NEW STEGANOGRAPHY ENCRYPTION ALGORITHM BASED ON DISCRET TOMOGRAPHY APPLIED ON ULTRASOUND IMAGE

RANDRIANARISOA Voahary Finaritra[1], RANDRIAMITANTSOA Paul Auguste[2], RAKOTOMALALA Mamy Alain[3]

[1] *PhD student, TASI, ED-STII, Antananarivo, Country*
[2] *Thesis director, TASI, ED-STII, Antananarivo, Country*
[3] *Thesis co-director, TASI, ED-STII, Antananarivo, Country*

ABSTRACT

Information security covers many areas such as steganography, watermarking or cryptography. This article proposes a security method that combines scrambling, steganography and cryptography. The secrecy to be protected is a digital image which will be scrambled, encrypted then the encryption key would be inserted within that image encrypted via steganography. Inversely, there is a perfect reconstruction by means of an infinite PSNR value and the algorithm execution delay is not exceeding 0.1 seconds.

Keyword: - *steganography, scrambling, cryptography, Mojette transform, block cipher.*

1. Introduction

The purpose of information security is to guarantee the protection of information against different types of attackers. In this article, the objective is to ensure the confidentiality of secret image by using a steganography encryption algorithm based on Mojette transform.
Encryption uses the ECB (Electronic Code Book) mode whose key is generated with one-to-one correspondence bins of Mojette transform. In order to compensate the block effect due to that technique, secret image will beforehand be subject to a scrambling code.
Encrypted image would be bearer of the secret key which will allow the recipient to decrypt the information.

2. Methodology

2.1 Stage of security algorithm

The methodology could be summarized by the following diagram:

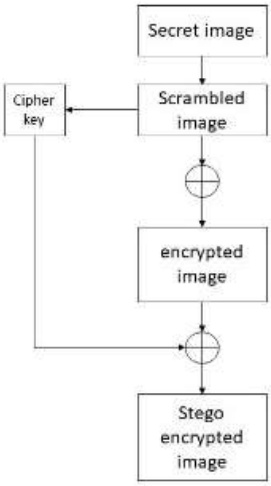
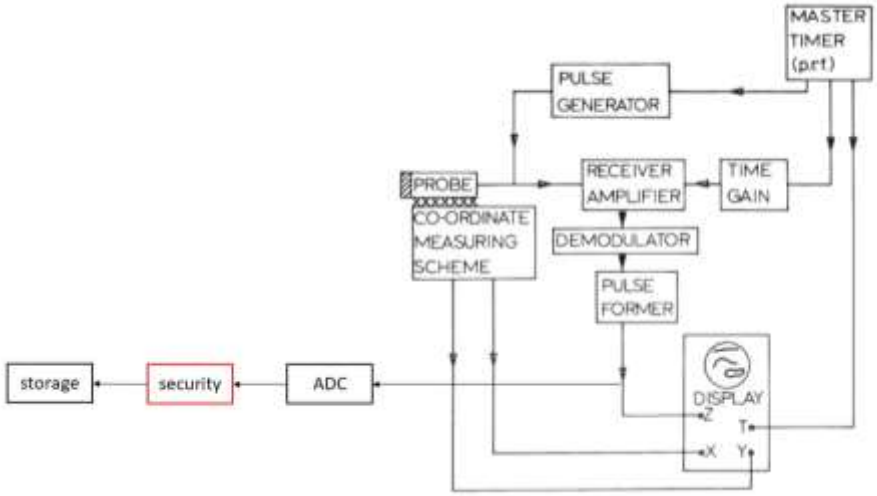


Fig -1 Stage of security algorithm

It must follow through three stages :
- Scrambling stage
- Encryption stage
- Steganography encryption stage

This diagram could be regarded as a diagram which will integrate the block diagram of an ultrasound device as illustrated below:



A block diagram of a basic B-mode scanning instrument [1], with the security block

2.2 Scrambling code

Scrambling code consists of the permutation of an image pixels position in order to make its reading unintelligible. Many methods has already been presented in the literature as a scrambling scheme using a conventional Fibonacci number of [2] or a scrambling method of chaos in the field of wavelet of [3] or a scrambling method using two types of Fibonacci p-code of [4].
The proposed scrambling system is described on this equation:

$$\forall M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad M_s = scrambling\,(M) = \begin{bmatrix} D & C \\ B & A \end{bmatrix} \tag{1}$$

This figure represents the first round of scrambling scheme; then in the second round, the same function should be reapplied on each quadrant.

