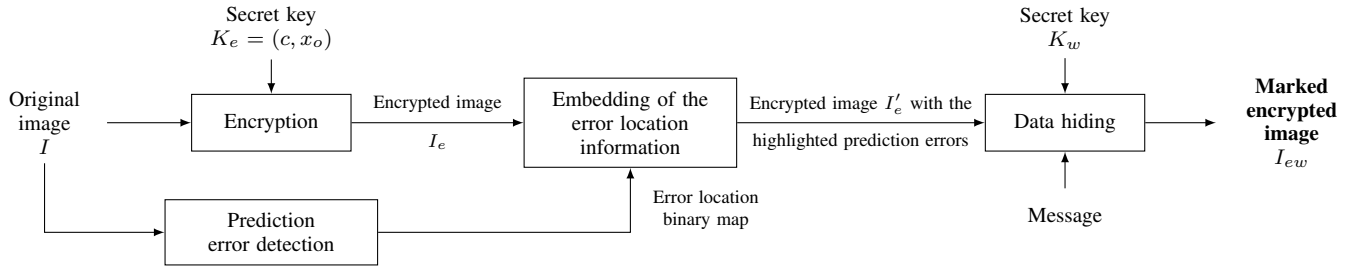


Fig. 6: EPE-HCRDH approach encoding phase.

data hiding as described in Section III-A3. 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. The loss of embedding capacity is also then less important since the flags are used for more than one prediction error. Note that it is possible to consider blocks of smaller size but statistically the risk that a part of the secret message should be taken for a flag will increase. With blocks of eight pixels, there is a good trade-off between the loss of embedding capacity and the false alarm rate. In fact, there are few unmarked pixels and the probability that a part of the message seems to a flag is very small ($\frac{1}{2^8}$).

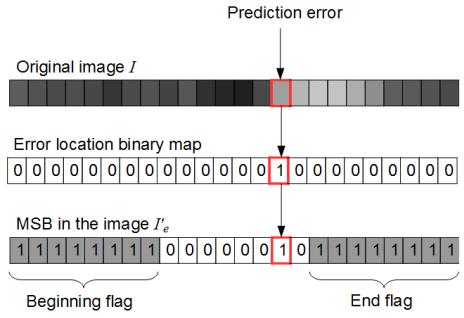


Fig. 7: Prediction error highlighting.

Then, the encrypted image I'_e is obtained, where the prediction errors are highlighted. Using this technique, during the data hiding phase, the person who wants to mark the image can extract the MSB value of each pixel and use the error location information to detect where it is possible to embed bits of the secret message (*i.e.* in all the blocks where there is no prediction error and which do not serve as flags). All the available pixels are then marked to obtain the marked encrypted image I_{ew}, by using Eq. (3).

3) Data extraction and image recovery: During the decoding step, the secret message can be extracted by following these steps:

- The pixels of the marked-encrypted image I_{ew} are scanned in the scan line order and for each pixel, the MSB value is extracted, according to Eq. (4), and stored. We assume that 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. Since the next pixels are not marked during the data hiding step, pixels are scanned until the next sequence where eight MSB are equal to 1, which indicates the end of the error sequence.
- This process is repeated until the end of the image.

Conversely, as this method is fully reversible, the original image I can be perfectly reconstructed. Firstly, the marked encrypted image I_{ew} is decrypted to recover the seven LSB of each pixel, by using Eq. (5). Then, the MSB values of the pixels are predicted with Eq. (6) and Eq. (7).

IV. Experimental results

In this section, we present the results we obtained by applying our method with the CPE-HCRDH approach (high-capacity reversible data hiding approach with correction of prediction errors) and the EPE-HCRDH approach (high-capacity reversible data hiding approach with embedded prediction errors). Section IV-A gives a full example for the two approaches and shows the obtained results on 10,000 images from the BOWS-2 database [1]. Then, in Section IV-B, we perform a statistical analysis in order to test the visual security of our method. Finally, in Section IV-C, we compare our two approaches with related methods and discuss its efficiency.

For data hiding in encrypted images, we have to measure different performances which are the number of incorrect extracted bits, the payload (*i.e.* embedding rate) and the reconstructed image quality after data extraction. We are interested to discover the best trade-off between all these parameters.

