A commutative digital image watermarking and encryption method in the tree structured Haar transform domain

M. Cancellaro, F. Battisti, M. Carli, G. Boato, F.G.B. De Natale, A. Neri

Abstract

In this paper a commutative watermarking and ciphering scheme for digital images is presented. The commutative property of the proposed method allows to cipher a watermarked image without interfering with the embedded signal or to watermark an encrypted image still allowing a perfect deciphering. Both operations are performed on a parametric transform domain: the Tree Structured Haar transform. The key dependence of the adopted transform domain increases the security of the overall system. In fact, without the knowledge of the generating key it is not possible to extract any useful information from the ciphered-watermarked image. Experimental results show the effectiveness of the proposed scheme.

1. Introduction

Multimedia communications and information security are two active areas in both academia and industry. These two separate worlds are expected to continue playing important roles in the information era. The trend shows a fusion between them to allow a secure delivery of multimedia data. According to ITU-T, Rec. X.800 [1] and IETF RFC 2828 [2], the data security is pursued by assuring, among others, the following services: authentication, to verify the identity claimed by or for any system entity; data confidentiality, to protect data against unauthorized disclosure; data integrity, to verify that data have not been changed, destroyed, or lost in an authorized or accidental manner. To satisfy these constraints several methods have been proposed in literature, such as watermarking and cryptography.

Watermarking techniques are suitable for copyright protection. In this case the cover is the object of communication and the protection of its ownership is the goal of the hiding technique. To be effective a watermark should not introduce perceivable artifacts in the host data and it should be detectable also if unintentional or intentional modification of the watermarked signal occur. Effective methods, presented in literature, embed the watermark bits into the most significant portions of the digital image or video, so that they cannot be removed without impairing the original content. As cryptography is concerned, the messages are scrambled so that they cannot be understood. The aim is to make the information not intelligible to any unauthorized entity who might intercept them. In this case, the data content is kept secret and the security of the methods lays in the secret key involved in the process. Actually, encrypted data need an additional level of protection in order to keep control on them after the decryption phase. In fact, when the ciphered data is deciphered by the authorized user, it is unprotected and it can be easily modified, tampered, or stolen. Moreover, one may want or need to check the presence of a watermark without deciphering the data, thus increasing the security of the system.

The scientific community started focusing on the possibility of providing both security services simultaneously. The two protection levels can be combined in many different ways. Bas et al. [3] give an overview on the
possible scenarios where the combination of both level of protection can be exploited, while Merhav [4] presents a theoretic analysis of this problem.

Although the most challenging goal would be to process the data directly in the encrypted domain, this is a very ambitious task and therefore we usually distinguish between secure watermark detection and secure watermark embedding [5]. In the first case the aim is to avoid removal of the watermark by a cheating verifier, since the crucial weakness of the detection procedure is to require the knowledge of the secret key used for embedding [3]. The two proposed solutions are asymmetric watermarking [6,7] and zero-knowledge detector [8,9]. In the second scenario, the goal is to protect both the original content and the watermark even if the insertion is operated by an untrusted embedder. Consequently, the challenge is to avoid the access to the non watermarked data by jointly deciphering and fingerprinting sensitive data [10,11].

In the last decade a slightly different approach for content protection has been introduced. To overcome the computational complexity and cost of encrypting digital images or videos, and to deal with the high transmission rate vs. low available bandwidth, many authors proposed the idea of applying cryptographic methods only to relevant portions of the data. Basically the protection level of the data is adapted according to the particular application. As a general guide, some constraints should be satisfied: the original content (full resolution or quality) could not be obtained without correctly deciphering the data, sufficient security for the considered application should be provided, the bitstream size should not be increased, and the computational complexity with respect to full encryption should be reduced.

Selective encryption of digital data may refer to the partial encryption of the data (i.e., only some bands of the image wavelet decomposition, or some bit plane of its binary representation [28] are ciphered), to localized image wavelet decomposition, or some bit plane of its partial encryption of the data (i.e., only some bands of the bitstream size should not be increased, and the security for the considered application should be provided, the original content (full resolution or quality) could not be obtained without correctly deciphering the data, sufficient security for the considered application should be provided, the bitstream size should not be increased, and the computational complexity with respect to full encryption should be reduced.

