Reversible data hiding in encrypted images based on adaptive local entropy analysis

Pauline Puteaux, William Puech

Reversible Data Hiding in Encrypted Images based on Adaptive Local Entropy Analysis

Pauline Puteaux and William Puech
LIRMM Laboratory, UMR 5506
CNRS, Univ. Montpellier
Montpellier, France
e-mail: {pauline.puteaux, william.puech}@lirmm.fr

***Abstract*— With the development of cloud computing, the growth in information technology has led to serious security issues. For this reason, a lot of multimedia files are stored in encrypted forms. Methods of reversible data hiding in encrypted images (RDHEI) have been designed to provide authentication and integrity in the encrypted domain. The original image is firstly encrypted to ensure confidentiality, by making the content unreadable. A secret message is then embedded in the encrypted image, without the need of the encryption key or any access to the clear content. The challenge lies in finding the best trade-off between embedding capacity and quality of the reconstructed image. In 2008, Puech *et al.* suggested using the AES algorithm to encrypt an original image and to embed one bit in each block of 16 pixels (payload = 0.0625 bpp) [12]. During the decryption phase, the original image is reconstructed by measuring the standard deviation into each block. In this paper, we propose an improvement to this method, by performing an adaptive local entropy measurement. We can achieve a larger payload without altering the recovered image quality. Our obtained results are very good and better than most of the modern state-of-the-art methods, whilst offering an improved security level with the use of the AES algorithm, defined as the encryption standard by the NIST.**

***Keywords*— Multimedia security, reversible data hiding, image encryption, image recovery, statistical analysis.**

I. Introduction

Reversible data hiding (RDH) is particularly suitable for authentication and data enrichment. In a RDH scheme, a secret message is concealed in a cover image, without altering its global meaning. After the message extraction, it is also possible to reconstruct losslessly the original image. In the state of the art, three general strategies have been proposed: lossless compression [5], [3], difference expansion (DE) [17] and histogram shifting (HS) [11], [6]. The first RDH methods used lossless compression. The idea is to losslessly compress a part of the cover image and to reversibly embed a secret message in the released part. Based on the original scheme by Fridrich *et al.* [5], Celik *et al.* proposed to compress the least significant bit (LSB) planes [3] and to use unaltered bit-planes as side information. However, the efficiency of this kind of method is limited. Indeed, since there is few redundancy in the LSB planes, it is not possible to have a high embedding capacity – also called payload. Tian suggested to use the Haar transform to design a technique based on difference expansion [17]. He suggested computing the difference between a pixel and its neighbors to determine where information had to be concealed. In their paper [24], Zhang and Wang proposed to exploit the set of modification direction for a pixel (EMD). In techniques based on histogram modification, the histogram of the image is built according to grey-level values [11], [16] or by using statistical data [6]. Today, methods tend to use a combination of these different schemes. In order to exploit the correlation between a pixel and its adjacent neighbors, a prediction error (PE) analysis can be performed [8]. These methods achieve a better performance in comparison to previous ones.

Some applications, like in the medical or military world, require a high security level. By randomizing an original content of the original image, encryption provides visual confidentiality. Cryptosystems can be divided into two groups, depending if a stream cipher or a block cipher is used [18]. They can be symmetric – when the same secret key is used during the encryption and decryption (AES, DES) or asymmetric – when they involve both public and private keys. Moreover, encryption can be selective, when only certain details are hidden in the encrypted image [13], or full when the global meaning of the image is kept secret [10].

