Fuzzy Transforms and Fragile Watermarking Tamper Detection on Coded Images

Ferdinando Di Martino Salvatore Sessa

Università degli Studi di Napoli Federico II, Dipartimento di Costruzioni e Metodi Matematici in Architettura, Via Monteoliveto 3, 80134 Napoli, Italy

Abstract

An original image is compressed with fuzzy transforms and is divided in images of sizes 2×2 on which we apply a known fragile watermarking process. A preprocessing phase is considered to determine the better compression rate for the coding process. We test this scheme in tamper detection analysis on a sample of images. The results are good in terms of accuracy for tamper detection with compressed images and in terms of dimension of the image dataset used for storage of the original images.

Keywords: fragile watermarking, fuzzy C-Means, fuzzy transform, tamper detection

1. Introduction

The sensibility of the fragile watermarking algorithms to the destruction of marks by attacks makes easier the detection and localization of tampered zones. Then many fragile watermarking algorithms have been proposed in literature (cfr., for example, [1, 3, 4, 5, 9, 13, 15, 16, 17, 24, 28]). The insertion of the watermark, the tamper detection and the tamper localization are the three procedures which compose a fragile watermark and below described.

1) The insertion of the watermark is a procedure which applies a secret key on the source image before its distribution. The marked image is substantially identical to the original image for an external user.

2) The tamper detection is based principally on statistical processes and can be tested on a sample image suitably modified, measuring the True (resp. False) Positive (TP, resp. FP) rate on the tampered image, given from the proportion of pixels detected as tampered with respect to the pixels really tampered (resp., from the proportion of non tampered pixels wrongly detected and considered as tampered from the algorithm).

.

3) The tamper localization locates the tampered areas on the image. A result of the tamper localization process can be a two-level image showing the ground of the tampered regions.

The image can be distributed after the insertion of the watermark.

The source image must be stored in an image dataset to be used for successive operations of tamper detections and localizations. In some applications the owner who marks the image can be different by who makes both detection and localization. To determine if an image has been tampered, it is necessary to extract the source image from the image dataset applying the marking key. Then the tamper detection makes a comparison of the marked image with the image suspected to have been altered and the tamper localization finds the tampered zones. The dimension of the image dataset containing the original images can be very large and can grow quickly in the course of time. Some authors [9, 25, 26, 27] explore the idea of marking compressed images. In [25] the authors experiment a simple watermarking scheme on images compressed in the transform domain. In [26] the authors propose a watermarking scheme based on table lookup in frequency domain in which the marked image is stored in compressed form. In [9] the authors apply the watermark on images coded via fuzzy relation equations [7, 8, 12, 18].

Our scope is to experiment an efficient application in which the watermark is inserted on compressed images on which both tamper detection and localization are performed. The main advantage of this process consists in the reduction of the memory necessary to the storage of the image dataset. In Fig.1 we schematize the process which performs the storage of the compressed image dataset:

- the coding image realizes the compression;

- the watermark insertion marks the coded image which is stored in the new compressed image dataset;

- finally, the marked coded image is decompressed and ready to be distributed.

1.1. Tamper analysis

In Figure 2 we schematize the process of the tamper analysis. The input of the tamper detection and the localization functions are the image supposed tampered and the corresponding marked compressed source image which is searched in the image dataset. The two input compressed images are compared from the tamper detection function (which calculates the TF and FP indexes above defined) and from the tamper

localization function, which produces the two levels of the tamper localization image.