In this section we describe the design of a layered architecture for cryptography and watermarking. Given a digital image $X$, let $f_W$ be the function used to hide the digital watermark $W$ into $X$, i.e.,

$$X_W = f_W(X,W),$$

and let $f_C$ be the function employed to cipher the original data $X$

$$X_E = f_C(X,\xi),$$

given an encryption key $\xi$. Watermarking and encryption commute whenever the two functions $f_W$ and $f_C$ satisfy the following rule:

$$X_{W,E} = f_W(f_C(X,\xi),W) = f_C(f_W(X,W),\xi).$$

A simple commutative watermarking and encryption scheme can be obtained by separately watermarking a subset of the bit-planes of the image representation in a suitable transformed domain and encrypting the remaining bit-planes. However, correlation between watermarked and encrypted bit-planes can introduce some security breach in the form of plain text attacks. To enhance security we propose to pseudo-randomly select the representation domain among a wide set of parametric transformations. Knowledge about the actual transform is conveyed by an additional secret key shared by means of a secure channel. In particular, here, we employ the discrete TSH transform,
recently developed in [15], which is a generalization of the Discrete Haar transform. To illustrate the TSH transform let us first consider the classical Haar discrete transform.

Given the discrete interval \( I = [1, N] \) with \( N=2^k \), let us denote with \( \phi^H(t) \) the Haar scaling function:

\[
\phi^H(t) = \begin{cases} 
\frac{1}{\sqrt{N}} & \text{if } t \in [1,N], \\
0 & \text{otherwise},
\end{cases}
\]  

(4)

To construct the representation basis, we split the interval \( I \) into two halves \( I^H_0 = [1,2^{−1}N] \) and \( I^H_1 = [2^{−1}N+1,N] \). Then, we introduce the (wavelet) function \( \psi^H(t) \) defined as follows:

\[
\psi^H(t) = \begin{cases} 
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_0, \\
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_1, \\
0 & \text{otherwise},
\end{cases}
\]  

(5)

It can be verified that \( \phi^H(t) \) and \( \psi^H(t) \) are orthogonal and \( \|\phi^H(t)\| = 1 \) and \( \|\psi^H(t)\| = 1 \) where \( \| \cdot \| \) denotes the \( L_2 \) norm.

The set \( \{\phi^H(t), \psi^H(t)\} \) can be extended by splitting \( I^H_0 \) into \( I^H_0,0 \) and \( I^H_1 \) into \( I^H_0,1 \) and \( I^H_1,0 \), having denoted with subscript 0 and 1 the left and the right halves, respectively, thus obtaining

\[
\phi^H_j(t) = \begin{cases} 
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_{j,0}, \\
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_{j,1}, \\
0 & \text{otherwise},
\end{cases}
\]  

\[
\psi^H_j(t) = \begin{cases} 
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_{j,0}, \\
\frac{1}{\sqrt{2^{−1}N}} & \text{for } t \in I^H_{j,1}, \\
0 & \text{otherwise},
\end{cases}
\]  

(7)

This procedure is iterated \( L-1 \) times, until each subinterval contains only one integer. More specifically, for a given resolution level \( k \) (\( 1 \leq k < l \)), there are \( 2^k \) intervals \( I^H_{a_1,a_2,\ldots,a_k} = [a^H_{a_1,a_2,\ldots,a_k}, b^H_{a_1,a_2,\ldots,a_k}] \), with \( a_n \in \{0,1\} \), with

\[
a^H_{a_1,a_2,\ldots,a_k} = \sum_{n=1}^{k} a_n 2^{−n−1} + 1,
\]

\[
b^H_{a_1,a_2,\ldots,a_k} = \sum_{n=1}^{k} a_n 2^{−n−2} + 2^{−k},
\]  

