Noisy Encrypted Image Correction based on Shannon Entropy Measurement in Pixel Blocks of Very Small Size

Pauline Puteaux, William Puech

Noisy Encrypted Image Correction based on Shannon Entropy Measurement in Pixel Blocks of Very Small Size

Pauline Puteaux and William Puech
LIRMM, UMR 5596 – CNRS, Univ. Montpellier, Montpellier, France
{pauline.puteaux, william.puech}@lirmm.fr

***Abstract*—Many techniques have been presented to protect image content confidentiality. The owner of an image encrypts it using a key and transmits the encrypted image across a network. If the recipient is authorized to access the original content of the image, he can reconstruct it losslessly. However, if during the transmission the encrypted image is noised, some parts of the image can not be deciphered. In order to localize and correct these errors, we propose an approach based on the local Shannon entropy measurement. We first analyze this measure as a function of the block-size. We provide then a full description of our blind error localization and removal process. Experimental results show that the proposed approach, based on local entropy, can be used in practice to correct noisy encrypted images, even with blocks of very small size.**

***Index Terms*—Image encryption, image denoising, statistical analysis, multimedia security.**

I. Introduction

The aim of encryption methods is to guarantee data privacy by fully or partially randomizing the content of an original image. Cryptosystems can be symmetric, when the same key is used during the encryption and the decryption phases, like in AES or DES, or asymmetric, when there are public and private keys, like in RSA or in the Paillier cryptosystem. Moreover, in symmetric cryptography, data can be encrypted independently of the last operation or by utilizing previously encrypted content [1]. During the transmission or the archiving of the encrypted digital data, it is often necessary to analyze or process it, without knowing the original content or the secret key used during the encryption phase. In recent years, this topic has attracted increasing research attention and different image processing methods in the encrypted domain have been developed [2], such as visual secret sharing (VSS) schemes, recompression of crypto-compressed digital images, indexation and search techniques in encrypted databases and reversible data hiding in encrypted images (RDHEI).

Furthermore, encrypted data can be damaged during its transmission through a noisy channel or by watermarking. Even if the secret key is known during the decryption phase, it becomes difficult to reconstruct the original image without errors. In order to deal with this problem, error correction methods for noisy encypted images have been proposed. Classical error correction codes introduce redundancy in the digital data [3]. After detection, error correction can carried out in two different ways: automatic repeat request (ARQ) or forward error correction (FEC). In the first instance, the error detection scheme is combined with requests for retransmission of erroneous data until all the data can be verified. In the second, the sender of the image encodes it by using an error-correcting code (ECC) before the transmission phase. After transmission, redundancy is used to check consistency of the delivered message and to recover initial data. Privacy-preserving error correction schemes are also proposed. Hu *et al.* described a technique where a double cipher is used to perform non-local means (NLM) denoising [4]. Some authors suggested resorting to secret sharing, like SaghaianNejadEsfahani *et al.* in [5]. Recently, Pedrouzo-Ulloa *et al.* presented an error correction scheme based on 2-ring learning with errors (2-RLWE) where they combined homomorphic polynomial equations and thresholding [6]. Other methods allow the removal of noise by completing a statistical analysis of each block of the encrypted image during the decryption process to determine if it has been decrypted or if it is still encrypted. Puech *et al.* proposed a RDHEI scheme where they performed an analysis of the standard deviation of the marked encrypted image, in order to reconstruct the original version without any errors during the decryption step [7]. Islam *et al.* described an effective means to correct noisy AES-encrypted images by calculating three statistical measurements: global variance method (GVM), mean local variance method (MLVM) and sum of the squared derivative method (SSDM) [8].

Although some papers were interested by pixel block entropy calculation [9], none of the previous noisy encrypted image correction methods are based on Shannon entropy [10]. In fact, due to the sparsity of the sample when a small block-size is considered, the direct use of entropy is not possible.

For this reason, in this paper, we are interested in analyzing the signification of this statistical measurement according to the considered block-size and by adapting its calculation in order to be able to use it for noisy encrypted image correction.

