CrossMark

Hybrid blind robust image watermarking technique based on DFT-DCT and Arnold transform

Mohamed Hamidi[1] · Mohamed El Haziti[1,2] ·
Hocine Cherifi[3] · Mohammed El Hassouni[1,4]

Abstract In this paper, a robust blind image watermarking method is proposed for copyright protection of digital images. This hybrid method relies on combining two well-known transforms that are the discrete Fourier transform (DFT) and the discrete cosine transform (DCT). The motivation behind this combination is to enhance the imperceptibility and the robustness. The imperceptibility requirement is achieved by using magnitudes of DFT coefficients while the robustness improvement is ensured by applying DCT to the DFT coefficients magnitude. The watermark is embedded by modifying the coefficients of the middle band of the DCT using a secret key. The security of the proposed method is enhanced by applying Arnold transform (AT) to the watermark before embedding. Experiments were conducted on natural and textured images. Results show that, compared with state-of-the-art methods, the proposed method is robust to a wide range of attacks while preserving high imperceptibility.

✉ Mohamed Hamidi
   hamidi.medinfo@gmail.com

   Mohamed El Haziti
   elhazitim@gmail.com

   Hocine Cherifi
   hocine.cherifi@u-bourgogne.fr

   Mohammed El Hassouni
   mohamed.elhassouni@gmail.com

[1] LRIT-CNRST URAC 29, Rabat IT Center, Faculty of Sciences, Mohammed V University,
   Rabat, Morocco

[2] LRIT-CNRST URAC 29, Rabat IT Center, EST, Mohammed V University, Rabat, Morocco

[3] Laboratoire Electronique, Informatique et Image (Le2i) UMR 6306 CNRS,
   University of Burgundy, Dijon, France

[4] LRIT-CNRST URAC 29, Rabat IT Center, FLSH, Mohammed V University, Rabat, Morocco

1 Introduction

With the rapid growth of Internet and new technologies for multimedia services and the proliferation of digital devices, multimedia data can be modified, duplicated and distributed very easily. Therefore, preventing unauthorized use of these contents has become more and more important. To overcome this issue, digital image watermarking, especially robust image watermarking, is an efficient solution. The underlying concept of image watermarking is to embed a watermark within the cover image to protect it from illegal usage. The watermark must be imperceptible, so that it should not degrade the quality of the host image and it should be difficult or even impossible to counterfeit or remove it.

In general, the process of image watermarking must satisfy four requirements which are imperceptibility, robustness, capacity and security [6]. A good watermarking scheme must provide the best tradeoff between these four properties according to the requirements of the aimed application. The first important requirement of an image watermarking system is imperceptibility. It refers to perceptual similarity between the original image and the watermarked image. Indeed, an efficient watermarking scheme should produce no artifacts or quality loss in the images. If the watermarking scheme fails to achieve this requirement, it will not be suitable for practical applications. The second property is the robustness. It represents the ability of detecting the watermark even if the watermarked image has incurred changes in its distribution process. Consequently, the watermark needs to be robust against common signal processing operations such as filtering, noise addition, lossy compression, cropping, etc. The third requirement is the capacity which refers to the maximum number of bits that can be embedded in a given host data. The fourth requirement is the security of watermark. It refers to its ability to resist hostile attacks, so that unauthorized users cannot remove the watermark. In order to achieve a minimum level of security, a secret key is required in watermarking systems.

Digital image watermarking can be used in a wide variety of applications. In Copyright protection, the goal is to secure digital images in unsecured networks like Internet. The ownership can be proven in the case of dispute, by extracting the owner's copyright information embedded invisibly into the host image. Authentication is also an interesting application of image watermarking which aims to detect if any modification has been applied to the host image and then localizes exactly the tampered region. Another application is tamper detection. The presence of tampering is achieved by embedding a fragile watermark. If the watermark is degraded or destroyed, it indicates that the image cannot be trusted. This process is used in applications involving sensitive data such as medical imagery, satellite imagery, etc. Machine learning techniques can be widely used in several applications such as recognition [18, 19]. In digital image watermarking, machine learning approaches and artificial intelligence mechanisms such as neuro computing, fuzzy techniques as well as evolutionary algorithms are used in the watermarking field [10].

There are different classification criteria for watermarking systems. Based on the resistance to attacks, watermarking algorithms are divided into three main categories; fragile [41], semi-fragile [23] and robust watermarking [43]. Fragile watermarking schemes have been proposed especially for image authentication and integrity verification. They are used to detect any unauthorized modification at all. Semi-fragile techniques are implemented

for detecting any unauthorized modification, while allowing at the same time some image-processing operations. Robust watermarking algorithms are designed to survive arbitrary, malicious attacks such as image scaling, cropping, and lossy compression. They are usually used for copyright protection with the aim of declaring rightful ownership. The existing algorithms can be also classified into spatial and transform domains. Spatial domain techniques [36] embed the watermark by directly modifying the image pixels, whereas in frequency domain techniques [5] a transformation is first performed and then the watermark is embedded into discrete cosine transform (DCT) [7, 11, 32, 42], discrete wavelet transform (DWT) [2, 29–31] or discrete Fourier transform (DFT) coefficients [4, 37]. For applications, such as authentication, tamper detection, copyright protection, it is desirable to be able to extract the watermark without the original image. This requirement, introduces a very challenging issue especially if robustness is also needed. We distinguish between non-blind [25], semi-blind [34] and blind [12, 24] watermarking systems depending on whether or not the host image is needed during watermark extraction. In non-blind techniques, the original image is needed; Semi-blind methods require the watermark and some side information; Blind approaches neither need the original image nor the watermark.

In this paper, we propose a blind robust image watermarking technique for copyright protection. The watermark is embedded in middle band coefficients of DCT of the magnitude after carrying out the DFT of the original image. The choice of using DFT magnitude has been driven by the gain in terms of imperceptibility. However, it has been found that the robustness of the proposed scheme is weak when the DFT magnitude is used only. To overcome this problem, we choose to apply the DCT to the DFT magnitude thanks to its advantages especially its robustness against signal processing attacks. In addition, to enhance the security of the proposed method, the Arnold transform is used to encrypt the watermark. The gain obtained after jointing these two transforms in terms of imperceptibility and robustness is clearly illustrated in experimental results.

To evaluate the proposed scheme, we compare its DFT counterpart in terms of imperceptibility and robustness. Furthermore, comparative experiments are performed with alternative methods presented in [17, 26, 29, 34, 38] and [33].

This paper is organized as follows. Section 2 discusses the related works. Section 3 gives a description of used terminologies followed by the proposed watermarking scheme illustrated in Section 4. Section 5 sketches the experimental setup. Section 6 reports the experimental results. Finally, Section 7 concludes the paper.

2 Related work

In the literature, several watermarking methods have been proposed in the transform domain. One of the most popular watermarking scheme is introduced in [5], by Cox et al. where a pseudo-random Gaussian sequence is embedded into the largest 1000 AC coefficients in the DCT domain. Das et al. in [7] proposed a DCT watermarking method based on correlation between DCT coefficients in the same position of adjacent blocks.

Poljicak et al. [21] the watermark is inserted in the magnitude of the Fourier transform taking the advantage of minimizing the impact of the watermark implementation on the overall quality of an image. In [38], Wang et al. proposed a wavelet-tree-based blind watermarking scheme for copyright protection. The watermark bits are embedded by quantizing super trees. In [26], a blind watermarking method based on quantization of distance between wavelet coefficients is proposed. In [17], Lin et al. proposed a wavelet-tree-based

watermarking method using distance vector of binary cluster for copyright protection. The watermark is embedded into insignificant coefficients of a wavelet tree.

Note that all these solutions are based on a single transform domain and that they all try to insert the watermark in selected values in order to increase robustness and imperceptibility. The main motivation of the majority of existing watermarking schemes is to improve the robustness against a wide range of attacks while preserving a good visual quality of images. Therefore, the need to develop hybrid methods that combine two or more transforms to use the characteristics of these transforms and achieve the required aims has increased considerably [35].

In [1] a digital image watermarking scheme based on DCT and SVD is proposed. This approach used differential evolution (DE) to select adaptively the strength of the watermark and Arnold transform (AT) in order to enhance security. First, the host image is divided into 8×8 square blocks and then the DCT is applied on each block. Afterwards, the DC components of DCT coefficients are collected with the aim of constructing the low resolution approximation matrix. Finally, the watermark image is scrambled using the Arnold transform (AT) and embedded into the diagonal matrix S using a scaling factor obtained with a differential evolution algorithm. In [9], a blind watermarking algorithm based on Discrete Wavelet Transform (DWT) and Discrete Cosine Transform (DCT) is proposed. The watermark is scrambled by Arnold transform and embedded in a spread spectrum pattern using pseudo random in the mid frequency coefficients of the corresponding DCT blocks of DWT LL sub-band. Experimental results show that combining the two transforms gives better results than using DCT only. In [27], a blind robust image watermarking method based on DWT-SVD and DCT using Arnold Cat Map encryption for copyright protection is proposed. The DCT coefficients of the watermark image are embedded into the middle singular value of each block having size 4×4 of the host image's one level Discrete Wavelet Transform (DWT) sub-bands. This scheme is secure, imperceptible and robust against common signal processing operations.

In [28], Amit Kumar Singh et al. proposed a hybrid method based on DWT-DCT-SVD. The original image is decomposed into first level DWTs. Next, the DCT and SVD are applied to the low frequency band (LL). Afterwards, the watermark image is transformed also using the DCT and SVD. Then, the S component of watermark is inserted in the S component of the host image. The method is robust against signal processing attacks.

In [24], Soumitra Roy et al. proposed a RDWT-DCT based blind image watermarking scheme using Arnold scrambling. First, the original image is decomposed into non overlapping blocks and the RDWT (Redundant Discrete Wavelet Transform) is carried out to each block. Second, the watermark is encrypted using Arnold chaotic map to increase the security. Then, the DCT is applied to each LH subband of the non-overlapping host image block. Finally, the watermark is embedded by modifying middle significant AC coefficients using repetition code. Soumitra's method is shown to be robust against geometric attacks, jpeg compression among others.

In [34], a semi-blind gray scale image watermarking technique in redundant wavelet domain using the combination of non sub-sampled contourlet transform (NSCT), redundant discrete wavelet transform (RDWT) and SVD is proposed. Singh's et al. used Arnold transform encryption to enhance the security of the watermarking system. This method is shown to be robust to geometrical and signal processing attacks. In [29], a secure DWT-DCT-SVD based image watermarking technique is proposed. First, the host image is decomposed up to second level of DWT. Second, the DCT and SVD have been applied to the low frequency ll of the original image. The watermark medical image is also transformed by DCT and SVD. The watermark embedding is performed by inserting the singular value of watermark

image in the singular value of the original image. In [33], an hybrid image watermarking technique based on Nonsubsampled contourlet transform(NSCT), Multiresolution Singular value decomposition (MSVD), discrete cosine transform (DCT) and Arnold transform is proposed. In this method, three image watermarks have been added into the cover image exploiting the advantages of combining these transforms to enhance robustness, capacity and imperceptibility requirements.

Recently, an interesting survey of image watermarking techniques and their application have been proposed [14]. Kumar et al. have discussed recent state-of-art watermarking techniques issues and potential solutions. The work can be useful for secure e-governance applications.

The major limitation of the existing image watermarking schemes for copyright protection is the difficulty to ensure a good tradeoff between imperceptibility and robustness. To take full advantage of image transforms, we design a novel combination of DFT and DCT for blind robust image watermarking. The reason behind this choice is due to the fact that the DFT magnitude shows ability to ensure high imperceptibility while the DCT can improve the robustness of the proposed technique to common signal processing attacks. Furthermore, Arnold transform is used to enhance the security of the proposed watermarking system.