(7)

the following two basis functions are associated to each interval:

\[
\phi^H_{a_1,a_2,\ldots,a_k}(t) = \begin{cases} 
\frac{1}{\sqrt{2^{−k}N}} & \text{for } t \in I^H_{a_1,a_2,\ldots,a_k}, \\
0 & \text{otherwise},
\end{cases}
\]  

(8)

\[
\psi^H_{a_1,a_2,\ldots,a_k}(t) = \begin{cases} 
\frac{1}{\sqrt{2^{−k}N}} & \text{for } t \in I^H_{a_1,a_2,\ldots,a_k,0}, \\
\frac{1}{\sqrt{2^{−k}N}} & \text{for } t \in I^H_{a_1,a_2,\ldots,a_k,1}, \\
0 & \text{otherwise}.
\end{cases}
\]  

(9)

In the TSH discrete transform the interval \( I \) is split into two intervals \( I^TSH_0 = [1,v_0] \) and \( I^TSH_1 = [v_0+1,N] \) with \( 1 \leq v_0 < N \); however \( N \) does not need to be a power of 2. The basis function \( \psi^{TSH}(t) \) is defined as follows:

\[
\psi^{TSH}(t) = \begin{cases} 
\sqrt{\frac{v_1}{v_0N}} & \text{for } t \in I^TSH_0, \\
-\sqrt{\frac{v_0}{v_1N}} & \text{for } t \in I^TSH_1, \\
0 & \text{otherwise},
\end{cases}
\]  

(10)

where \( v_1 = N-v_0 \). This construction can be iterated by splitting \( I^TSH_0 \) into \( I^TSH_{0,0} \) and \( I^TSH_{0,1} \) and \( I^TSH_1 \) into \( I^TSH_{1,0} \) and \( I^TSH_{1,1} \), respectively, and so on. In general, given the interval \( I^TSH_{a_1,a_2,\ldots,a_k} = [a^TSH_{a_1,a_2,\ldots,a_k}, b^TSH_{a_1,a_2,\ldots,a_k}] \), with \( a_j = \{0,1\} \), of length \( v_{2^j,a_2,\ldots,a_k} = b^TSH_{a_1,a_2,\ldots,a_k} - a^TSH_{a_1,a_2,\ldots,a_k} + 1 \), if \( v_{2^j,a_2,\ldots,a_k} > 1 \) we can partition \( I^TSH_{a_1,a_2,\ldots,a_k} \) into two intervals \( I^TSH_{a_1,a_2,\ldots,a_k,0} \) and \( I^TSH_{a_1,a_2,\ldots,a_k,1} \) of length \( v_{2^j,a_2,\ldots,a_k,0} \geq 1 \) and \( v_{2^j,a_2,\ldots,a_k,1} = v_{2^j,a_2,\ldots,a_k} - v_{2^j,a_2,\ldots,a_k,0} \geq 1 \). The following recursive relationships hold

\[
a^TSH_{a_1,a_2,\ldots,a_k,0} = a^TSH_{a_1,a_2,\ldots,a_k},
\]

\[
b^TSH_{a_1,a_2,\ldots,a_k,0} = b^TSH_{a_1,a_2,\ldots,a_k} - v_{2^j,a_2,\ldots,a_k,0} - 1,
\]

(11)

and

\[
a^TSH_{a_1,a_2,\ldots,a_k,1} = a^TSH_{a_1,a_2,\ldots,a_k} + v_{2^j,a_2,\ldots,a_k,0},
\]

\[
b^TSH_{a_1,a_2,\ldots,a_k,1} = b^TSH_{a_1,a_2,\ldots,a_k},
\]

(12)

with initial conditions \( a^TSH_{0,1} = 1 \), \( b^TSH_{0,0} = v_0 \), \( a^TSH_{0,0} = v_0+1 \), \( b^TSH_{0,1} = N \). The functions \( \phi^{TSH}_{a_1,a_2,\ldots,a_k}(t) \) and \( \psi^{TSH}_{a_1,a_2,\ldots,a_k}(t) \) are defined as follows:

\[
\phi^{TSH}_{a_1,a_2,\ldots,a_k}(t) = \begin{cases} 
\frac{1}{\sqrt{v_{2^j,a_2,\ldots,a_k}}} & \text{for } t \in I^TSH_{a_1,a_2,\ldots,a_k}, \\
0 & \text{otherwise},
\end{cases}
\]  