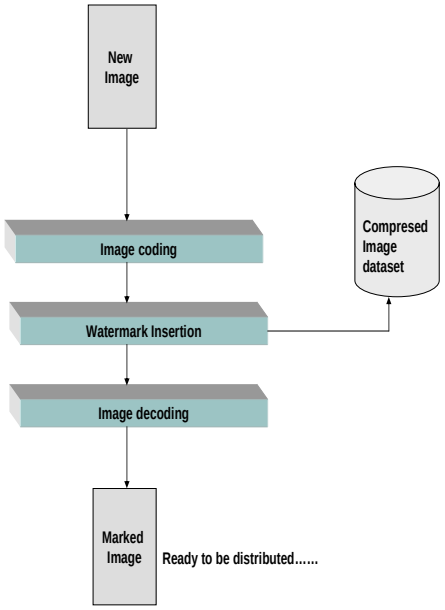

Fig. 1: Coded watermarked images and their storage

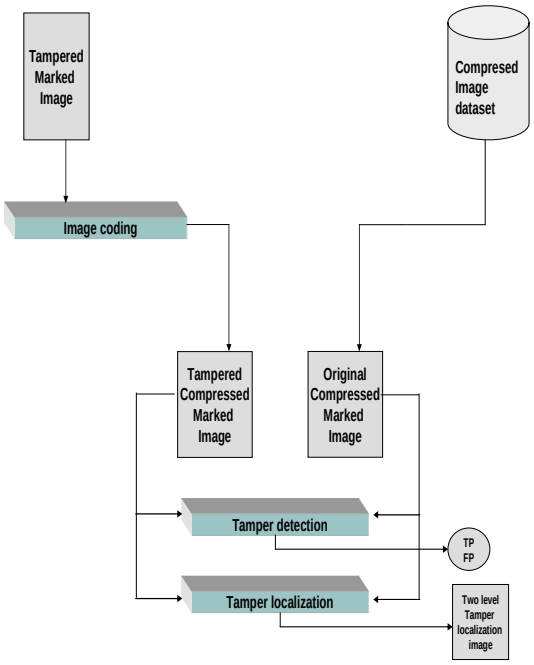

Fig. 2: Process of the tamper analysis

We have used the Fuzzy Transforms (shortly, F-transforms) [10, 11, 22] for coding/decoding images. In [11] we showed that the F-transforms give better results with respect to the fuzzy relation equations and give results comparable with the ones obtained by using the JPEG compression method in terms of image quality and coding/decoding time. The watermark insertion is made with the block-based watermarking scheme described in [5]. In this method a fuzzy partition of the blocks is performed by using the Fuzzy C-Means (shortly, FCM) algorithm [2] to realize the blockwise independency and a relation between blocks. An authentication data is generated for each block by using a pseudo random sequence seeded with a secret key.
In [5] the authors shown that the blockwise independency watermarking scheme can detect tamper on various types of attack, as "cut and paste" and "vector quantization counterfeiting" attacks. The authors show that the tamper detection results are better that those ones obtained with the approaches described in [4,16].

Here we experiment the watermarking architecture described in Figure 2. In the marker insertion process the block based watermarking scheme is applied to the blocks (which are images of sizes 2×2) of the image coded via the F-transforms.
To determine the better compression rate to be used in the image marking process, we realize a pre-processing phase in which we analyze the trend of the Peak Signal to Noise Ratio (shortly, PSNR) with respect to the compression rate of images coded, marked and decoded as in Figure 1. We consider a threshold value of the PSNR as that value such that the corresponding Root Mean Square Error (shortly, RMSE) is increased by a factor 2.5 with respect to the PSNR obtained by marking directly the original image. For compressions in which the resulting PSNR is less than the threshold value, we consider the RMSE so high to invalidate both tamper detection and localization analysis. Many tests are performed for both tamper detection and localization processes and for different types of manipulations. The blockwise independency watermarking scheme [5] is applied to the compressed source image; the tampered image is compressed and compared with the coded source marked image.
Section 2 describes the image compression method realized with the F-transforms, in Section 3 we recall briefly the FCM method. In Section 4 we describe the Chen and Wang [5] watermarking scheme and in Section 5 we present our fragile watermarking process. Section 6 contains the results of our tests made on a well known image and final considerations are reported in Section 7.

2. Coding/decoding by F-transforms

We recall some main definitions of F-transforms [22]. Let $n \geq 3$ and $x_1, x_2, \ldots, x_n$ be points of the interval $[a,b]$ such that $x_1 = a < x_2 < \ldots < x_n = b$. We say that the fuzzy sets $A_1,\ldots,A_n : [a,b] \rightarrow [0,1]$ form a fuzzy partition of $[a,b]$ if

(1) $A_i(x_i) = 1$ for every $i = 1,2,\ldots, n$;

(2) $A_i(x) = 0$ if $x \notin (x_{i-1}, x_{i+1})$ for every $i = 2,\ldots, n-1$;

(3) $A_i(x)$ is a continuous function on $[a,b]$;

(4) $A_i(x)$ is strictly increasing on the interval $[x_{i-1}, x_i]$ for $i = 2, \ldots, n$ and is strictly decreasing on the interval $[x_i, x_{i+1}]$ for $i = 1,\ldots, n-1$;

(5) for every $x \in [a,b]$, $\sum_{i=1}^{n} A_i(x) = 1$.