3 Preliminaries

3.1 Discrete Fourier transform (DFT)

Discrete Fourier transform of an image leads to magnitude and phase representation. This transformation has several characteristics. An important property of the DFT is its translation invariance. In fact, spatial shifts doesn't affect the magnitude but affect the phase component [20]. DFT is also robust to cropping. In fact, when the watermark is embedded in the magnitude, even if the spectrum is blurred, the synchronization is not needed. The discrete Fourier transform of an image $f(x, y)$ of size $M \times N$ and the inverse DFT (IDFT) are defined respectively as follows:

$$
\begin{aligned}
F(u, v) &= \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-2\pi j \left(\frac{ux}{M} + \frac{vy}{N} \right)} \\
&= R(u, v) + j I(u, v)
\end{aligned}
\tag{1}
$$

$$
F(u, v) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{2\pi j \left(\frac{ux}{M} + \frac{vy}{N} \right)}
\tag{2}
$$

Where $R(u, v)$ and $I(u, v)$ are the real and the imaginary parts of the Fourier transform, respectively. Equation shows the polar representation of the Fourier transform:

$$
F(u, v) = |F(u, v)| e^{j\phi(u, v)}
\tag{3}
$$

Where $|F(u, v)|$ and $\phi(u, v)$ are respectively the Fourier magnitude and the Fourier phase. They are represented as follows:

$$
M(u, v) = |F(u, v)| = [R^2(x, y) + I^2(x, y)]^{1/2}
\tag{4}
$$

$$
\phi(u, v) = tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]
\tag{5}
$$

Where $R(u, v)$ and $I(u, v)$ are respectively the real and imaginary parts of $F(u, v)$.

3.2 Discrete cosine transform (DCT)

The discrete cosine transform (DCT) is one of the famous transformation technique that transforms an image from the spatial domain to the frequency domain [13]. It has been widely applied in image processing exploiting both the decorrelation and the energy compaction properties. Generally, DCT watermarking approaches used square matrix of 8×8 as block size. The mathematical expressions of the 2D-DCT and inverse 2D-DCT are respectively:

$$
\begin{aligned}
C(u, v) = & \frac{2}{\sqrt{mn}} \alpha(u)\alpha(v) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \\
& \times cos \frac{(2x+1)u\pi}{2m} \times cos \frac{(2y+1)v\pi}{2n}
\end{aligned}
\tag{6}
$$

$$
\begin{aligned}
f(x, y) = & \frac{2}{\sqrt{mn}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v) C(u, v) \\
& \times cos \frac{(2x+1)u\pi}{2m} \times cos \frac{(2y+1)v\pi}{2n}
\end{aligned}
\tag{7}
$$

Where $f(x, y)$ and $C(u, v)$ are respectively the pixel values in the spatial domain and the DCT coefficients. m, n represent the block size.

$$
\alpha(u)\alpha(v) = \begin{cases} \frac{1}{\sqrt{2}} & if (u, v) = 0 \\ 1 & else \end{cases}
\tag{8}
$$

3.3 Arnold transform

The idea behind using scrambling algorithm is to enhance the security of the watermark in order to avoid unauthorized person to counterfeit or remove it. Therefore, it guarantees more safety and reliability for the image in the transmission process. Arnold scrambling is widely used in digital image watermarking due to its simplicity and periodicity [8, 16]. According to this periodicity, after several cycles, the host image can be easily restored. The obtained watermark image after the scrambling process is chaotic, so without the scrambling algorithm and the key, the attacker cannot decrypt it even if it has been extracted from the watermarked image. Moreover, the spatial relationships between the pixels have been destroyed which ensures more security. We note that the period of Arnold transform should less than $\frac{N^2}{2}$ with N the image size [8].

The result of applying the Arnold transform is randomly organizing the pixels of the image. The principle idea is that if iterated enough times, the host image reappears. Note that the parameters of Arnold transform serve as the additional key for enhancing the security.

Arnold transform, also called cat map transform, is defined as:

$$
\begin{bmatrix} a' \\ b' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} mod(n)
\tag{9}
$$

Where (a, b) and (a', b') are the pixel coordinates of the original watermark and the encrypted watermark, respectively. Let A the left matrix in the right part of (9), $I(a, b)$ and $I(a', b')^{(n)}$ represent pixels in the original watermark and the encrypted one obtained

by performing Arnold transform n times, respectively. Thus, using n times the Arnold transformation, the watermark encryption can be written as:

$$I(a', b')^{(p)} = AI(a, b)^{(p-1)} mod(n) \qquad (10)$$

Where $p = 1, 2, \ldots, n$, and $I(a', b')^{(0)} = I(a, b)$. In fact, to obtain $I(a, b)^{(p-1)}$ we multiply the inverse matrix of A at each side of (10). That is to say, by iteratively calculating the formula (11) n times, the encrypted watermark can be decrypted easily.

$$J(a, b)^{(p)} = A^{-1} J(a', b')^{(p-1)} mod(n) \qquad (11)$$

Where $J(a', b')^{(0)}$ is a pixel representation of the encrypted watermark and $J(a, b)^{(p)}$ is the decrypted pixel after p iterations. Figure 1 shows the watermark image encryption using Arnold transforms, where (a) is the original watermark sized 64×64, and (b) is the encrypted one with $n = 50$.

4 Proposed scheme

In this work, we propose a blind robust image watermarking method for copyright protection. The watermark is inserted in the DCT middle band of the DFT magnitude. The reason behind the choice of DFT magnitude has been driven by the gain in terms of watermark imperceptibility. Nevertheless, the scheme shows robustness weakness when the DFT magnitude is used only. Since the DCT is very robust against signal processing attacks, we believe it is an excellent solution to consider in our scheme. For this reason, the DCT is applied to the DFT magnitude to enhance the watermark robustness. Furthermore, the watermark is encrypted with the Arnold transform to increase the security of the proposed method.

4.1 Watermark embedding

The proposed embedding mechanism is illustrated in Fig. 2. First of all, a prepossessing is applied to the watermark. It consists on Arnold transformation which is used to encrypt the
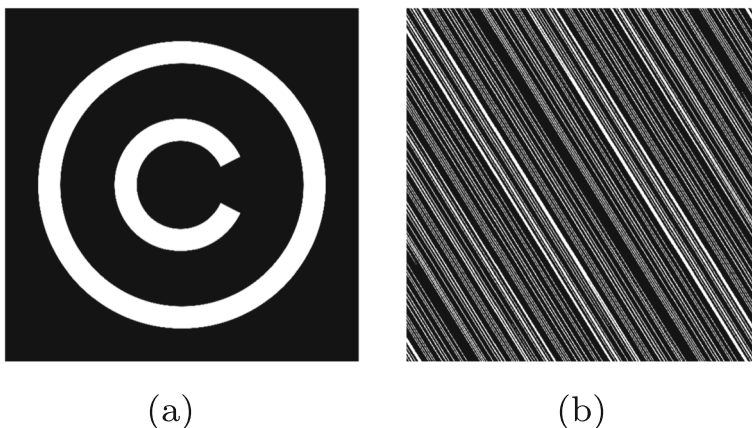

(a)   (b)

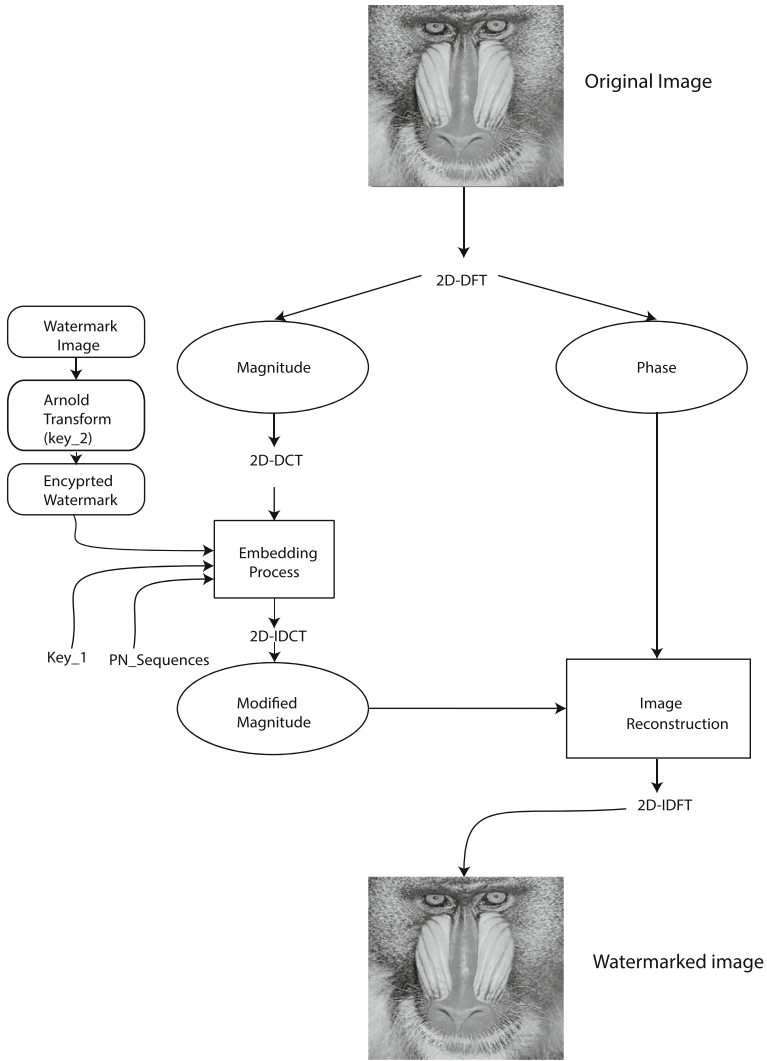**Fig. 1** **a** Original watermark, **b** Encryped watermark

Fig. 2 The embedding process of the proposed scheme

watermark as a security enhancement. The idea behind this transformation is to make difficult even impossible for an unauthorized person, without knowing the scrambling algorithm and parameters that represent an additional secret key (key 2), to detect the real watermark even if it's correctly extracted. Initially, the DFT is applied to the original image then the the magnitude $M(u, v)$ and the phase $\phi(u, v)$ are calculated using (4) and (5). Afterwards, the magnitude matrix is divided into square blocks of size 8×8. Then, the DCT is computed on every block of the magnitude. Next, using the secret key 1, two uncorrelated pseudo-random

sequences are generated: one sequence for "0" bits (PN_Seq_0) and another sequence for the "1" bits (PN_Seq_1). Note that PN sequences must have the same size than the middle band coefficients.

The embedding process consists of inserting the PN-Sequences according to the bit value of the watermark using (12). $M(u, v)$ is obtained after applying the DCT to the DFT magnitude. The watermark $W(u, v)$ that consists of two PN sequences, is inserted in the middle band coefficients. The strength of embedding is adjusted by the parameter k which controls the tradeoff between the robustness and imperceptibility. The middle band coefficients of the DCT transform of the DFT magnitude are used for watermark embedding to avoid modifying the important visual parts of image. The original DFT magnitude and the modified one are depicted in Fig. 9.

$$M_w(u, v) = \begin{cases} M(u, v) + k * W(u, v) & u, v \in F_M \\ M(u, v) & u, v \notin F_M \end{cases} \qquad (12)$$

Where $M_w(u, v)$ is the watermarked magnitude block, $M(u, v)$ represents the 8×8 DCT block of DFT magnitude, $W(u, v)$ represents the watermark which consists of two PN sequences, F_M refers to the middle frequency band which is modified during the watermark embedding, and k is the watermark strength that controls the tradeoff between imperceptibility and robustness requirements.

The watermark consists of two pseudo-random sequences PN_Seq_0 and PN_Seq_1 (see Algorithm 1). Each sequence is a vector composed by $\{-1, 1\}$ values with a normal distribution having zero mean and unity variance. The motivation behind this choice (normally distributed watermark) is the robustness to the attacks trying to produce an unwatermarked document by averaging multiple differently watermarked copies of it [22]. In the detection side, it is important that the PN sequences are statistically independent. This constraint is granted by the pseudo-random nature of the sequences. In addition, such sequences could be easily regenerated by providing the correct seed (key 1).