The payload is expressed in bit per pixel (*bpp*) and is expected to be as large as possible in order to conceal the maximum amount of information. To evaluate the reconstructed image quality in comparison to the original one, we use two metrics with full reference which are peak-signal-to-noise ratio (PSNR) and structural similarity (SSIM).

A. A detailed example for the two proposed approaches

We first applied our two approaches on the same original image of 512×512 pixels, from the BOWS-2 database [1], illustrated in Fig. 8. Fig. 9 shows the results obtained with the CPE-HCRDH approach and Fig. 10, with the EPE-HCRDH approach. For the two scenarios, we used the secret key $K_e = (c, x_0) = (0.123456789, 0.567894123)$. In Fig. 9.a and Fig. 10.a, in white, we can see the location of all the pixels with prediction errors. We can observe that, in these

Fig. 8: Original image I from the BOWS-2 database [1].

two approaches, we have neither the same prediction errors, nor the same number of errors, because we do not use the same predictor, as explained in Section III-B1 and Section III-C1. But globally they are in the same order of magnitude.

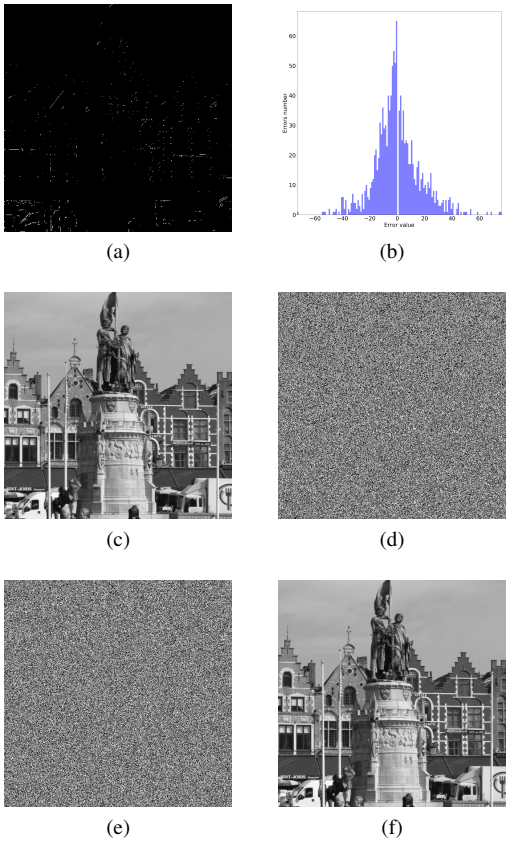


Fig. 9: Experiment using our CPE-HCRDH approach, with an embedding rate equal to 1 bpp: a) Errors' location, number of errors = 1242 (0.47%), b) Histogram of the estimated prediction errors, c) Pre-processed image I', PSNR = 46.87 dB, d) Encrypted image I'_e, e) Marked encrypted image I'_{ew}, f) Reconstructed image I', PSNR = 46.87 dB, SSIM = 0.9997.

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. 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

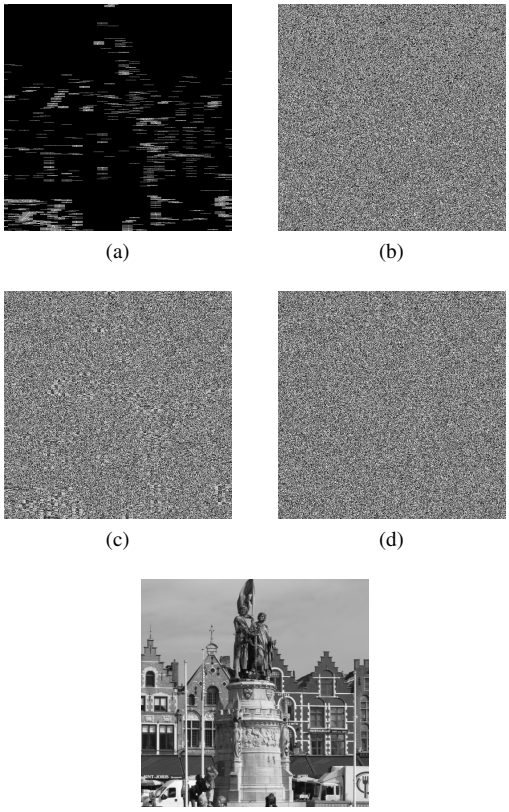


Fig. 10: Illustration of our EPE-HCRDH approach: a) Unmarked pixels' location (errors and flags), number of errors = 1225 (0.46%), b) Encrypted image I_e, c) Encrypted image I'_e with the highlighted prediction errors, d) Marked encrypted image I_{ew} with an embedding rate = 0.9220 bpp, e) Reconstructed image I, PSNR $\rightarrow +\infty$, SSIM = 1.