If the following additional properties hold:

(6) $x_i = a+h\cdot(i-1)$ for i=1, 2,…, n, where h= (b-a)/(n-1) (thus the nodes are equidistant),

(7) $A_i(x_i - x) = A_i(x_i + x)$ for $x \in [0,h]$ and i=2,…, n-1,

(8) $A_{i+1}(x) = A_i(x - h)$ for $x \in [x_i, x_{i+1}]$ and i=1,2,…, n-1,

then we say that the fuzzy sets $\{A_1,\ldots, A_n\}$ constitute an uniform (or symmetric) fuzzy partition. We limit ourselves to consider only the discrete case. Let $m \geq 3$, $y_1, y_2, \ldots, y_m \in [c,d]$ be m assigned points such that $y_1 = c < \ldots < y_m = d$. Let $B_1,\ldots,B_m : [c,d] \rightarrow [0,1]$ be a fuzzy partition of [c,d] and f be a function in two variables assuming prefixed values in the finite set $P \times Q = \{p_1,\ldots,p_N\} \times \{q_1,\ldots,q_M\} \subseteq [a,b] \times [c,d]$, where P (resp. Q) is sufficiently dense with respect to the chosen fuzzy partition $\{A_1, A_2, \ldots, A_n\}$ of [a,b] (resp. $\{B_1,\ldots,B_m\}$ of [c,d]), that is if N>n (resp. M>m) and for each k = 1,…,n (resp., l=1,…,m) there exists at least an index $i \in \{1,\ldots,N\}$. (resp., $j \in \{1,\ldots,M\}$) such that $A_k(p_i) > 0$ (resp. $B_l(q_j) > 0$). Then the n × m fuzzy matrix $[F_{kl}]$ is defined as the direct F-transform of f with respect to $\{A_1,\ldots, A_n\}$ and $\{B_1,\ldots,B_m\}$ if

$$F_{kl} = \frac{\sum_{j=1}^{M}\sum_{i=}^{N} f(p_i,q_j)A_k(p_i)B_l(q_j)}{\sum_{j=1}^{M}\sum_{i=}^{N} A_k(p_i)B_l(q_j)} \qquad (1)$$

for k = 1,…,n and l = 1,…,m. Afterwards we can define the inverse F-transform of f with respect to $\{A_1, A_2, \ldots, A_n\}$ and $\{B_1,\ldots,B_m\}$ to be the function $f^F_{nm}: (p_i,q_j) \in P \times Q \rightarrow f^F_{nm}(p_i,q_j) \in [0,1]$ defined as

$$f^F_{nm}(p_i,q_j) = \sum_{k=1}^{n}\sum_{l=1}^{m} F_{kl}A_k(p_i)B_l(q_j) \qquad (2)$$

As in [22], Theorem 5, it is possible to prove that the following result:

Theorem . Let f(x,y) be a function assigned over the set $P \times Q = \{p_1,\ldots,p_N\} \times \{q_1,\ldots,q_M\} \subseteq [a,b] \times [c,d]$. Then for every $\varepsilon > 0$, there exist two integers $n(\varepsilon)$, $m(\varepsilon)$ with $n(\varepsilon) < N$, $m(\varepsilon) < M$ and related fuzzy partitions $\{A_1, A_2, \ldots, A_{n(\varepsilon)}\}$ of [a,b] and $\{B_1, B_2, \ldots, B_{m(\varepsilon)}\}$ of [c,d] such that P and Q are sufficiently dense with respect to $\{A_1, A_2,\ldots,A_{n(\varepsilon)}\}$ and $\{B_1, B_2,\ldots,B_{m(\varepsilon)}\}$, respectively, and for every $(p_j,q_j) \in P\times Q$, $i \in \{1,\ldots,N\}$ and $j \in \{1,\ldots,M\}$, the following inequality holds:

$$\left| f(p_i,q_j) - f^F_{n(\varepsilon)m(\varepsilon)}(p_i,q_j) \right| < \varepsilon \qquad (3)$$

Let R be a gray image of sizes M × N, with R(i,j)= P(i,j)/Lt, i.e. we consider it as a fuzzy relation R: $(i,j) \in \{1,\ldots,M\}\times\{1,\ldots,N\} \rightarrow [0,1]$, R(i,j) being the normalized value of the pixel P(i,j) with respect to the length Lt of the gray scale. For brevity of notation, we put $p_i = i$, $q_j = j$, a = c = 1, b = N and d = M. Moreover, we define the fuzzy sets $A_1,\ldots,A_m : [1,M] \rightarrow [0,1]$ (resp., $B_1,\ldots,B_n : [1,N] \rightarrow [0,1]$) with m < M (resp., n < N), forming a fuzzy partition of [1,M] (resp., [1,N]). Hence R is divided in submatrices R_D of sizes M(D)×N(D), $R_D : (i, j) \in \{1,\ldots,M(D)\}\times\{1,\ldots,N(D)\} \rightarrow R_D(i, j) \in [0,1]$, called blocks compressed to other submatrices of sizes m(D)×n(D) (with m(D)< M(D), n(D)<N(D)) via the direct F-transform $[F^D_{kl}]$ given by

$$F^D_{kl} = \frac{\sum_{j=1}^{N(D)}\sum_{i=1}^{M(D)} R_D(i,j)A_k(i)B_l(j)}{\sum_{j=1}^{N(D)}\sum_{i=1}^{M(D)} A_k(i)B_l(j)} \qquad (4)$$