The watermark strength is handled by the gain factor k which controls the tradeoff between robustness and imperceptibility. In fact, an increase of the gain factor increases the watermarking robustness while it decreases the imperceptibility of the watermark. Thus, we choose empirically the value of k so that we have a good tradeoff between robustness and imperceptibility. After, inverse DCT is applied to obtain the modified magnitude. Finally, the watermarked image is reconstructed with the unchanged phase and the modified magnitude using (13).

$$I_w(u, v) = M_w(u, v) * e^{(j\phi(u,v))} \qquad (13)$$

Afterwards, the inverse discrete Fourier transform (IDFT) is performed to obtain the watermarked image.

Figure 2 sketches the watermark embedding process which is described in detail in Algorithm 1.

Algorithm 1 Watermark embedding

Require: Original image, Watermark, key 1, key 2, PN_Seq_0, PN_Seq_1.
Ensure: Watermarked image.
   1. Perform Arnold transform to the watermark image using the secret key 2.
   2. Apply DFT to the original image and calculate the magnitude and the phase.
   3. Generate two uncorrelated PN sequences for middle frequency band coefficients using the secrete key 1.
   4. Divide the magnitude of DFT into 8×8 blocs and then apply the DCT.
   5. Insert the two PN sequences bits according to watermark bits using the (12)
   if Watermark (bit) $= 0$ **then** $W(u, v) = $ PN_Seq_0
      else $W(u, v) = $ PN_Seq_1.
   6. Perform IDCT on each watermarked magnitude block.
   7. Reconstruct the watermarked image with the modified magnitude using the (13) [22].
   8. The final watermarked image is obtained by performing the IDFT.

4.2 Watermark extraction

With the knowledge of the secret key 1 used during the embedding process and the key 2 used during the pretreatment of watermark, the extraction process is blind and quite simple, as shown in Fig. 3. Thus, the proposed method is blind since only two private keys (key 1 and key 2) are needed.

It is sufficient to perform the 2D-DFT of the watermarked image and calculate the DFT magnitude. With the same secret key (key 1) than in the embedding process two PN sequences are generated. Thereby, we obtain the same PN sequences. Then, the 2D-DCT is applied to the DFT magnitude. In the extracting process, as shown in Fig. 3, the middle-band frequency coefficients of each 8×8 DCT bloc are extracted. Afterwards, for each bloc, the correlation between the middle band frequencies coefficients and the two PN sequences is computed. Then, the encrypted i^{th} watermark bit is extracted using (14). Finally, the inverse Arnold transform using the key 2 is applied to extract the watermark. The proposed extraction scheme is further described in Algorithm 2.

Algorithm 2 Watermark extracting

Require: Watermarked image, key 1, key 2.
Ensure: Watermark.
   1. Apply DFT to the watermarked image and calculate the magnitude.
   2. Generate two PN sequences (PN_Seq_0 and PN_Seq_1) using the same secrete key used in the embedding process.
   3. Apply DCT to the Magnitude of DFT and extract the middle frequency band coefficients.
   4. Calculate the correlation between the middle frequency band coefficients F_M and the two PN sequences.
   5. Extract the ith Watermark bit W_i as follows:

$$W_i = \begin{cases} 0 & if & Corr(0) > Corr(1) \\ 1 & if & Corr(1) > Corr(0) \end{cases} \tag{14}$$

Where $Corr(0)$ is the correlation between the middle frequency band coefficients of ith block and PN_Seq_0, and $Corr(1)$ is the correlation between the middle frequency band coefficients of ith block and PN_Seq_1.
   6. Perform the inverse Arnold transform and extract watermark image.
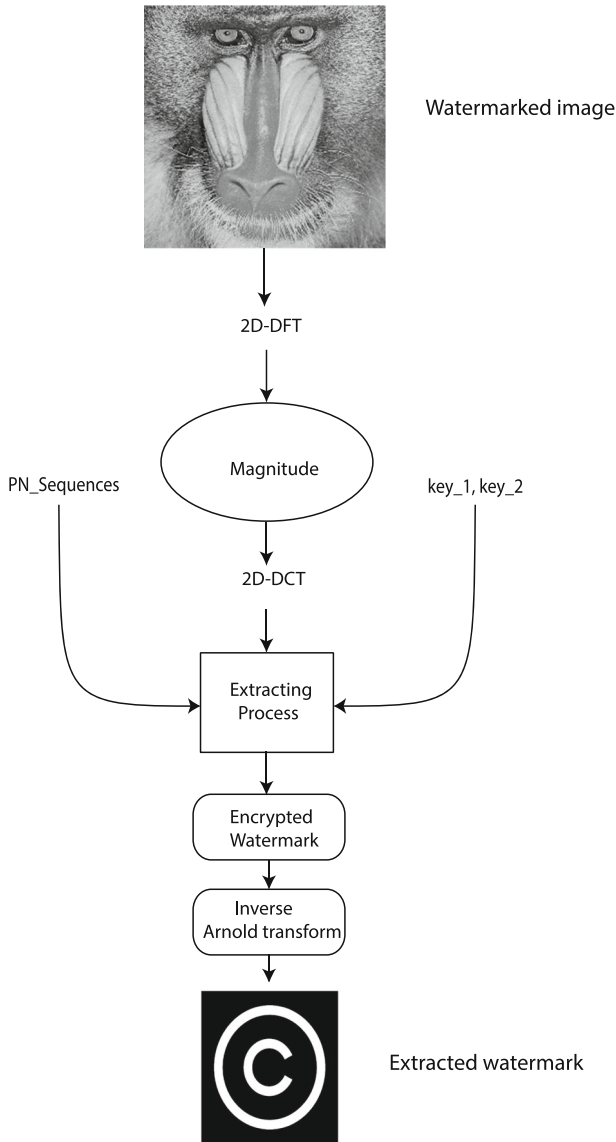

Fig. 3 The extracting process of the proposed scheme

5 Experimental setup

5.1 Studied images

To evaluate the performance of the proposed watermarking scheme, several experiments have been conducted on 10 natural 8-bit grayscale images of size 512×512 as depicted in Fig. 7a–j ("Mandril", "Peppers", "Cameraman", "Lena", "Goldhill", "Walkbridge", "Woman_blonde", "Livingroom", "Pirate", and "Lake") and a set of 39 textured images

provided by the University of Southern California [40]. The majority of these images come from the standard texture image Brodatz database [3]. Fig. 4a–j shows a sample of 10 test textured images taken from [3]. A (19 × 52) binary logo is used as watermark as shown in Fig. 12a. Another binary logo of size 64 × 64 is used for comparison purpose as shown in Fig. 1. To increase the security and the safety of the watermarking method, the watermark image logo has been scrambled using Arnold transform. Parameter value of Arnold transform, which refers to the secret key (key 2), is taken as $n = 24$, where n denotes the cycle of Arnold scrambling. The parameter k which denotes the embedding strength of the embedded watermark is chosen in such a way that ensure the best tradeoff between imperceptibility and robustness. To this end, extensive experiments have been conducted using empirically several values of k to find out the value ensuring this tradeoff. According to these experiments, the best found value is $k = 9600$ (see Figs. 5 and 6). Note that this value is the best for the proposed work with or without attack. All the experiments were coded by MATLAB R2013a and implemented on a PC with CPU Intel(R) Core(TM) i5-3470 @ 3.2 GHz with 4-GB of RAM.
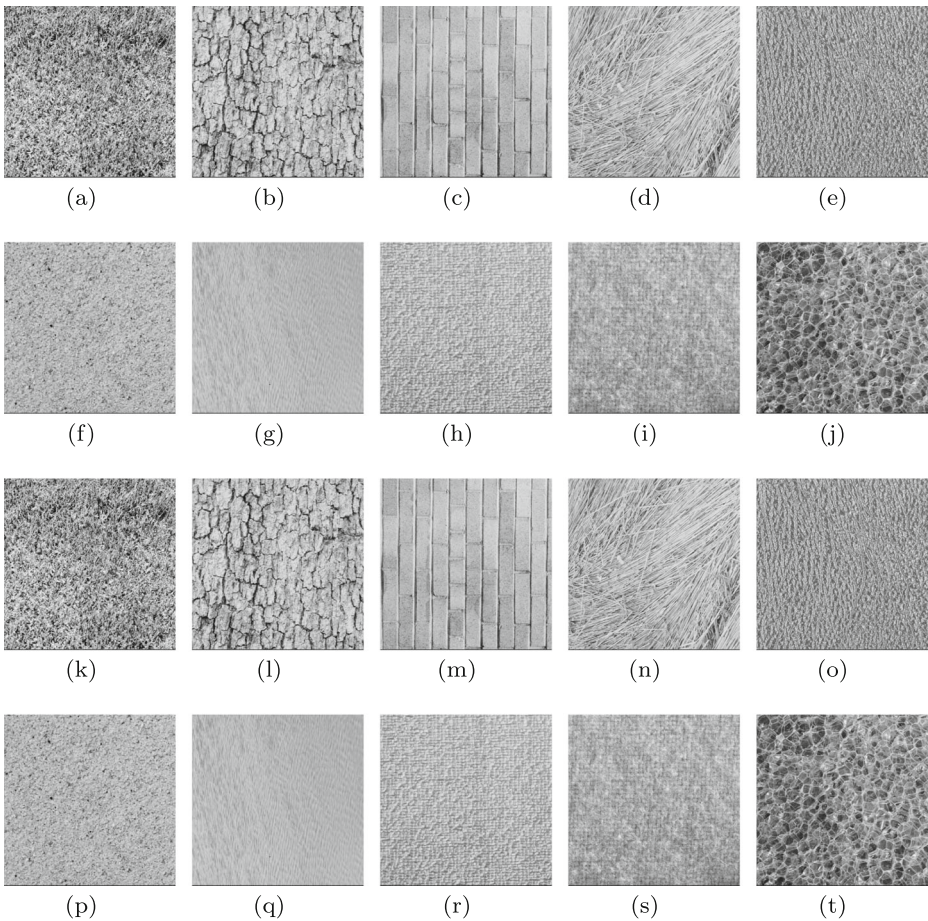

Fig. 4 Sample of original textured images from Brodatz [3]: (**a–j**). Watermarked textured images: (**k–t**)
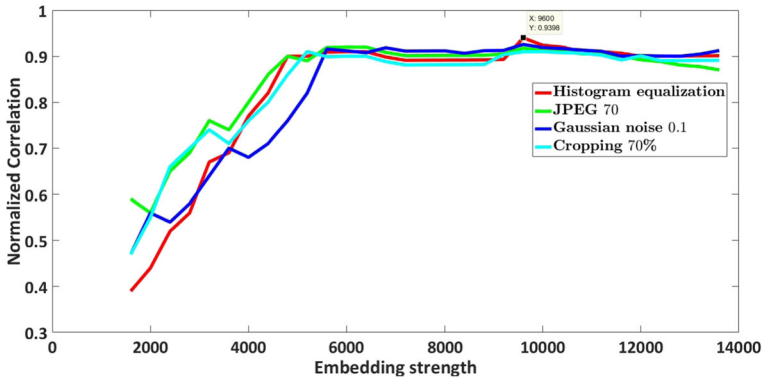
Fig. 5 NC values after variation of embedding strengths under some attacks for Lena

5.2 Evaluation metrics

Numerous metrics have been proposed in the literature to evaluate the quality of images [15]. When the original image is known, a distance between the original image and the processed one is usually computed. The challenge is to perceptually tune the distance such that the predicted quality is in agreement with human quality judgments. Peak Signal to Noise Ratio (PSNR) is the most widely used metric in the watermarking literature to measure the distance between the original image and the watermarked one. Althought, it is well recognized that it is does not correlate with human perception we use it for comparative purposes it in this work. It is defined as follows:

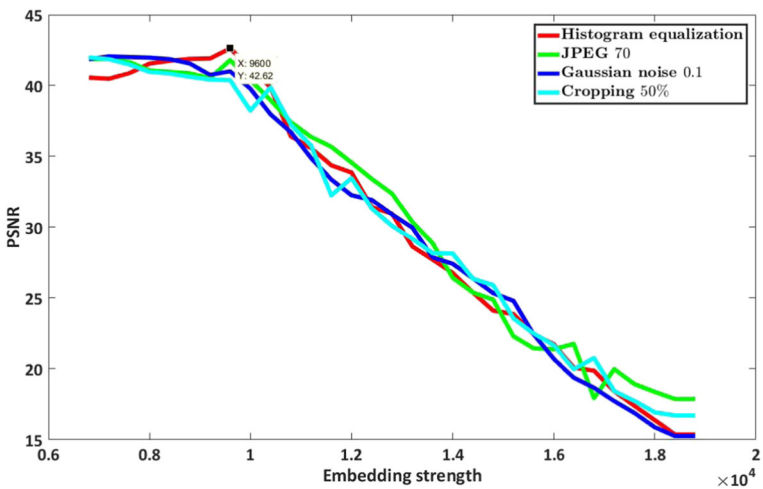$$PSNR = 10 \log \left(\frac{MAX^2}{MSE} \right) \tag{15}$$

Fig. 6 The obtained PSNR values using different embedding strength values after several attacks for Lena image

Where MAX is the maximum possible pixel value of the image which is equal to 255 for an $8-$bit per pixel representation, and MSE is given by:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \tag{16}$$

Where $I(i, j)$ and $K(i, j)$ refers to the original image and the watermarked image respectively. Basically, when the distortions decreases the PSNR increases.