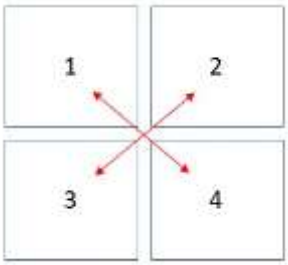


Fig -3 Principle of scrambling code

Finally, after the nth round, the result is featured with following picture :

Fig -4 On the left : original image; on the right: scrambled image with the system

2.3 Encryption

Encryption key is generated as per the following scheme :

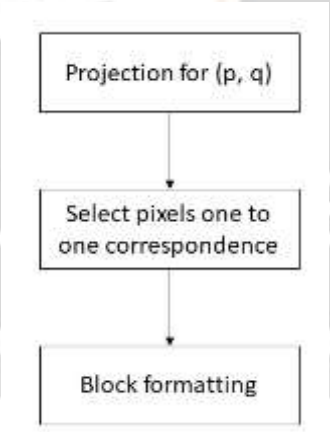


Fig -5 key generation

Mojette transform is defined by a set of projections. A discrete projection is characterized by the direction of projection (p,q) such as [5] [6] [7]:

$$pgcd(p,q) = 1 \quad (2)$$

And:

$$\theta = arctan\frac{q}{p} \quad (3)$$

Where :

θ: projection angle

Projection is composed of bins whose values correspond to the sum of pixels verifying :

$$b = pl - kq \quad (4)$$

With :
b : bin index
k : like « kolumn »
l : like « line »
(p, q) : direction of projection
In the image, at least one pixel contribute to a bin. There is some bins called one to one correspondence bins where a bin is composed with a single pixel. In our algorithm, these pixels of one to one correspondence bins will serve as encryption key element.
To perform the block cipher of the secret image, the secret key size should match an elementary block of that image. Therefore, the one to one correspondence bin vector which serves to the secret key creation must be reduced in order to address that constraint.

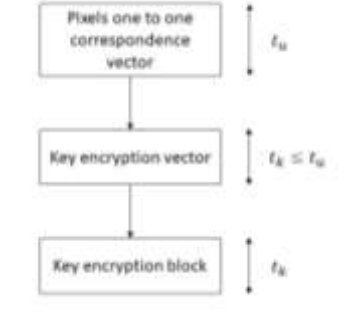


Fig -6 secret key blocking

The block cipher mode ECB (Electronic Code Book) consist to encrypt independently each block. For any plaintext block mi, ciphertext is given by [8]:

$$c_i = E_k(m_i) \quad (5)$$

With :
ci : ciphertext
mi : plaintext
Ek : encryption function
The proposed encryption function is the XOR operator :

$$c_i = E_k(m_i) = m_i \oplus k \quad (6)$$

With :
ci : ciphertext
mi : plaintext
k : encryption key
Ek : encryption function
The decryption function is equivalent to the encryption function :

$$m_i = D_k(c_i) = c_i \oplus k \quad (7)$$

2.3 Steganography encryption

The idea consists to replace the one to one correspondence bins which has been served for the cipher key within the image. Hence, the recipient could extract from the encrypted image the key which will allow to reconstruct the original image.

