BLIND TATOO OF DIGITAL IMAGE IN THE KSVD DOMAIN

Ramafiarisona Hajasoa Malalatiana[1], Randriamitantsoa Paul Auguste[2], Zafimarina Reziky[3]

[1] PhD, TASI, Antananarivo, Madagascar
[2]Thesisdirector, TASI, ED-STII, Antananarivo, Madagascar
[3] PhD, TASI, Antananarivo, Madagascar

ABSTRACT

This paper Digital image watermarking involves introducing a brand, signature, name or logo of the author into the image to protect it from counterfeit, various illegal copies and shares. There are several watermark methods, most of them based on mathematical transformations. In this document, many methods can be used to realize a watermarking on an image. One of them is the Ksvd algorithm. It consists on overcomplete dictionary that contains prototype signal-atoms, signal are described by sparse linear combinations of these atoms.

Keywords: *sparse coding, watermarking, Ksvd, parsimonious.*

1. INTRODUCTION

Currently, there is a switch to a digital world where via the internet access to information is very simple and very fast. However, scanned documents can be copied, edited and then re-posted on the internet without worrying about copyrights.
Even though some organizations are fighting piracy this has not solved the problem of copyrights, so we opted for tattooing. It consists of inserting a mark called "signature" that only authorized persons can detect thus proving the integrity of any document.

2. PRINCIPLE AND METHODOLOGY

2.1 Instructing phase

In general, all tattooing methods are based on an identical principle. For the tattoo scheme given in figure 1, a message m containing L bits of information is transformed according to a key k into a mark w which is then inserted in the document x also called "host" (can be an image, a sound , or a video), to give a tattooed document y, this is the insertion phase. Here, w is expressed as a noise that is added to the document, the deformation depending on the power of the noise. k is secret and specific to the tattoo artist. It is then copied and etched, which is modeled by transmission in a noise-laden channel. After deformation, the received document is called z. [1] [2]

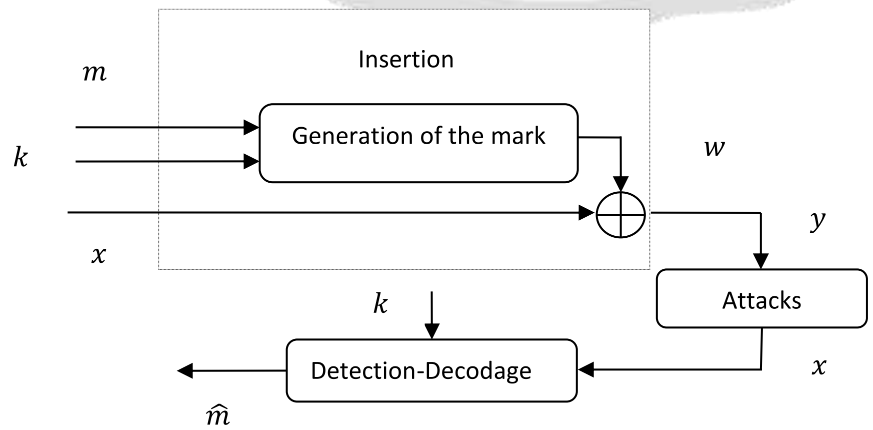


Fig - 1: Principle of image tattoo

2.2Extraction phase

This phase makes it possible to detect the inserted signature. Between the insertion phase and the phase of detecting the tattooed image may have undergone modifications.
The goal is to check if the mark inserted in a specific document (the copyright information) is present or not.