The structural similarity (SSIM) index performs similarity measurement using a combination of three heuristic factors luminance comparison, contrast comparison, and structure comparison. It is the most influential perceptual quality metric [39]. It is defined by (17).

$$SSIM(I_0, I_w) = \frac{(2\mu_{I0}\mu_{Iw} + c_1)(2\sigma_{I_0 I_w} + c_2)}{(\mu_{I0}^2 + \mu_{Iw}^2 + c_1)(\sigma_{I0}^2 + \sigma_{Iw}^2 + c_2)} \tag{17}$$

Where, I_0 and I_w are respectively the original image and the watermarked image, μ_{I0} and μ_{Iw} are respectively the local means of $I0$ and Iw, σ_{I0}^2 is the variance of $I0$ whereas σ_{Iw}^2 is the variance of Iw, c_1 and c_2 are two variables to stabilize the division with weak denominator.

Robustness measures the ability of the watermark to resist against removal due to intentional or unintentional attacks. Indeed, Watermarks should survive standard data processing, such as would occur in a creation and distribution process and also to malicious attack. The normalized correlation (NC) is a widely used attribute for quantifying the robustness of the underlying watermarking technique against various attacks. It measures the similarity between the extracted watermark and the original watermark. It is defined by:

$$NC = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} [W(i, j) \times W'(i, j)]^2}{\left(\sqrt{\sum_{i=1}^{P} \sum_{j=1}^{Q} [W(i, j)]^2} \ \sqrt{\sum_{i=1}^{P} \sum_{j=1}^{Q} [W'(i, j)]^2} \right)} \tag{18}$$

Where, W and W' are the original and the extracted watermarks, respectively.

Several common attacks have been applied to these images in order to evaluate the performance of the proposed watermarking algorithm in terms of robustness and imperceptibility. Furthermore, we choose to compare our scheme with schemes presented in [26, 38], and [17] in terms of imperceptibility and robustness because they provide a clear presentation and description of their experimental results.

6 Experimental results

6.1 Imperceptibility

In order to evaluate the imperceptibility of the proposed scheme, we calculate the PSNR and the SSIM between the original image and the watermarked image, respectively. Moreover, the absolute difference between watermarked images and original images has been calculated for all test images. For the brevity of space we have given only two, corresponding to the images "Mandrill" and "D94".

Table 1 shows the imperceptibility results in terms of PSNR and SSIM using a (19×52) logo as watermark. From Fig. 7 and Table 1, it can be seen that the watermarked images preserve good visible quality and thus there is no visual distortion. Besides, all the obtained PSNR values are above 57 dB and the SSIM values are close to 1. In addition, the PSNR

Table 1 Watermark imperceptibility measured in terms of PSNR (dB) and SSIM

Natural images	PSNR	SSIM	Textured images	PSNR	SSIM
Mandrill	61.28	1.0	D9	58.97	1.0
Lena	61.97	0.9998	D12	58.11	1.0
Peppers	65.97	1.0	D94	58.26	0.9999
Cameraman	63.54	0.9999	D15	58.18	1.0
Goldhill	66.37	0.9999	D24	57.95	1.0
Walkbridge	59.24	0.9999	D29	58.03	1.0
Woman_blonde	57.31	0.9998	D38	57.00	0.9998
Livingroom	59.37	0.9999	D84	57.96	1.0
Pirate	58.82	0.9999	D19	58.38	0.9999
Lake	58.67	1.0	D112	58.41	1.0
Average	61.25	0.99991	Average	58.12	0.99996

average value of natural images slightly exceeds the PSNR average of the textured images. These results demonstrate that the proposed method is very insensitive to the image nature.

It can be observed from the difference between the watermarked images and the original images shown in Fig. 8 that the modified part are spread out over the image. This is due to the fact that the watermark is embedded in all the coefficients of the middle band of DCT. It can be concluded from the above figures that all the obtained values after calculating the difference between original images and watermarked images are close to 0. In addition, as shown in Fig. 9, there is no visual difference between the original magnitude and the modified one after watermark embedding. Furthermore, we have compared the histogram of the original and watermarked images to check if there is any clue that the image has been watermarked. Indeed, in some watermarking techniques the distribution of the watermarked image is unbalanced, suggesting the presence of a watermark. Due to space limitations, we report only two typical results, corresponding to the images "Lena" and "D19" in Fig. 10 . We can observe from this Figure the similarity between the shape of the histograms of the host image and the watermarked image.

In order to quantify the impact of using two transforms in the proposed scheme, we compare the performance in terms of PSNR and SSIM between the DFT-DCT and the DFT-only based approach. It can be seen from Table 2 that the combination of the two transforms DFT-DCT gives better results in terms of imperceptibility than the DFT-only based approach for all the test images.

6.2 Robustness

In order to evaluate the robustness of the proposed scheme, we calculate the normalized correlation (NC) between the original watermark and the extracted one. Before applying attacks, it can be observed that the watermark was extracted perfectly with a correlation NC = 1. To test the algorithm robustness, the watermarked images are exposed to various attacks: 1) noising attack: Gaussian Noise (GN) and salt & pepper noise (SPN); 2) format-compression attack: JPEG and JPEG2000 compression; 3) image-processing attack: low-pass Gaussian Filtering (LPGF), Gaussian smoothing(GS), and histogram equalization (HE); 4) Geometric distortion: cropping (Cropp) and rotation. 5) Combined attacks: histogram equalization & Gaussian noise (HE+GN), histogram equalization and salt & pepper
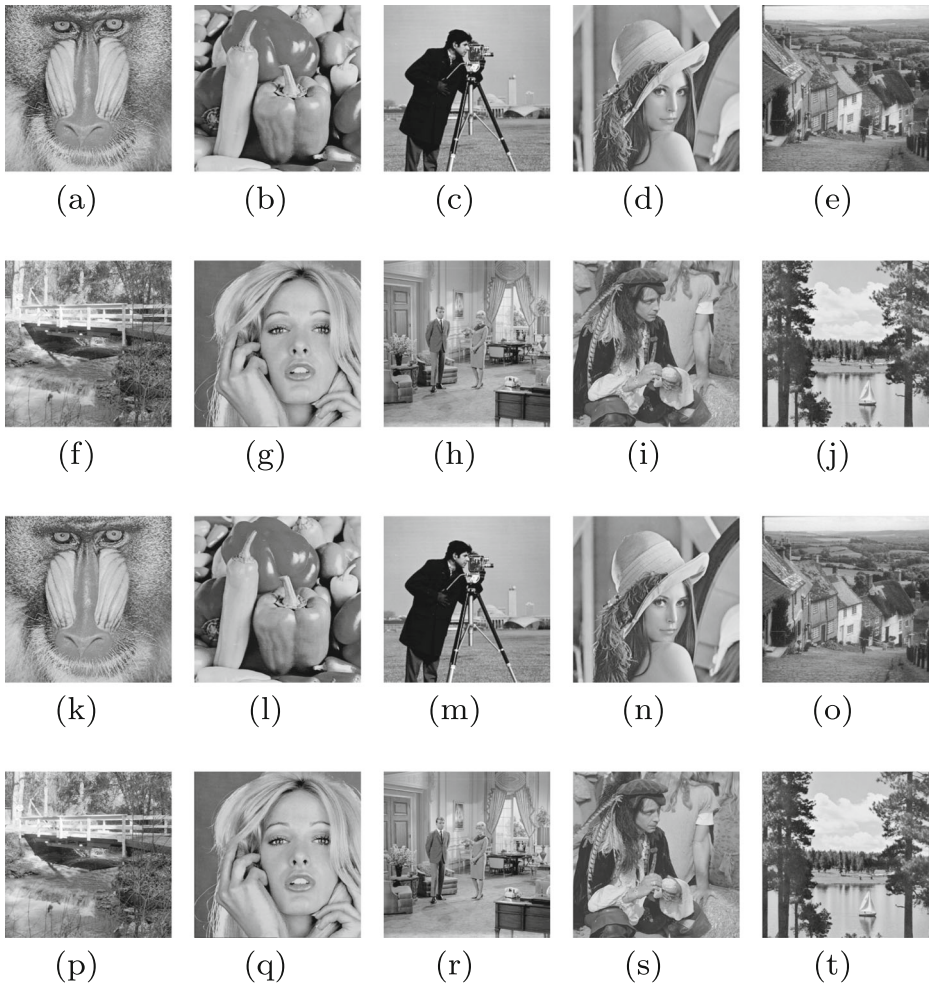
Fig. 7 Original images: **a** Mandril, **b** Peppers, **c** Cameraman, **d** Lena, **e** Goldhill, **f** Walkbridge, **g** Woman_blonde, **h** Livingroom, **i** Pirate, **j** Lake; **k** Watermarked Mandrill **l** Watermarked Peppers, **m** Watermarked Cameraman, **n** Watermarked Lena, **o** Watermarked Goldhill, **p** Watermarked walkbridge **q** Watermarked woman_blonde, **r** Watermarked Livingroom, **s** Watermarked Pirate, **t** Watermarked Lake

Fig. 8 Absolute difference of original image and watermarked image: **a** Mandrill, **b** D94

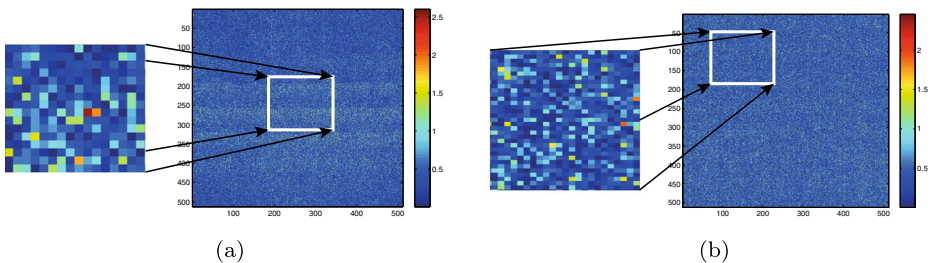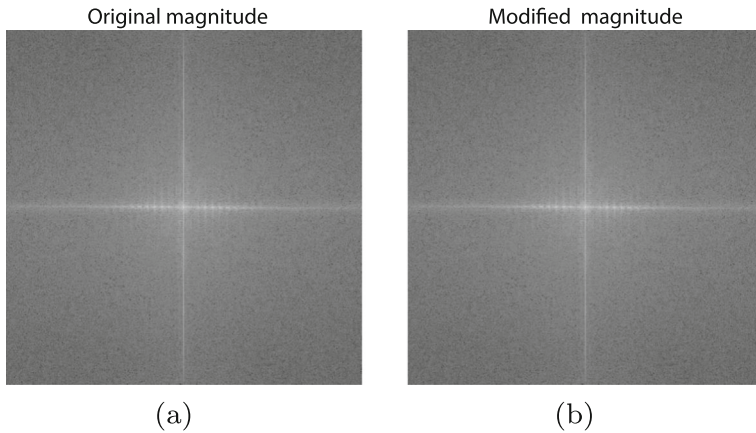**Fig. 9** **a** Original DFT magnitude **b** modified DFT magnitude of Mandrill with ($k = 9600$)

noise (HE+SPN), Gaussian noise & JPEG (GN+JPEG), salt & pepper noise and JPEG (SPN+JPEG), low-pass Gaussian filtering and salt & pepper noise (LPGF+SPN). Figure 11 represents the obtained NC values after a wide rang of attacks applied to a simple of textured images taken from Brodatz database [3]. It can be seen from Tables 5, 3 and Fig. 11 that the proposed scheme is very robust to histogram equalization regardless of the image nature. Figure 12 displays the extracted watermarks after several attacks (Histogram equalization , Salt & Pepper noise, JPEG compression, Gaussian smoothing, cropping,etc.). We can see visually that although the watermarked images are exposed to these attacks, the watermark is almost extracted perfectly (Fig. 12).