Section II depicts our analysis on the entropy measurement as a function of pixel block-size in an image and our method of error localization and removing in a noisy encrypted image. In Section III, experimental results and discussion are presented. Finally, the conclusion is drawn in Section IV.

II. Entropy analysis as a function of block-size and number of grey-levels

In this section, we first perform an entropy analysis as a function of block-size and number of grey-levels in an image. We study the zero-order entropy measurement, and then, we

ISBN 978-90-827970-1-5

exploit the redundancy between pixels by performing a distance-map entropy analysis. In a second phase, we present our blind method based on local entropy measurement to detect and correct the errors in a noisy encrypted image.

A. Zero-order entropy

Let X be an image with a size of $m \times n$ pixels with l grey-levels α_i $(0 \leq i < l)$, with the associated probability $p(\alpha_i)$. Zero-order entropy of an image X, in *bit per pixel* (*bpp*) is:

$$H(X) = -\sum_{i=0}^{l-1} p(\alpha_i) \log_2(p(\alpha_i)). \quad (1)$$

In the particular case where the l grey-levels α_i have the same probability, zero-order entropy value is maximal:

$$H(X) = -\sum_{i=0}^{l-1} \frac{1}{l} \log_2\left(\frac{1}{l}\right) = \log_2(l) \ bpp. \quad (2)$$

If the encryption algorithm is effective, the pixel values of an encrypted image are pseudo-randomly generated. Therefore, the grey-level distribution tends to be uniform. Then, entropy value of an encrypted image with l grey-levels (H_e) is very close to the maximal entropy value:

$$H_e \simeq \log_2(l) \ bpp. \quad (3)$$

In the clear domain, we assume that an image follows a normal law. For discrete values, the equivalent to the normal distribution is the binomial distribution, according to the de-Moivre-Laplace theorem. Therefore, in the clear domain, entropy value of an image with l grey-levels (H_c) is approximated by the binomial law $\mathcal{B}(l,p)$ entropy value:

$$H_c \simeq \frac{1}{2} \log_2\left[2\pi e(l-1)p(1-p)\right] \ bpp, \quad (4)$$

with e the base of the exponential function and $0 \leq p \leq 1$.

If we compare the value of zero-order entropy of a clear image and those of an encrypted one, we would like to have:

$$\begin{aligned} \frac{1}{2} \log_2\left[2\pi e(l-1)p(1-p)\right] &< \log_2(l), \\ 2\pi e(l-1)p(1-p) &\leq l^2. \end{aligned} \quad (5)$$

If l is large, this inequality is always true because $\lambda l \ll l^2$, with $\lambda < l$. If l is large enough, then, the entropy value in a clear image is smaller than those of an encrypted image:

$$H_c < H_e. \quad (6)$$

Therefore, we can consider blocks of k pixels in an image of l grey-levels α_i, instead of the full image (*i.e.* we examine a much smaller pixel sample), in order to define the concept of local entropy. Let B be a block of k pixels in an image of l grey-levels, with the associated probability $p(\alpha_i)$. Local entropy (*i.e.* inside the block) is increased by the minimal value between its block-size k and the number of grey-levels l:

$$H_{(k,l)}(B) \leq \log_2(\min(k,l)) \ bpp. \quad (7)$$

Indeed, if the block-size is larger than the number of grey-levels, maximal entropy corresponds to equiprobability between all the grey-levels. Otherwise, if there are more grey-levels than pixels in the block, the maximal value is reached when each pixel value is different. In this case, the pixels sample is sparse, because some grey-level values are not present in the block B. For this reason, the entropy measurement may be erroneous and a block in the clear domain may be considered as encrypted.

When the number of grey-levels is much greater than the block-size, the maximum entropy value is thus often reached in the clear domain and we cannot distinguish a clear block from an encrypted one by using the standard zero-order entropy value. Therefore, we propose to quantize the number of grey-levels for the entropy measurement in order to decrease the value of l. The idea is to find the best trade-off between the block-size k and the number of grey-levels l in the image.