3. CHOICE OF TECHNIQUES AND ALGORITHMS FOR TATOOS

3.1 Algorithm using Ksvd

- Insertion of the mark

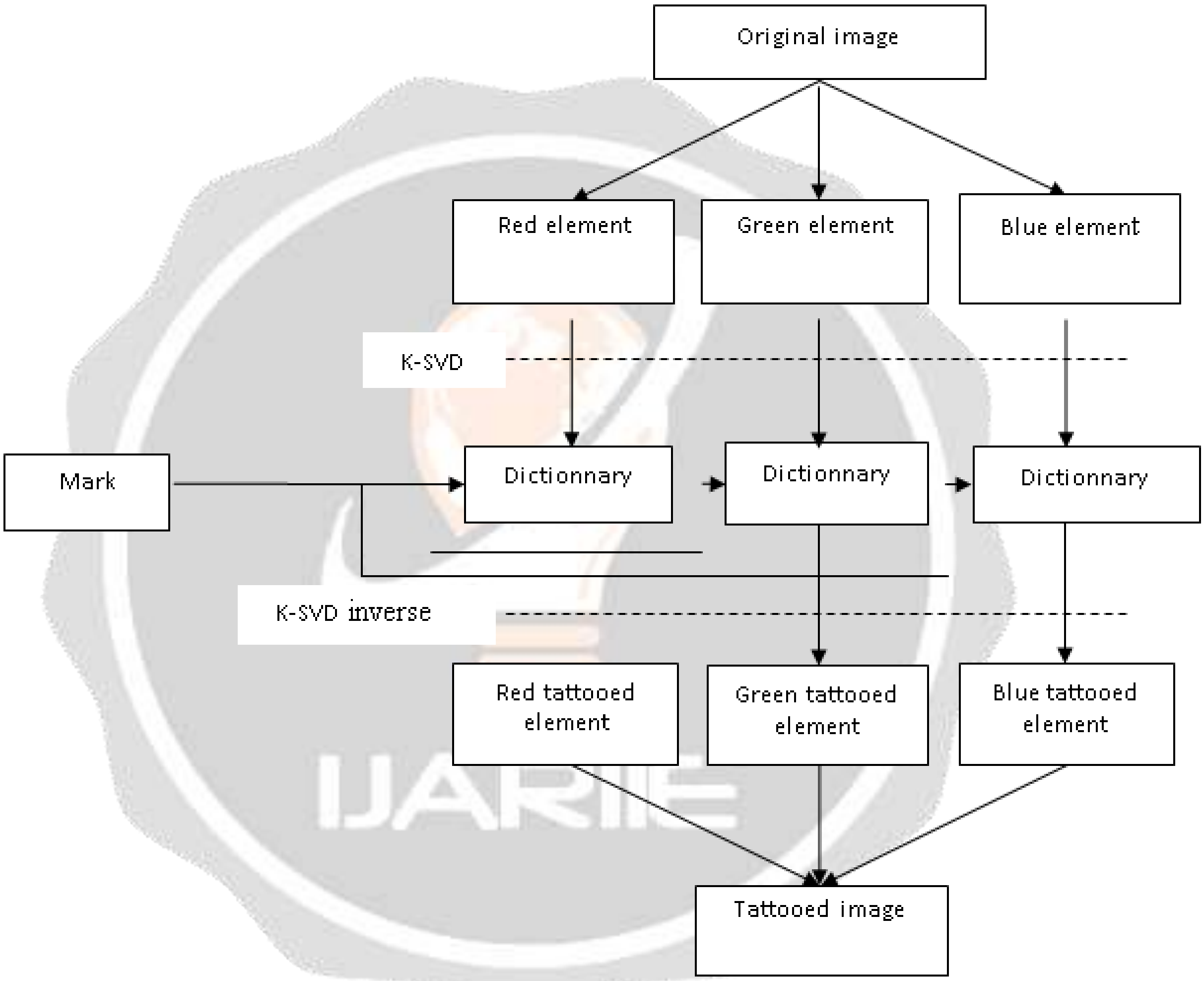


Fig - 2: Diagram of the different stages of insertion of the brandin the Ksvd domain

A dictionary is a set of elements $\{d_1, \ldots, d_k\}$ c IR^n , where k is greater than or equal to n. It is generally written in the form of a matrix D of dimensions n $\times$ k, and all its columns are $d_i$elements that we call "atoms". [2]
A learning dictionary is defined for a given input signal y_i, consisting of $\{y_1, \ldots, y_N\}$ elements, with $N >> K$, this makes it possible to produce a learning dictionary D^* where: $x_i^* \approx D^* y_i$ and x_i are parsimonious codes or sparse codes. [2] [3]