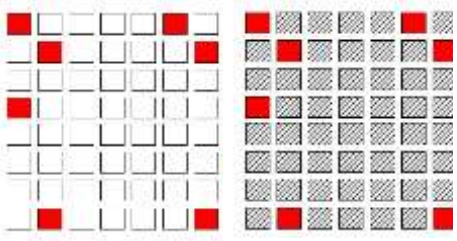


Fig -7 Secret key positioning within the encrypted image(on the left: scrambled image, on the right: cipher-image)

3. Results

The simulation proceeds as follows :

- Steganography encryption algorithm :
  - choose test image
  - scramble image in order to make it unintelligible
  - choose the projection direction (p, q) which will constitute the key to be shared between the sender and the receiver
  - identify the pixels of one to one correspondence (see equation (2)) then we reconstruct cipher key (figure 4)
  - cipher image with cipher key (XOR operation)
  - replace the pixels of unequivocal bins (which have constituted the encryption key) within the encrypted image
- The inverse algorithm :
  - use the projection direction (p,q) (secret key) for identifying the pixels of unequivocal bins in order to reconstitute the decryption key
  - decrypt image
  - replace the pixels of one to on correspondence
  - descramble the image in order to retrieve the original one

Test image choosed for the simulation is as follows :

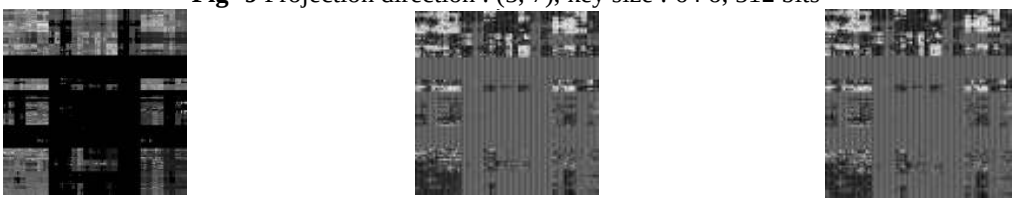
Fig -8 Test image size 512 * 512

The following figures show the visual result of our simulation for each stage :

- (a)scrambling,
- (b)encryption ,
- (c) steganography encryption.

(a) (b) (c)

Fig -9 Projection direction : (5, 7), key size : 64 o, 512 bits

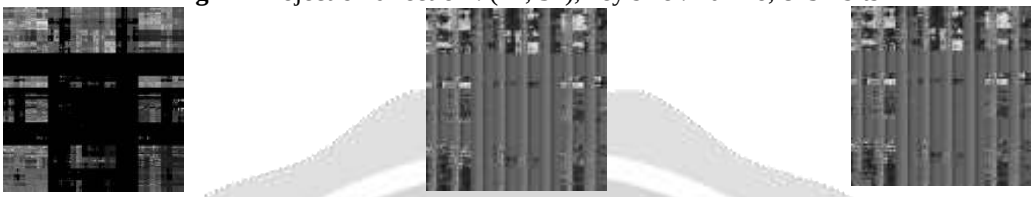


(a) (b) (c)

Fig -10 Projection direction : (11, 13), key size : 256 o, 2048 bits

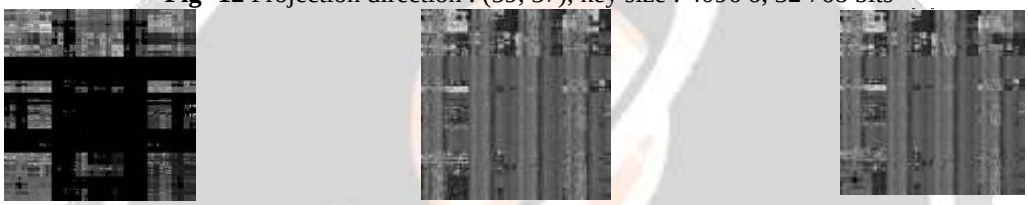
(a) (b) (c)

Fig -11 Projection direction : (17, 31), key size : 1024 o, 8192 bits



(a) (b) (c)

Fig -12 Projection direction : (59, 37), key size : 4096 o, 32 768 bits



(a) (b) (c)

Fig -13 Projection direction : (89, 97), key size : 16384 o, 131 072 bits

The following table outline the execution delay result of our simulation, we have a perfect reconstruction (PSNR=infinite, SSIM=1) after inverse algorithm.