Encrypted image analysis and processing have received a lot of attention within the scientific and business communities in the last few years, in particular due to the development of cloud computing [4]. Methods of reversible data hiding in encrypted images (RDHEI) have been designed to combine image encryption and RDH. In a RDHEI scheme, image encryption is executed in the first place and it is possible to insert a secret message in an encrypted image without knowing the original content of the image or secret key. Many methods have been recently designed in order to achieve the best trade-off between payload and quality of the reconstructed image. The space to embed the message may be vacated after [22], [19] or before [9], [20] the encryption phase. Furthermore, encryption and data hiding can be joint, when data extraction and image reconstruction are completed at the same time, or separately. Recently, Cao *et al.* proposed a sparse coding technique [2]. They managed to vacate a large space to hide bits of the secret message by exploiting the local correlation between pixels. Qian and Zhang suggested compressing some

978-1-5386-1842-4/17/$31.00

bits of the MSB planes to release room for additional data [14]. Zhang *et al.* designed a system based on public key cryptography with probabilistic and homomorphic properties, allowing to recover losslessly the original image [23]. Other methods managed to remove the embedded message and to recover the original image by completing a statistical analysis of each block of the encrypted image to determine if it is decrypted or still encrypted. Zhang proposed to exploit the spatial correlation in natural images. By measuring the fluctuation of each block, the embedded data is fully extracted and the original image is perfectly reconstructed [21]. Hong *et al.* improved this technique by considering the local complexity for each pixel during the encryption step [7]. Puech *et al.* suggested analyzing the local standard deviation into blocks of the marked encrypted image, in order to recover their original version without any errors during the decryption step [12]. Although this method is effective, the embedding capacity is quite low, because only one bit is concealed in each block (payload = 0.0625 *bpp*).

In this paper, we propose an improvement to the method by Puech *et al.* [12]. As in the previous method, the original image is firstly encrypted with AES cryptosystem in ECB mode to ensure image content confidentiality. However, we are able to embed a much larger amount of information in the encrypted image, without degrading the reconstructed image quality. In fact, we suggest to adapt the Shannon entropy measurement in order to be able to perform a significant local analysis. During the reconstruction phase, we use this statistical metric instead of standard deviation to losslessly recover each block of 4×4 pixels from the original image.

The rest of this paper is organized as follows. Section II describes our RDHEI method based on local entropy analysis. Experimental results are provided in Section III. Finally, the conclusion is drawn and future work is proposed in Section IV.

II. Proposed method

In this section, we first introduce the concept of adaptive local entropy. After that, we present our joint reversible data hiding method in encrypted images based on this statistical measurement. The encoding phase includes two classic steps: image encryption with a secret key K_e, and secret message embedding using a data hiding key K_w. The overview of this encoding method is presented in Fig. 1. For the decoding scheme, if the recipient has only the data hiding key K_w, they can extract the secret message. However, to reconstruct an original image, they have to know both the encryption key K_e and the watermarking key K_w, because image decryption and message extraction are executed at the same time, as shown in Fig. 2. During this phase, they perform a local entropy analysis to recover original image blocks.

A. Adaptive local entropy analysis

The notion of entropy was introduced by Shannon in 1948 [15]. This statistical metric measures the expected value of information contained in a message. Despite its popularity in image processing and information theory, none of the previous methods of reversible data hiding in encrypted domain use Shannon entropy during the image reconstruction phase. In fact, due to the sparsity of the sample when a small block size is considered, direct use of entropy is difficult.

Let B be a block of k pixels in an image encoded on l grey-levels α_j, with the associated probability $p(\alpha_j)$. Local entropy (*i.e.* inside block B) is increased by the minimal value between its block size k and the number of grey-levels l:

$$\begin{aligned} H_{(k,l)}(B) &= -\sum_{i=0}^{l-1} p(\alpha_j)\log_2(p(\alpha_j)), \\ &\leq -\min(k,l)\cdot\frac{1}{\min(k,l)}\log_2\left(\frac{1}{\min(k,l)}\right), \\ &\leq \log_2(\min(k,l))\ \textit{bpp}. \end{aligned} \tag{1}$$

If the block size is larger than the number of grey-levels, like in a full image, maximal entropy value corresponds to uniform distribution, *i.e.* equiprobability between all the grey-levels. Otherwise, if there are more grey-levels in an image than pixels in a block, the maximal entropy value is reached when each pixel value is different. In this case, the pixel sample is sparse, because some grey-level values are not present in block B. For this reason, entropy measurement may be erroneous: a block relatively homogeneous may seem pseudo-randomly generated, with regards to its entropy value.