- Extraction of the mark

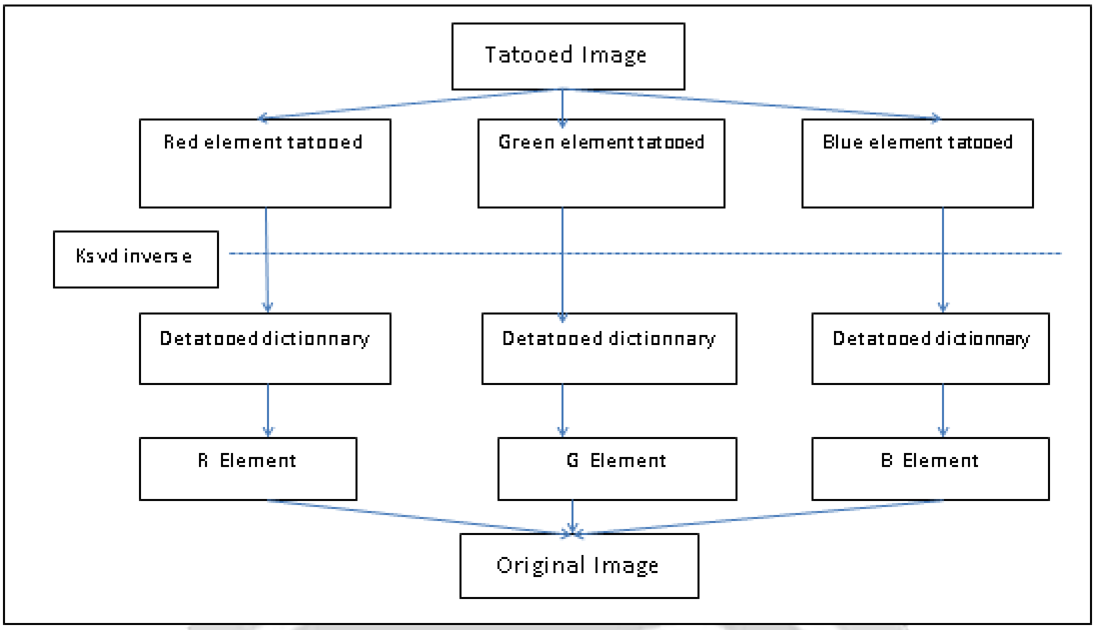


Fig - 3: Diagram of the different stages of extraction of the brandin the Ksvd domain

The extraction of the bits of the message according to the values of the predefined DCT coefficients:

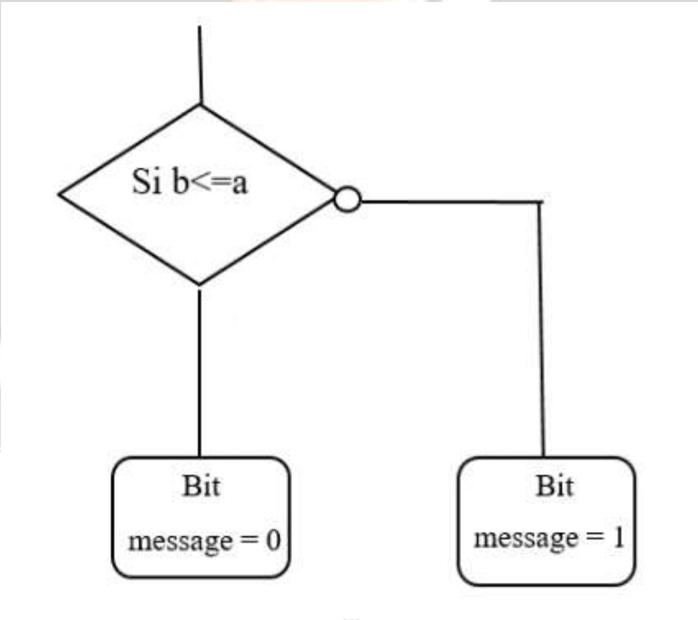


Fig - 4: Schema of extraction of the bits of the message

3.2 Algorithm using Ksvd

The reconstruction of the dictionary *D* is performed by minimizing the reconstruction error and satisfying the sparsity constraint.
The K-SVD algorithm is an iterative approach to minimize the energy of the previous equation and learns from the reconstructive learning dictionary for parsimonious representation of the signal. [2]