often on the edges and there are sometimes more than one error in the same block and, in these cases, the loss in terms of embedding capacity decreases. The histogram in Fig. 9.b illustrates the distribution of the prediction errors when the CPE-HCRDH approach is used and then shows the necessary modifications of the pixel values to avoid all the prediction errors and Fig. 9.c represents the pre-processed image based on Algorithm 1. We can observe that the pre-processed image is very similar to the original one, which is indicated by a PSNR equal to 46.87 dB and a SSIM of 0.9997. In Fig. 9.d, we can see the encrypted pre-processed image by using the encryption key. Fig. 10.b is the encrypted image in the EPE-HCRDH approach and Fig. 10.c corresponds to this image when the highlighted prediction errors are embedded. The content of the original image and the error location information are not visible anymore. Fig. 9.e and Fig. 10.d are the marked encrypted images, obtained in the final step of the encoding. For the CPE-HCRDH approach, each pixel of the encrypted pre-processed image is used to conceal one bit of the secret message (payload = 1 bpp). For the EPE-HCRDH approach, we mark the pixels according to the error location information

and even if the embedding rate is smaller, it is quite high with a payload equal to 0.9220 *bpp*. Fig. 9.f and Fig. 10.e present the reconstructed images after data extraction. Fig. 9.f is the same as the pre-processed image (PSNR = 46.87 dB) and, with the EPE-HCRDH approach, the original image is perfectly recovered, as shown by a PSNR which tends to $+\infty$ and a SSIM equal to 1 (Fig. 10.e). Note that the secret message is always extracted without error in both approaches.

We have applied our proposed approaches on $10,000$ 512×512 grey-level images of the BOWS-2 database [1], which has a strong statistical variability in the image content. Table I illustrates the results obtained for this database. In 6.3% of cases, when there is no prediction error (*i.e.* all the differences between original pixel values and their predictors are below or equal to 64), the two approaches are fully reversible. In this case, original images are recovered without any errors, as indicated by a PSNR which tends to $+\infty$ and a SSIM equal to 1. Moreover, it is possible to mark all the pixels of the images in order to have the highest possible payload of 1 *bpp*. In the other cases, for the CPE-HCRDH approach, we keep this payload, but the original image is not perfectly recovered, as we remove the prediction errors by changing some pixel values. Furthermore, for low contrast images, the reconstructed image quality is high. Indeed, on the average, the PSNR is equal to 57.4 dB and the SSIM is very close to 1 (0.9998); in 98.64% of cases, the PSNR is higher than 40 dB, which indicates a very good image quality. Concerning the EPE-HCRDH approach, which is totally reversible for all the images, the PSNR tends then to $+\infty$ and the SSIM is equal to 1. Even if all the pixels are not marked because there are some MSB prediction errors (in particular in the worst case), the payload remains high and on the average, we have a payload of 0.9681 *bpp*; in 92.19%, it is larger than 0.9 *bpp*.

TABLE I: Performance measurements of our two approaches on the BOWS-2 database (10,000 images) [1].

		Best case (6.3%)	Worst case	Average
CPE HCRDH approach	Percentage of MSB prediction errors in the original image	0%	4.9%	0.2%
	Payload (*bpp*)	1	1	1
	PSNR (dB)	$+\infty$	29.0	57.4
	SSIM	1	0.9872	0.9998
EPE HCRDH approach	Percentage of MSB prediction errors in the original image	0%	5.3%	0.2%
	Payload (*bpp*)	1	0.3805	0.9681
	PSNR (dB)	$+\infty$	$+\infty$	$+\infty$
	SSIM	1	1	1

In order to better visualize the distribution of the different image payloads, in Fig. 11, we randomly selected 500 images among the 10,000 tested images [1] and applied our EPE-HCRDH approach.