In order to improve further the robustness Fig. 13 shows the watermarked Mandrill image after several attacks of our approach, we compare the performance in terms of NC between the DFT-DCT and the DFT-only based method after carrying out several kind of attacks. It can be observed from Table 5 that the DFT-DCT approach improved the robustness performance considerably when compared to the DFT-only watermarking method.
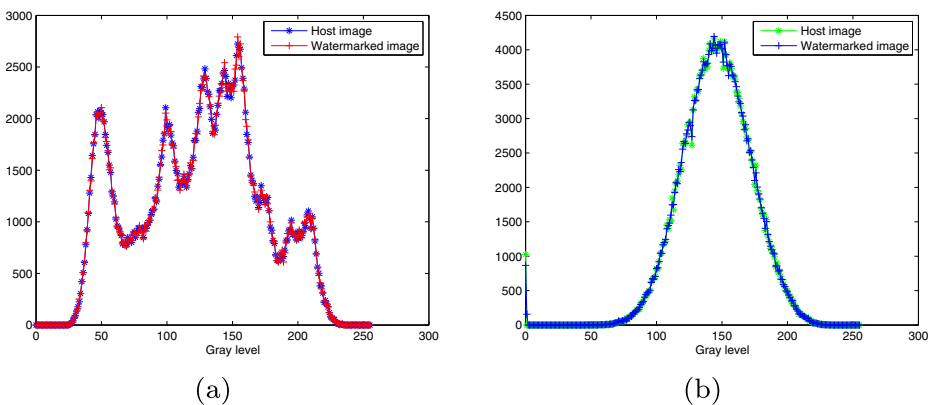

Fig. 10 **a**: Imperceptibility illustration through histograms taking image Lena, **b**: Imperceptibility illustration through histograms taking image D19 taken from Brodatz

Table 2 Comparison of the imperceptibility between the DFT only and the DFT-DCT based algorithm for several images

Cover image	Watermarking methods			
	DFT only		DFT-DCT	
	Imperceptibility metric			
	PSNR	SSIM	PSNR	SSIM
Mandrill	45.58	0.9871	61.28	1.0
Lena	47.31	0.9785	61.97	0.9998
Peppers	48.21	0.9745	65.97	1.0
Cameraman	46.15	0.9803	63.54	0.9999
Goldhill	49.73	0.9762	66.37	0.9999
Walkbridge	41.02	0.9778	59.24	0.9999
Woman_blonde	39.87	0.9812	57.31	0.9998
Livingroom	43.66	0.9788	59.37	0.9999
Pirate	44.81	0.9801	58.82	0.9999
Lake	44.05	0.9796	58.67	1.0
Average	45.039	0.9794	61.25	0.9991
D9	44.37	0.9692	58.97	1.0
D12	44.63	0.9655	58.11	1.0
D94	44.58	0.9745	58.26	0.9999
D15	44.66	0.9758	58.18	1.0
D24	43.92	0.9678	57.95	1.0
D29	43.98	0.9754	58.03	1.0
D38	43.45	0.9620	57.00	0.9998
D84	44.01	0.9685	57.96	1.0
D19	44.85	0.9588	58.38	0.9999
D112	45.02	0.9734	58.41	1.0
Average	44.347	0.9691	58.125	0.9996

6.2.1 Noising attack

First, we carried out the addition of Gaussian noise with zero mean ($\mu = 0$) and several variance values (GN1 : $\sigma = 0.001$, GN2 : $\sigma = 0.005$, $\sigma = 0.01$, $\sigma = 0.02$ and $\sigma = 0.1$) in order to better understand the limitations of the proposed method. As reported in Fig. 11 and Table 4, it appears that for variance values below 0.02 the proposed scheme is quite robust to noise addition (NC = 1.0). For higher variance values, the NC values decreases slightly but the results are still good ($\sigma = 0.1$, NC = 0.9417). Second, Salt & Pepper noise (SPN) has also been applied with zero mean ($\mu = 0$) and several intensities ($\sigma = 0.001$, $\sigma = 0.005$, $\sigma = 0.01$ and $\sigma = 0.02$) with the aim of analyzing the limitation of the proposed work. It clearly appears from Fig. 11 and Table 4 that the method show good robustness against salt & pepper noise for intensity values less than 0.02. For an intensity value equal to 0.02 the results are encouraging (NC = 0.9843).
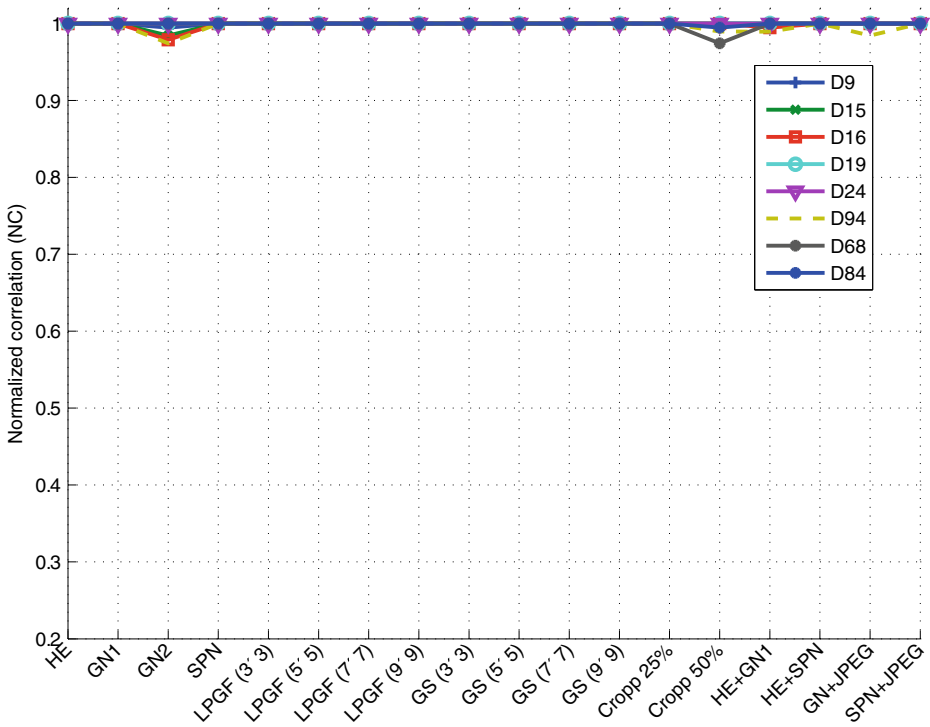
Fig. 11 Robustness in terms of NC after several attacks applied to a simple of textured images

6.2.2 Compression attack

Robustness against lossy compression is of crucial importance due to the wide diffusion of lossy compression tools and the huge use of this image format. To assess the performance from this point of view, we iteratively applied JPEG compression to the watermarked images, each time increasing the quality factor, i.e decreasing the compression ratio, ranging from 10 to 90. Note that the quality factor for images is an integer value ranging from 1 to 99, which denotes the predetermined image quality. Moreover, robustness against JPEG2000 using different compression ratios has been investigated. The compression ratio (CR) is varied from 1 to 10. Fig. 14 shows the robustness in terms of NC against JPEG2000 compression for both natural and textured images. It can be seen from Fig. 14, Tables 3 and 5 that the NC values are close to 1 when CR < 5. By increasing the compression ratio, the NC values decreases slightly but the results are still encouraging (NC = 0.9 when CR = 10).

Afterwards, we compare the proposed method to [17, 26, 29, 34, 38] and [33].

Figures 15 and 16 summarized the results obtained in terms of NC after JPEG compression using several quality factors for a sample of textured images taken from Brodatz [3] and five natural images, respectively. The first observation to make after looking at the results is that there is a small difference between the NC values, from natural images to textured images even if each kind of image has its specific characteristics. As depicted in Tables 5 and 3, the NC value is equal to 1 when the quality factor greater than 75. Otherwise, the NC value decreases but the results are encouraging and proved that the proposed technique is still robust to JPEG compression (quality factor = 30, NC = 0.9344) for D9

Table 3 SSIM and NC values after several attacks applied to D9 image taken from the Brodatz database [3]

Attacks	SSIM	NC
Histogram equalization	0.9995	1.0
Gaussian noise ($\mu = 0, \sigma = 0.001$)	0.9998	1.0
Gaussian noise ($\mu = 0, \sigma = 0.005$)	0.9928	0.9947
Salt & pepper noise ($\mu = 0, \sigma = 0.001$)	0.9997	1.0
Low-pass Gaussian filtering		
($\sigma = 0.5, 3 \times 3$)	0.9999	1.0
($\sigma = 0.5, 5 \times 5$)	0.9999	1.0
($\sigma = 0.5, 7 \times 7$)	0.9998	1.0
($\sigma = 0.5, 9 \times 9$)	0.9998	1.0
($\sigma = 0.6, 3 \times 3$)	0.9998	1.0
($\sigma = 0.6, 5 \times 5$)	0.9997	0.9997
($\sigma = 0.6, 7 \times 7$)	0.9996	0.9989
($\sigma = 0.6, 9 \times 9$)	0.9996	0.9984
Gaussian smoothing		
($\sigma = 0.5, 3 \times 3$)	1.0	1.0
($\sigma = 0.5, 5 \times 5$)	0.9999	1.0
($\sigma = 0.5, 7 \times 7$)	0.9999	0.9999
($\sigma = 0.5, 9 \times 9$)	0.9999	0.9999
($\sigma = 0.6, 3 \times 3$)	0.9996	0.9954
($\sigma = 0.6, 5 \times 5$)	0.9996	0.9938
($\sigma = 0.6, 7 \times 7$)	0.9996	0.9883
($\sigma = 0.6, 9 \times 9$)	0.9995	0.9802
JPEG compression		
90%	0.9997	1.0
80%	0.9986	1.0
75%	0.9967	1.0
70%	0.9956	0.9959
60%	0.9915	0.9918
50%	0.9871	0.9874
40%	0.9623	0.9626
30%	0.9341	0.9344
20%	0.7887	0.7890
10%	0.6740	0.6743
JPEG2000 compression		
CR = 2	0.9997	1.0
CR = 4	0.9997	1.0
CR = 6	0.9762	0.9765
CR = 8	0.9460	0.9463
CR = 10	0.9002	0.9003
Cropping		
25%	1.0	1.0
50%	1.0	1.0

Table 3   (continued)

Attacks	SSIM	NC
Combination attacks		
HE + GN ($\sigma = 0.001$)	0.9995	1.0
HE + GN ($\sigma = 0.01$)	0.9995	0.9882
HE + SPN ($\sigma = 0.001$)	0.9995	1.0
HE + SPN ($\sigma = 0.01$)	0.9995	0.9796
GN ($\sigma = 0.001$) + JPEG compression (90%)	0.9995	1.0
GN ($\sigma = 0.001$) + JPEG compression (70%)	0.9994	0.9704
SPN ($\sigma = 0.001$) + JPEG compression (90%)	0.9997	1.0
SPN ($\sigma = 0.001$) + JPEG compression (70%)	0.9996	0.9946
LPGF ($\sigma = 0.5$, window size (9 × 9)) + SPN ($\sigma = 0.001$)	0.9996	0.9947
LPGF ($\sigma = 0.6$, window size(9 × 9)) + SPN ($\sigma = 0.001$)	0.9994	0.9694

image, and (quality factor = 30, NC = 0.9261) for Mandrill image. As shown in Table 5 the results obtained in terms of robustness against JPEG of the DCT-DFT method are good and outperform the DFT-only method.