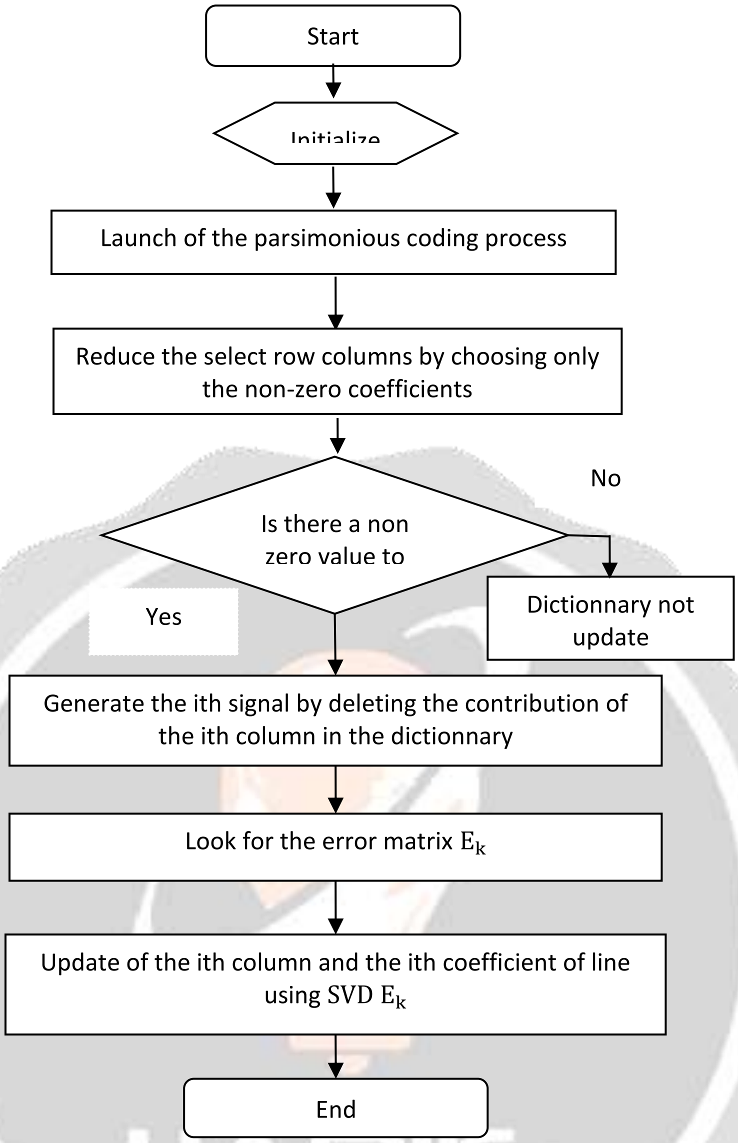


Fig - 5: K-SVD algorithm

The K-SVD algorithm is an algorithm for solving the parsimonious dictionary learning problem. For reasons of efficiency, an optimized version of the algorithm has recently been proposed: the OMP. [3][4]
The algorithm consists of an alternating optimization. Iteratively solve a partial problem in X, at D fixed. This step is called sparse coding step, with Frobenius's norm, in which we try to solve: [5] [6]

$$\min_X \{\|Y - DX\|_F^2\} \tag{1}$$

Such than: $\forall i \in \{1\,; N\}, \|x_i\|_0 \le T_0$
The second step of K-SVD is to optimize on D to X fixed. We therefore seek to solve:

$$\min_D \{\|Y - DX\|_F^2\} \tag{2}$$

The K-svd algorithm will successively optimize each atom of the dictionary d_k independently of the others, supposed to be fixed. The objective function is written as follows: [6] [7]

$$f(D) = \|Y - DX\|_F^2 = \left\| Y - \sum_{j=1}^{K} d_j x_T^j \right\|_F^2 \tag{3}$$

$$= \left\| \left(Y - \sum_{j \neq k}^{K} d_j x_T^j \right) - d_k x_T^k \right\|_F^2$$

$$f(D) = \left\| E_k - d_k x_T^k \right\|_F^2$$

Each product $d_j x_T^j$is a matrix of size nxN. The term E_k is the data reconstruction error made from the $K - 1$ elements of the dictionary$\{d_j\}$.
Since we suppose that these elements are fixed, we try to minimize $f(d_k)$. It is possible to minimize $f(d_k)$by least squares, but we want to ensure that the updated solution always allows parsimonious decomposition. To ensure this property, we will define: [8] [9]

$$w_k = i | 1 \leq i \leq N, x_T^k(i) \neq 0 \tag{4}$$

w_k represents the set of indices that use the atom d_k (those for which $x_T^k(i)$ = 0).

The method used to optimize d_k is then:

- We restrict the matrix E_k, to form E_k^R considering only the columns corresponding to w_k. If we write $\Omega_k$the size matrix $N \times |w_k|$ corresponding to copy N times the line vector w_k, we have:

$$E_k^R = E_k \Omega_k \text{ et } x_R^k = x_T^k \Omega_k \tag{5}$$

- The solution will actually minimize $\left\| E_k^R - d_k x_R^k \right\|_F^2$with respect to d_k and x_R^k,, ensuring that the support of x_T^kremains unchanged.[10] [11][12] [13]

4. CREATION OF DICTIONARY

Le principal problème dans le travail est la consommation en mémoire vive sur l'utilisation de Ksvd. Par exemple, avec un processeur core i5 à 4Go de RAM, pour construire un dictionnaire avec 100 itérations et 50000 signaux (patchs d'images), Matlab prend environ 20 minutes pour une seule itération. Ainsi, nous avons dû réduire les valeurs des paramètres qui vont malheureusement augmenter les différentes erreurs sur la construction des dictionnaires.
Nous allons commencer la construction et l'apprentissage du dictionnaire pour chaque composante. Avec une itération K=10 et un patch de 8 pixels, le dictionnaire ainsi obtenu est de taille, chacun de 8*64=572 pixels ou 572 atomes. On parvient aux résultats suivants pour chaque composante. Pour illustrer ceci, prenons l'exemple de la composante bleue.