To solve this problem, we propose to adapt, by quantizing the image histogram for the entropy measurement in order to decrease the value of l. Actually, the idea is to find the best trade-off between block size k and the number of grey-levels l in an image.

B. Image encryption

In the proposed method, the original image I is first encrypted, block by block, to obtain the encrypted image I_e. Each original block X_i of 128 bits, which corresponds to $k = 16$ grey-level pixels, is encrypted using the AES encryption algorithm $E_{K_e}(\cdot)$ in ECB mode with the secret key K_e (128 bits) producing the encrypted block Y_i:

$$Y_i = E_{K_e}(X_i), \tag{2}$$

As the AES encryption function is efficient, the local entropy value measured into each encrypted block Y_i must be close to the maximal entropy value and larger than the entropy value measured in the clear domain:

$$H_{(k,l)}(X_i) \leq H_{(k,l)}(Y_i). \tag{3}$$

C. Data embedding

During the data hiding step, in each encrypted block, we replace p bits of Y_i with bits of the secret message to obtain

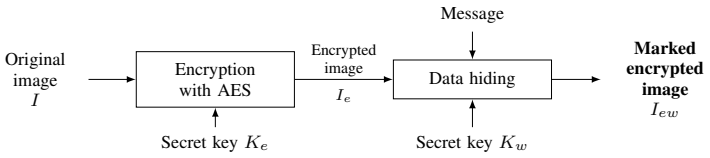


Fig. 1. Overview of the encoding method.

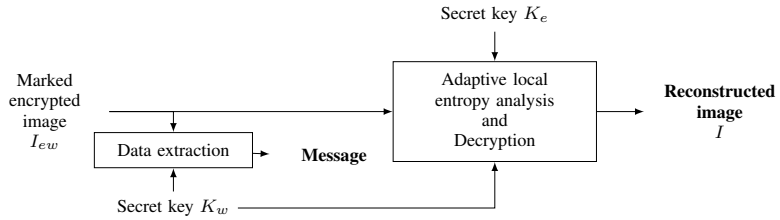


Fig. 2. Overview of the decoding method.

marked encrypted blocks Y_{w_i} of the marked encrypted image I_{ew}:

$$Y_{w_i} = DH_{K_w}(Y_i), \tag{4}$$

where $DH_{K_w}(\cdot)$ is the data hiding function involving the secret key K_w, different to the secret key K_e used during the encryption phase.

K_w is used as a seed of a pseudo-random generator to get a list of positions pos in the encrypted block Y_i to insert bits of the to-be-embedded message, depending on a given embedding rate. Without the key K_w, it is impossible to guess where the bits of the secret message are hidden, especially as the set of positions is different for each block.

D. Data extraction and image recovery

During the reconstruction phase, there are two possible schemes. If the recipient only has the data hiding key, they can extract the secret message easily. With the help of K_w, they generate the pseudo-random sequence corresponding to the position of the embedded bits and read them to recover the hidden message. If they have both keys, they can obtain the secret message and reconstruct the original image. After message extraction, each encrypted block Y_{w_i} is still marked. If they directly decrypt the marked encrypted image, they obtain an image which can be very different to the original one, especially when the embedding rate is high. For this reason, they have to perform the proposed adaptive local entropy analysis in each block to jointly remove the message and decrypt the encrypted content. The detailed method is presented in Algorithm 1. With the help of the data hiding key, the location pos of all the potential erroneous bits is known. For each of them, it is necessary to test two possible values (0 or 1): if p bits in the block were altered, there are also 2^p possible configurations. In other words, the higher the payload, the more the algorithm complexity is important. For each combination c, the recipient has to decrypt the current block $Y_{w_{i(pos,c)}}$ with encryption key K_e and to measure local entropy value by adapting the number of grey-levels, as explained previously in Section II.A. At the end of the process, the decrypted block with smaller entropy value is considered as the expected original one.