6.2.3 Low-pass Gaussian filtering

The low-pass Gaussian filtering attack is also one of the common manipulations in image processing. It aims to remove high frequency components from the image. The watermarked

Fig. 12 Extracted watermarks after attacks: **a** Embedded Watermark, **b** HE, **c** GN($\sigma = 0.001$), **d** SPN, **e** LPGF (window size: (9 × 9)), **f** JPEG (Q = 60), **g** JPEG (Q = 65), **h** JPEG (Q = 70), **i** JPEG (Q = 75), **j** Gaussian Smoothing (window size: (9 × 9)), **k** Cropping 50%, **l** Cropping 25%, **m** HE + SPN ($\sigma = 0.001$), **n** GN ($\sigma = 0.001$) + JPEG (QF = 90), **o** SPN ($\sigma = 0.001$) + JPEG (QF = 90), **p** HE + GN ($\sigma = 0.001$)
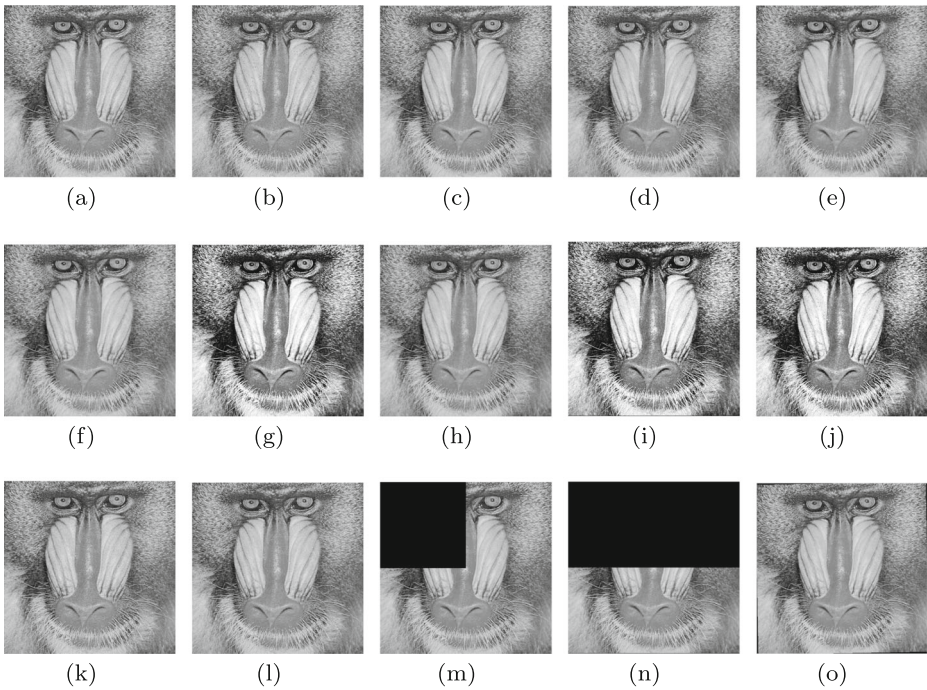
Fig. 13 Sample of attacked watermarked images: **a** Gaussian noise addition with mean zero and standard deviation 0.001, **b** Salt & pepper noise with noise density 0.01, **c** JPEG compression with quality factor 90, **d** JPEG compression with quality factor 60, **e** Gaussian low passe filtering with window size 3×3, **f** Gaussian low passe filtering with window size 9×9, **g** Histogram equalization, **h** Gaussian smoothing with window size 9×9, **i** Histogram equalization + Gaussian noise addition with mean zero and standard deviation 0.001, **j** Histogram equalization + Salt & pepper noise with noise density 0.001, **k** Gaussian noise addition with mean zero and standard deviation 0.001 + JPEG compression with quality factor 90, **l** Salt & pepper noise with noise density 0.01 + JPEG compression with quality factor 90, **m** Cropping 25%, **n** Cropping 50%, **o** Rotation 1°

images were filtered with a low-pass Gaussian filter using several window sizes (3×3), (5×5), (7×7) and (9×9) and two standard deviation values ($\sigma = 0.5$, $\sigma = 0.6$). The results shown in Tables 4 and 3 in terms of NC and PSNR are obtained after applying the low-pass Gaussian filtering to Mandrill natural image and D9 textured image taken from Brodatz [3]. From Tables 4 and 3 , it is clear that our approach is robust to low-pass Gaussian filtering. The results show that the robustness is still good even with larger filter size. As depicted in Table 4, in the case of Mandrill image, with a standard deviation $\sigma = 0.6$ and filter size (9×9), the obtained NC is 0.9765. Similarly, in the case of D9 textured image, it can be observed from Table 3 that with a standard deviation $\sigma = 0.6$ and filter size (9×9), the obtained normalized correlation (NC) is 0.9984.

6.2.4 Gaussian smoothing

Gaussian smoothing is a very common operation in image processing. It consists of removing detail and noise. We have applied the Gaussian smoothing attack to the test images with different standard deviations and window sizes. From Tables 4 and 3 it can be seen that the proposed method is robust against Gaussian smoothing attack for several filter sizes ((3×3),
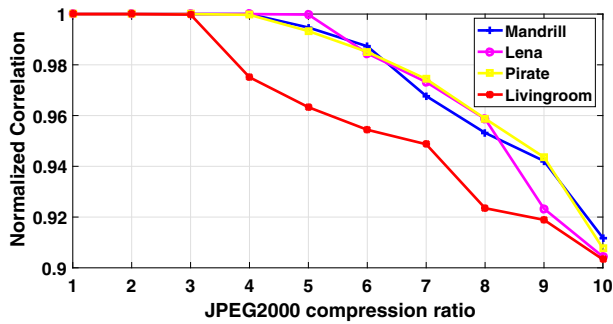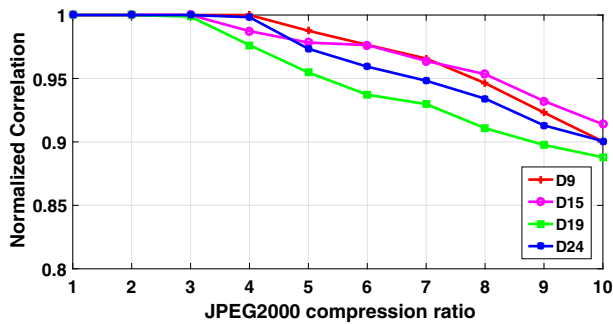
Fig. 14 **a**: robustness in terms of NC after JPEG2000 compression attack applied to a simple of natural images, **b**: robustness in terms of NC after JPEG2000 compression attack applied to a simple of textured images

(5×5), (7×7) and (9×9)). The results obtained in terms of NC are close to 1. In fact, even with ($\sigma = 0.6$) and size window (9×9), NC $= 0.9741$. In addition, as depicted in Table 5, the proposed system not only shows good resistance against Gaussian smoothing but outperforms the DFT-only approach.

6.2.5 Cropping

Image cropping is one of the most common manipulations in digital image. It's the most severe geometric distortion to be applied against an image. It consists of cropping off a rectangular region of the image by setting its pixels to zero value. To check the robustness of our proposal, we apply cropping attacks with several proportions (10, 20, 25, 40 and 50%) to the watermarked images then watermark is extracted. It can be concluded from Fig. 17 that the proposed scheme is very robust to the cropping attack. Tables 4 and 3 show the obtained results after applying cropping to "Mandrill" and "D9" respectively. The above results in terms of NC, under cropping 50%, show that our method is able to withstand this attack (NC $= 1.0$). The main reason stands on the fact that the effect of cropping leads to the blurring of spectrum. Therefore, there is no need of any synchronization since the watermark is embedded in the DFT magnitude.

Table 4 NC and PSNR values under various attacks for Mandrill

	Proposed scheme
Attacks	NC
No Attack	1.0
Histogram equalization	1.0
Gaussian noise	
$(\mu = 0, \sigma = 0.001)$	1.0
$(\mu = 0, \sigma = 0.005)$	1.0
$(\mu = 0, \sigma = 0.01)$	1.0
$(\mu = 0, \sigma = 0.02)$	0.9857
$(\mu = 0, \sigma = 0.1)$	0.9417
Salt & Pepper	
$(\mu = 0, \sigma = 0.001)$	1.0
$(\mu = 0, \sigma = 0.005)$	1.0
$(\mu = 0, \sigma = 0.01)$	0.9997
$(\mu = 0, \sigma = 0.02)$	0.9843
Low-pass Gaussian filtering	
$(\sigma = 0.5, 3 \times 3)$	1.0
$(\sigma = 0.5, 5 \times 5)$	1.0
$(\sigma = 0.5, 7 \times 7)$	1.0
$(\sigma = 0.5, 9 \times 9)$	0.9999
$(\sigma = 0.6, 3 \times 3)$	0.9834
$(\sigma = 0.6, 5 \times 5)$	0.9818
$(\sigma = 0.6, 7 \times 7)$	0.9794
$(\sigma = 0.6, 9 \times 9)$	0.9765
Gaussian smoothing	
$(\sigma = 0.5, 3 \times 3)$	1.0
$(\sigma = 0.5, 5 \times 5)$	1.0
$(\sigma = 0.5, 7 \times 7)$	1.0
$(\sigma = 0.5, 9 \times 9)$	1.0
$(\sigma = 0.6, 3 \times 3)$	0.9903
$(\sigma = 0.6, 5 \times 5)$	0.9838
$(\sigma = 0.6, 7 \times 7)$	0.9784
$(\sigma = 0.6, 9 \times 9)$	0.9741
Cropping	
(10%)	1.0
(25%)	1.0
(50%)	0.9999

Table 4 (continued)

	Proposed scheme
Attacks	NC
Combined attacks	
HE + GN ($\mu = 0$, $\sigma = 0.001$)	1.0
HE + GN ($\mu = 0$, $\sigma = 0.01$)	0.9875
HE + SPN ($\sigma = 0.001$)	1.0
HE + SPN ($\sigma = 0.01$)	0.9886
GN ($\mu = 0$, $\sigma = 0.001$) + JPEG compression (QF = 90)	1.0
GN ($\mu = 0$, $\sigma = 0.001$) + JPEG compression (QF = 70)	0.9715
SPN ($\sigma = 0.001$) + JPEG compression (QF = 90)	1.0
SPN ($\sigma = 0.001$) + JPEG compression (QF = 70)	0.9892
LPGF ($\sigma = 0.5$, window size (9×9)) + SPN ($\sigma = 0.01$)	0.9999
LPGF ($\sigma = 0.6$, window size (9×9)) + SPN ($\sigma = 0.01$)	0.9703

6.2.6 Combined attacks

The goal of this experiment is to check whether this kind of combination attack is able to remove the watermark of the proposed scheme. To test further the robustness of our method, different combinations of attacks composed by several kinds of attacks have been carried out. Tables 3 and 4 sketch the NC values where we can see the robustness of our method both for textured and natural images.