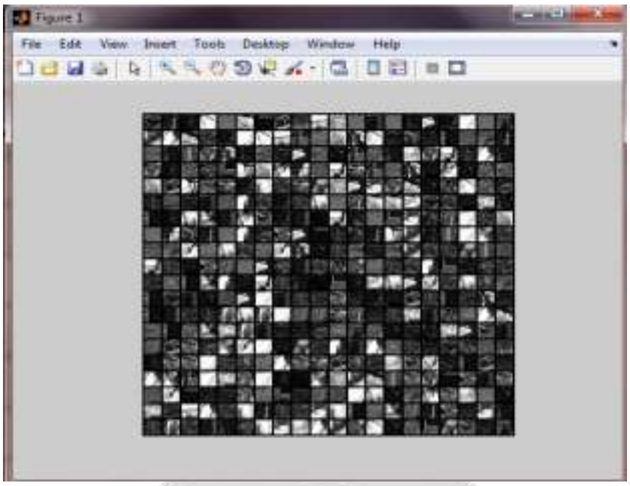


Fig - 6: Blue component image patches

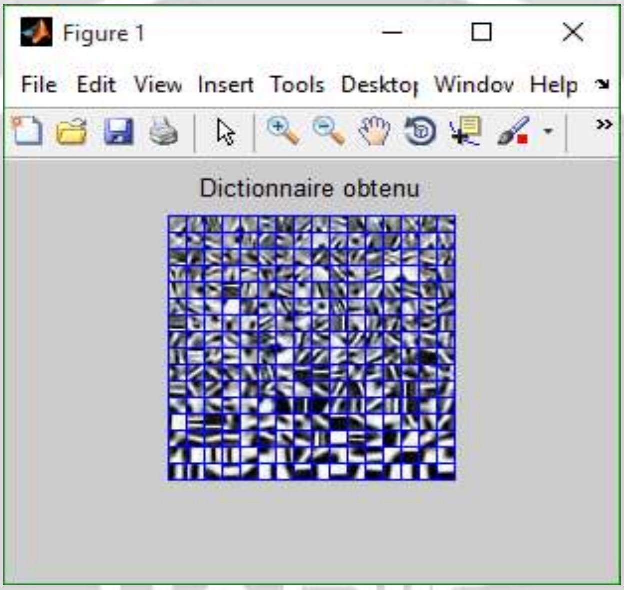


Fig - 7: Dictionary from the blue component

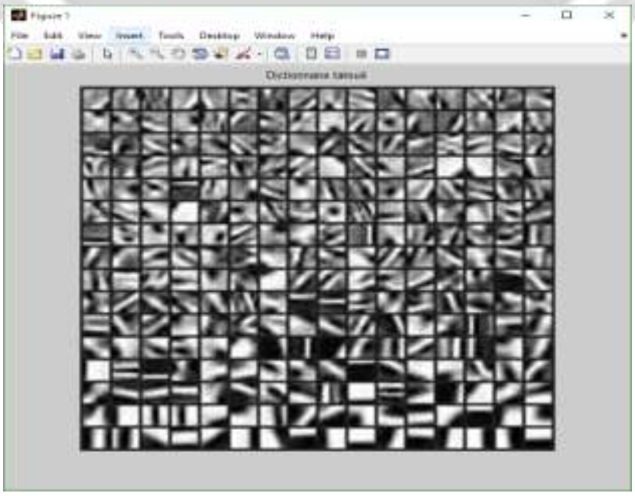


Fig - 8: Dictionary of the blue component tattooed by DCT

Once we have transformed the original image by KSVD, we go through the step of inserting a mark by the DCT algorithm of previous dictionaries. The tattooed dictionaries shown in Figure 8 are obtained.

5. RESULTS OF THE SIMULATION AND DISCUSSIONS

We will use the image of "Lena" as a host image size: 512 x 512 and the image "Copyright" that will be served as a trademark with a size of 256 x 256.