Projection direction	Key size (octets)	Steganography encryption time (seconds)	Reconstruction time (seconds)
(5, 7)	64	0.0849	0.0778
(11, 13)	256	0.0736	0.0932
(17, 31)	1024	0.0603	0.0720
(59,37)	4096	0.0640	0.0686
(89, 97)	16384	0.0658	0.0691

Results are modelled by subsequent diagrams.

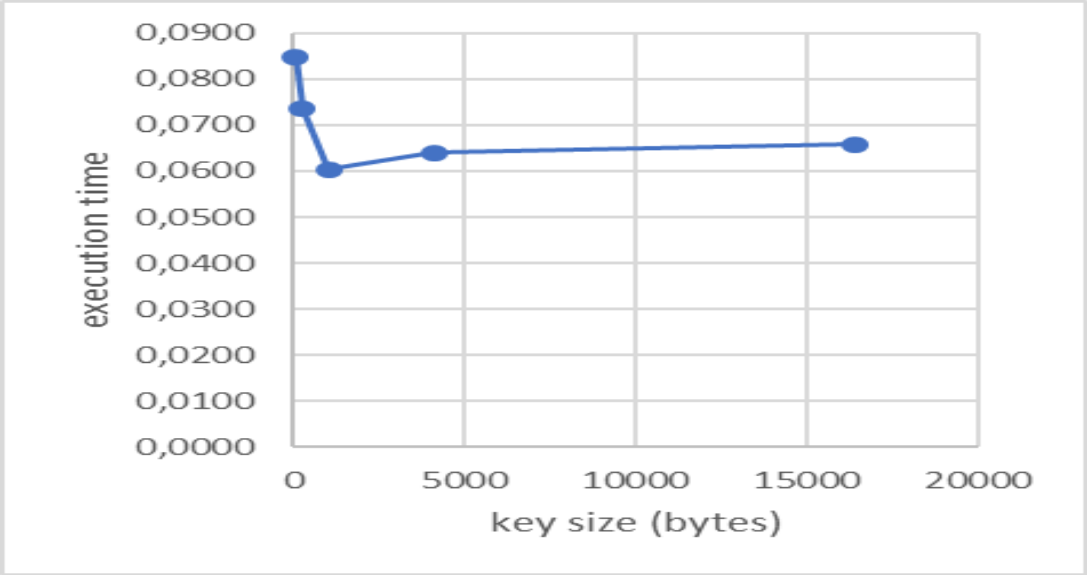


Fig -13 Algorithm execution time as a function of key size

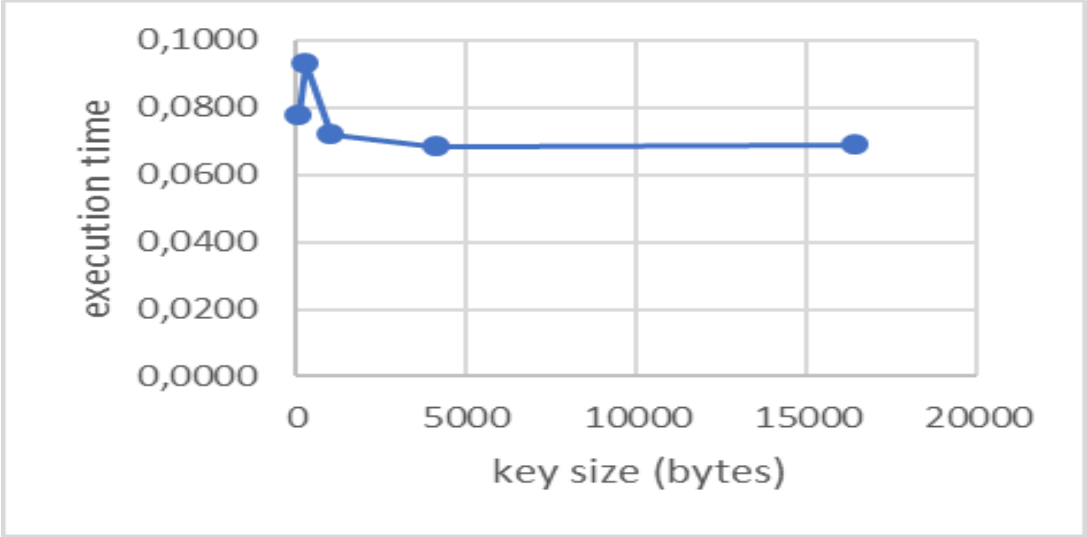


Fig -15 Inverse algorithm execution time as a function of key size

4. Observations

Visual results show that the algorithm result could not allow by no means the original image to be recognizable. Scrambling code has already transform the image to be unintelligible, and encryption makes the system more complex.

Larger the key size is, bigger the difference between the encrypted image and the steganography encrypted image is. Indeed, the steganography encrypted image (right image on the previous figures) present smooth zones at the lower left corner and the higher right corner and which jump out because it differ from other zones characterized by noise.

The algorithm execution time as well as its inverse operation are almost equal. For every key sizes, the time will not exceed 0.1 seconds.

5. Interpretations

Within the encryption algorithm, it is recommended to use a large key as much as possible in order to avoid an attacker breaking it easily. For our case, the use of a smaller key allows us to reduce the smooth zones size of the steganography encrypted image and therefore avoid that an attacker could guess the key position.
For this scheme, the key to be shared with the sender and the receiver is p and q number which are the projection directions of Mojette transform. From these numbers, the receiver could research key positions which served for encryption. The weight and the key size is thus reduced and it facilitate the key sharing.
If the encryption key were deducted from the original image, it is possible that it was constituted with homogeneous zone, given that the unequivocal pixels correspond to peripheral zone of the image. However, as the image were scrambled first before reconstructing the key encryption, result delivers a key with an almost random value.
