Journal of Ambient Intelligence and Humanized Computing (2019) 10:2041–2061
https://doi.org/10.1007/s12652-018-0806-3

ORIGINAL RESEARCH

Fragile watermarking tamper detection via bilinear fuzzy relation equations

Ferdinando Di Martino[1] · **Salvatore Sessa**[1,2]

Received: 22 January 2018 / Accepted: 20 April 2018 / Published online: 28 April 2018

Abstract
We present a fragile color image watermarking based on the greatest solution of a bilinear fuzzy relation equation. The original image is coded with fuzzy transforms and divided in sub-images of sizes 2×2 called blocks. The watermark is applied on these blocks. A pre-processing phase is used to determine the best compression rate for the coding process. We test this scheme in tamper detection analysis on a sample of color images having different sizes. The results show that the proposed algorithm is better than that one obtained by using our previous method. Furthermore comparisons with various block-based fragile watermarking methods are made in our tests.

Keywords Fragile watermarking · Block-wise Scheme · Bilinear fuzzy relation equation · Fuzzy transform · Tamper detection · Tamper localization

1 1. Introduction

Today's availability of powerful image processing software allows to manipulate and to alter an image maliciously without making others aware of this manipulation. Some known image analysis techniques can detect these manipulations, but an expert attacker can make his manipulations unrecognizable by these algorithms.

Digital watermarking techniques can be applied to prevent unauthorized alteration and to detect tampers on the published image. Generally they are classified in three categories (Cox et al. 2008; Shih 2007):

- *robust watermarking*, applied to preserve the image copyright. The information encapsulated in the image information cannot be destroyed by any attack;
- *fragile watermarking*, applied to detect and localize alterations in the image; the information encapsulated in the image can be easily destroyed and is used to detect and localize tampered zones;
- *semi-fragile watermarking*, applied to detect only malicious manipulations of the image, ignoring manipulations due to "routine processes" such as, for example, lossy compressions, brightness adjustments or filtering operations.

Fragile watermarking scheme is further classified into two sub-categories:

- *block-wise scheme*, in which the image is partitioned into blocks; in each block a secret random signature is inserted in pixels of that block;
- *pixel-wise scheme* in which a binary authentication watermark was produced by difference between pixels.

Block-wise fragile algorithms localize the tampered blocks, but they cannot localize precisely the tampered pixels. Pixel-wise algorithms (Al-Otum and Al-Taba'a 2009; Barni 2002; MeenakshiDevi et al. 2009; Qinet al. 2017) can localize precisely the tampered pixels, but they can be too expensive in terms of CPU time and memory storage.

A first block-wise image watermark scheme was presented by Walton (1995): a pseudo-random walk is applied

✉ Salvatore Sessa
sessa@unina.it

Ferdinando Di Martino
fdimarti@unina.it

[1] Dipartimento di Architettura, Università degli Studi di Napoli Federico II, Via Toledo 402, 80134 Napoli, Italy

[2] Centro Interdipartimentale di Ricerca "A. Calza Bini", Università degli Studi di Napoli Federico II, Via Toledo 402, 80134 Napoli, Italy

on the image dependent from a secret key (SK). The checksum is built from the 7 most significant bits and is inserted in the least significant bit (LSB) of image data. However, Holliman and Memon (2000) show that this algorithm cannot detect Vector Quantization (VQ) counterfeiting attacks. Since each block is authenticated by itself, the tampering image appears authentic to the block – based watermarking scheme. Many variations of Walton (1995) were proposed in literature for solving this problem as, for instance, in the following papers:

- Chang et al. (2006) and Li (2004) presented a block-wise scheme in which the authentication data of each block is generated by using a cryptographic hash function;
- Suthaharan (2004) gave a block-wise scheme where the watermark is generated by different combinations of circular shifts and permutations on a gradient image, that is a 256 grey-scale image has continuous intensity changes and this scheme is robust to VQ attacks;
- Li et al. (2012) proposed a new block-wise fragile watermarking scheme, robust to VQ attack, in which some high frequency coefficients in the quantized Discrete Cosine Transform (DCT) block are selected as the recovery information and the inverse DCT transform is applied for image recovery;
- Celik et al. (2002) studied a hierarchical block-wise watermarking scheme partitioning an image into blocks as multi-level hierarchy on which the block signatures are constructed;
- Chang et al. (2006) presented an authentication data to be inserted into some Least Significant Bit (LSB) of the central pixel block;
- Li and Yuan (2006) proposed a new block-wise scheme where a non-deterministic dependency relationship between blocks is applied;
- Ni et al. (2013) studied a new block-wise scheme based on partitions of the image into irregular image regions which is also resistant to VQ attacks;
- Chen and Wang (2009) proposed a new block-wise scheme is realized by Fuzzy C-Means (FCM) clustering algorithm (Bezdek 1981): the image is partitioned in sub-images called blocks of sizes 2×2 and an authentication data is generated for each block by using a pseudo random sequence seeded with a SK. In other words, each block can be considered as fourth dimensional pattern of a dataset in which the four pixels are its features. After setting the number of clusters C, the FCM algorithm is applied for finding the partition matrix U of dimensions $N_B \times C$, where N_B is the number of blocks. The image is marked by applying the authentication data to the two LSB's of each pixel in any block. The authors show that this scheme can detect many types of attack as VQ counterfeiting, cut and paste ta2009ing. Chen and Wang (2009) show that their algorithm is better than that obtained by Chang et al. (2006) and Holliman and Memon (2000).

Recently some block-wise scheme variations were proposed to improve the tamper detection and localization precision as, for instance, in the following papers:

- - Tong et al. (2013) proposed a new block-based scheme where the 2×2 blocks are scrambled via a chaotic map and moreover it has high tamper localization accuracy;
- - Ansari et al. (2016) divided the into 4×4 blocks and the Singular Value decomposition technique is applied to each block: the trace of the singular matrix is mapped in order to increase the tamper detection accuracy;
- - Singh and Singh (2017) studied a new block-based scheme based on DCT in which the three LSB's of each pixel are replaced with a watermark, so increasing the accuracy of the tamper localization.

Due to the large size that today reaches the dataset of images published on WEB sites, it is necessary to preserve the watermark from the compression of the image to be published. Some authors investigate approaches to make the watermark more robust to the compression as proposed by Wolfgang et al. (1998) and those based on fuzzy relation equations (Di Martino and Sessa 2006, 2012a; Nobuhara et al. 2002).

Di Martino and Sessa (2012b) proposed a new block-wise scheme in which the watermark is applied directly on the image compressed via fuzzy transforms (F-transforms). The watermark insertion is performed by applying the block-based watermarking scheme described by Chen and Wang (2009) where a fuzzy partition of the 2×2 blocks of the compressed image is performed by using an FCM algorithm: a block-wise independency and a relationship between blocks is realized as well. An authentication data is generated for each block by using a pseudo-random sequence seeded with a SK. Here we describe in Sect. 2 the watermark insertion process and the tampere analysis due Martino and Sessa (2012b) and an example of watermark applied to a 2×2 block is also presented. In Sect. 3 we recall the F-transform imag2012bing-decoding algorithm (Di Martino and Sessa 2012b). In Sect. 4 the algorithm for finding the greatest solution of a bilinear fuzzy relation equation is described (Di Martino and Sessa 2017). In Sect. 5 the BFRE image watermarking method is presented together to the detailed pre-processing phase. In Sect. 6 we point out many tests comparing the detection performances of the BFRE method and F-transform based algorithm, moreover we show comparisons with other block-wise fragile watermarking algorithms. Final considerations are reported in Sect. 7.

2 Watermark processing in 2×2 block

Here we recall the process (Di Martino and Sessa 2012a) for marking a new image and storing the compressed marked image in an image dataset (Figs. 1, 2). The process is partitioned in the following activities:

- *Image coding*: the new image is compressed by using the direct F-transform method;
- *Watermark insertion*: the watermark insertion marks the coded image and stored in the new compressed image dataset;
- *Image decoding*: the marked image is decompressed by using the inverse F-transform, ready to be distributed.

The tampered image is compressed and compared with the compressed original marked image. The tamper localization function localizes the tampered regions, producing the two levels of the tamper localization image.

In this paper we present a new color image watermarking algorithm in which we apply a Bilinear Fuzzy Relation Equation (BFRE) (Li 1992) on the 2×2 blocks of the compressed image to be marked. The BFRE algorithm was used by Di Martino and Sessa (2017) for image comparison, but here our objective is to improve the tamper detection process. Formally, the BFRE algorithm finds the greatest solution of a system of n bilinear fuzzy relation equations with n unknowns. We consider two fuzzy matrices A and B of dimensions n×n, where $A = [a_{ij}]$ and $B = [b_{ij}]$, a_{ij}, b_{ij} in [0,1] and i,j = 1,2,…,n. We calculate the greatest solution of a *system of fuzzy bilinear equations* given by $A \bullet \mathbf{x} = B \bullet \mathbf{x}$, where "•" is the known max–min composition in [0,1], with the vector solution $\mathbf{x}=(x_1,x_2,\ldots,x_n)^T$, being $0 \le x_j \le 1$, j = 1,.2,…,n. The general form of an above mentioned system is the following:

$$\vee_{i=1}^{n}\left(a_{ij} \wedge x_j\right) = \vee_{i=1}^{n}\left(b_{ij} \wedge x_j\right) \tag{1}$$

where ∨and ∧ are the max and min operators, respectively. Obviously the least solution is $\mathbf{x_0} = (0,0,\ldots,0)^T$. If A = B, we