Algorithm 1: Image reconstruction algorithm.

Data: Marked encrypted image I_{ew} of size $m \times n$ pixels,
Block size k, number of grey-levels l,
Secret keys K_e and K_w,
Number p of embedded pixels into each block
Result: Reconstructed image I of size $m \times n$ pixels
foreach $Y_{w_i} \in I_{ew}$, Y_{w_i} of size $k = 16$ pixels **do**
  $H_{min} \leftarrow \log_2(\min(k, l))$;
  /* Minimal entropy initialization */
  $X_i \leftarrow D_{K_e}(Y_{w_i})$;
  /* Clear block X_i initialization */
  /* D_{K_e} is the decryption function */
  $pos \leftarrow$ potential erroneous bits location, using K_w;
  for $c \leftarrow 0$ **to** $2^p - 1$ **do**
    /* Search of the combination which minimizes the entropy value */
    if $H_{min} < H_{(k,l)}(D_{K_e}(Y_{w_{i(pos,c)}}))$ **then**
      $H_{min} \leftarrow H_{(k,l)}(D_{K_e}(Y_{w_{i(pos,c)}}))$;
      $X_i \leftarrow D_{K_e}(Y_{w_{i(pos,c)}})$;

Note that the number of error cases (*i.e.* blocks which are not deciphered well because there exists at least one combination with a smaller entropy than the true one, in the clear domain) is quite high using zero-order entropy, by considering a reduced number of grey-levels during the entropy measurement. In order to obtain better results, it is possible to perform entropy measurement on the distance map of an image. The idea is to exploit the correlation between neighboring pixels in the clear domain. In fact, in the clear domain, values of adjacent pixels are very close, which is not the case in the encrypted domain. Even if there is a contour in a block in clear, this frontier delimits two relatively homogeneous regions. To compute the distance map D from an image X, we have to compute the absolute difference between a pixel x and its predictor $pred(x)$, computed according to the

values of its neighbors:

$$\forall d \in D, d = d(x, pred(x)) = |x - pred(x)|, \quad (5)$$

Note that the predictor $pred(x)$ can be one of the neighbors of x (left, top or left-top pixel) or a combination of these values (for example, the average value), in order to take into account the different directions.

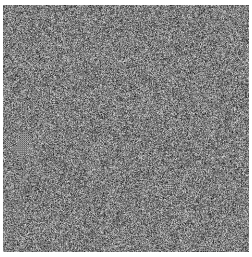
As with zero-order entropy, during the reconstruction of each block of an image, we try all possible configurations and search the minimal entropy of the distance map value.

III. Experimental results

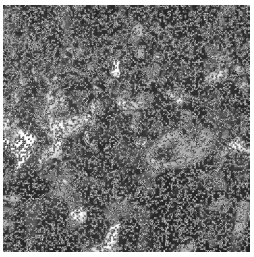
We applied our method on the *Crowd* image of 512×512 pixels, illustrated in Fig. 3. Fig. 4 and Fig. 5 show the obtained results with a payload of 0.0625 *bpp* – when one bit of the message by block is embedded – and 0.5 *bpp* – when eight bits by block are embedded – respectively. The original image is firstly encrypted using the AES algorithm in ECB mode, with encryption key K_e. After that, the secret message is concealed in the encrypted image, according to data hiding key K_w. Note that the higher the payload, the more it is possible to embed a larger amount of information. For the local entropy measurements, we use zero-order entropy and distance map entropy, with 256 grey-levels and 8 grey-levels. Indeed, we also performed an analysis on $1,000$ images from the BOWS-2 database [1] in order to find the optimal number of grey-levels to consider during the entropy calculation, according to the block size. Results have shown that, for blocks of 4×4 pixels, best results are achieved with 8 grey-levels. On average, there is only 0.1066% of possible errors with zero-order entropy and 0.0379%, with distance map entropy.