The algorithm execution time allows its using within a transmission chain or a real-time application.

6. Conclusion and perspectives

In this article, we proposed a steganography encryption algorithm based on the Mojette transport. The objective was to reduce the key to be shared and make the encrypted image to also be bearer of key encryption. Mojette transform has permitted to select pixels which has served as elements of that encryption key.
The result in terms of execution time is deemed to be reasonable for a real-time application. Globally, in terms of visual aspect, it is impossible to sense the content of the original image from the steganography encrypted image.
For the inverse algorithm, PSNR and SSIM value shows that the reconstruction is authentic to the original image and thereby, there is no loss information at all.
As perspective, it is possible to choose another variants of discrete tomography (comprising the Mojette transform) in order to complicate the choice of encryption key elements.

7. References

[1]M. Hussey, « Basic physics and technology of Medical diagnostic ultrasound », Macmillan, ISBN 978-0-333-36605-9, 1985

[2]J. Zou, R. K. W., D. Qi, « a new digital image scrambling method based on Fibonacci numbers », in ISCAS 2004, 2004

[3]G. Gu, g. H., « the application of chaos and DWT in image scrambling », in Proceeding of the Fifith International Conference on Machine Learning and Cybernetics, 2006, Dalian

[4]Y. Zhou, S. Agaian, V. M. Joyner, K. Panetta, « two Fibonnacci P-code Based Image Scrambling Algorithms », proceedings of SPIE, volume 6812(1), 2008

[5]JP. Guédon, « The MojetteTransform : Theory and Applications », Wiley– ISTE, 2009

[6]M. Servières, « reconstruction tomographique Mojette », Ecole Doctorale Sciences et Technologie de L'Information et Des Matériaux, 07 Décembre 2005

[7]B. Recur, « précision et qualité en reconstruction tomographique : algorithmes et applications », Ecole doctorale de mathématiques et informatique, 29 Novembre 2010

[8]P. Fouque, « cryptographie appliquée », sécurité des systèmes d'informations, éditions T. I., H5210, 2003