for each k = 1,…, m(D) and l = 1,…, n(D). Successively we decompress the above blocks with the inverse F-transform $R^F_{m(D)n(D)} : (i, j) \in \{1,\ldots,M(D)\}\times\{1,\ldots,N(D)\} \rightarrow R^F_{m(D)n(D)}(i, j) \in [0,1]$ defined as

$$R^F_{m(D)n(D)}(i,j) = \sum_{l=1}^{n(D)}\sum_{k=1}^{m(D)} F^D_{kl}A_k(i)B_l(j) \qquad (5)$$

which approximates R_D in the sense of the above Theorem. Certainly two integers n(D)=n(D,ε), m(D)=m(D,ε) exist for every block R_D and $\varepsilon = \varepsilon(D) > 0$ such that the inequality $\left|R_D(i,j) - R^F_{m(D)n(D)}\right| < \varepsilon$ holds true for every $(i, j) \in \{1,\ldots,M(D)\}\times\{1,\ldots,N(D)\}$, but the cited Theorem does not give a method which determines such integers m(D) and n(D). Thus we assign several values to m(D) and n(D) with m(D) < M(D) and n(D) < N(D), hence we have various values of the compression rate ρ(D) = (m(D)·n(D))/(M(D)·N(D)). The experiments of [10, 11] have shown that the best performances are obtained with the following fuzzy sets $A_1,\ldots,A_{m(D)} : [1,M(D)] \rightarrow [0,1]$ and $B_1,\ldots,B_{n(D)} : [1,N(D)] \rightarrow [0,1]$ (forming a symmetric fuzzy partition):

$$A_1(i) = \frac{1}{2}\cos\left(\frac{\pi}{h}(i-1)+1\right) \text{ if } i \in [1,x_2]$$
or $A_1(i) = 0$ otherwise, (6)

$$A_k(i) = \frac{1}{2}\cos\left(\frac{\pi}{h}(i-x_k)+1\right) \text{ if } i \in [x_{k-1},x_{k+1}]$$
or $A_k(i) = 0$ otherwise, (7)

$$A_{m(D)}(i) = \frac{1}{2}\cos\left(\frac{\pi}{h}(i-x_{m(D)-1_h})+1\right) \text{ if}$$
$i \in [x_{m(D)-1},M(D)]$ or $A_{m(D)}(i)=0$ otherwise (8)

and moreover

$$B_1(j) = \frac{1}{2}\cos\left(\frac{\pi}{s}(j-1)+1\right) \text{ if } j \in [1,y_2]$$
or $B_1(j) = 0$ otherwise, (9)

$$B_t(j) = \frac{1}{2}\cos\left(\frac{\pi}{s}(j - y_t) + 1\right) \quad \text{if } j \in [y_{t-1}, y_{t+1}]$$

or $B_t(j) = 0$ otherwise, (10)

$$B_{n(D)}(j) = \frac{1}{2}\cos\left(\frac{\pi}{s}(j - y_{n(D)-1}) + 1\right)$$

if $j \in [y_{n(D-1)}, N(D)]$ or $B_{n(D)}(j)=0$ otherwise, (11)

where $k = 2,\ldots, m(D)$, $h = (M(D)-1)/(m(D)-1)$, $x_k = 1 + h\cdot(k-1)$, $t = 2,\ldots, n(D)$, $s = (N(D)-1)/(n(D)-1)$, $y_t = 1 + s\cdot(t-1)$.

3. An FCM's overview.

The FCM is the most known fuzzy clustering method [2]. Let $X=\{\mathbf{x_1},\ldots,\mathbf{x_N}\}\subset R^n$ be the dataset composed by N vectors in R^n defined with the following matrix:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1N} \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ x_{n1} & x_{n2} & \ldots & x_{nN} \end{pmatrix} \quad (12)$$

where $\mathbf{x_j} = (x_{1j}, x_{2j},\ldots,x_{nj})^T$ is the j-th feature vector for $j = 1,\ldots, N$. We must minimize the following objective function:

$$J(\mathbf{X},\mathbf{U},\mathbf{V}) = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m d_{ij}^2 \quad (13)$$

where u_{ij} is the degree to which $\mathbf{x_j}$ belongs to the i - th cluster, $I = 1,\ldots,C$, $V = \{\mathbf{v_1},\ldots,\mathbf{v_C}\} \subset R^n$ is the set of the centers of the C clusters (point prototypes) represented by the matrix:

$$V = \begin{pmatrix} v_{11} & v_{12} & \ldots & v_{1C} \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ v_{n1} & v_{n2} & \ldots & v_{nC} \end{pmatrix}, \quad (14)$$