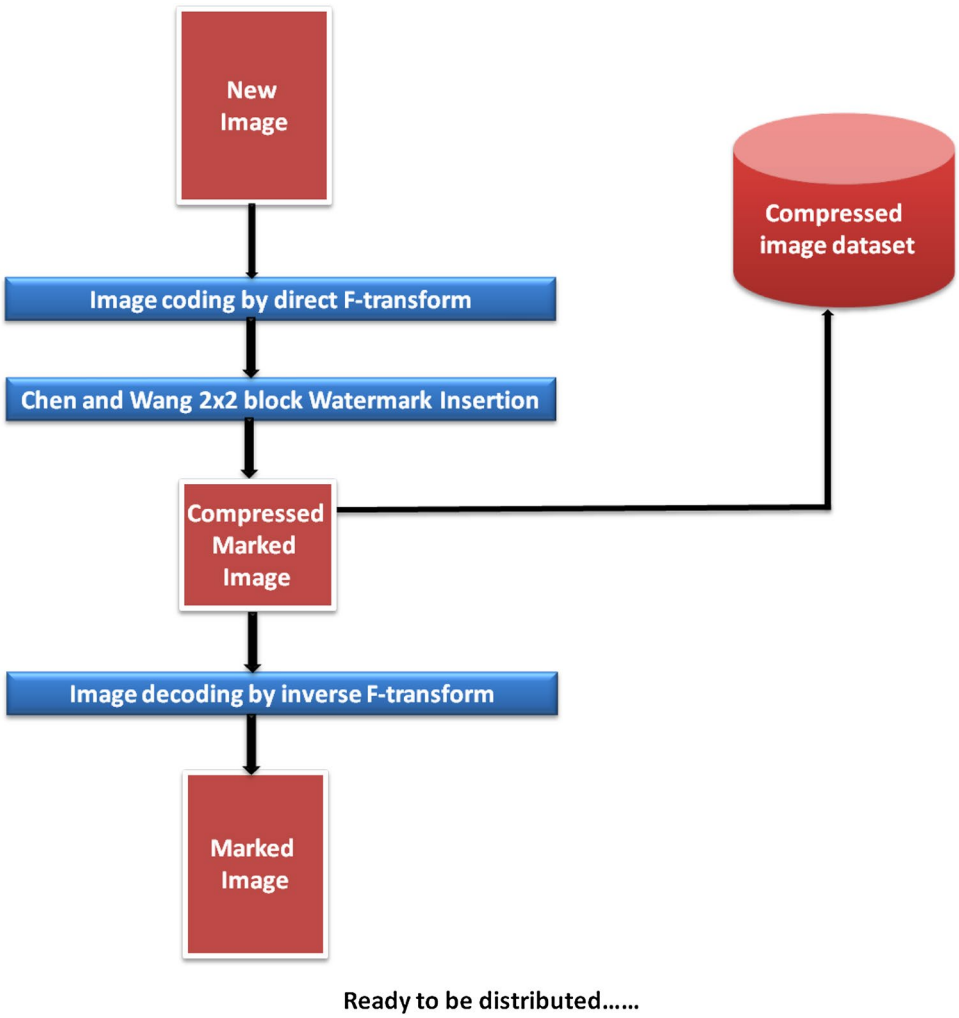

Fig. 1 Watermark insertion process

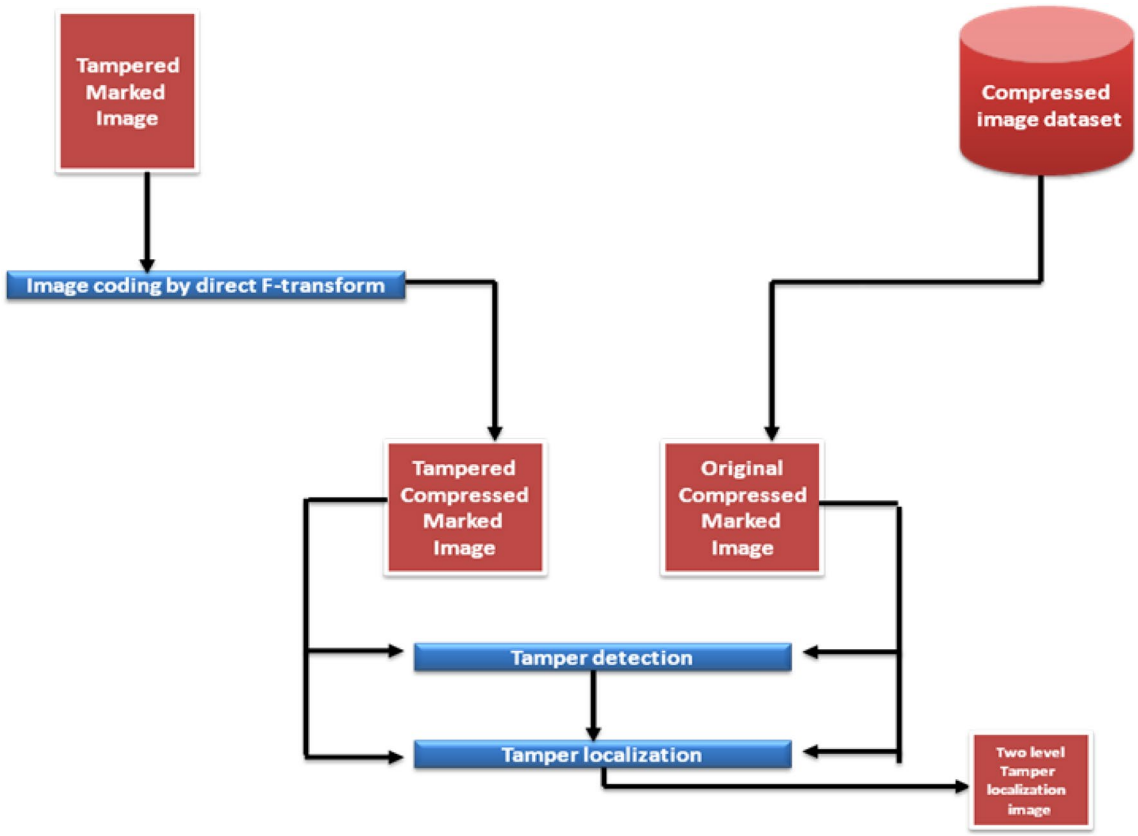

Fig. 2 Tamper analysis process

obtain the trivial greatest solution $\mathbf{x_1} = (1,1,\ldots,1)^{\mathrm{T}}$. Hence from now on, we suppose $\mathrm{A} \neq \mathrm{B}$ and an algorithm to find the greatest solution was given by Lin (1992) recalled further on.

As proposed by Di Martino and Sessa (2017), here the original image is compressed by the direct F-transform and the compressed image is stored in the image dataset. Then the watermark is applied on the compressed image by using the BFRE algorithm and the marked decompressed image is published as well. We consider an $\mathrm{N} \times \mathrm{M}$ color image compressed with the F-transform algorithm. Each band of the compressed image is partitioned in K blocks of dimensions 2×2. Let $\hat{\mathbf{x}}_\mathbf{h} = \left(\hat{x}_{1h}, \hat{x}_{2h}\right)$, $\mathrm{h} = 1,\ldots,\mathrm{K}$, the greatest vector solution obtained for the h*th* block. We calculate the mean value of the two components given as $\bar{x}_h = (\hat{x}_{1h} + \hat{x}_{2h})/2$, then we take the integer $\mathrm{S_h} = [255 \cdot \bar{\mathrm{x}}_\mathrm{h}]$, where $\mathrm{s_h} \in \{0,1, 2, \ldots, 255\}$.

For each block we apply the Chen and Wang scheme [8], in which the authentication data is embedded in the 8 LSB's for each image block. For achieveing this aim, a random sequence, $(\mathrm{r_1}, \mathrm{r_2},\ldots,\mathrm{r_K})$, $\mathrm{r_h} \in \{0,1,\ldots,255\}$, $\mathrm{h} = 1,\ldots,\mathrm{K}$, is considered creating a pseudo-random number generator seeded with a SK. For the h*th* block the corresponding authentication data is constructed as

$$d_h = s_h \oplus r_h \tag{2}$$

where the operator $\oplus$ is the XOR operator. Each two bit couple in the 8 bit authentication data d_h is embedded in the two LSB's of the corresponding pixel of the block. In Fig. 3 we show an example of this watermark insertion process.

This process is repeated in the compressed images of the two bands G, B, applying the BFRE algorithm over each 2×2 block. The compressed original unmarked image is stored in the image dataset with the information necessary to obtain the marked compressed image. The BFRE watermark insertion process is schematized in Fig. 4.

Following Di Martino and Sessa (2017), in order to find the best compression rate ρ to be applied for coding the original image, a pre-processing phase is performed in where the trend of the mean Peak Signal to Noise Ratio (PSNR) is obtained for the marked image in any band. The PSNR index of any marked image is defined as

$$PSNR = 20\log_{10}\frac{255}{RMSE} \tag{3}$$

where RMSE is the Root Mean Square Error calculated by comparing the decompressed marked image and the original image in any band. Di Martino and Sessa (2017) optimized

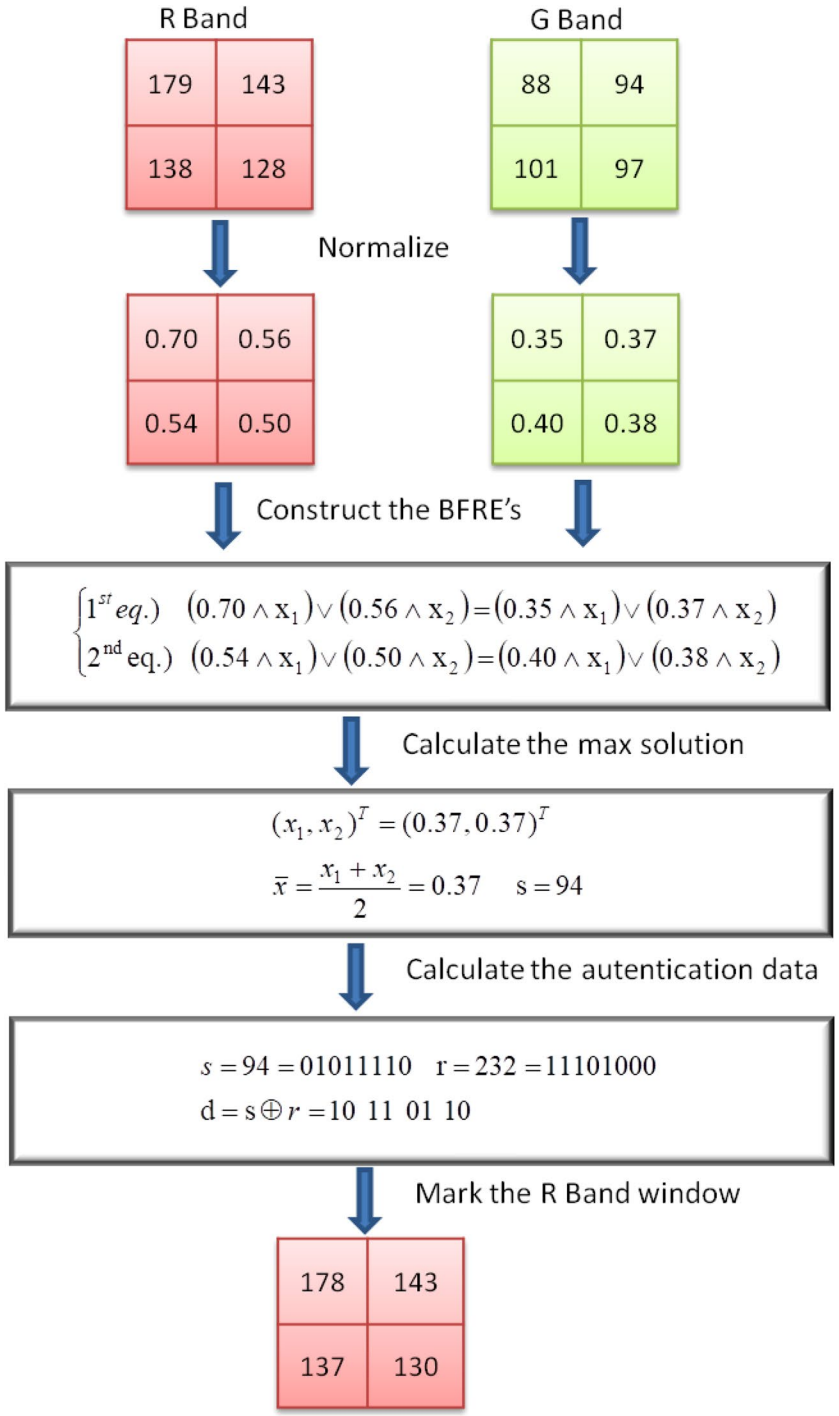

Fig. 3 Example of watermark insertion in a block of the compressed R-band image

the compression rate as the smallest ρ for which the RMSE is not greater than the value $2.5 \cdot (\text{RMSE})_0$, where $(\text{RMSE})_0$ is the RMSE obtained without compression of the image ($\rho = 1$). For a value of RMSE equals to $2.5 \cdot (\text{RMSE})_0$, we obtain a threshold given as

$$(PSNR)_{TH} = 20\log_{10}\frac{255}{2.5 \cdot (RMSE)_0} \tag{4}$$

For compression rates such that the PSNR is less than the threshold (4), the loss of information is considered enough to invalidate both tamper detection and localization analysis. For color images the threshold $(\text{PSNR})_{\text{TH}}$ is given by the arithmetic average of the thresholds obtained in any band. In order to ensure high tamper detection performance, we set the maximum compression rate for which the PSNR index results greater or equal the threshold (4),