Fig. 3. Original *Crowd* image.

Fig. 4.a and Fig. 5.a present marked encrypted images. Note that, even with a large payload, we cannot distinguish any information from the original content of the image. In fact, marked encrypted images are pseudo-randomly generated: entropy of these images is very high and close to the maximal value of $\log_2(\min(512 \times 512,\ 256))$ *bpp*. As shown in Fig. 4.b and Fig. 5.b, if we directly decrypt the marked encrypted images without correction, there are many erroneous blocks in the reconstructed images. In Fig. 4.b, as only one bit is concealed into each block, approximately half of the blocks are perfectly reconstructed (PSNR = 11.28 dB, SSIM = 0.124). On the other hand, in Fig. 5.b, the reconstructed image looks like the marked encrypted one: only one block out of 256 may be correctly recovered (PSNR = 8.23 dB, SSIM = 0.003). Since the number of possible combinations by block during the reconstruction phase is equal to $2^8 = 256$ bits. Fig. 4.c-d and Fig. 5.c-d are the reconstructed images when selecting the combination which allows us to have the smaller entropy value among all the possible combinations, in each block of $4 \times 4 = 16$ pixels. If we do not perform a quantization from the image, the number of grey-levels l, equals to 256, is too high in comparison to its block size. In this case, grey-level distribution is sparse. Due to this problem, local entropy measurement can be erroneous. Indeed, as illustrated in Fig. 4.c and Fig. 5.c, many blocks are badly reconstructed. For a small payload of 0.0625 *bpp*, PSNR value between the original image and the reconstructed one is equal to 22.76 dB and SSIM is equal to 0.885. For a large payload of 0.5 *bpp*, PSNR is equal to 14.88 dB and SSIM is equal to 0.499.



(a) Marked encrypted image.



(b) Directly reconstructed image (PSNR = 11.28 dB, SSIM = 0.124).

(c) Reconstructed image using zero-order entropy with $l = 256$ grey-levels (PSNR = 22.76 dB, SSIM = 0.885).

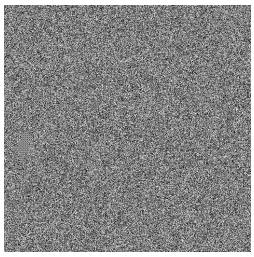
(d) Reconstructed image using zero-order entropy with $l = 8$ grey-levels (PSNR = 38.57 dB, SSIM = 0.885).

(e) Reconstructed image using distance map entropy with $l = 256$ grey-levels (PSNR = 31.85 dB, SSIM = 0.988).

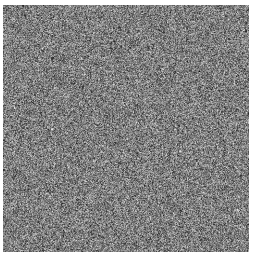
(f) Reconstructed image using distance map entropy with $l = 8$ grey-levels (PSNR $\rightarrow +\infty$ dB, SSIM = 1).

Fig. 4. Application of our new RDHEI method based on local entropy measurement with a payload of 0.0625 *bpp*, with two values of grey-levels and two types of entropy (zero-order and distance map).

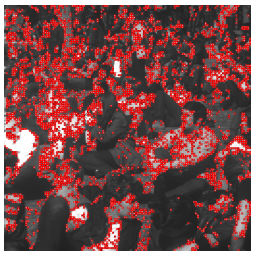
Conversely, in Fig. 4.d and Fig. 5.d, if we consider a number of grey-levels smaller than the block size during entropy measurement ($l = 8$), almost all blocks are correctly decrypted, but some errors still remain (PSNR = 38.58 dB, SSIM = 0.998 for a payload of 0.0625 bpp, and PSNR = 23.67 dB, SSIM = 0.927 for a payload of 0.5 bpp). As displayed in Fig. 4.e-f and Fig. 5.e-f, the use of distance map entropy allows us to obtain better results. As we exploit the correlation between neighboring pixels in the clear domain, the number of erroneous blocks is significantly reduced. Best performances are achieved by using distance map entropy and quantization (Fig. 4.f and Fig. 5.f). With a small payload of 0.0625 bpp, the original image is perfectly reconstructed (PSNR $\rightarrow +\infty$ dB, SSIM = 1). With a high capacity, even if the reconstructed image is not exactly the same as the original one, its quality is very high, which is indicated by a PSNR equal to 44.41 dB and a SSIM equal to 0.999.