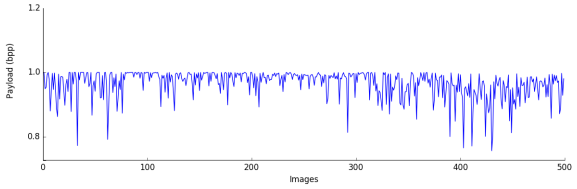


Fig. 11: Payload measurements, for the EPE-HCRDH approach, on a sample of 500 images from the BOWS-2 database [1].

B. Statistical analysis of our proposed method

We perform a statistical analysis of our two approaches, in order to verify that they achieve a high visual security level. We use different statistical metrics: horizontal and vertical correlation coefficients, Shannon entropy, χ^2 test, number of changing pixel rate (NPCR), unified averaged changed intensity (UACI) and PSNR between the original image and the encrypted or marked encrypted images.

a) Horizontal and vertical correlation coefficients:

$$corr_{p,p_N} = \frac{E\{|p - E(p)|\,|p_N - E(p_N)|\}}{\sqrt{V(p)V(p_N)}}, \tag{11}$$

where p_N refers to the considered neighbor of p (*i.e.* the left pixel when the horizontal correlation is computed and the top pixel when the vertical correlation is computed), $E(x)$ is the sample mean of x ($E(x) = \frac{1}{S}\sum_{k=1}^{S} x_k$), $V(x)$ is the sample variance of x ($V(x) = \frac{1}{S}\sum_{k=1}^{S} |x_k - E(x)|^2$) and S is the size of the considered sample.

b) Shannon entropy:

$$H(I) = -\sum_{l=0}^{255} P(\alpha_l)\log_2(P(\alpha_l)), \tag{12}$$

where I is a $m \times n$ image with 256 grey-levels α_l ($0 \leq l < 256$) and $P(\alpha_l)$ is the probability of α_l.

c) χ^2 test:

$$\chi^2 = 256 \cdot (m \times n)\sum_{l=0}^{255}\left(P(\alpha_l) - \frac{1}{256}\right)^2. \tag{13}$$

d) Number of changing pixel rate (NPCR):

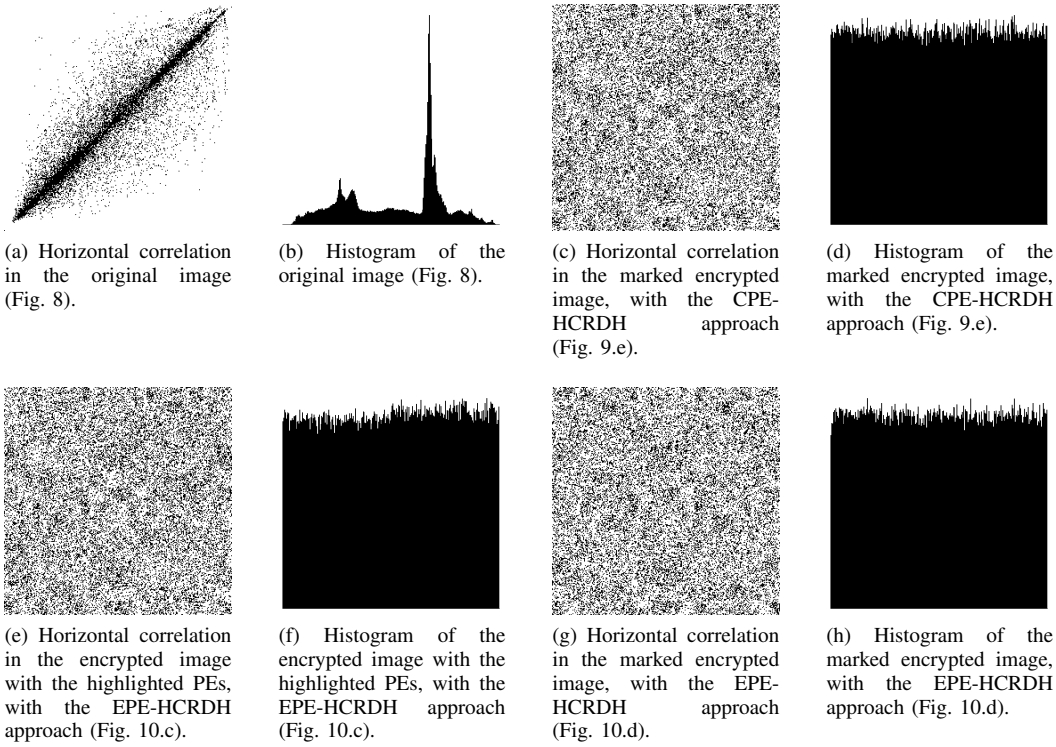
$$NPCR = \frac{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} d(i,j)}{m \times n} \times 100, \tag{14}$$