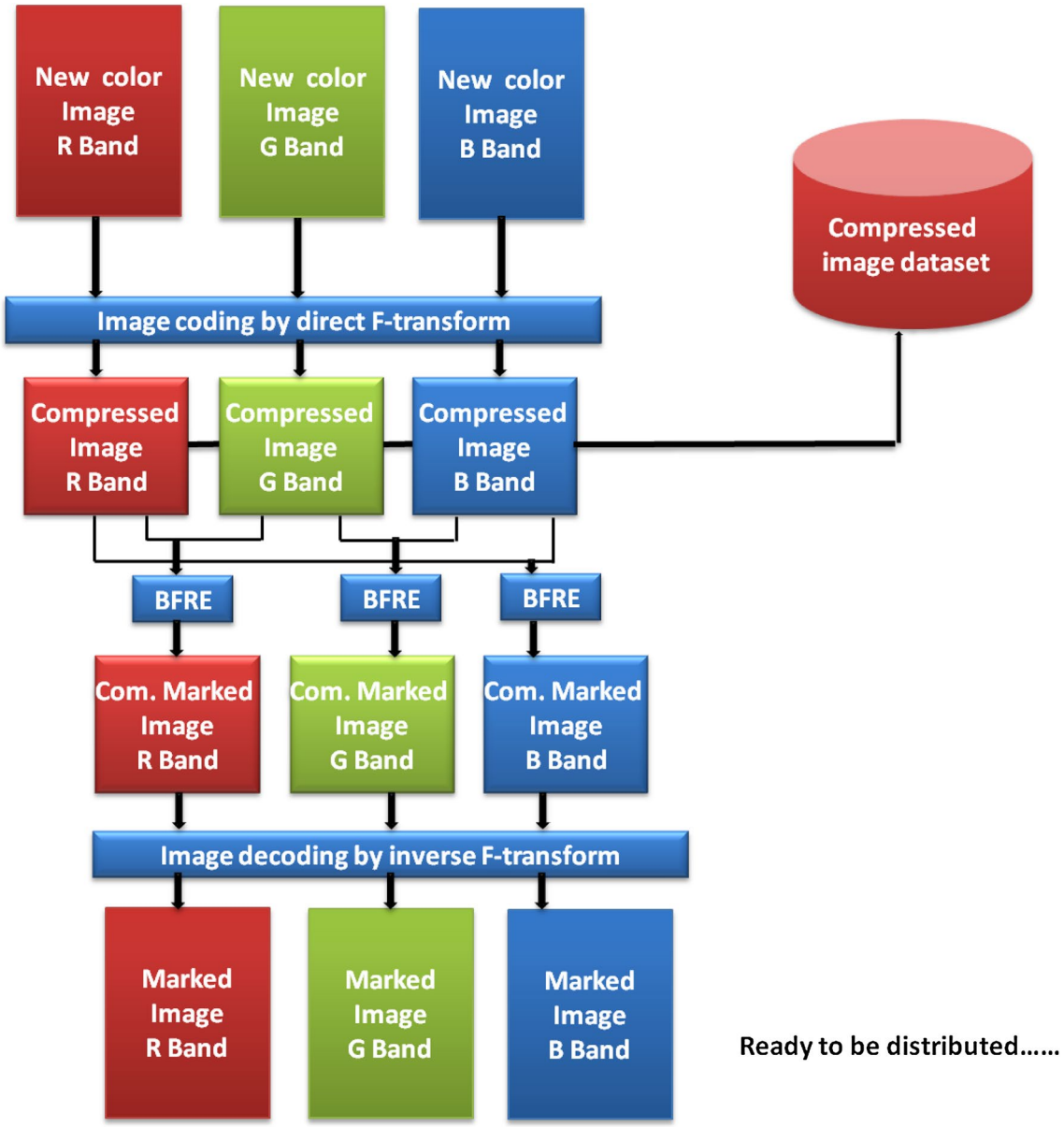

Fig. 4 BFRE watermark insertion process

then we set as compression rate the minimum of the compression rates found in the three bands. So we ensure that in no band the loss of information due to compression can affect the results of the tamper analysis.

A tampered image is compressed via direct F-transform and the corresponding compressed image in the three bands is extracted from the image dataset. Afterwards the BFRE watermark insertion is applied and finally the tamper localization function localizes the tampered zones, producing the two level tamper localization binary images for each band. In Fig. 5 this process is schematized in detail.

The BFRE watermarking algorithm preserves the advantages of the F-transform based algorithm. Indeed we have that

- the advantage of the F-transform tamper detection is preserved in terms of storage of the published image dataset. F-transform based algorithm is used for coding the source image, moreover the compressed image is tampered and stored in the dataset;
- the CPU time performance of the F-transform tamper analysis is preserved and the tamper detection and localization processes are made on the compressed images;

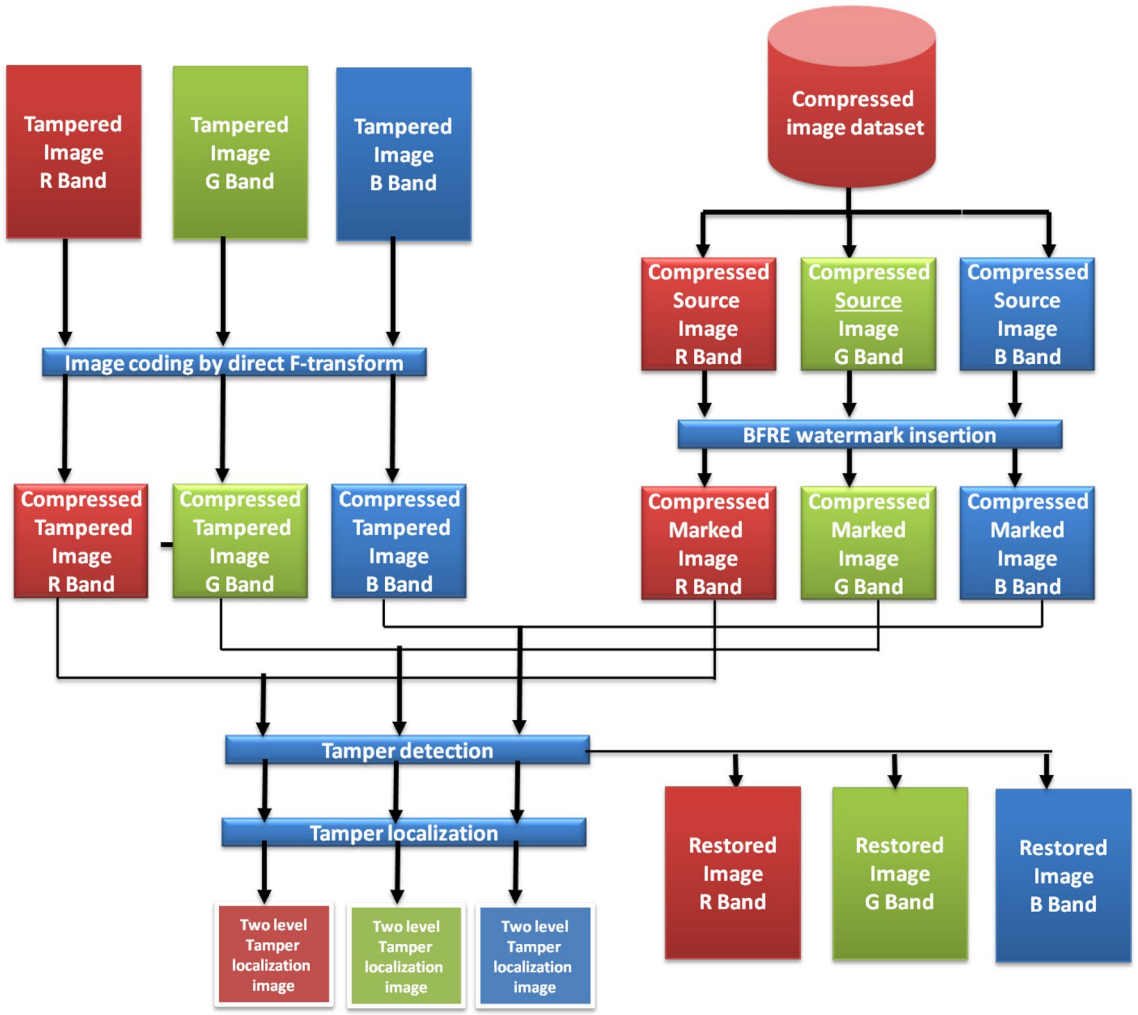

Fig. 5 BFRE tamper analysis process

- the block-wise independency watermarking scheme due to Chen and Wang (2009) is applied to the compressed source image. Then the tampered image is compressed and compared with the coded source marked image.

3 The F-transform based coding/decoding method

We recall some main definitions of F-transforms (Perfilieva 2006). 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 $x \in [a,b]$, $\sum_{i=1}^{n} A_i(x) = 1$.
6. If the following additional properties hold:
7. $x_i = a + h \cdot (i-1)$ for $i = 1, 2, \ldots, n$, where $h = (b-a)/(n-1)$ (equi-distance of nodes),
8. $A_i(x_i - x) = A_i(x_i + x)$ for every $x \in [0,h]$ and $i = 2,\ldots, n-1$,
9. $A_{i+1}(x) = A_i(x - h)$ for every $x \in [x_i, x_{i+1}]$ and $i = 1,2,\ldots, n-1$,

we say that the fuzzy sets $\{A_1,\ldots, A_n\}$ constitute an uniform (or symmetric) fuzzy partition. Considering the discrete case, let f be a function predefined in a finite set $P = \{p_1,\ldots,p_N\} \subset [a,b]$ which is sufficiently dense with respect to a fuzzy partition $\{A_1, A_2, \ldots, A_n\}$ of [a,b] (that is, if $N > n$ and for every $k = 1,\ldots,n$, there exists at least an index $i \in \{1,\ldots,N\}$ such that $A_k(p_i) > 0$). Then we define the direct F-transform of f with

respect to $\{A_1, A_2, \ldots, A_n\}$ as the vector $(F_1, F_2, \ldots, F_n)$ with components defined as

$$F_k = \frac{\sum_{i=1}^{N} f(p_i)A_k(p_i)}{\sum_{i=1}^{N} A_k(p_i)} \tag{5}$$

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

$$f_n^F(p_i) = \sum_{k=1}^{n} F_k A_k(p_i) \tag{6}$$

Perfilieva (2006) proved that the inverse F-transform can approximate f with an arbitrary precision and the greater the dimension n of the fuzzy partition is, the greater the precision of this approximation is.