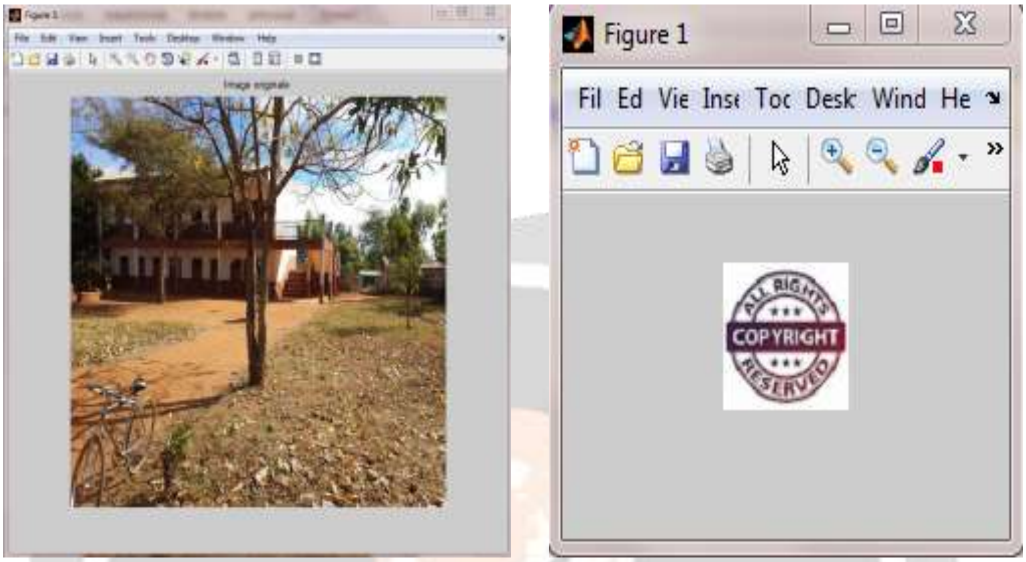


Fig - 9: Host image on the left and image mark on the right

Below we have the result, that is to say the original image and the tattooed image.

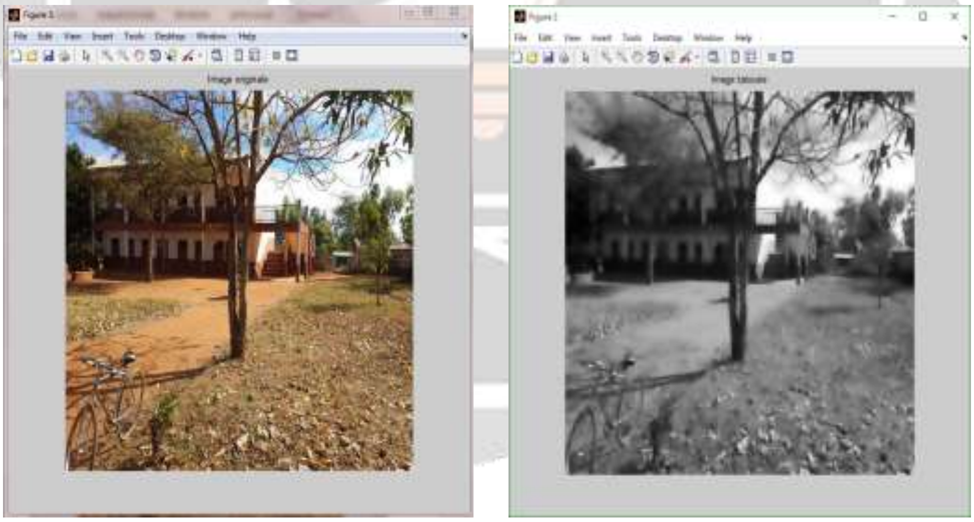


Fig - 10: Image original on the left, tattooed image on the right

To evaluate the tattoo, we will calculate the PSNR between the original image and the tattooed image, we got about 6dB. Note that this value is low for a digital tattoo. This value shows that the KSVD algorithm is not effective for use in a digital tattoo. The insertion of the mark in the dictionary will greatly affect the reconstruction of the image.