where $d(i,j)$ is defined as:

$$d(i,j) = \begin{cases} 1, \text{ if } \ p(i,j) = p'(i,j), \\ 0, \text{ otherwise.} \end{cases} \tag{15}$$

e) Unified averaged changed intensity (UACI):

$$UACI = \frac{100}{m \times n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\frac{|p(i,j) - p'(i,j)|}{255}. \tag{16}$$



(a) Horizontal correlation in the original image (Fig. 8).

(b) Histogram of the original image (Fig. 8).

(c) Horizontal correlation in the marked encrypted image, with the CPE-HCRDH approach (Fig. 9.e).

(d) Histogram of the marked encrypted image, with the CPE-HCRDH approach (Fig. 9.e).

(e) Horizontal correlation in the encrypted image with the highlighted PEs, with the EPE-HCRDH approach (Fig. 10.c).

(f) Histogram of the encrypted image with the highlighted PEs, with the EPE-HCRDH approach (Fig. 10.c).

(g) Horizontal correlation in the marked encrypted image, with the EPE-HCRDH approach (Fig. 10.d).

(h) Histogram of the marked encrypted image, with the EPE-HCRDH approach (Fig. 10.d).

Fig. 12: Statistical representations (correlation and histogram) for the original, encrypted and marked encrypted images obtained with our two approaches.

TABLE II: Quality evaluation of the obtained images with our two approaches.

Image	Horizontal correlation	Vertical correlation	Entropy (*bpp*)	χ^2 test (square root)	NPCR (%)	UACI (%)	PSNR (dB)
Original image (Fig. 8)	0.9388	0.9436	7.3227	668.628	/	/	/
Encrypted image, with the CPE-HCRDH approach (Fig. 9.d)	-0.0057	-0.0035	7.9994	14.8342	99.6143	30.1338	8.7081
Marked encrypted image, with the CPE-HCRDH approach (Fig. 9.e)	-0.0062	-0.0015	7.9994	15.1188	99.6082	30.1521	8.7069
Encrypted image, with the EPE-HCRDH approach (Fig. 10.b)	0.0071	-0.0017	7.9994	14.8806	99.6136	30.1344	8.7081
Encrypted image, with the highlighted PEs, with the EPE-HCRDH approach (Fig. 10.c)	0.0362	0.0147	7.9991	18.2620	99.6071	30.1238	8.6834
Marked encrypted image, with the EPE-HCRDH approach (Fig. 10.d)	-0.0016	0.0037	7.9994	14.8299	99.6059	30.1569	8.7039

f) Peak-signal-to-noise ratio (PSNR):

$$PSNR = 10 \cdot \log_{10} \frac{255^2}{\frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (p(i,j) - p'(i,j))^2}. \tag{17}$$

As we can see in Fig. 12, the correlation between horizontal pixels in the original image is very high (Fig. 12.a) while there is no correlation between adjacent pixels in the marked encrypted images (Fig. 12.c and Fig. 12.g) and the encrypted image with the highlighted PEs (Fig. 12.e). Moreover, the histogram of the marked encrypted image obtained with our CPE-HCRDH approach (Fig. 12.d) and the histograms of the encrypted image with the highlighted PEs (Fig. 12.f) and the marked encrypted image (Fig. 12.h) obtained with our EPE-HCRDH approach are uniformly distributed in comparison with the original image (Fig. 12.b). It is not possible to exploit them to obtain information about the original content of the image. Indeed, our image encryption scheme allows us to make pseudo-random dependence of the statistical properties between the encrypted images and the original image and this characteristic is conserved after the insertion of the secret message or of the error location information, as presented in the Table II. In the original image (Fig. 8), there is a high correlation between adjacent pixels, as indicated by values close to one (0.9388 and 0.9436). In the encrypted or marked encrypted images, these values are low and close to zero, which means that there is no correlation between the pixel values. Moreover, we can see that the value of the entropy is very high for the encrypted or marked encrypted images ($\sim$ 7.9995 bpp) and close to the maximal value, which