This concept is extendable to the case of two variables (Perfilieva 2006). Indeed let f be a function in two variables assuming prefixed values in the finite set $P \times Q = \{p_1,\ldots,p_N\} \times \{q_1,\ldots,q_M\} \subset [a,b] \times [c,d]$, 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]), where $2 < n < N$ and $2 < m < M$. Then the $n \times m$ 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

$$F_{kl} = \frac{\sum_{j=1}^{M}\sum_{i=1}^{N} f(p_i, q_j)A_k(p_i)B_l(q_j)}{\sum_{j=1}^{M}\sum_{i=1}^{N} A_k(p_i)B_l(q_j)} \tag{7}$$

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_{nm}^F:$(p_i, q_j) \in P \times Q \rightarrow f_{nm}^F(p_i, q_j) \in [0,1]$ defined as

$$f_{nm}^F(p_i, q_j) = \sum_{k=1}^{m}\sum_{l=1}^{n} F_{kl}A_k(p_i)B_l(q_j) \tag{8}$$

Let I be a $N \times M$ grey image and P: $(i,j) \in \{1,\ldots,N\} \times \{1,\ldots,M\} \rightarrow [0,1]$, P(i,j) being the normalized value of the pixel I(i,j) of the image with respect to the length of the grey scale. For brevity of notation, we put $p_i = i$, $q_j = j$, $a = c = 1$, $b = N$, $d = M$. Moreover, we define the fuzzy sets $A_1,\ldots,A_n$: $[1,N] \rightarrow [0,1]$ (resp., $B_1,\ldots,B_m$: $[1,M] \rightarrow [0,1]$) with $n < N$ (resp., $m < M$), forming a fuzzy partition of [1,N] (resp., [1,M]). Following (Di Martino and Sessa 2012b), then P is divided in sub-matrices P_D of sizes $N(D) \times M(D)$, that is P_D : $(i, j) \in \{1,\ldots,N(D)\} \times \{1,\ldots,M(D)\} \rightarrow P_D(i, j) \in [0,1]$, called blocks, compressed to blocks of sizes $n(D) \times m(D)$ (with $n(D) < N(D)$, $m(D) < M(D)$) via the direct F-transform $[F_{kl}^D]$ defined as

$$F_{kl}^D = \frac{\sum_{j=1}^{M(D)}\sum_{i=1}^{N(D)} P_D(i,j)A_k(i)B_l(j)}{\sum_{j=1}^{M(D)}\sum_{i=1}^{N(D)} A_k(i)B_l(j)} \tag{9}$$

for any k = 1,…, m(D) and l = 1,…, n(D). Then we decode the blocks with the following inverse F-transform $P_{m(D)n(D)}^F$: $(i, j) \in \{1,\ldots,M(D)\} \times \{1,\ldots,N(D)\} \rightarrow P_{m(D)n(D)}^F(i, j) \in [0,1]$:

$$P_{m(D)n(D)}^F(i,j) = \sum_{l=1}^{m(D)}\sum_{k=1}^{n(D)} F_{kl}^D A_k(i)B_l(j) \tag{10}$$

which approximates the matrix P_D of sizes $N(D) \times m(D)$. The following fuzzy sets $A_1,\ldots,A_{n(D)}$:$[1,N(D)] \rightarrow [0,1]$ and $B_1,\ldots,B_{m(D)}$:$[1,N(D)] \cdot [0,1]$ constitute an uniform fuzzy partition:

$$A_1(i) = \begin{cases} 0.5\left(\cos\frac{\pi}{h}(i-1)+1\right) & \text{if } i \in [1,x_2] \\ 0 & \text{otherwise} \end{cases} \tag{11a}$$

$$A_k(i) = \begin{cases} 0.5\left(\cos\frac{\pi}{h}(i-x_k)+1\right) & \text{if } i \in [x_{k-1},x_{k+1}] \\ 0 & \text{otherwise} \end{cases} \tag{11b}$$

$$A_{n(D)}(i) = \begin{cases} 0.5\left(\cos\frac{\pi}{h}(i-x_{n(D)-1})+1\right) & \text{if } i \in [x_{n(D)-1},N(D)] \\ 0 & \text{otherwise} \end{cases} \tag{11c}$$

where k = 2,…, m(D), $h = (M(D)-1)/(m(D)-1)$, $x_k = 1 + h\bullet(k-1)$ and

$$B_1(j) = \begin{cases} 0.5\left(\cos\frac{\pi}{s}(j-1)+1\right) & \text{if } j \in [1,y_2] \\ 0 & \text{otherwise} \end{cases} \tag{12a}$$

$$B_t(j) = \begin{cases} 0.5\left(\cos\frac{\pi}{s}(j-y_t)+1\right) & \text{if } j \in [y_{t-1},y_{t+1}] \\ 0 & \text{otherwise} \end{cases} \tag{12b}$$

$$B_{m(D)}(j) = \begin{cases} 0.5\left(\cos\frac{\pi}{s}(j-y_{m(D)-1})+1\right) & \text{if } j \in [y_{m(D)-1},M(D)] \\ 0 & \text{otherwise} \end{cases} \tag{12c}$$

where t = 2,…, n(D), $s = (N(D)-1)/(n(D)-1)$, $y_t = 1 + s\bullet(t-1)$.

4 The BFRE algorithm

Let $A = [a_{ij}]$ and $B = [b_{ij}]$ two fuzzy relations of dimensions $m \times n$, (i = 1,…,m and j = 1,…,n) and a vector $x = (x_1, x_2, \ldots, x_n)$. We consider the following system of fuzzy relation equations:

$$a_i \vee (\vee_{j=1}^{n}(a_{ij} \wedge x_j)) = b_i \vee (\vee_{j=1}^{n}(b_{ij} \wedge x_j)) \tag{13}$$

for any i = 1,2,…,m and a_i, $b_i \in [0,1]$ are assigned real numbers. The Eqs. (13) form a so-called *system of external fuzzy bilinear equations* (Li19922). If $a_i = b_i = 0$ for

i = 1,2,…,m, the system (13) becomes (1). In the sequel we deal with the following quantities:

$$\rho_i = \min\Big(a_i \vee (\vee_{j=1}^{n}(a_{ij}), b_i \vee (\vee_{j=1}^{n}(b_{ij})\Big) \tag{14}$$

for any i = 1,2,…,m. Let us consider the sets $\Delta_i^1 = \{j \in \{1,2,\ldots,n\} : b_{ij} > \rho_i\}$, $\Delta_i^2 = \{j \in \{1,2,\ldots,n\} : a_{ij} > \rho_i\}$. Let $\rho_k = \min\{\rho_i : i = 1,\ldots,m\}$. For finding the greatest solution $\hat{x} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)$ of the system (13), the following theorem holds 19921992):

Theorem 1 Let be either $\Delta_k := \Delta_k^1$ or $\Delta_k := \Delta_k^2$. If $\Delta_k = \{j_1, j_2, \ldots, j_t\}$, then $\hat{x}_{j_1} = \hat{x}_{j_2} = \ldots = \hat{x}_{j_t} = \rho_k$. If $\Delta_k = I_n$, then $\hat{x} \in [0,1]^n$ with $\hat{x}_i = \rho_k$ for i = 1,…,n.

In other words, the following recursive algorithm holds:

Step 1 We calculate ρ_k and the corresponding set $\Delta_k = \{j_1, j_2, ..., j_t\}$.

Step 2 If t = n, we have $\hat{x} = (\rho_k, ..., \rho_k) \in [0,1]^n$ and the process stops. Otherwise the system (13) becomes a new system of m-t external fuzzy bilinear equations with m-t variables by replacing the variables $x_{j_1}, x_{j_2}, \ldots, x_{j_t}$ with ρ_k.

Step 3 Repeat steps 1 and 2 for finding each component $\hat{x}_j$.

5 The BFRE image watermarking algorithm

Let's consider a N × M color original image for applying the image watermarking algorithm schematized in Fig. 4. We use the block F-transform algorithm described in Sect. 2 for coding the image. Then the compressed image is partitioned in blocks 2 × 2. The greatest solution of a BFRE system is found for each block of a band, following the process of Fig. 3. For marking the R band (resp., G-band) compressed image, each 2 × 2 block is normalized to form the bilinear fuzzy relation Eq. (1). The greatest solution is de-normalized and the Chen and Wang (2009) scheme is applied embedding an authentication data in the LSB's for each block. The same process is applied to the corresponding 2 × 2 blocks of the G and B bands (resp., B and R bands) of the compressed image to mark the G (resp., B) block. The marked compressed image is then stored in the image dataset and the images are decompressed by using the inverse F-transform (10) and ready to be published. Strictly speaking, the BFRE watermarking insertion consists of the following steps:

BFRE Watermarking insertion

Step 1: The original image is compressed in any band with a compression rate ρ by using the block F-transform compression method. The image is partitioned in K blocks of sizes N(D) × M(D), compressed in blocks of size n(D) × m(D), being the compression rate equal to (N(D) × M(D))/(n(D) × m(D)). The direct F-transform (9) is calculated by using the uniform fuzzy partitions (11) and (12)

Step 2: The compressed images in the R and G bands are partitioned in 2 × 2 blocks and the pixels are normalized in [0,1] in every block

Step 3: For each 2×2 block in the R and G bands of the compressed image, a BFRE system composed by two equations is constructed. The BFRE algorithm is applied for finding the greatest solution $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2)$

Step 4: Let $\hat{\mathbf{x}}_\mathbf{h} = (\hat{x}_{1h}, \hat{x}_{2h})$, h = 1,…,K, the greatest solution vector obtained for the h*th* window. We calculate the mean value $\bar{x}_h = \frac{\hat{x}_{1h} + \hat{x}_{2h}}{2}$, and the integer $s_h = [255 \cdot \bar{x}_h]$, where $s_h \in \{0,1,\ldots,255\}$. Then the Chen and Wang scheme is applied, generating a random sequence($r_1, r_2, \ldots, r_K$), $r_h \in \{0,1,\ldots,255\}$ by creating a PRNG seeded with a SK. The corresponding authentication data is embedded into each LSB's of the four pixels of the h*th* compressed 2×2 block in the R and G bands. This step is repeated for h=1,...,K

Step 5: Step 3 and 4 are repeated by considering the compressed images in the G, R (resp., B, R) bands for marking the compressed image in the G (resp., in B) band

Step 6: A copy of the unmarked compressed original image is stored in the image dataset in which the information necessary to mark is preserved (number of blocks, dimensions of the original image, compression rate ρ, random sequence (r_1, r_2, …, r_K)

Step 7: The marked compressed image is decompressed in every band by calculating the inverse F-transform. Then the marked decompressed image is ready to be published

In accordance to Di Martino and Sessa (2012b), we apply a pre-processing phase to finding the optimal compression rate ρ. In each band we calculate the threshold (4) for the PSNR index for the decompressed marked image such the corresponding RMSE is given by 2.5•$(RMSE)_0$, where $(RMSE)_0$ is the RMSE obtained marking the image without compression (i.e., ρ = 1).

In order to ensure high tamper detection performances, we impose that the PSNR of the decompressed marked image must be greater or equal to the PSNR threshold in every pre-fixed band. The BFRE watermarking pre-processing consists of the following steps:

BFRE watermarking insertion in the pre-processing phase	
Step 1:	In every band we set the thresholds $(PSNR)^R_{TH}$,$(PSNR)^G_{TH}$, $(PSNR)^B_{TH}$, applying the watermark directly to the original image. $(PSNR)_0$ is obtained comparing the original and the marked images and computing the threshold with (4). We set an initial strong ρ
Step 2:	The source image is compressed in each band. Then the BFRE watermarking insertion process is applied to the compressed image. Finally, we calculate the PSNR of the decompressed marked image in each band by using (3), that is we obtain $(PSNR)^R$, $(PSNR)^G$, $(PSNR)^B$
Step 3:	If $(PSNR)^R \geq (PSNR)^R_{TH}$, $(PSNR)^G \geq (PSNR)^G_{TH}$, $(PSNR)^B \geq (PSNR)^B_{TH}$, we set ρ and the process stops, otherwise we set a smaller ρ and go to Step 2

In the tamper analysis process the compressed original image is extracted from the image dataset along with the information necessary to obtain the marked compressed image. The tampered image is compressed with the same compression rate of the corresponding compressed marked image in the image dataset. Then the tamper detection function applies the BFRE algorithm on the original compressed image: the pixels (which are not corresponding in the tampered images) are marked as invalid pixels. Thus the published marked image is reconstructed. Finally, the tamper localization identifies the invalid pixels, detecting the tampered zones. The BFRE tamper analysis process is composed by the following steps:

BFRE Tamper analysis	
Step 1:	From the image dataset the compressed original image is extracted the corresponding one to the published image tampered. The BFRE watermarking insertion is applied to it for obtaining the compressed marked image
Step 2:	The tampered image is compressed by using the F-transform compression method to the blocks. The image is partitioned in K blocks of size N(D) × M(D) and compressed with rate ρ=(N(D)×M(D))/(n(D)×m(D))
Step 3:	The tampered image can be reconstructed and republished decoding the compressed marked image
Step 4:	The two compressed images are compared and the tampered pixels are detected
Step 5:	The tampered zones are detected in every band

For measuring the performances of the algorithm we calculate the Sensitivity, Specificity and Accuracy indices. The *Sensitivity* index measures the ability to detect correctly the tampered pixels. The *Specificity* index measures the ability to evaluate correctly the non-tampered pixels. The *Accuracy* index measures the overall ability to detect correctly a pixel.