B. Distance-map entropy

In the zero-order entropy measurement, we do not take into account the local correlation between neighboring pixels in the clear domain. Indeed, values in adjacent pixels are very close, which is not the case in the encrypted domain: the correlation is very small since pixels are pseudo-randomly generated. Note that even if there is an edge in a block in clear, this frontier delimits two relatively homogeneous regions.

In order to exploit this property, we generate the distance-map D from the original image X. We compute the absolute difference between a pixel x and its predictor $pred(x)$, computed according to the values of its neighbors:

$$\forall d \in D, \ d = d(x, pred(x)) = |x - pred(x)|. \quad (8)$$

Like the original image, the distance-map is also encoded on l grey-levels. By using Eq. (1), since each distance value d_i $(0 \leq i < l)$ has the probability $p(d_i)$, the distance-map entropy is:

$$H(D) = -\sum_{i=0}^{l-1} p(d_i) \log_2(p(d_i)). \quad (9)$$

In the encrypted domain, the distance distribution is not uniform: it depends on the value of x. For example, if x is equal to 128, the range of the distance value is $[\![0, 128]\!]$: $\forall \ pred(x), d(x, pred(x)) \leq 128$, and then, $P(D > 128 \mid X = 128) = 0$. Therefore, theoretical distance-map entropy (H_e^D) in the encrypted domain is:

$$H_e^D = \left[-\sum_{i=1}^{l-1} \frac{2i}{l^2} \log_2\left(\frac{2i}{l^2}\right)\right] - \frac{1}{l} \log_2\left(\frac{1}{l}\right). \quad (10)$$

In a clear image, the distance value distribution seems to be a geometric one. Consequently, theoretical distance-map entropy (H_c^D) in the clear domain is:

$$H_c^D = -\sum_{i=1}^{l-1} \left((1-q)^{i-1}q\right) \log_2\left((1-q)^{i-1}q\right), \quad (11)$$

with $0 < q < 1$.

According to Eq. (10) and Eq. (11), we have:

$$H_c^D < H_e^D. \quad (12)$$

Entropy value of the distance-map in the clear domain is then smaller than its value in the encrypted domain. In case of local distance-map entropy, the trade-off between the block-size k and the number of grey-levels l still respects Eq. (7).

ISBN 978-90-827970-1-5 © EURASIP 2018

C. Application: noisy encrypted image correction

Although encryption algorithms are useful to preserve the content confidentiality of an original image, they are also extremely noise-sensitive. In case of noisy encrypted image, the knowledge of the encryption key is not sufficient to reconstruct the original content without error. Indeed, even if only one bit of the encrypted image is altered, the reconstructed image can be quite different from the original one. Hence, during the decryption phase, it is necessary to perform a correction of the noisy encrypted image. According to the presented analysis, we propose to achieve a local entropy analysis during the error removal process.

Suppose that an original image has been encrypted using a block cipher, such as the AES algorithm in ECB mode. If the encrypted image is noised during its transmission across a channel, some of its pixel blocks cannot be correctly decrypted without correction. However, we have seen previously that zero-order (or distance-map) entropy is smaller in the clear domain than in the encrypted domain (Eq. (6) and Eq. (12)). Therefore, this property can be used to differentiate a clear block from a badly decrypted one, which seems still encrypted. In fact, local entropy value of the expected block in clear (*i.e.* when the encrypted block is correctly denoised before decryption) should be much smaller than in case of erroneous block reconstruction. As there is no information about both location and number of altered bits, all bits of the encrypted image can be possibly erroneous. For each block of k pixels, there are also 2^{8k} possible combinations to test. Consequently, this is not practically feasible, due to the computational complexity, even if k is small. This is actually equivalent to perform a brute-force attack. Nevertheless, in practice, the amount of noise which affects the image is low and relies on the transmission type. The bit error rate (BER) is also between 10^{-12} for optical fiber transmission, and 10^{-4} for wireless transmission. Moreover, as noise is a pseudo-random phenomenon, we can presume that it uniformly alters an encrypted image. Considering these two assumptions, during the encrypted image denoising, we can suppose that one bit at most has been flipped into each block of k pixels (with a small k), which corresponds to a BER of $\frac{1}{8k}$. Associated computational complexity is greatly reduced, because there are also $8k+1$ configurations to test for each block of k pixels.