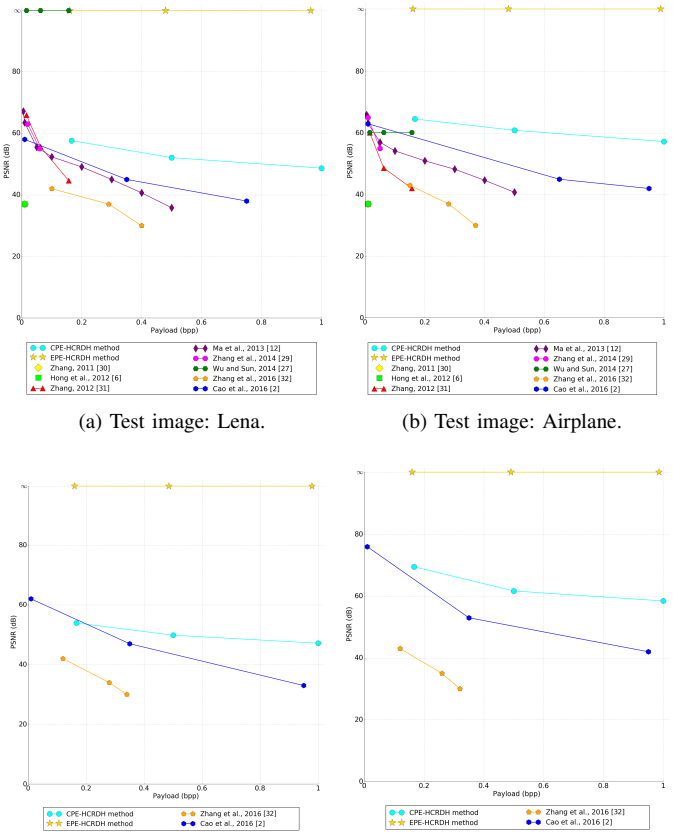
indicates that the grey-level distribution tends to be uniform. In comparison, the value measured in the original image is smaller (7.3227 *bpp*). Then, if we consider the values obtained with the χ^2 test, we observe that it is very high for the original image (668.628) while they are much lower in the encrypted or marked encrypted images (~ 15). This means that data in our encrypted or marked encrypted images are disordered, non-uniform and uncorrelated: our scheme is resistant to statistical attacks. We also measure NPCR, UACI and PSNR between the original image and the encrypted or marked encrypted images. NPCR values are very high and close to the maximal value ($\sim 99.6\%$), UACI rates are close to 30.15% and PSNR values are very low ($\sim 8.7\ dB$), which indicates that the original and the encrypted or marked encrypted images are, as expected, very different.

C. Comparisons with related methods and discussion

We made several comparisons, in terms of embedding rate and reconstructed image quality, between our two proposed approaches and eight state-of-the-art methods: very recent methods proposed by Zhang *et al.* [32] and Cao *et al.* [2] (Fig. 13.a–d) and other methods proposed by Zhang [30], Hong *et al.* [6], Zhang [31], Ma *et al.* [12], Zhang *et al.* [29] and Wu and Sun [27] (Fig. 13.a–b).