We calculate the parameters *True Positive (TP)* (resp., *False Negative (FN)*), given by the number of tampered (resp., non-tampered) pixels in the image correctly detected; *True Negative (TN)* (resp., *False Positive (FP)*) given by the number of tampered (resp., non-tampered) pixels in the image incorrectly (resp., correctly) detected as non-tampered (resp., tampered). Thus we have $sensitivity = TP/(TP + FN)$, $specificity = TN/(TN + FP)$, $accuracy = (TP + TN)/(TP + TN + FP + FN)$.

In our tests we measure the sensitivity, specificity and accuracy in each band and calculate the final sensitivity, specificity and accuracy as mean of the values obtained in each band.

6 Test results

Now we show the results of tests in which the BFRE watermarking algorithm was applied on a set of 300 color images having different sizes extracted from web pages at https://www5.cs.fau.de/research/data/image-manipulation. For each image we apply the BFRE tamper analysis process, measuring the sensitivity, specificity and accuracy indices. Comparison tests are pointed out with other fragile watermarking algorithms.

For brevity of exposition, we present the detailed results for three images: “Baboon”, “Rose” and “Infusion”. The original “Baboon” of sizes 256×256 is shown in Fig. 6a. Figure 6b (resp., 6c, 6d) contains the same image in the R (resp., G, B) band.

In the pre-processing phase we use the BFRE algorithm marking the original image (i.e., without compression). We obtain the values of $(\text{PSNR})_0$ and $(\text{PSNR})_{\text{TH}}$ shown in Table 1 for each band.

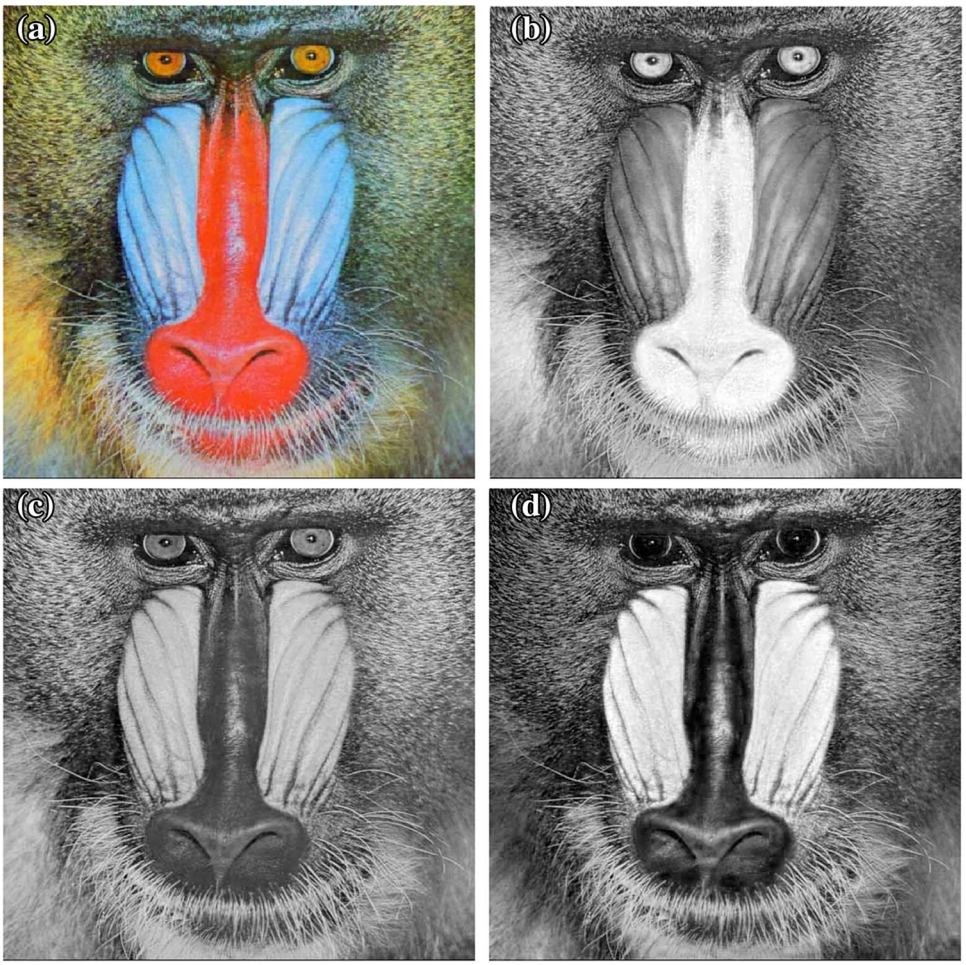

Fig. 6 **a** Original image "Baboon", **b** R band, **c** G band, **d** B band

Table 1 $(PSNR)_0$ and $(PSNR)_{TH}$ obtained in each band for the image "Baboon" ($\rho = 1$)

Band	$(PSNR)_0$	$(PSNR)^i_{TH}$, i = R, G, B
R	31.72	23.76
G	31.67	23.71
B	31.65	23.69

Table 2 PSNR in each band obtained for the marked image "Baboon" ($\rho = 0.25$)

Band	*PSNR*	$(PSNR) - (PSNR)^i_{TH}$
R	23.90	0.14
G	23.90	0.19
B	23.84	0.15

Then we have $\rho = (2 \times 2)/(4 \times 4) = 0.25$. By applying the BFRE watermarking algorithm, we obtain the following values of the PSNR in each band (Table 2):

In Fig. 7a we show the image of Fig. 6 marked by using the BFRE watermarking insertion process. Figure 7b (resp., 7c, 7d) shows the marked image in the R (resp., G, B) band.

The marked image of Fig. 7a has been tampered as shown in Fig. 8a. Figure 8b (resp., 8c, 8d) shows the tampered image in the R (resp., G, B) band.

We apply the BFRE detection algorithm to the tampered image of Fig. 8a. The marked image is extracted, decompressed and compared with the tampered image. Figure 9a (resp., 9b, 9c) shows the tamper localization zone detected in the R (resp., G, B) band.

In Table 3 we show the Sensitivity, Specificity and Accuracy values obtained in the R,G, B bands for the tampered image of Fig. 8a under several compression rates.

The BFRE algorithm gives the best results with respect to the F-transform algorithm for any compression rate also in the mean values of sensitivity, specificity and accuracy given in Table 4.

Now we present the results obtained for the color image "Rose" of sizes 800×600 (Fig. 10a). Figure 10b (resp., 10c, 10d) contains the same image in the R (resp., G, B) band. In

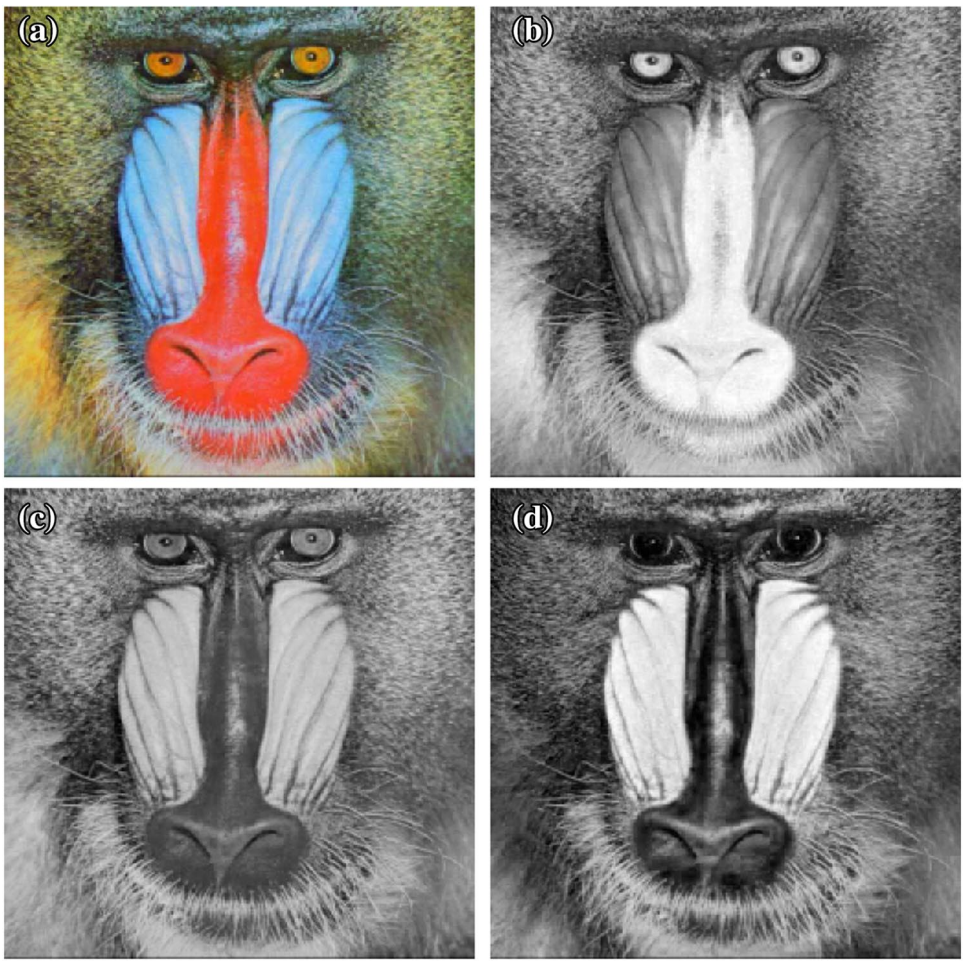

Fig. 7 **a** Marked "Baboon" ($\rho = 0.25$), **b** R band, **c** G band, **d** B band

the pre-processing phase we use the BFRE algorithm marking the original image without compression. The values of $(PSNR)_0$ and $(PSNR)^i_{TH}$, i = R, G, B, are shown in Table 5.

Then we find the optimal compression rate given by $\rho = 0.25$. Applying the BFRE watermark insertion algorithm to the compressed image, we obtained the values for the PSNR shown in Table 6.

In Fig. 11a we show the original marked image and in Fig. 11b (resp., 11c, 11d) the marked image in the R (resp., G, B) band.

In Fig. 12a we show the marked image of Fig. 10a which has been tampered: another rose appears on the left. Moreover, the blob on the right has disappeared. Figure 12b (resp., 12c,12d) shows the tampered image in the R (resp., G, B) band.

We apply the BFRE detection algorithm to the tampered image of Fig. 10a. The marked image is extracted, decompressed and compared with the tampered image. Figure 13a (resp., 13b, 13c) shows the tamper localization zones detected in the R (resp., G, B) band.