d_{ij} is the distance of $\mathbf{v_i} = (v_{1i}, v_{2i},\ldots,v_{nj})^T$ from $\mathbf{x_j}$ calculated by using a specific metric and $m \in [1,+\infty)$ (usually, m=2) is the "fuzzifier" parameter which determines the degree of fuzziness of the partition. The distance d_{ij} is given by

$$d_{ij} = \sqrt{(\mathbf{x_j} - \mathbf{v_i})^T \mathbf{N} (\mathbf{x_j} - \mathbf{v_i})} \quad (15)$$

where $\mathbf{N}$ is a positive and symmetric norm matrix. Moreover the following assumptions are made:

$$\sum_{i=1}^{C} u_{ij} = 1 \qquad \forall\, j \in \{1,\ldots,N\} \quad (16)$$

$$0 < \sum_{j=1}^{N} u_{ij} < N \qquad \forall\, i \in \{1,\ldots,C\} \quad (17)$$

The search of a solution for Equation (10), subject to the constraints (13) and (14), with Lagrange method, leads to the following

$$\mathbf{v_i} = \frac{\sum_{j=1}^{N} u_{ij}^m \mathbf{x_j}}{\sum_{j=1}^{N} u_{ij}^m} \quad (18)$$

for $i = 1,\ldots,C$. Thus we deduce that

$$u_{ij} = \frac{1}{\left(\sum_{k=1}^{c} \frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{2}{m-1}}} \quad (19)$$

The iterative process stops at the s-th iteration if

$$\left\|\mathbf{U}^{(s)} - \mathbf{U}^{(s-1)}\right\| = \max_{i,j}\left|u_{ij}^{(s)} - u_{ij}^{(s-1)}\right| < \in \quad (20)$$

where $\varepsilon > 0$ is a parameter assigned a priori to stop the iteration process and $\mathbf{U}^{(s)} = \left(u_{ij}^{(s)}\right)$ is the matrix $\mathbf{U}$ of the membership degrees calculated at the s-th step.

4. The Chen and Wang algorithm

In the Chen and Wang watermarking algorithm proposed in [5], an image R with M×M pixels is considered in a 8-bit gray scale and is divided in blocks of sizes 2×2. The j-th block is considered as a fourth dimensional feature vector $\mathbf{X_j}$ with components $(x_{1j}, x_{2j},\ldots,x_{nj})^T$ which are its pixel values. The FCM algorithm is applied to the dataset formed from $(M/2)^2$ blocks of the image. In the resulting membership matrix $\mathbf{U}$ of dimensions $C\times(M/2)^2$, where C is the number of clusters, the membership values of each column $(u_{1j}, u_{2j},\ldots,u_{Cj})^T$, $j = 1,\ldots, (M/2)^2$, are rearranged in a descending order for obtaining a new matrix $\hat{\mathbf{U}}$, where the j-th column is given by $\left(\hat{u}_{1j}, \hat{u}_{2j},\ldots,\hat{u}_{Cj}\right)^T$ with $\hat{u}_{hj} \geq \hat{u}_{kj}$ if $h < k$. Then we obtain the following value for each j-th column:

$$f_j = (\hat{u}_{1j} - \hat{u}_{Cj}) \cdot Lt \quad (21)$$

where $\hat{u}_{1j}$ and $\hat{u}_{Cj}$ are the maximum and minimum values of the j-th column in the matrix $\hat{\mathbf{U}}$, respectively. The generated sequence $F = (f_1, f_2,\ldots, f_{(M/2)^2})$ is used to derive the 8-bit authentication data embedded in the 8 Least Significant Bit (shortly, LSB) for each image block. To this aim the random sequence $R = (r_1, r_2,\ldots, r_{M/2^2})$, $r_j \in \{0,1,\ldots,Lt\}$, is considered by creating a Pseudorandom Number Generator (shortly, PRNG) seeded with a secret key (shortly, SK). For each block of sizes 2×2, the corresponding authentication data is constructed as

$$a_j = f_j \oplus r_j \qquad (22)$$

where the operator $\oplus$ is the XOR operator. Each two bit couple in the 8 bit authentication data a_j is embedded in the two LSB's of the corresponding pixel value in the block of sizes 2×2. Strictly speaking, the authentication data embedding method [5] can be divided in the following seven steps:

1) The original image is divided in $(M/2)^2$ blocks of sizes 2×2; the two LSB's of each block are set to zero.

2) Each block is considered as a fourth dimensional feature vector of a dataset of dimension $(M/2)^2$; the FCM algorithm is used on this dataset determining the matrix **U** of dimensions $C\times(M/2)^2$, where C is the number of clusters.

3) The matrix **U** is changed in a matrix $\hat{\mathbf{U}}$ in which the membership values of the C columns are rearranged in a decreasing order.

4) The sequence $F = (f_1, f_2, ..., f_{(M/2)^2})$ is generated, where f_j is obtained with (21).