(a) Marked encrypted image.



(b) Directly reconstructed image (PSNR = 8.23 dB, SSIM = 0.003).



(c) Reconstructed image using zero-order entropy with $l = 256$ grey-levels (PSNR = 14.88 dB, SSIM = 0.499).

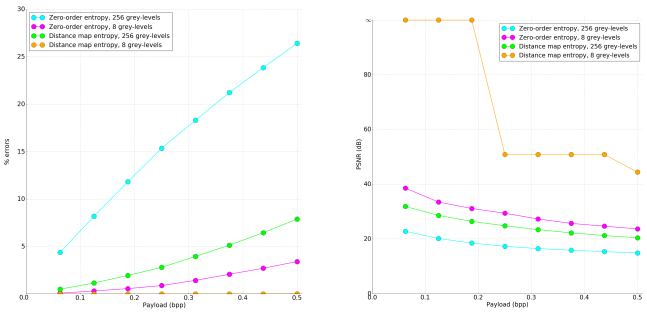
(d) Reconstructed image using zero-order entropy with $l = 8$ grey-levels (PSNR = 23.67 dB, SSIM = 0.927).

(e) Reconstructed image using distance map entropy with $l = 256$ grey-levels (PSNR = 20.41 dB, SSIM = 0.825).

(f) Reconstructed image using distance map entropy with $l = 8$ grey-levels (PSNR = 44.41 dB, SSIM = 0.999).

Fig. 5. Application of our new RDHEI method based on local entropy measurement with a payload of 0.5 bpp, with two numbers of grey-levels and two types of entropy (zero-order and distance map).

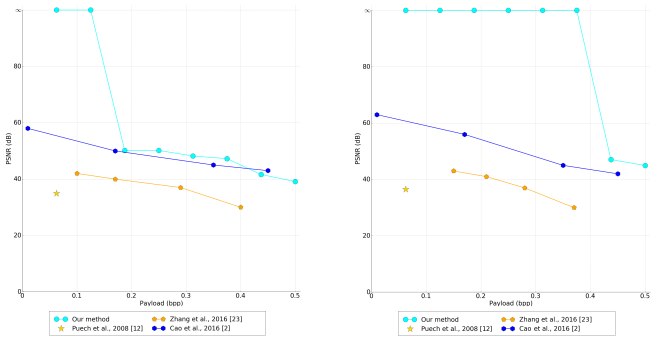
In order to better visualize the influence of the type of entropy and necessity to perform a quantization for the measurement, we plot the percentage of errors (Fig. 6.a) and the PSNR value (Fig. 6.b) between the original *Crowd* image (Fig. 3) and the reconstructed image, by using zero-order entropy / distance map entropy and without quantization / with 8 grey-levels. We consider different payloads (from 0.0625 bpp to 0.5 bpp) during this experimentation. As shown previously, we can see that the percentage of errors using distance map entropy and 8 grey-levels is equal to zero (from 0.0625 bpp to 0.1875 bpp) or close to this value. For small payloads, the reconstructed image is exactly the same as the original one (PSNR $\rightarrow +\infty$ dB, SSIM = 1). In other cases, PSNR and SSIM values are high (PSNR larger than 40 dB and SSIM close to 1), which indicates high reconstructed image quality. Note that these very good results are not possible with zero-order entropy and without quantization, regardless of the embedding capacity. In particular, if we use zero-order entropy without quantization, more than a quarter of the blocks are badly reconstructed and PSNR is smaller than 20 dB.