In Table 7a we show the comparisons obtained applying the BFRE and F-transform tamper detection algorithms for several compression rates.

These results confirm the results obtained for the image Baboon (i.e., Table 4). The best performances are obtained by using the BFRE algorithm. The difference between the mean sensitivity, specificity and accuracy in both algorithms increases for strong compressions. Now we examine the original color image "Infusion" of sizes 800×600. The original image is given in Fig. 14a, in Fig. 14b (resp., 14c, 14d) the same image in the R (resp., G, B) band.

Applying the BFRE to the original image, we get $(PSNR)_0$ and $(PSNR)^i_{TH}$ (Table 8).

Following the steps of the pre-processing phase, then we find the optimal compression rate, obtained compressing blocks 5×5 in blocks 2×2 ($\rho = (2 \times 2)/(5 \times 5) = 0.16$). Applying the BFRE watermark algorithm to the compressed image, we obtained the following values for the PSNR (Table 9):

In Fig. 15a we show the original marked image and in Fig. 15b (resp., 15c, 15d) the marked image in the R (resp., G, B) band.

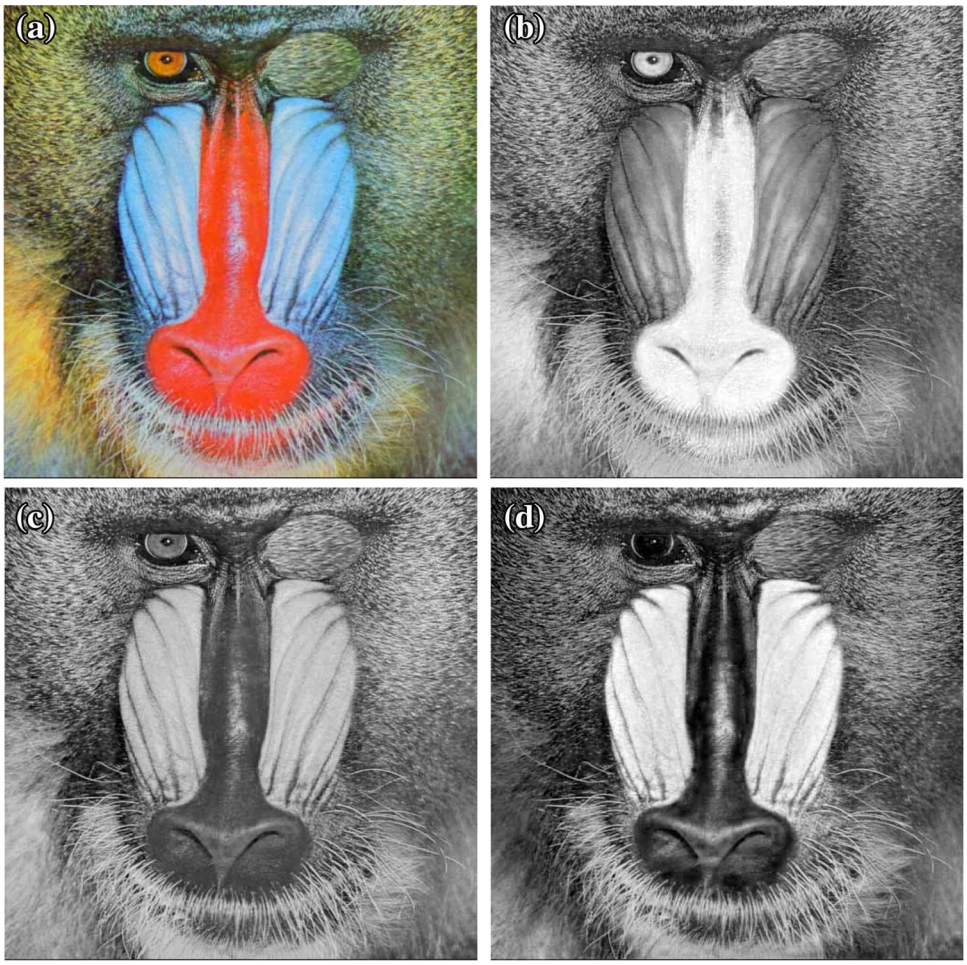

Fig. 8 **a** The tampered image "Baboon". **b** R band. **c** G band. **d** B band

In Fig. 16a the marked image of Fig. 14a has been tampered: other berries appear near the cup. Figure 16b (resp., 16c, 16d) shows the tampered image in the R (resp., G, B) band.

We apply the BFRE tamper detection algorithm to the tampered image in Fig. 16a. The marked image is extracted, decompressed and compared with the tampered image. Figure 17a (resp., 17b, 17c) shows the tamper localization zones detected in the R (resp., G, B) band.

In Table 10a we show the comparisons obtained applying the BFRE and F-transform tamper detection algorithms for several compression rates.

Finally we made the comparison test by considering a set of 200 color images extracted from the above dataset. Figure 18 (resp., Figs. 19, 20) shows the mean sensitivity (resp., specificity, accuracy) differences obtained by using the BFRE and F-transform image watermarking algorithms by varying the compression rate.

Figure 18 (resp., 19, 20) shows that difference between the sensitivity (resp., specificity, accuracy) index (calculated by applying the BFRE algorithm and the corresponding one calculated by applying the F-transform algorithm) increases by augmenting the compression rate of the image. For $\rho < 0.2$, this trend becomes approximately exponential. In Table 11 we show the mean values of the three indices obtained for the 200 images above considered by applying the BFRE algorithm and other block-wise fragile watermarking algorithms FCM, Hierarchical, DCT, Chaos, SVD appeared in [1, 7, 8, 26, 29], respectively.

Generally speaking, the results of Table 11 show that the tamper detection and localization performances obtained by using the BFRE algorithm are acceptable and comparable with the ones obtained by using other block-wise fragile watermarking methods without compression of the original image. In addition, the BFRE method represents an efficient tool for memory storage because it preserves the marked images and compressed under a suitable compression rate.

7 Conclusions

We present a new fragile block-based watermarking algorithm in which the greatest solution of a bilinear fuzzy relation equation is applied on 2×2 blocks for marking

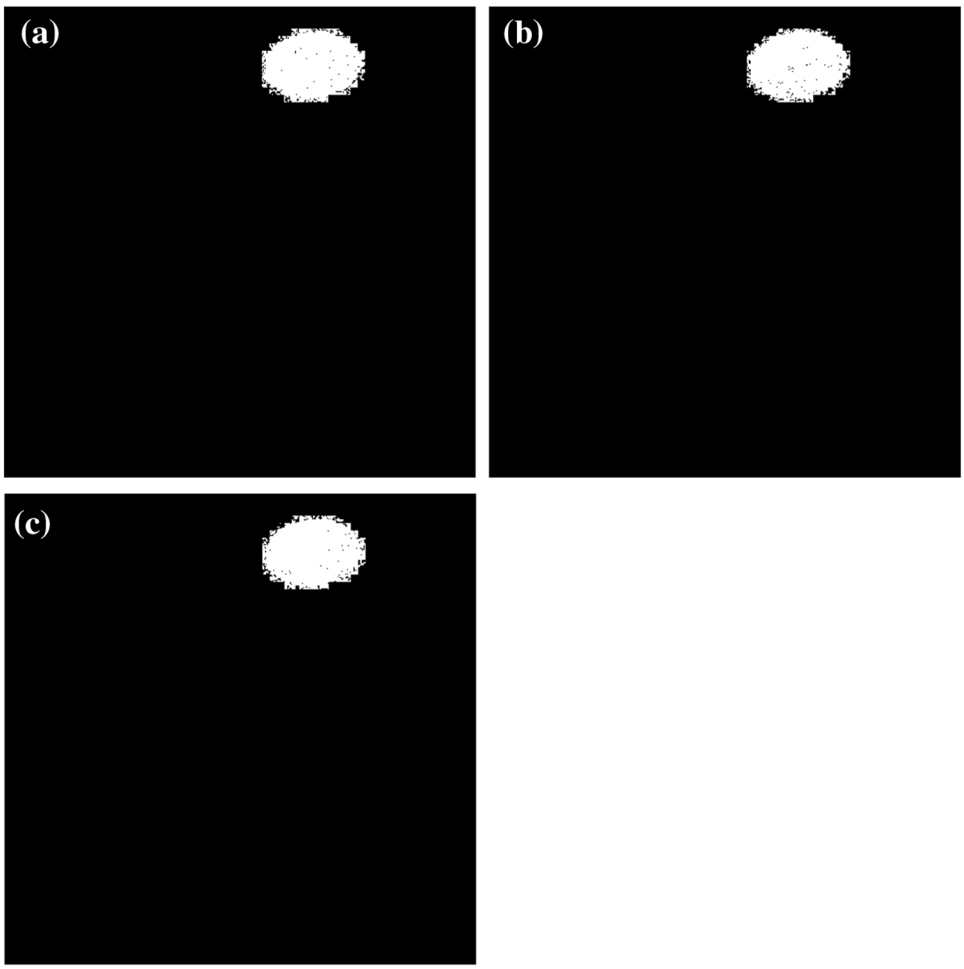

Fig. 9 **a** Tampered zone—R band. **b** Tampered zone—G band. **c** Tampered zone—B band

Table 3 Sensitivity, specificity and accuracy for several ρ in each band (Baboon)

ρ	Band	Sensitivity (%)	Specificity (%)	Accuracy (%)
1 (No compression)	R	99.47	99.96	99.72
	G	99.41	99.97	99.69
	B	99.38	99.97	99.68
0.25	R	95.93	99.94	97.94
	G	94.66	99.95	97.30
	B	95.97	99.95	97.96
0.0625	R	94.97	99.92	97.10
	G	94.73	99.93	97.22
	B	95.39	99.93	97.60
0.015625	R	92.65	99.63	96.14
	G	92.89	99.58	96.24
	B	93.14	99.73	96.44

color images. This algorithm preserves the advantage of storing the marked compressed images in the dataset and contains a pre-processing phase for determining the optimal compression of the marked image. Comparison results show that the proposed method improves the algorithm of Di Martino and Sessa (2012b) in terms of sensitivity, specificity and accuracy and moreover it is comparable with other known block-wise fragile watermarking methods in which no compression of images is realized (Ansari

Table 4 Mean sensitivity, specificity and accuracy for ρ given in Table 3 (Baboon)

ρ	BFRE algorithm			F-transform algorithm		
	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)
1.0000	99.42	99.97	99.70	99.39	99.96	99.68
0.2500	95.52	99.95	97.74	95.05	99.93	97.49
0.0625	95.03	99.93	97.50	94.50	99.91	97.21
0.015625	92.89	99.65	96.27	91.11	99.22	95.17

Fig. 10 **a** Original image "Rose". **b** R band. **c** G band. **d** B band

Table 5 $(PSNR)_0$ and $(PSNR)_{TH}$ obtained in each band for the image "Rose" (ρ = 1)

Band	$(PSNR)_0$	$(PSNR)^i_{TH}$, i = R,G,B
R	42.75	34.79
G	43.31	35.35
B	43.68	35.72

Table 6 PSNR in each band obtained for the marked image "Rose" (ρ = 0.25)

Band	*PSNR*	$PSNR - (PSNR)^i_{TH}$
R	35.31	0.52
G	35.97	0.62
B	36.53	0.58

Fig. 11 **a** "Rose" ($\rho = 0.25$). **b** R band. **c** G band. **d** B band

Fig. 12 **a** The tampered image "Rose". **b** R band, **c** G band. **d** B band