5) The random sequence $R = (r_1, r_2, ..., r_{(M/2)^2})$, $r_j \in \{0,1,...,Lt\}$, is generated using a PRNG seeded with a SK.

6) For each block is generated the authentication data a_j with (22).

7) The marked image is obtained embedding each authentication data in the 8 LSB's of the image blocks.

Tamper detection is made by marking the tampered image with the above authentication data embedding method and by comparing it with the marked original image. Numerous tests [5] show that this method provides more accurate tamper detection and localization than other fragile watermarking methods [4, 16, 17].

5. F-transforms and fragile watermarking

We consider an M×N input image compressed with the F-transforms: as described in Section 2, the original image is normalized in a matrix R, hence it is divided in blocks R_D of sizes M(D)×N(D), where we have that $R_D : (i,j) \in \{1,\ldots,M(D)\times\{1,\ldots,N(D)\} \to R_D(i,j) \in [0,1]$.

Each block R_D is compressed to blocks of sizes m(D)×n(D) (with m(D) < M(D), n(D) <N(D)) via the direct F-transform defined from (21). Without loss of generality, we choice square blocks of dimension M(D) = N(D) = K compressed to blocks of dimension m(D) = n(D) = 2, in order to apply the Chen and Wang authentication data embedding method to the compressed image. To determine the better compression rate, that is to set the value of K, we consider a pre-processing phase in which the image is compressed with a greater compression rate; we plot the PSNR with respect to the compression rate, considering acceptable a value for K such that the corresponding PSNR don't have an appreciable fall. The PSNR is defined as

$$PSNR = 20\log_{10}\frac{Lt}{RMSE} \qquad (23)$$

where RMSE is given from the following formula:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(R(i,j) - R^F(i,j)\right)^2}{M\times N}} \qquad (24)$$

In such formula R(i,j) is the pixel value of the matrix R in the i-th row and j-th column and $R^F(i,j)$ is the same pixel value of the decoded image R^F. To determine the optimal value of K we consider the $(PSNR)_0$ deduced by applying the mark directly on the original image. The experiments have suggested to calculate a threshold for the PSNR, namely $(PSNR)_1$; beyond this threshold the loss of data is considered too high and consequently the tamper detection and localization processes could not be validated. Our tests suggest as threshold that value of $(PSNR)_1$ for which the corresponding Root Mean Square Error $(RMSE)_1$ is increased by 2.5 times with respect to the $(RMSE)_0$ corresponding to $(PSNR)_0$, that is $(RMSE)_1 = 2.5\cdot(RMSE)_0$. Thus we have the following formula:

$$(PSNR)_1 = \frac{1}{\log_{10}\left(\frac{1}{2.5}\right)}\left(\frac{(PSNR)_0}{1-(PSNR)_0}\right) \qquad (25)$$

Strictly speaking, our F-transforms authentication data embedding method consists of the following steps:

1) The parameter K is chosen by coding the source image with many compression rates ρ and analyzing the trend of the PSNR with respect to ρ; we take a value of K such that the corresponding compression rate $\rho = 4/K^2$ produces also an acceptable decompressed image, that is the PSNR is not greater then the value $(PSNR)_1$ given from (22).

2) The original image is compressed using the F-transforms. The image R of sizes M×N is divided in (M/K)×(N/K) blocks R_D of dimensions M(D) = N(D) = K. Each block R_D of the image is compressed in a block of sizes m(D)=n(D)=2; the resulting compression rate is given by $\rho = 4/K^2$. Note that the value of K is such that the number of blocks N_{blocks} = (M/K)×(N/K) is an integer.

3) The Chen and Wang authentication data embedding method [5] is applied on the compressed image. The dimensions of the resulting membership values of the matrix **U** in the FCM algorithm applied to the compressed image is C×(M/K)×(N/K), where C is the number of clusters. Applying the steps described in Section 4, the marked image is obtained embedding each authentication data in the 8 LSB's of the blocks of the compressed image.

In the tamper detection and localization processes the tampered image is compressed and compared with the marked original image, both coded with the F-transforms and a compression rate $\rho = 4/K^2$. Note that the compressed original image is marked by using the above steps 2 and 3, moreover the TP and FP indexes are determined and the tampered zone is localized.

6. Results

Without loss of generality, here we assume that all the images are represented from square matrices, that is M=N, and moreover suppose that Lt=255.

We take from well known datasets the color image "Baboon" of sizes M=N=512 shown in Figure 3; in the Figure 3a (resp. 3b, 3c) the image is shown in the band R (resp., G, B).