(a) Percentage of errors. (b) PSNR value.

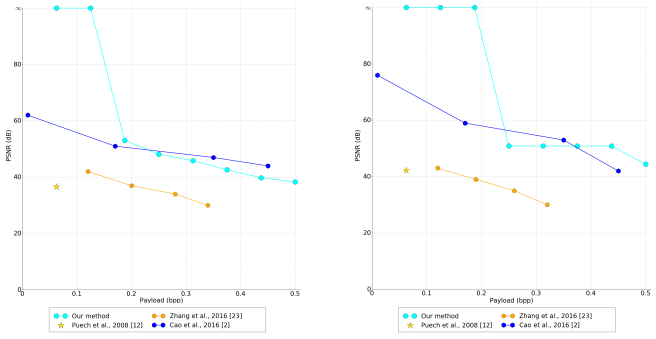
Fig. 6. Reconstructed *Crowd* image quality as a function of the payload, the type of entropy (zero-order and distance map) and the considered number of grey-levels ($l = 256$ and $l = 8$).

We also made some comparisons between our proposed method using the best settings (distance map entropy and 8 grey-levels), and the method of Puech *et al.* [12] and two recent approaches: Cao *et al.*'s [2] and Zhang *et al.*'s [23]. We used well-known images like *Lena*, *Airplane*, *Man* and *Crowd*. Firstly, we can observe that in all cases, our method succeeds in obtaining better results than the original method of Puech *et al.*. Indeed, our method is fully reversible for small payloads, as indicated by a PSNR value which tends to infinity, while the method of Puech *et al.* does not allow to have a PSNR value greater than approximately 40 dB. We can also see that it achieves a better performance than the method of Zhang *et al.* in terms of reconstructed image quality, even with large payloads. Moreover, our results are better than those obtained by Cao *et al.* for small payloads and comparable for larger ones.



(a) Test image: Lena.

(b) Test image: Airplane.



(c) Test image: Man.

(d) Test image: Crowd.

Fig. 7. Performance comparisons between our proposed approach and similar state-of-the-art methods for four test images.

IV. Conclusion

In this paper, we proposed a new reversible data hiding method in encrypted images based on an adaptive local Shannon entropy analysis. During the reconstruction phase, we performed an entropy analysis in each block of 4×4 pixels. Indeed, entropy value of a block of pixels in a clear image is smaller than the value in the encrypted domain. For this reason, it is possible to recover the original pixel values by comparing entropy values for different possible pixel configurations. As we have small block sizes, pixel sample is sparse and we have to quantize the number of grey-levels for entropy measurement in order to make it significant. Thanks to this practical tip, we achieved a very good trade-off between embedding capacity and reconstructed image quality, which enables us to considerably improve on the previous method [12]. Moreover, we obtain better results than most of the state-of-the-art methods while offering a higher security level. Future work on this method includes improving image recovery. Indeed, in some cases, even if the global meaning is conserved, we fail to perfectly reconstruct all the pixel blocks of the original image. For this reason, we are interested in using joint entropy between image pixel values and distance map information in order to reduce the number of error cases. However, in our method, the original image characteristics can be analyzed before encryption in order to use a payload value allowing perfect reversibility.

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] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-lsb data embedding," *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253–266, 2005.

[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] J. J. Fridrich, M. Goljan, and R. Du, "Invertible authentication." *Security and Watermarking of Multimedia Contents*, vol. 3, pp. 197–208, 2001.

[6] 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.

[7] 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.

[8] B. Ma and Y. Q. Shi, "A reversible data hiding scheme based on code division multiplexing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 9, pp. 1914–1927, 2016.

[9] 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.

[10] 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*. IEEE, 2012, pp. 261–264.

[11] 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.

[12] 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.

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

[14] 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.

[15] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[16] 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.

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

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

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

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

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

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

[23] 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.

[24] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Communications Letters*, vol. 10, no. 11, 2006.