In our proposed error correction method, for each block, we are interested by defining the set of the possible correct configurations in clear, according to the local entropy value. In most cases, the configuration which has the lowest local entropy value is the expected block in clear. Indeed, the associated local entropy value is much smaller than for the other configurations, where the local entropy value is close to be maximal. However, in rare cases, some block configurations have close local entropy values after decryption. This problem arises when the local entropy value of the expected block in clear is high – especially when there is a edge into the block – or when the local entropy value of a badly decrypted block is quite low. In these two cases, the expected block in clear corresponds to one of the configurations which minimize the local entropy value, but has not necessarily the lowest value.

Algorithm 1: Identification of the possible correct configurations in clear for one block.

Data: Block $B[i]$ of size k, number of grey-levels l, threshold Δ
Result: Set L of the possible correct configurations in clear
`/* Step 1: Search of the decrypted block combination which minimizes entropy */`
$H_{min} = H_{(k,l)}(B[i])$;
$B_{min} = D_{AES}(B[i])$;
for $j = 0$ **to** $8k-1$ **do**
  if $H_{(k,l)}(D_{AES}(B[i]_{j_{\bar{b}}})) < H_{min}$ **then**
    $H_{min} = H_{(k,l)}(D_{AES}(B[i]_{j_{\bar{b}}}))$;
    $B_{min} = D_{AES}(B[i]_{j_{\bar{b}}})$;
$L \leftarrow B_{min}$;
`/* Step 2: Search of all possible correct configurations in clear, according to Δ */`
if $\left|H_{(k,l)}(D_{AES}(B[i])) - H_{min}\right| < \Delta$ **then**
  $L \leftarrow D_{AES}(B[i])$;
for $j = 0$ **to** $8k-1$ **do**
  if $B[i]_{j_{\bar{b}}} \neq B_{min}$ **then**
    if $\left|H_{(k,l)}(D_{AES}(B[i]_{j_{\bar{b}}})) - H_{min}\right| < \Delta$ **then**
      $L \leftarrow D_{AES}(B[i]_{j_{\bar{b}}})$;

Algorithm 1 presents the steps to define the set of the possible correct configurations in clear, for one block $B[i]$ of size k of a noisy encrypted image I_{ne} with $m \times n$ pixels ($0 \leq i < \frac{m \times n}{k}$). First step consists to identify the decrypted block combination B_{min} which has the lowest local entropy value H_{min}, according to a number of grey-levels l chosen in accordance with the block-size k in order to perform a significant measurement. Indeed, we consider the $8k+1$ possible combinations of the input block: the initial configuration, and the $8k$ others by modifying only one bit b by its inverse value $\bar{b}$. Then, the decryption function of the AES algorithm $D_{AES}(\cdot)$ is applied to each configuration, and the local entropy measurement is performed. During a second step, according to a threshold Δ and the minimal local entropy value H_{min} (computed in Step 1), the set L of the possible correct configurations in clear for the block $B[i]$ is defined. After applying Algorithm 1 to all the blocks $B[i]$ ($0 \leq i < \frac{m \times n}{k}$), almost all the blocks of the original image are correctly reconstructed, when there is just one possible correct configuration in clear ($|L| = 1$). For the remaining blocks, when $|L| > 1$, the expected correct configuration in clear is necessarily in the set of the possible correct configurations. It is not possible to distinguish it from the other possible configurations using only the local entropy, because values are too close and not discriminating. However, note that this proposed method allows to obtain a confidence index on the error location and possible correct configurations in clear for each block. In fact, this is not the case for the previous noisy encrypted image correction methods, where some blocks remain badly decrypted. Their location is unknown without resorting to the original image, and there are some artifacts on the reconstructed image which cannot be removed.