To do this, we used the well known images of Lena, Airplane, Man and Crowd. First of all, we can see that our approaches allow us to have a larger payload than the others in all cases. In fact, the maximal payload value for the state-of-the-art methods, obtained by Cao *et al.* is 0.95 *bpp*. With our CPE-HCRDH approach, we can embed 1 *bpp* and with the EPE-HCRDH one, we achieve results very close to this high value. In fact, since we do not need to use overhead for our two approaches, in the EPE-HCRDH approach, we have to decrease the payload and the cost is 0.0359 bpp for Lena, 0.0111 bpp for Airplane, 0.0212 bpp for Man and 0.0145 bpp for Crowd. When we examine the reconstructed image quality, our EPE-HCRDH approach is the only scheme which allows us to perfectly reconstruct the original image with the only knowledge of the encryption key and without the need of the data hiding key (PSNR $\rightarrow +\infty$). None of the other methods obtain such results in any situation. Only the Lena image is exactly the same as the original one by using Wu and Sun's method. Moreover, for the other images, we can see that we outperform all the other methods with our proposed ones, even when we hide more information in the image. This is especially true when we choose an embedding capacity comparable to the other methods. When we mark regularly one pixel every six (0.1667 *bpp*) or one every two (0.5 *bpp*), our results, in terms of recovered image quality, are better than those obtained by the other state-of-the-art methods, even the most recent.

In conclusion, in addition to being error-free during data extraction, our method, whatever the adopted approach, allows us to have a very good trade-off between the embedding rate and the recovered image quality after data extraction, by using only the encryption key. From the security point of



(a) Test image: Lena. (b) Test image: Airplane.

(c) Test image: Man. (d) Test image: Crowd.

Fig. 13: Performance comparisons between our proposed approaches and similar state-of-the-art methods for four test images.

view, the statistical analysis shows that there is no information about the content of the original image in the encrypted or marked encrypted version. Moreover, if a small part of the secret message is modified by an attacker, as this message is encrypted, it cannot be decrypted and thus exploited for authentication. Moreover, in the EPE-HCRDH approach, if the message is modified or removed, then the clear image could not be reconstructed. Note that even if in the proposed method the hidden message can be used for several applications for authentication and data enrichment, it can been also used as an alternative of ECC (Error Correcting Code) for integrity check for example.

V. Conclusion

In this work, we proposed an efficient method of reversible data hiding in encrypted images based on MSB prediction with a very high embedding capacity, which outperforms the last state-of-the-art methods. From our knowledge this is one of the first methods which proposes to use MSB instead of LSB for a RDHEI. Due to the fact that MSB prediction is easier than LSB prediction in original domain and because image quality deterioration is not a problem in the encrypted domain, we are then able to have a very high capacity. By analyzing the original content of the image, the prediction errors are

highlighted and an error location binary map is built. In the CPE-HCRDH approach, the original image is slightly modified in order to avoid all the prediction errors. After that, by substituting all MSB in the image, it is possible to hide one bit per pixel. In addition to this maximal payload equal to 1 bpp, the reconstructed image quality is high (SSIM close to 1, PSNR $=$ 57.4 dB on the average). In the EPE-HCRDH approach, information about the location of the prediction errors is stored in the encrypted image according to the error location binary map. Note that using overhead such an additional map is not necessary for this proposed approach. Rather than that, we used some MSB values instead of embedding bits from the hidden message. Thus, by substituting most of the MSB values in the encrypted image, a large message can be hidden (payload close to 1 bpp) and during the decoding phase, the original image can be recovered losslessly (PSNR $\rightarrow +\infty$). In addition, we have seen that the proposed scheme provides a good security level and can be used to preserve the original image content confidentiality, while offering at the same time authenticity or integrity check.

In future work, we are interested in hiding more than one bit per pixel. In fact, we think that it is possible to use, for example, the second MSB of each pixel to enlarge the amount of embedded information. Further research directions include testing other error predictors in order to reduce the number of prediction errors and, in this same manner, improve the reconstructed image quality (for CPE-HCRDH) or the payload (for EPE-HCRDH). Indeed, with the CPE-HCRDH approach, the more the payload is increased, the more the number of prediction errors is important and so therefore, more the recovered image is altered. Moreover, with the EPE-HCRDH approach, we are also involved in the search for a new prediction error highlighting mechanism which will allow us to improve the embedding capacity.

REFERENCES

[1] P. Bas and T. Furon, "Image database of BOWS-2," http://bows2.ec-lille.fr/.

[2] 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.

[3] 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.

[4] Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni, "Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing," *EURASIP Journal on Information Security*, vol. 2007, p. 17, 2007.

[5] 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*, vol. 21, no. 8, pp. 1061–1070, 2011.

[6] 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.

[7] 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.