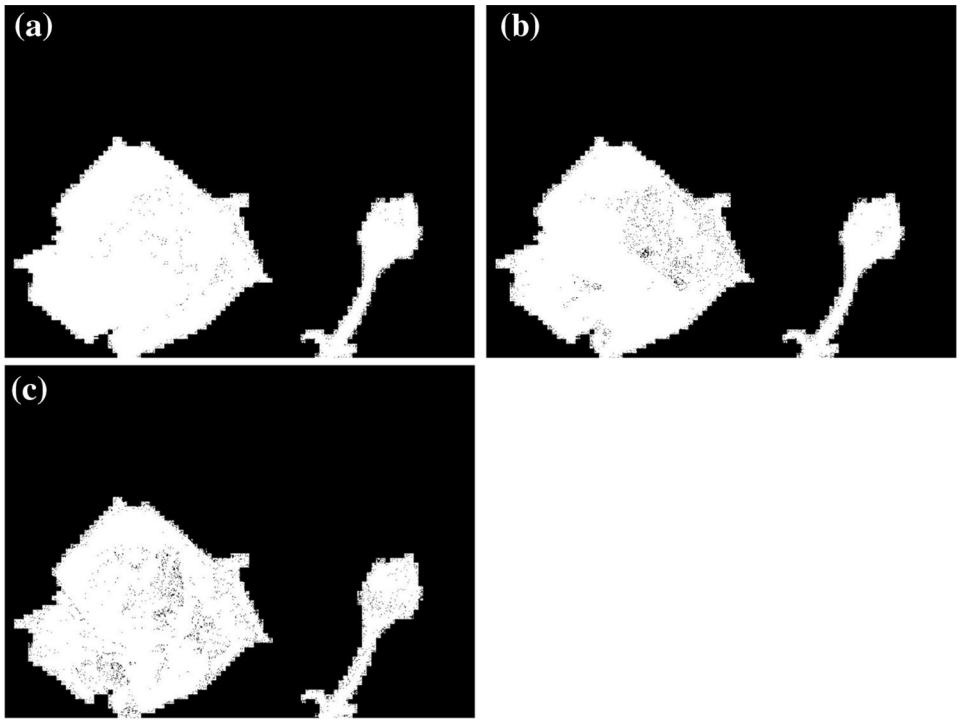

Fig. 13 **a** Tampered zone—R band. **b** Tampered zone—G band. **c** Tampered zone—B band

Table 7 Mean sensitivity, specificity and accuracy for several ρ (Rose)

ρ	BFRE algorithm			F-transform algorithm		
	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)
No compr	99.31	99.96	99.64	99.16	99.95	99.56
0.25	95.46	99.93	97.70	94.27	99.91	97.09
0.16	95.13	99.92	97.53	93.88	99.90	96.89
0.0625	94.62	99.91	97.27	93.21	99.88	96.55
0.015625	92.01	99.59	95.80	90.18	99.03	94.61

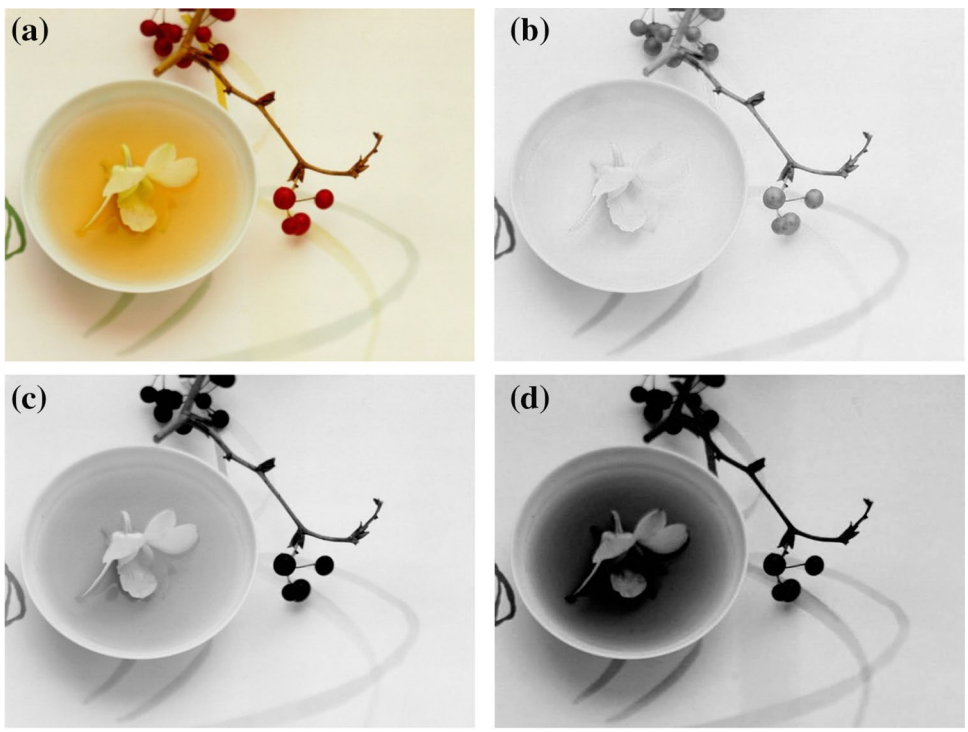

Fig. 14 **a** The original image "Infusion". **b** R band. **c** G band. **d** B band

Table 8 $(PSNR)_0$ and $(PSNR)_{TH}$ in each band for the image "Infusion"

Band	$(PSNR)_0$	$(PSNR)^i_{TH}$ i = R,G,B
R	44.06	36.10
G	42.19	34.23
B	41.85	33.89

Table 9 PSNR in each band obtained for the marked image "Infusion" (ρ = 0.16)

Band	*PSNR*	$PSNR - (PSNR)^i_{TH}$, i = R,G,B
R	36.80	0.70
G	34.91	0.68
B	34.37	0.48

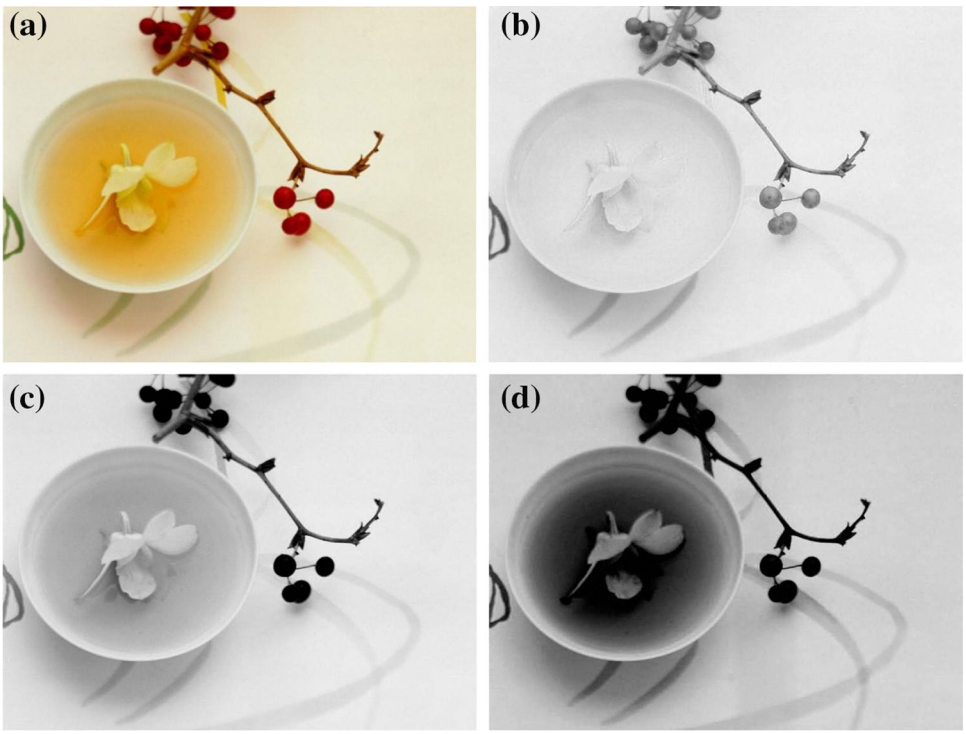

Fig. 15 **a** The marked "Infusion" (ρ = 0.16). **b** R band. **c** G band. **d** B band

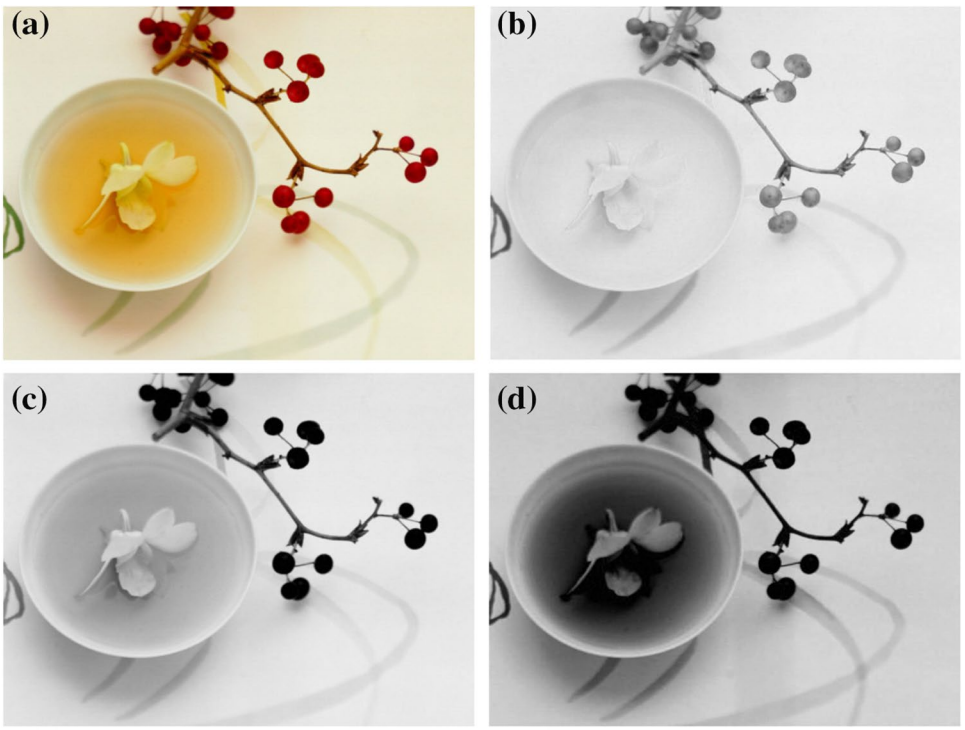

Fig. 16 **a** The tampered image "Infusion". **b** R band. **c** G band. **d** B band