ISBN 978-90-827970-1-5 © EURASIP 2018

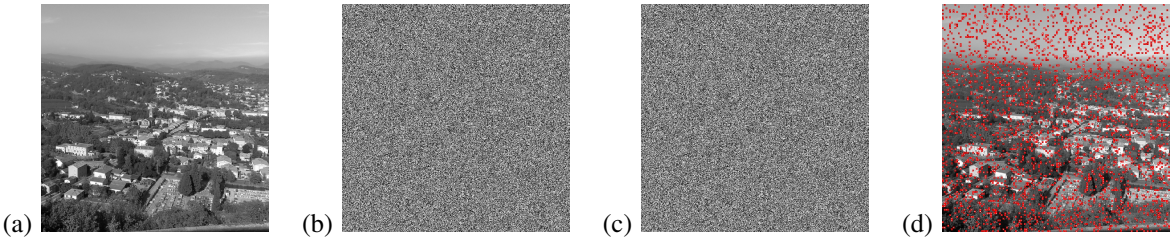


Fig. 1. Illustration of the problem of noisy encrypted image decryption: a) Original *Village* image (512×512 pixels, 256 grey-levels), b) Encrypted image, using the AES algorithm in ECB mode with blocks of 4×4 pixels (PSNR with the original image $= 8.86\ dB$), c) Noisy encrypted image, BER = 2.6×10^{-3} (PSNR with the original image $= 8.85\ dB$, PSNR with the encrypted image $= 33.71\ dB$), d) Directly decrypted image (PSNR with the original image $= 16.51\ dB$).

III. Experimental results and discussion

In Fig. 1, we first present an illustration of the problem of noisy encrypted image decryption. The original *Village* image, of size 512×512 pixels and encoded on 256 grey-levels, is presented in Fig. 1.a. This image is encrypted using the AES algorithm in ECB mode, with blocks of 4×4 pixels, illustrated in Fig. 1.b. Note that there is no information about the original image content in the encrypted image, as indicated by a very low PSNR of $8.86\ dB$. During the transmission, the encrypted image has been randomly noised with a BER of 2.6×10^{-3}, which randomly corrupts approximately one bit, every three blocks (see Fig. 1.c). PSNR between the noisy encrypted image and the original image remains very low ($8.85\ dB$). Moreover, the encrypted image and its noisy version are quite different (PSNR $= 33.71\ dB$). As shown in Fig. 1.d, if we directly decrypt the noisy encrypted image without correction, there is a large number of erroneous blocks (framed in red for more visibility). Therefore, the original image cannot be recovered after decryption, even with the correct secret key. In fact, the reconstructed image is very different from the original version, according to a low PSNR value of $16.51\ dB$. This highlights the necessity to apply a method of error localization and correction in the noisy encrypted image, during the decoding phase.

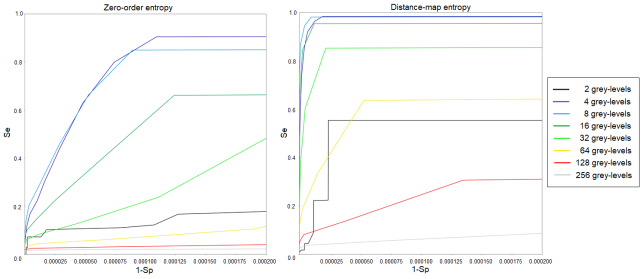


Fig. 2. ROC curves in order to select the best parameters and the best threshold Δ to use in our method of correction of noisy encrypted image, for a block-size $k = 4 \times 4$ pixels, depending if zero-order entropy or distance-map entropy is used and for different numbers l of grey-levels ($2 \leq l \leq 256$).