In conclusion, it can be observed from Tables 3 and 4 that, in all the cases, our method achieves good watermark extraction capability against several kind of attacks independently of the image nature. That is illustrated by the obtained values of the NC calculated between the original watermark and the extracted one which are above 0.9694. Moreover, regardless of the attack type, it can be concluded that the obtained results in terms of NC of the DFT-DCT method outperforms the performances of the DFT-only method.
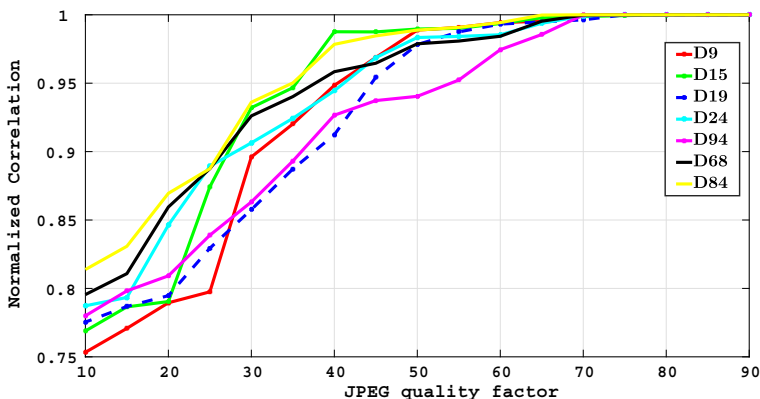

Fig. 15 Robustness in terms of NC after JPEG compression attack applied to a simple of textured image

Table 5 Robustness comparison between the DFT only and the DFT-DCT based algorithm for Mandrill in terms of NC

Attacks	Watermarking methods	
	DFT only Robustness metric	DFT-DCT
	NC	NC
No attack	1.0	1.0
GN ($\mu = 0$, $\sigma = 0.001$)	0.7140	1.0
GN ($\mu = 0$, $\sigma = 0.005$)	0.6872	1.0
SPN ($\mu = 0$, $\sigma = 0.001$)	0.7365	0.9947
JPEG (90%)	0.6854	1.0
JPEG (80%)	0.6476	1.0
JPEG (75%)	0.6324	1.0
JPEG (70%)	0.6003	0.9998
JPEG (60%)	0.5243	0.9843
JPEG (50%)	0.5102	0.9788
JPEG (40%)	0.5035	0.9584
JPEG (30%)	0.4690	0.9261
JPEG (20%)	0.4198	0.7795
JPEG (10%)	0.3268	0.6433
JPEG2000 (CR = 2)	0.6876	1.0
JPEG2000 (CR = 4)	0.6134	1.0
JPEG2000 (CR = 6)	0.6068	0.9873
JPEG2000 (CR = 8)	0.5846	0.9532
JPEG2000 (CR = 10)	0.5243	0.9118
LPGF ($\sigma = 0.5$, 3×3)	0.7865	1.0
LPGF ($\sigma = 0.5$, 5×5)	0.7369	1.0
LPGF ($\sigma = 0.5$, 7×7)	0.7166	1.0
LPGF ($\sigma = 0.5$, 9×9)	0.7087	0.9999
GS ($\sigma = 0.5$, 3×3)	0.7750	1.0
GS ($\sigma = 0.5$, 5×5)	0.7434	1.0
GS ($\sigma = 0.5$, 7×7)	0.7190	1.0
GS ($\sigma = 0.5$, 9×9)	0.7003	1.0
HE	0.9720	1.0
Cropp (10%)	0.9986	1.0
Cropp (20%)	0.9932	1.0
Cropp (25%)	0.9890	1.0
Cropp (40%)	0.9878	1.0
Cropp (50%)	0.9689	0.9999
Rotation ($\theta = 0.25°$)	1.0	1.0
Rotation ($\theta = 0.75°$)	1.0	0.9999
Rotation ($\theta = -0.25°$)	1.0	1.0
Rotation ($\theta = -0.75°$)	1.0	0.9998
HE+GN ($\sigma = 0.001$)	0.7387	1.0

Table 5   (continued)

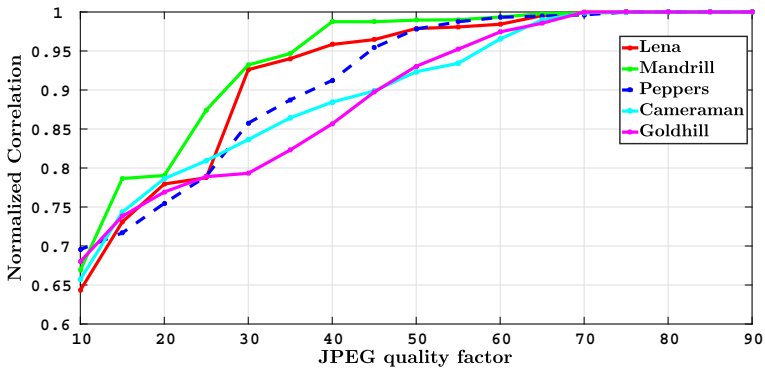| Attacks | Watermarking methods | |
| | DFT only | DFT-DCT |
| | Robustness metric | |
	NC	NC
HE+SPN ($\sigma = 0.001$)	0.7407	1.0
GN ($\sigma = 0.001$)+ JPEG (90%)	0.6084	1.0
SPN ($\sigma = 0.001$)+ JPEG (90%)	0.5435	1.0
LPGF (9×9)+ SPN ($\sigma = 0.001$)	0.5108	0.9999
Average	0.7331	0.9795

Fig. 16 Robustness in terms of NC after JPEG compression attack applied to a simple of natural images
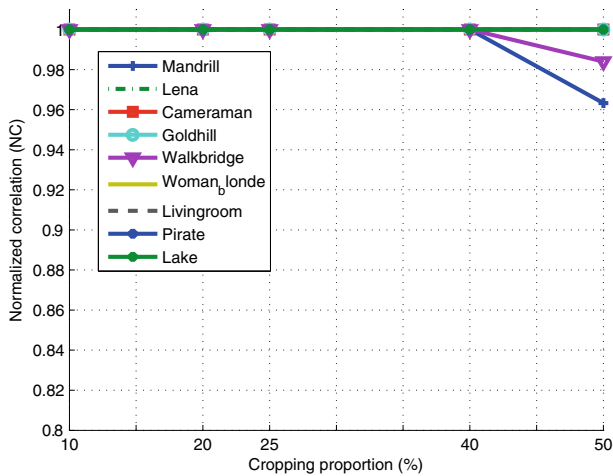

Fig. 17 Robustness in terms of NC after cropping attack applied to a simple of natural images

Table 6 Average CPU time for the proposed watermarking method

Computational time (seconds)	The proposed method
Embedding time	0.9483
Extraction time	0.8845
Total time	1.8328

6.3 Computational complexity

To evaluate the computational complexity of the proposed method, the complexity of Algorithm 1 (watermark embedding) and Algorithm 2 (watermark extraction) has been calculated using the big O notation. Due to the matrix multiplication involved in (13), the computational complexity of the proposed method is $O(n^3)$. In the extraction process, the complexity is $O(n^2)$ due to inverse Arnold transform calculation. Consequently, the global complexity of the proposed method is $O(n^3)$.

In order to analyze the time complexity of the proposed scheme, several experiments have been conducted on 10 natural images of size 512×512. Table 6 shows the average CPU time of the watermark embedding and extraction. The experiments are performed using MATLAB R2013a environment on a PC with CPU Intel(R) Core(TM) i5-3470 @ 3.2 GHZ with 4-GB of RAM.

6.4 Comparison with alternative methods

To further demonstrate the robustness of the proposed method, we compare it with schemes [17, 26, 29, 34, 38] and [33] in terms of imperceptibility and robustness as well as capacity.

6.4.1 Imperceptibility

In Table 7, is presented the comparison in terms of imperceptibility between the proposed scheme and the schemes in [26, 38], and [17]. The metric used in comparison is PSNR using Lena as test image. The PSNR values show the superiority of our method even if its capacity is bigger than the alternative methods. We believe that the main reason stands on the fact that the watermark is inserted in the DFT magnitude which ensures high imperceptibility.

According to Table 8, it can be seen that the proposed method gives good results in terms of imperceptibility with a capacity of 65536 bits and outperforms the Singh et al. method [34].

6.4.2 Robustness

The robustness comparison is performed in the case of Gaussian noise, salt & pepper noise, histogram equalization, JPEG compression, Cropping and rotation attacks.

Table 7 Watermark imperceptibility and capacity using Lena as test image

Watermarking methods	PSNR (dB)	Capacity (bits)
[38]	38.20	512
[26]	51.80	512
[17]	44.73	512
[34]	41.36	16834
Proposed method	61.97	988

Table 8 Comparison of the imperceptibility in terms of PSNR and SSIM between the proposed method and [34] for several images using a watermark of size 256×256

Cover image	Watermarking methods			
	Scheme in [34]		Proposed method	
	Imperceptibility metric			
	PSNR	SSIM	PSNR	SSIM
Lena	39.774	0.9937	40.93	0.9972
Cameraman	39.619	0.9868	42.03	0.9987
Mandrill	32.492	0.9611	39.85	0.9803
Peppers	39.031	0.9831	43.21	0.9990

For Salt & pepper attack, as shown in Table 9, it can also be observed that, compared to scheme [26], our approach is more robust. Moreover, it can be seen from Tables 10 and 11 that the proposed method shows high robustness to Salt & pepper noise compared to [29, 34] and [33].

Moreover, the proposed technique is robust to histogram equalization and outperforms the schemes in [17, 26, 29, 34] and [33].

Moreover, in the case of JPEG compression attack, it can be seen from Table 12 that our method gives better results than the approach in [26]. In addition, as depicted in Tables 13, 10 and 11, it can be observed that the proposed approach outperforms schemes [26, 38] and [17]. In addition, the proposed method shows relatively good robustness to JPEG compression when the quality factor is 50 except for the schemes [29, 34] and [33] which outperform the proposed method is this particular case.

For cropping attack , according to Table 13, the proposed scheme, compared with the other schemes, gives the best performance. Moreover, it can be seen from Table 13 that the results obtained after rotation attack are encouraging. In our method, the NC values are not so good for the rotation attacks with degree greater than ± 0.75; but it is far better than those in the listed methods (see Table 13).

Furthermore, it can be concluded that the proposed method shows good robustness against several kind of attacks compared with alternative methods.