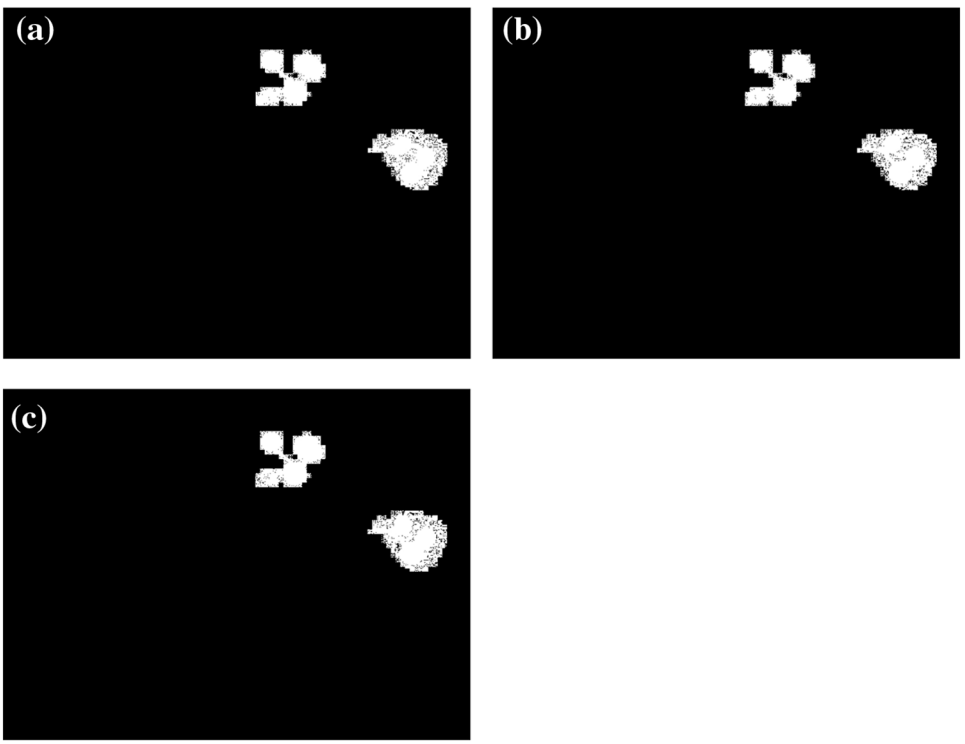

Fig. 17 **a**. Tampered zone—R band. **b** Tampered zone—G band. **c** Tampered zone—B band

Table 10 Mean sensitivity, specificity and accuracy for several ρ (Infusion)

ρ	BFRE algorithm			F-transform algorithm		
	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)
No compr.	99.45	99.97	99.71	99.32	99.96	99.64
0.25	95.51	99.94	97.73	94.67	99.93	97.30
0.16	95.17	99.93	97.55	93.94	99.91	96.93
0.0625	94.68	99.91	97.30	93.26	99.88	96.57
0.015625	92.25	99.57	95.91	90.20	99.11	94.66

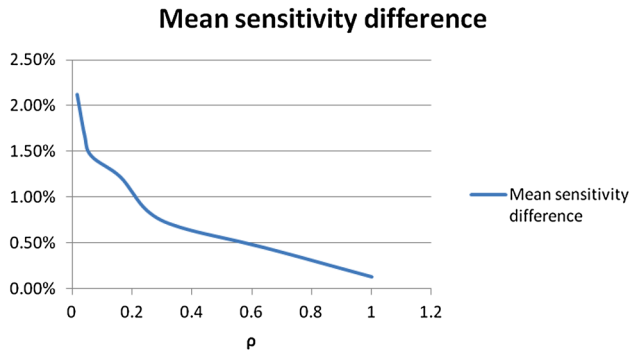

Fig. 18 Trend of the difference between the mean sensitivity values obtained by using the BFRE and F-transform algorithms for 200 images extracted from the above dataset

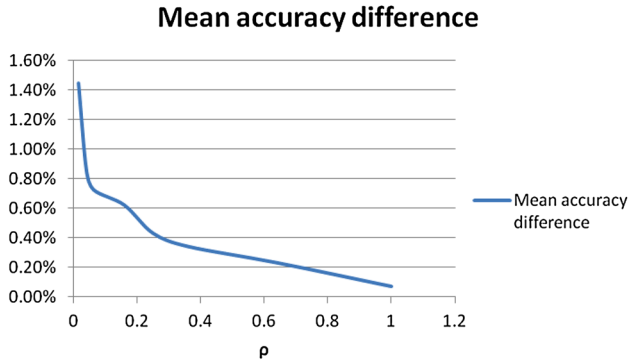

Fig. 20 Trend of the difference between the mean accuracy values obtained by using the BFRE and F-transform algorithms for 200 images extracted from the above dataset

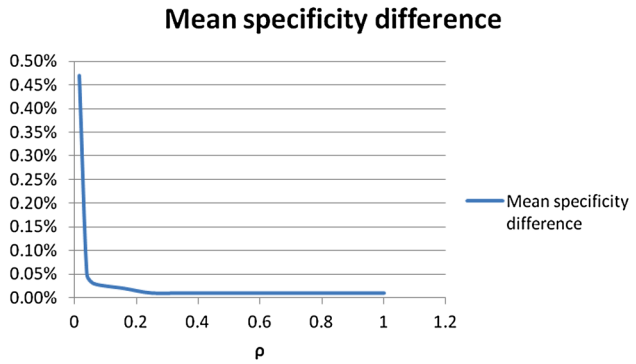

Fig. 19 Trend of the difference between the mean specificity values obtained by using the BFRE and F-transform algorithms for 200 images extracted from the above dataset

Table 11 Mean sensitivity, specificity and accuracy obtained by using various block-based fragile watermarking methods for 200 images extracted from the above dataset

Method	Mean sensitivity (%)	Mean specificity (%)	Mean accuracy (%)
BFRE	95.45	99.93	97.69
F-transform (Di Martino and Sessa 2012b)	94.31	99.91	97.11
FCM (Cen and Wang 2009)	96.12	99.96	98.04
Hierarchical (Chang and Tai 2013)	95.99	99.95	97.97
DCT (Singh and Singh 2017)	96.23	99.97	98.10
Chaos (Walton 1995)	96.04	99.96	98.01
SVD (Ansari et al. 2016)	96.28	99.97	98.13

et al. 2016; Chang and Tai 2013; Chen and Wang 2009; Singh and Singh 2017; Walton 1995).

Acknowledgements This paper was performed under the auspices of INDAM-GCNS.

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Al-Otum HA, Al-Taba'a AO (2009) Adaptive color image watermarking based on a modified improved pixel-wise masking technique. Comput Electr Engin 5:673–695. https://doi.org/10.1016/j.compeleceng.2009.01.007

Ansari IA, Pant M, Ahn CW (2016) SVD based fragile watermarking scheme for tamper localization and self-recovery. J Mach Learn Cyber 7:1225–1239. https://doi.org/10.1007/s1304201504551

Barni M (2002) Improved wavelet-based watermarking through pixel-wise making. IEEE Trans Image Process 10(5):783–791. https://doi.org/10.1109/83.918570

Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York, ISBN:0306406713

Celik MU, Sharma G, Saber E, Tekalp AM (2002) Hierarchical watermarking for secure image authentication with localization. IEEE Trans Image Process 11(6):585–595. https://doi.org/10.1109/TIP.2002.1014990

Chang YF, Tai WL (2013) A block-based watermarking scheme for image tamper detection and self-recovery. OPTO Electron Rev 21(2):182–190. https://doi.org/10.2478/s1177201300884

Chang CC, Hu YS, Lu TC (2006) A watermarking-based image ownership and tampering authentication scheme. Pattern Recogn Lett 27(5):439–446. https://doi.org/10.1016/J.PATREC.2005.09.006

Chen WC, Wang MS (2009) A fuzzy C-means clustering-based fragile watermarking scheme for image authentication. Expert Syst Appl 36:1300–1307. https://doi.org/10.1016/j.eswa.2007.11.018

Cox IJ, Miller M, Bloom J, Fridrich J, Kalker T (2008) Digital watermarking and stenography. Morgan Kaufmann, San Francisco, ISBN: 9780123725851

Di Martino F, Sessa S (2006) Digital watermarking in coding/decoding processes with fuzzy relation equations. Soft Comput 10:238–243. https://doi.org/10.1007/s0050000504779

Di Martino F, Sessa S (2012a) Digital watermarking strings with images compressed by fuzzy relation equations. In: Chatterjee PA (eds) Computational intelligence in image processing siarry. Springer, Berlin, pp 173–186. https://doi.org/10.1007/978364231062116

Di Martino F, Sessa S (2012b) Fragile watermarking tamper detection with images compressed by fuzzy transform. Inf Sci 195:62–90. https://doi.org/10.1016/j.ins.2012.01.014

Di Martino F, Sessa S (2017) Comparison between images via bilinear fuzzy relation equations. J Ambient Intell Humaniz Comput. https://doi.org/10.1007/s1265201705763

Hirota K, Pedrycz W (2002) Data compression with fuzzy relational equations. Fuzzy Sets Syst 126(3):325–335. https://doi.org/10.1016/S01650114(01)000094

Holliman M, Memon N (2000) Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes. IEEE Trans Image Process 9(3):432–441. https://doi.org/10.1109/83.826780

Li JX (1992) A new algorithm for the greatest solution of fuzzy bilinear equation. Fuzzy Sets Syst 46:193–210. https://doi.org/10.1016/0165-0114(92)90132-N

Li CT (2004) Digital fragile watermarking scheme for authentication of JPEG images. IEE Proc Vision Image Signal Process 151(6):460–466. https://doi.org/10.1049/ip-vis:20040812

Li CT, Yuan Y (2006) Digital watermarking scheme exploiting non deterministic dependence for image authentication. Opt Eng 45(12):127001. https://doi.org/10.1117/1.2402932

Li X, Zhang H, Chen M (2012) Self-recovery fragile watermarking based on superior block-wise tamper detection. In:Proceedings of 11*th* IEEE International Conference on Signal Processing, Beijng, pp. 1697–1700. https://doi.org/10.1109/ICoSP.2012.6491907

MeenakshiDevi P, Venkatesan M, Duraiswamy K (2009) Fragile watermarking scheme for image authentication with tamper localization using integer wavelet transform. J Comput Sci 5(11):831–837. https://doi.org/10.3844/jcssp.2009.831.837

Ni R, Zhao Y, Yang L, Qiao X (2013) Irregular region-wise watermarking scheme for image tampering detection and recovery. Res Notes Inform Sci 14:471–476. https://doi.org/10.4156/rnis.vol14.84

Nobuhara H, Pedrycz W, Hirota K (2002) A digital watermarking algorithm using image compression method based on fuzzy relational equations. In: Proceedings of FUZZ-IEEE 2002, Vol. 2, IEEE Press, pp. 1568–1573. https://doi.org/10.1109/FUZZ.2002.1006740

Perfilieva I (2006) Fuzzy transforms. Fuzzy Sets Syst 157 (8): 993 – 1023. https://doi.org/10.1016/j.fss.2005.11.012

Qin C, Ji P, Zhang X, Dong J, Wang J (2017) Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. Sig Process 138:280–293. https://doi.org/10.1016/j.sigpro.2017.03.033

Shih FY (2007) Digital watermarking and steganography: fundamentals and techniques. CRC Press (Taylor & Francis Group). Abingdon. ISBN: 9781420047578

Singh D, Singh SK (2017) DCT based efficient fragile watermarking scheme for image authentication and restoration. Multimedia Tools Appl 76:953–977. https://doi.org/10.1007/s110420153010x

Suthaharan S (2004) Fragile image watermarking using a gradient image for improved localization and security. Pattern Recogn Lett 25(16):1893–1903. https://doi.org/10.1016/j.patrec.2004.08.017

Tong X, Liu Y, Zhang M, Chen Y (2013) A novel chaos-based fragile watermarking for image tampering detection and self-recovery. Signal Process Image Comm 28:301–308. https://doi.org/10.1016/j.image.2012.12.003

Walton S (1995) Information authentification for a slippery new age. Dr Dobbs J 20(4):18–26

Wolfgang RB, Podilciuk CI, Delp EJ (1998) The effect of the matching watermark and compression transforms in compressed color images. In: Proceedings IEEE International conference in image processing (ICIP1), Chicago, pp. 440–444. https://doi.org/10.1109/ICIP.1998.723519