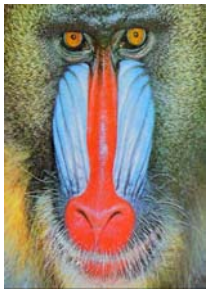

Fig. 3: Original image "Baboon"

Fig. 3a: R band

Fig. 3b: G band

In Figure 4 the original image of Figure 3 has been tampered; Figure 4a (resp., 4b, 4c) shows the tampered image in the band R (resp., G, B). In the pre-processing phase we have marked the original image in Figure 3 calculating the related $(PSNR)_0$ and the threshold $(PSNR)_1$ via formula (22) for each band and hence making the arithmetical mean of the respective corresponding values. Since no misunderstanding can arise, we continue to denote with $(PSNR)_0$ and $(PSNR)_1$ such mean values. More generally, from now on, with PSNR we denote the mean arithmetical of the PSNR

Fig. 3c: B band

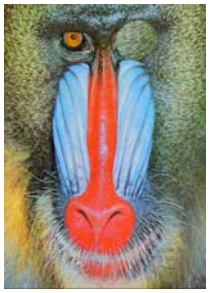

Fig. 4: The tampered image "Baboon"

Fig. 4a: Tamper in R band

Fig. 4b: Tamper in G band

Fig. 4c: Tamper in B band

calculated in the three bands for every color image.
Then we have applied the mark on the image coded with various compression rates ρ in order to set the better value of K. Figure 5 shows the trend of the PSNR with respect to ρ.

In order to apply the Chen and Wang fragile watermarking scheme [5], we must express K as a power of 2

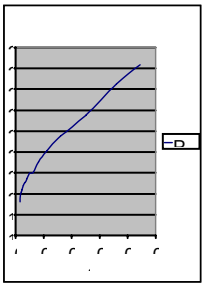

Fig. 5: PSNR with respect to ρ for "Baboon"

to get a value integer for the number of blocks N_{blocks}. and clearly we assume K= ∞ for ρ = 0.

In Table 1 we report the results for K = ∞,4,8,16, the respective compression rates, N_{blocks} and PSNR.

K	ρ	N_{blocks}	PSNR
∞	0.000000	65536	42.683
4	0.250000	16384	23.752
8	0.062500	4096	21.081
16	0.015625	1024	19.620

Table 1. PSNR and compression rates for "Baboon"

For ρ = 0, we have $(PSNR)_0$ = 42.683, then we obtain a value $(PSNR)_1$ = 23.08. Therefore the optimal value is K = 4 (ρ = 0.25) to which it is correspondent a PSNR still greater than the threshold $(PSNR)_1$. Then we test the process described in Figure 2 by studying the tampered image of Figure 4. In Table 2 we show the TP and FP indexes obtained in the three bands; for completeness, we also show the results obtained for K = 8, 16; we compare these results with those ones obtained by applying the method [5] directly on the original image.

K	ρ	TP R Band	FP R Band	TP G Band	FP G Band	TP B Band	FP B Band
∞	0.000000	99.425	0.049	99.332	0.042	99.418	0.032
4	0.250000	95.491	0.073	94.254	0.064	95.389	0.065
8	0.062500	94.189	0.078	94.467	0.069	94.920	0.067
16	0.015625	97.867	0.492	98.592	0.475	97.389	0.451

Table 2. FP and TP in the three bands for "Baboon"

In Table 3 we report the mean values of the TP and FP indexes; for completeness of presentation, we give the mean values of the True Negative (TN) and False Negative (FN) indexes, the mean sensitivity and specificity indexes (for simplicity, we denote all these mean values with the same symbols) defined as

$$sensitivity = \frac{TP}{TP + FN} \qquad (26)$$

$$specificity = \frac{TN}{TN + FP} \qquad (27)$$

7. Conclusions

In this work the block-based fragile watermarking has been applied on images compressed via the F-transforms by using blocks of sizes 2×2, marked with the scheme of [5] and stored in the image dataset. Our method preserves the benefits of the use of the FCM algorithm, assures the break blockwise independency described in [5] and reduces the problem of the dimension of the image dataset.

ρ	TP	FP	TN	FN	sensitivity	specificity
0.000	99.39	0.041	99.95	0.60	0.99	1.00
0.250	95.04	0.067	99.92	4.95	0.95	0.99
0.062	94.52	0.071	99.93	5.47	0.94	0.99
0.015	97.94	0.473	99.77	6.17	0.94	0.99

Table 3. Mean indexes for "Baboon"

We also define a pre-processing phase to determine the better compression rate. Many tests (for brevity, here not presented) made on a large sample of gray and color images prove that the use of this method do not invalidate the tamper detection and localization results.

References

[1] M. Barni, Improved wavelet-based watermarking through pixel-wise making, *IEEE Transactions of Image Processing*, 10 (5): 783–791, 2002.