In Section II-C, we proposed to analyze the local entropy value, for each block of a noisy encrypted image. The idea is to use the fact that, in theory, local entropy value of a block in the clear domain is much smaller than in the encrypted domain. However, in practical cases, this is not always true in textured areas. In addition, when an original image is encrypted using the AES algorithm, the block-size k to consider for local entropy measurements is very small (4×4 pixels). With this block-size and by supposing that one bit at most has been flipped in a noisy encrypted block, there is also one correct configuration among 129. For this reason, making the assumption that the clear block corresponds to the configuration which minimizes the local entropy value is not practicable to correct all the errors in a noisy encrypted image. In fact, there is a significant risk to consider a badly decrypted configuration as the expected one and, even worse, this would not be possible to identify this error. Therefore, the best practice consists to find the set of the possible correct configurations for each block. Note that these possible configurations are always close to the evaluated minimal entropy value, which is more or less high. Then, the set of the possible correct configurations is composed of the configurations whose the difference with the evaluated minimal entropy value is smaller than a threshold value. In Fig. 2, in order to be sure of always having the correct configuration in this set for each block, we analyze the threshold Δ which has to be consider for a block-size of 4×4 pixels by ROC curves analysis, depending if zero-order entropy or distance-map entropy is used and for different numbers l of grey-levels ($2 \leq l \leq 256$). We perform these tests on $352,256$ blocks of 4×4 pixels with strong statistical variability. Moreover, we categorize the configurations as following:

- Positive: Configuration B, $H_{(k,l)}(B) - H_{min} \leq \Delta$.
  TP: B is positive and is the configuration in clear.
  FP: B is positive and is not the configuration in clear.
- Negative: Configuration B, $H_{(k,l)}(B) - H_{min} > \Delta$.
  TN: B is negative and is not the configuration in clear.
  FN: B is negative and is the configuration in clear.

Note that the number of false negative (FN) has to be null, in order to be sure that the expected configuration in clear is in the set of possible correct configurations. Moreover, we are also interested by minimizing the number of false positives to reduce the size of the set. Therefore, the threshold value is associated to the point on the curve with an abscissa equal to zero and an ordinate as higher as possible. Firstly, we can remark that results are better with distance-map entropy than with zero-order entropy, according to the area under the ROC curves. Then, if we observe the number of grey-levels, best results are achieved with $l = 8$ grey-levels (curve in sky blue). In conclusion, for our error localization and correction method, we will consider these parameters. The associated threshold value is therefore $\Delta = 0.25$ (after normalization of the entropy measurement by dividing the computed value by the maximal entropy value considering (k, l)).

In Fig. 3, according to this threshold, we applied our method

ISBN 978-90-827970-1-5 © EURASIP 2018

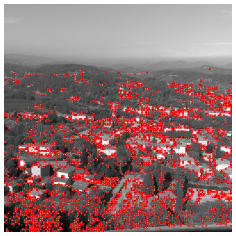
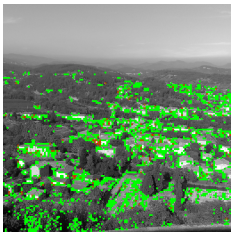
(a)  (b)  (c)  (d) 

Fig. 3. Example of our method, applied to the noised encrypted image in Fig. 1.c: a) Correction of the noisy encrypted image using distance-map entropy without image quantization ($l = 256$) and considering for each block, the configuration which minimizes the entropy value as the expected value in clear (PSNR with the original image $= 16.37\ dB$), b) Correction of the noisy encrypted image using distance-map entropy after image quantization ($l = 8$) and considering for each block, the configuration which minimizes the entropy value as the expected value in clear (PSNR with the original image $= 29.73\ dB$), c) Location of the blocks where there are more than one possible configuration for the block in clear, applying the Algorithm 1 using distance-map entropy, $l = 8$ grey-levels and $\Delta = 0.25$, d) Approximation of the original image, according to Fig. 3.c.