[8] V. Itier and W. Puech, "How to recompress a JPEG crypto-compressed image?" in *Electronic Imaging, Media Watermarking, Security, and Forensics 2017*, vol. 2017, no. 7, 2017.

[9] 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.

[10] 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.

[11] W. Lu, A. Swaminathan, A. L. Varna, and M. Wu, "Enabling search over encrypted multimedia databases," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2009, pp. 725 418–725 418.

[12] 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*, vol. 8, no. 3, pp. 553–562, 2013.

[13] K. Minemura, Z. Moayed, K. Wong, X. Qi, and K. Tanaka, "JPEG image scrambling without expansion in bitstream size," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*, 2012, pp. 261–264.

[14] M. Naor and A. Shamir, "Visual cryptography," in *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer, 1994, pp. 1–12.

[15] 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.

[16] 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.

[17] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images," in *Electronic Imaging 2008*. International Society for Optics and Photonics, 2008, pp. 68 191E–68 191E.

[18] W. Puech and J. M. Rodrigues, "Crypto-compression of medical images by selective encryption of DCT," in *Signal Processing Conference (EUSIPCO), 2005 13th European*, 2005, pp. 1–4.

[19] P. Puteaux, D. Trinel, and W. Puech, "High-capacity data hiding in encrypted images using MSB prediction," in *Image Processing Theory Tools and Applications (IPTA), 2016 6th IEEE International Conference on*, 2016, pp. 1–6.

[20] 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.

[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.

[22] 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.

[23] J. Tian, "Reversible watermarking by difference expansion," in *Proceedings of Workshop on Multimedia and Security*, vol. 19, 2002.

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

[25] W. Trappe and L. C. Washington, *Introduction to cryptography with coding theory*. Pearson Education India, 2006.

[26] M. Van Droogenbroeck and R. Benedett, "Techniques for a selective encryption of uncompressed and compressed images," in *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS) 2002, Ghent, Belgium*, 2002, pp. 90–97.

[27] X. Wu and W. Sun, "High-capacity reversible data hiding in encrypted images by prediction error," *Signal Processing*, vol. 104, pp. 387–400, 2014.

[28] D. Xu and R. Wang, "Separable and error-free reversible data hiding in encrypted images," *Signal Processing*, vol. 123, pp. 9–21, 2016.

[29] W. Zhang, K. Ma, and N. Yu, "Reversibility improved data hiding in encrypted images," *Signal Processing*, vol. 94, pp. 118–127, 2014.

[30] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 4, pp. 255–258, 2011.

[31] ——, "Separable reversible data hiding in encrypted image," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 826–832, 2012.

[32] 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*, vol. 26, no. 9, pp. 1622–1631, 2016.

[33] J. Zhou, W. Sun, L. Dong, X. Liu, O. C. Au, and Y. Y. Tang, "Secure reversible image data hiding over encrypted domain via key modulation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 3, pp. 441–452, 2016.

Pauline Puteaux received the M.S. degree in Computer Science and Applied Mathematics, with specialization in Cybersecurity, from the University of Grenoble, France, in 2017. She is currently pursuing the Ph.D. degree with the Laboratory of Informatics, Robotics and Microelectronics of Montpellier, France. Her work has focused on multimedia security, and in particular, image analysis and processing in the encrypted domain.

William Puech received the diploma of Electrical Engineering from the University of Montpellier, France (1991) and the Ph.D. Degree in Signal-Image-Speech from the Polytechnic National Institute of Grenoble, France (1997) with research activities in image processing and computer vision. He served as a Visiting Research Associate to the University of Thessaloniki, Greece. From 1997 to 2008, he has been an Associate Professor at the University of Montpellier, France. Since 2009, he is full Professor in image processing at the University of Montpellier, France. His current interests are in the areas of image forensics and security for safe transfer, storage and visualization by combining data hiding, compression and cryptography. He is the head of the ICAR team (Image & Interaction) in the LIRMM, has published more than 40 journal papers and 120 conference papers and is associate editor for 5 journals (JASP, SPIC, SP, JVCIR and IEEE TDSC) in the areas of image forensics and security. Since 2017 is the general chair of the IEEE Signal Processing French Chapter and since 2008 he is member of the IEEE Information Forensics and Security TC.