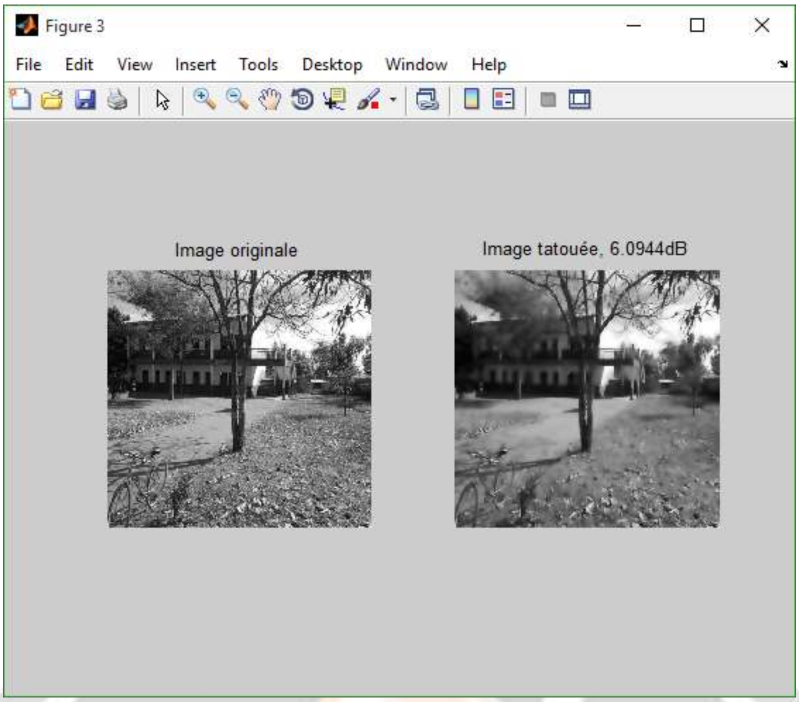


Fig - 11:PSNR between tattooed image and original image

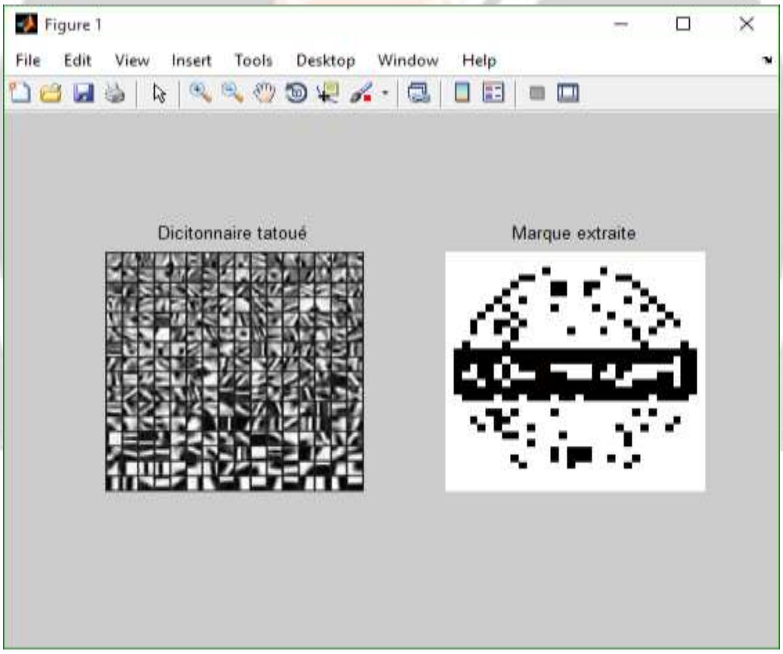


Fig - 12: Extraction of the mark of tattooed dictionary

To evaluate our tattooing methods we used different attacks, and we get the following figure:

The simulated attacks are:

- Image sound effects
- JPEG compression
- stetching

We will take an example of attack as compression, here is the result obtained by a 70% JPEG compression.

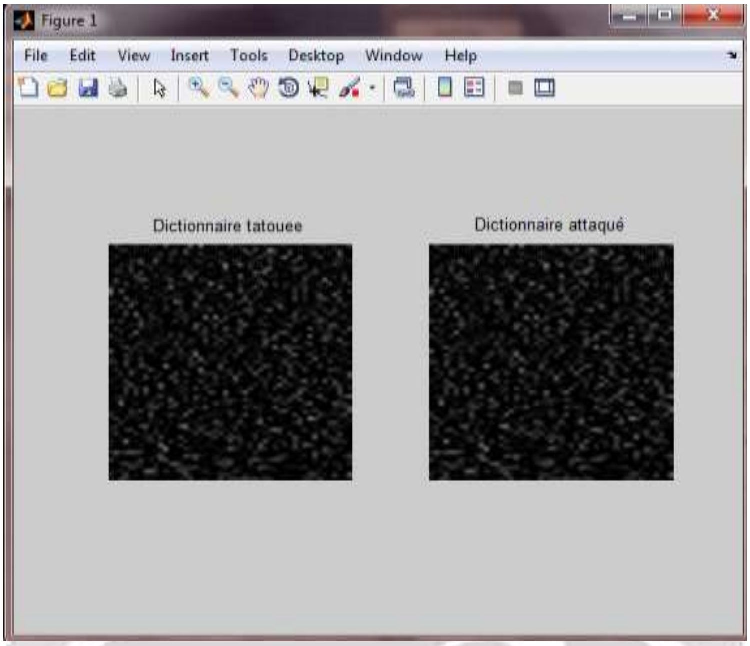
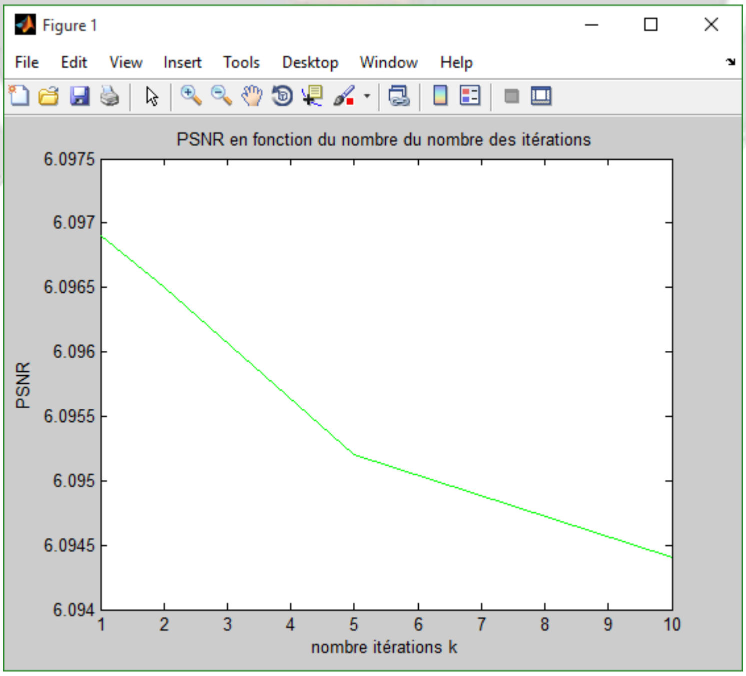