on the noisy encrypted *Village* image displayed in Fig. 1.c. In Fig. 3.a and Fig. 3.b, we illustrate the obtained results by making the assumption that the clear block corresponds to the configuration which minimizes the local entropy value, by considering respectively $l = 256$ and $l = 8$ grey-levels during the distance-map entropy calculation. As expected, we can see that there is a non-negligible amount of blocks which are badly decrypted (PSNR $= 16.37\ dB$ without quantization *vs* PSNR $= 29.73\ dB$ after quantization with $l = 8$). However, by comparing the results obtained in these two cases, we can see the interest to perform a quantization step before the entropy measurement, because this allows to significantly decrease the number of badly decrypted blocks. In Fig. 3.c, we can see the obtained results applying the Algorithm 1, using distance-map entropy, $l = 8$ grey-levels and $\Delta = 0.25$. When there is only one possible correct configuration in clear, blocks of the original image are perfectly reconstructed. Moreover, all blocks which are badly decrypted in Fig. 3.b are localized in Fig. 3.c. Indeed, they form a sub-set of the blocks for which there are more than one possible correct configuration in clear. Unfortunately, they are not the only ones, as indicated by the blocks represented in green, which are not framed in red. Therefore, we are sure that a set with a size equal to one consists of the correct configuration. In fact, in the context of blind approach, we can localize the correctly decrypted blocks and of the possibly badly decrypted ones, using local entropy. Finally, in Fig. 3.d, using a simple interpolation to estimate values of the blocks in clear where there are more than one possible correct configuration, we can obtain a very good approximation of the original image, without visual artifact such as badly decrypted blocks.

In order to provide average results, we applied our method on a set of 100 images with various statistical properties. If we consider the configuration which minimizes the entropy value as the expected block value in clear, we have 0.2% of blocks which cannot be correctly decrypted. Using our confidence index on the error location, 4.5% of the blocks are identified as blocks with more than one possible configuration in clear, on average. Therefore, note that there is no false negative with the threshold $\Delta = 0.25$ for all blocks from the dataset. Finally, we are sure that more than 95% of the blocks are correctly decrypted, without reference to the original image.

IV. Conclusion

In this paper, we performed an analysis of the use of Shannon entropy to correct noisy encrypted images. As zero-order entropy value in a block of pixels in a clear image is generally smaller than the value in the encrypted domain, it is possible to know if a block has been correctly decrypted or not during the decoding phase. However, there are some misconfigurations when blocks in the clear domain are highly textured. In this case, entropy value in the clear domain can be close to the values measured for badly decrypted configurations (and even higher), in particular when we consider very small block-sizes. A first idea to reduce the number of misconfigurations is to adapt the number of grey-levels by image quantization. Moreover, using distance map entropy, we show that we can significantly reduce the number of possible correct configurations in clear for each block, since we exploit the natural correlation between neighboring pixels in the clear domain. Furthermore, with our approach, we obtain a confidence index on the error location and the possible correct configurations in clear for each block.

In future work, we are also involved in the extension of the proposed method to noisy encrypted images using AES in CBC mode in order to detect if there is more than one error into a block.

References

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

[2] 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.

[3] S. B. Wicker, *Error control systems for digital communication and storage*. Prentice hall Englewood Cliffs, 1995, vol. 1.

[4] X. Hu, W. Zhang, H. Hu, and N. Yu, "Non-local denoising in encrypted images," in *International Conference on Internet of Vehicles*. Springer, 2014, pp. 386–395.

[5] S. M. SaghaianNejadEsfahani, Y. Luo, and S.-C. S. Cheung, "Privacy protected image denoising with secret shares," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pp. 253–256.

[6] A. Pedrouzo-Ulloa, J. R. Troncoso-Pastoriza, and F. Pérez-González, "Image denoising in the encrypted domain," in *Information Forensics and Security (WIFS), 2016 IEEE International Workshop on*, pp. 1–6.

[7] 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.

[8] N. Islam, Z. Shahid, and W. Puech, "Denoising and error correction in noisy AES-encrypted images using statistical measures," *Signal Processing:Image Commun.*, vol. 41, no. C, pp. 15–27, Feb. 2016.

[9] Y. Wu, Y. Zhou, G. Saveriades, S. Agaian, J. P. Noonan, and P. Natarajan, "Local Shannon entropy measure with statistical tests for image randomness," *Information Sciences*, vol. 222, pp. 323–342, 2013.

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

ISBN 978-90-827970-1-5 © EURASIP 2018