[2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[3] M. U. Celik, G. Sharma, E. Saber and A. M. Te kalp, Hierarchical watermarking for secure image authentication with localization, *IEEE Transactions on Image Processing*, 11 (6)585–595, 2000.

[4] C. C. Chang, Y. S. Hu and T. C Lu, A watermark ingbased image ownership and tampering authen tication scheme, *Pattern Recognition Letters*, 27 (5): 439–446, 2006.

[5] W. C. Chen and M. S. Wang, A fuzzy c-means clustering-based fragile watermarking scheme for image authentication, *Expert Systems with Applications*, 36: 1300–1307, 2009.

[6] I. J. Cox, M. Miller, J. Bloom, J. Fridrich and T. Kalker, *Watermarking and Steganography*, Morgan Kaufmann, San Francisco, 2007.

[7] F. Di Martino, V. Loia and S. Sessa, A method for coding/decoding images by using fuzzy relation equations. In T. Bilgic, B. De Baets and O. Kaynak, editors, *Proceedings of 10th IFSA World Congress*, Lecture Notes in Artificial Intelligence 2715, pages 436-441, Springer-Verlag, 2003.

[8] F. Di Martino, V. Loia and S. Sessa, A method in the compression/decompression of images using fuzzy equations and fuzzy similarities, *Proceedings of 10th IFSA World Congress*, Istanbul, pages 524 –527, 2003.

[9] F. Di Martino and S. Sessa, Digital watermarking in coding/decoding processes with fuzzy relation equations, *Soft Computing*, 10 : 238–243, 2006.

[10] F. Di Martino and S. Sessa, Compression and de-

compression of images with discrete fuzzy trans forms, *Information Sciences*, 177 : 2349– 2362, 2009.

[11] F. Di Martino, V. Loia, I. Perfilieva and S. Sessa, An image coding/decoding method based on di rect and inverse fuzzy transforms, *Int. Journal of Approximate Reasoning*, 48 : 110–131, 2008.

[12] K. Hirota and W. Pedrycz, Data compression with fuzzy relational equations, *Fuzzy Sets and Systems*, 126 : 325–335, 2002.

[13] M. Holliman and N. Memon, Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes, *IEEE Transactions on Image Processing*, 9 (3) : 432–441, 2000.

[14] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, Chapman & Hall/CRC Press, Boca Raton, 2007.

[15] C. T. Li, Digital fragile watermarking scheme for authentication of JPEG images, *IEEE Proceedings on Vision Image and Signal Processing*, IEEE Press 151 (6), pages 460–466, 1994.

[16] C. T. Li and Y. Yuan, Digital watermarking scheme exploiting nondeterministic dependence for image authentication, *Optical Engineering*, 45 (12) : 127001–6, 2006.

[17] C.T. Li and F.M.Yang, One-dimensional neighbourhood forming strategy for fragile watermarking, *Journal of Electronic Imaging*, 12 (2) : 284–291, 2003.

[18] V. Loia and S. Sessa, Fuzzy relation equations for coding/decoding processes of images and videos, *Information Sciences*, 171 : 145–172, 2005.

[19] P. Meenakshi Devi, M. Venkatesan and K. Duraiswamy, Fragile watermarking scheme for image authentication with tamper localization using integer wavelet transform, *Journal of Computer Science*, 5 (11) : 831–837, 2009.

[20] A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, 1996.

[21] H. Nobuhara, W. Pedrycz and K. Hirota, A digital watermarking algorithm using image compression method based on fuzzy relational equations, *Proceedings FUZZ- IEEE 2002*, IEEE Press 2, pages 1568–1573, 2002.

[22] I. Perfilieva, Fuzzy transforms, *Fuzzy Sets and Systems*, 157 : 993–1023, 2006.

[23] F. Y. Shih, *Digital Watermarking and Steganography: Fundamentals and Techniques*, CRC Press, Boca Raton, 2007.

[24] S. Suthaharan, Fragile image watermarking using a gradient image for improved localization and security, *Pattern Recognition Letters*, 25 (16) : 1893–1903, 2004.

[25] R.B.Wolfgang and E.J.Delp, A watermark for digital images, *Proceedings IEEE of the International Conference on Image Processing*, IEEE Press 3, pages 219–222, 1996.

[26] P. W. Wong and N. Memon, Secret and public key image watermarking schemes for image authentication and ownership verification, *IEEE Transactions on Image Processing*, 10 (10): 1593–1601, 2001.

[27] M. Wu and B. Liu, Watermarking for image authentication, *Proceedings IEEE of International Conference on Image Processing*, IEEE Press 2, pages 437–441, 1998.

[28] H. Zhong, F. Liu and L. C. Jiao, A new fragile watermarking technique for image authentication, *Proceedings of International Conference on Signal Processing* 1, Beijing, pages 792–795, 2002.