Fig - 13: Illustration of an attacked tattooed image

Table - 01: Comparative table of results in PSNR several experiments

k	1	2	5	10
PSNR	6.0969	6.0965	6.0952	6.0944

For the one same image, we have realised differents test, and so, we can see at the figure 14, the illustration with differents values of the iteration K corresponding with their PSNR. And for other image in the test, the PSNR give us a value near 6dB. As long the iteration number increase, more the PSNR's value decrease, but with minimum difference.



❖ **Fig - 14:** PSNR curve of the K-SVD according to the iteration K

6. CONCLUSION

As a conclusion, there are several possibilities of techniques to achieve a digital image tattoo. Those using transforms are the most used. Its most important feature is its reversibility. The tattoo model that we made is part of the category of invisible tattoos, that is to say imperceptible to the naked eye.

The marginal approach provides greater robustness to the digital tattoo image system because the attacks will have to reach each component of the tattooed image to modify or remove the mark. The dictionary formed by the K-svd algorithm in each component is then tattooed before reconstituting the final image that will contain the inserted mark.

Only authorized persons will know the tattoo algorithms used and will be able to use them to extract the mark and authenticate the originality of the image. Thus, this technique can be used in different fields such as for copyright on digital images published on the Internet. However, KSVD is most often used when denoising or compression, its use in the field of digital tattoo is brand new.

Also, the largest amount of information is needed. The greater the amount of information inserted, the toughness and imperceptibility will decrease. It is possible to find the best possible compromise between these three parameters depending on the application.

7. REFERENCES

[1] R. Rubinstein, T. Peleg, M. Elad, « *Analysis K-SVD: A Dictionary-Learning Algorithm for the Analysis Sparse Model* », 2011.

[2] M. A. Ferrari, M. N. Dobigeon « *Quelques applications du codage parcimonieux et sa mise en œuvre* », Université de Grenoble.

[3] S. Ayache, F. Xavier, « *Apprentissage de dictionnaire pour de la restauration et de la classification d’images* », 2015-2016.

[4] M. Aharon, M. Elad, A. Bruckstein, « *K-SVD:an algorithm for Designing Overcomplete Dictionaries for Sparse Representation* », November 2006.

[5] « *Apprentissage de dictionnaires image* », Université Pierre et Marie Curie Master 2 informatique,Spécialité IMA,2011.

[6] M.Aharon,« *K-SVD:Designof dictionaries for sparse representation*», Israel Institute of Technology, Haifa 32000, Israel, 2009.

[7] H.P. Dang, P. Chainais, « *Une approche bayésienne non paramétrique pour l’apprentissage d’un dictionnaire de taille adaptive* », 2009, Ecole Centrale Lille.

[8] M. Elad, O. Bryt, « *Compression of facial images using the K-SVD algorithm* », 2008.

[9] J. Landré, « *Une petite introduction aux représentations parcimonieuses de signaux et d’images* », Université de Reims Champagne-Ardenne, 2008.

[10] J. Aghaei, C. Labit, « *Représentations parcimonieuses par un dictionnaire à structure adaptative* », Oct 2013.

[11] « *Apprentissage de dictionnaires parcimonieux utilisant l’algorithme K-SVD* », M2/TASI, ESPA, AU 2016/2017

[12] L. Xu, «*A Study of the K-SVD Algorithm For Designing Overcomplete Dictionaries for Sparse Representation* », March 2012.

[13] M.Lebrun,« *An implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm* », 2012.