6.4.3 Complexity analysis comparison

To further evaluate the computational complexity of the proposed method, a comparison with various related works [29, 33, 34] has been elaborated. The computational complexity of each method is expressed in big O notation. The scheme in [34] is based on NSCT-RDWT-SVD and Arnold transform while scheme in [29] used DWT, DCT and SVD transforms. The proposed schemes in [29, 34] have cubic complexity because of the use of SVD. The complexity of the scheme in [33] is $O(n^2)$ due to Arnold transform. Our cubic complexity is due to the matrix multiplication performed during image reconstruction in the embedding process (13). Although the complexity of the proposed method is much higher

Table 9 NC values after Salt & pepper noise attack

Salt & pepper	Proposed scheme	Scheme [26]
$(\mu = 0, \sigma = 0.01)$	1.0	0.83
$(\mu = 0, \sigma = 0.02)$	0.9843	0.76

Table 10 Comparison of the robustness of the proposed algorithm with Singh's method [34] against several attacks for Lena

	Watermarking methods	
Attacks	Our method	[34]
Histogram equalization	1.0	0.9902
JPEG		
QF = 50	0.9587	0.9951
Gaussian noise		
$(\mu = 0, \sigma = 0.001)$	0.9972	0.9965
Salt & pepper noise		
$(\mu = 0, \sigma = 0.001)$	0.9932	0.9912

Table 11 Comparison of the robustness of the proposed algorithm with different methods against several attacks for Lena

	Watermarking methods			
Attacks	Our method	[34]	[29]	[33]
Histogram equalization	1.0	0.9902	0.9208	0.9942
JPEG				
QF = 50	0.9384	0.9951	0.9994	0.9935
Gaussian noise				
$(\mu = 0, \sigma = 0.01)$	1.0	0.9965	0.9754	0.9828
$(\mu = 0, \sigma = 0.5)$	0.9803	0.9865	0.6565	0.8481
Salt & pepper noise				
$(\mu = 0, \sigma = 0.001)$	0.9997	0.9912	0.9952	0.9867

Table 12 NC values after JPEG compression with several quality factors

JPEG compression	Proposed scheme	Scheme [26]
(QF = 100)	1.0	0.96
(QF = 90)	1.0	0.95
(QF = 80)	1.0	0.87
(QF = 70)	0.9980	0.86
(QF = 50)	0.9384	0.79
(QF = 20)	0.7382	0.66
(QF = 10)	0.6545	0.61

Table 13 Comparison of the robustness of the proposed algorithm with different methods against several attacks for Lena

Attacks	Watermarking methods			
	Our method	[38]	[26]	[17]
HE	1.0	NA	0.83	0.79
JPEG				
QF = 70	1.0	0.51	0.86	1.0
QF = 90	1.0	1.0	0.95	1.0
Cropping				
(25%)	1.0	–	0.96	0.60
(50%)	0.9742	–	0.90	–
Rotation				
$\theta = 0.25°$	1.0	0.37	0.75	0.61
$\theta = 0.75°$	0.9999	0.26	0.67	0.34
$\theta = 1.0°$	0.55	0.24	–	0.27
$\theta = -0.25°$	1.0	0.32	0.76	0.65
$\theta = -0.75°$	0.9998	0.10	0.24	0.67
$\theta = -1.0°$	0.57	0.16	–	0.28

than scheme in [33], both the accuracy of watermark extraction and the robustness of the proposed scheme are much more important than scheme in [33].

7 Conclusion

In this work, a blind robust hybrid image watermarking scheme combining the two well known transformations DFT and DCT for Copyright protection is presented. The watermark is embedded in the middle band DCT coefficients of the DFT magnitude of the cover image using two secret keys for increasing security. The first one is used to generate the PN sequences to be inserted in the watermark embedding while the second one is to encrypt the watermark with Arnold transform. Taking the advantages of jointing DFT and DCT transforms, the obtained results show that the proposed scheme ensures good resistance to a wide variety of attacks for textured images as well as natural images while preserving high imperceptibility. Future work can be focused on investigating the proposed technique for another kind of image and enhancing its robustness against new variety of attacks.

References

1. Ali M, Ahn CW, Pant M (2014) A robust image watermarking technique using SVD and differential evolution in DCT domain. Optik-International Journal for Light and Electron Optics 125(1):428–434
2. Barni M, Bartolini F, Piva A (2001) Improved wavelet-based watermarking through pixel-wise masking. IEEE Trans Image Process 10(5):783–791
3. Brodatz P (1966) Textures: a photographic album for artists and designers. Dover, New York
4. Cedillo-Hernandez M, Garcia-Ugalde F, Nakano-Miyatake M, Perez-Meana H (2012) Robust digital image watermarking using interest points and DFT domain. In: 35th IEEE international conference on telecommunications and signal processing (TSP), pp 715–719
5. Cox IJ, Kilian J, Leighton F, Shamoon T (1997) Secure spread spectrum watermarking for multimedia. IEEE Trans Image Process 6(12):1673–1687

6. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2007) Digital watermarking and steganography. Morgan Kaufmann, San Mateo
7. Das C, Panigrahi S, Sharma VK, Mahapatra KK (2014) A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation. AEU-Int J Electron C 68(3):244–253
8. Dyson FJ, Falk H (1992) Period of a discrete cat mapping. Am Math Mon 99(7):603–614
9. Feng LP, Zheng LB, Cao P (2010) A DWT-DCT based blind watermarking algorithm for copyright protection. In: Third IEEE international conference on computer science and information technology (ICCSIT), vol 7, pp 455–458
10. Gangadhar Y, Akula VG, Reddy PC (2018) An evolutionary programming approach for securing medical images using watermarking scheme in invariant discrete wavelet transformation. Biomed Signal Process Control 43:31–40
11. Hsu CT, Wu JL (1999) Hidden digital watermarks in images. IEEE Trans Image Process 8(1):58–68
12. Jia SL (2014) A novel blind color images watermarking based on SVD. Optik-International Journal for Light and Electron Optics 125(12):2868–2874
13. Khayam SA (2003) The discrete cosine transform (DCT): theory and application. Michigan State University, p 114
14. Kumar C, Singh AK, Kumar P (2018) A recent survey on image watermarking techniques and its application in e-governance. Multimed Tools Appl 77(3):3597–3622
15. Kutter M, Petitcolas FA (1999) Fair benchmark for image watermarking systems. In: Electronic imaging, international society for optics and photonics, pp 226–239
16. Li M, Liang T, He YJ (2013) Arnold transform based image scrambling method. In: Third international conference on multimedia technology
17. Lin WH, Wang YR (2009) A wavelet-tree-based watermarking method using distance vector of binary cluster. Expert Systems with Applications 36(6):9869–9878
18. Liu L, Cheng L, Liu Y, Jia Y, Rosenblum DS (2016) Recognizing complex activities by a probabilistic interval-based model. In: AAAI, vol 30, pp 1266–1272
19. Liu Y, Nie L, Liu L, Rosenblum DS (2016) From action to activity: sensor-based activity recognition. Neurocomputing 181:108–115
20. Pereira S, Pun T (2000) Robust template matching for affine resistant image watermarks. IEEE Trans Image Process 9(6):1123–1129
21. Poljicak A, Mandic L, Agic D (2011) Discrete Fourier transform–based watermarking method with an optimal implementation radius. J Electron Imaging 20(3):033008-033008
22. Press WH (1989) Numerical recipes in Pascal: the art of scientific computing, vol 1. Cambridge University Press, Cambridge
23. Qi X, Xin X (2011) A quantization-based semi-fragile watermarking scheme for image content authentication. J Vis Commun Image Represent 22(2):187–200
24. Roy S, Pal AK (2017) A robust blind hybrid image watermarking scheme in RDWT-DCT domain using Arnold scrambling. Multimed Tools Appl 76(3):3577–3616
25. Saha BJ, Kabi KK, Pradhan C (2014) Non blind watermarking technique using enhanced one time pad in DWT domain. In: 2014 international conference on computing, communication and networking technologies (ICCCNT). IEEE, pp 1–6
26. Sahraee MJ, Ghofrani S (2013) A robust blind watermarking method using quantization of distance between wavelet coefficients. Signal, Image and Video Processing 7(4):799–807
27. Singh D, Singh SK (2017) DWT-SVD and DCT based robust and blind watermarking scheme for copyright protection. Multimed Tools Appl 76(11):13001–13024
28. Singh AK, Dave M, Mohan A (2014) Hybrid technique for robust and imperceptible image watermarking in DWT-DCT-SVD domain. Natl Acad Sci Lett 37(4):351–358
29. Singh AK, Dave M, Mohan A (2016) Hybrid technique for robust and imperceptible multiple watermarking using medical images. Multimed Tools Appl 75(14):8381–8401
30. Singh AK, Dave M, Mohan A (2015) Robust and secure multiple watermarking in wavelet domain. Journal of Medical Imaging and Health Informatics 5(2):406–414
31. Singh AK, Kumar B, Dave M, Mohan A (2015) Robust and imperceptible dual watermarking for telemedicine applications. Wirel Pers Commun 80(4):1415–1433
32. Singh AK, Kumar B, Singh SK, Ghrera SP, Mohan A (2016) Multiple watermarking technique for securing online social network contents using Back Propagation Neural Network. Future Generat Comput Syst. Elsevier. https://doi.org/10.1016/j.future.2016.11.023
33. Singh S, Rathore VS, Singh R (2017) Hybrid NSCT domain multiple watermarking for medical images. Multimed Tools Appl 76(3):3557–3575
34. Singh S, Rathore VS, Singh R, Singh MK (2017) Hybrid semi-blind image watermarking in redundant wavelet domain. Multimed Tools Appl 76(18):19113–19137

35. Singh AK, Kumar B, Singh G, Mohan A (eds) (2017) Medical image watermarking: techniques and applications. Springer, Berlin
36. Su Q, Chen B (2017) Robust color image watermarking technique in the spatial domain. Soft Comput 22(1):91–106
37. Urvoy M, Goudia D, Autrusseau F (2014) Perceptual DFT watermarking with improved detection and robustness to geometrical distortions. IEEE Trans Inf Forensics Secur 9(7):1108–1119
38. Wang SH, Lin YP (2004) Wavelet tree quantization for copyright protection watermarking. IEEE Trans Image Process 13(2):154–165
39. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
40. Weber AG (1997) The usc-sipi image database: http://sipi.usc.edu/services/database/Database.html. Signal and Image Processing Institute at the University of Southern California
41. Xiao D, Shih FY (2012) An improved hierarchical fragile watermarking scheme using chaotic sequence sorting and subblock post-processing. Opt Commun 285(10):2596–2606
42. Zear A, Singh AK, Kumar P (2018) A proposed secure multiple watermarking technique based on DWT, DCT and SVD for application in medicine. Multimed Tools Appl 77(4):4863–4882
43. Zhang Q, Li Y, Wei X (2012) An improved robust and adaptive watermarking algorithm based on dct. Journal of Applied Research and Technology 10(3):405–415

Mohamed Hamidi is a Ph.D student in Engineering Sciences at Mohammed V University in Rabat, Morocco. He received the Master's degree in 2013 from the same University. His area of research interests include digital watermarking and computer vision. His current work focuses on digital watermarking of images and 3D meshes.
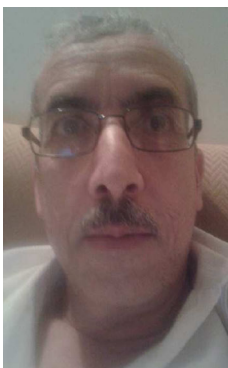
Mohamed El Haziti received the Doctorat de 3' Cycle and the Doctorat d'Etat-es-Sciences degrees in image processing from Mohammed V University (Rabat, Morocco) in 1997 and 2003, respectively. He is an Assistant Professor at Higher School of Technology, Sale, Morocco. His research interests include digital image, compression, watermarking and complex networks.

Hocine Cherifi is Full Professor of Computer Science at the University of Burgundy, Dijon, France since 1999. He completed his Ph.D. degree at the National Polytechnic Institute, Grenoble in 1984. Prior to moving to Dijon, he held faculty positions at Rouen University and Jean Monnet University. He has held visiting positions at Yonsei, Korea, University of Western Australia, Australia, National Pintung University, Taiwan, and Galatasaray University, Turkey. His recent research interests are in computer vision and complex networks. He has published more than 160 scientific papers in international refereed journals and conference proceedings. He held leading positions in more than 15 international conference organization (General chair, Program chair) and he served in more than 100 program committee. He is the founder of the international workshop on Complex Networks & their Applications. Currently, he is a member of the Editorial Board member of Computational social networks (Springer) and the Editor-in-Chief of the Journal of Applied Network Science (Springer).

Mohammed El Hassouni received the PhD in image and video processing from the University of Burgundy in 2005. He joined the University of Mohammed V as assistant professor since 2006 and associate professor since 2012. He also received the habilitation from University Mohammed V-Agdal in 2012. He was a visitor of several universities (Bordeaux I, Orleans, Dijon and Konstanz). He is member of IEEE and IEEE Signal Processing Society. His research focuses on image analysis, quality assessment and mesh processing. His application areas are medical imaging, biometry and QoE.