(13)

\[
\psi^{TSH}_{a_1,a_2,\ldots,a_k}(t) = \begin{cases} 
\frac{\sqrt{v_{2^j,a_2,\ldots,a_k,1}}}{v_{2^j,a_2,\ldots,a_k}} & \text{for } t \in I^TSH_{a_1,a_2,\ldots,a_k,0}, \\
\frac{\sqrt{v_{2^j,a_2,\ldots,a_k,0}}}{v_{2^j,a_2,\ldots,a_k}} & \text{for } t \in I^TSH_{a_1,a_2,\ldots,a_k,1}, \\
0 & \text{otherwise}.
\end{cases}
\]  

(14)

It has been demonstrated in [15] that the set of TSH functions is a set of orthogonal functions. We remark that each splitting scheme defines a different basis set and, for a given \( N \), each TSH basis is univocally defined by the set \( \{a^TSH_{a_1,a_2,\ldots,a_k}\} \), named Discontinuity Point Vector (DPV), that defines the splitting scheme. An example of the interval splitting procedure and the corresponding set \( \{\psi(t)\} \) is illustrated in Fig. 1. The DPV randomness allows to obtain a multiresolution analysis in which the details are spread all over the subbands. In particular for a signal of length \( N \), the possible number of DPV for the first order decomposition can be computed as

\[
\text{DPV Number} = (N−1) + \sum_{j=1}^{N-2} \prod_{i=N-1-j}^{N-1} i.
\]  

(15)
Let $T$ be the $(N \times N)$ orthogonal matrix, whose rows are constituted by the basis functions. Then, given a column vector $f$ of size $N$, its TSH transform $g$ is

$$g = Tf.$$  \hfill (16)

A detailed study of matrix representation of TSH transform of a general signal can be found in [23]. Here, we apply TSH transform to two dimensional signals. In this case different DPVs lead to different subband decompositions of the image (see in Fig. 2 examples of 3rd-level decomposition depending on two different DPVs). Since all the operations – watermark embedding and extraction, as well as encryption and decryption – are performed in the TSH domain, the knowledge of the chosen DPV is crucial. In fact, this becomes a further element to increase the security of the overall method, although the length of this secret key is not arbitrary (the maximum size of the vector correspond to the size of the signal that has to be decomposed).

2.1. Watermark embedding and encryption

In the following, the general scheme for the commutative watermark insertion and encryption is presented. Let $X$ be the original image, $W$ be the watermark (binary pseudorandom matrix of the same size of $X$), TSH$_{DPV}$ be the TSH decomposition depending by the secret key DPV, $N$ be a fixed integer

![Fig. 1. Splitting intervals (a) and associated TSH functions (b).](image1)

![Fig. 2. 3rd-level TSH transform of the image Lena: (a)–(b) with DPV=[230,504]; (c)–(d) with DPV=[27].](image2)
number, and $B$ the number of bits used to represent each TSH sample. Fig. 3 shows the case in which $N=3$.

The watermarking function $f_W$ of Eq. (1) is defined by the following operations:

1. Computation of $X$, the $n$-th order TSH$_{DPV}$ decomposition of $X$.
2. Quantization of the real coefficients of $X$, resulting into the bit-planes $BP_l$ with $l=1,2,...,B$ and the binary matrix $S_X$ that contains the sign of $X$.
3. Substitution of the least significant bit-plane $BP_B$ with $S_X$ (as for $f_W$).
4. Encryption of the $N$ most significant bit-planes $BP_1$ to $BP_N$. These planes are individually encrypted by using the AES. We propose to use this block cipher since it is widely accepted. However, any other secure block cipher can be used for encryption. In the performed tests we used 128-bit AES keys ($k_{AES}$).
5. Reconstruction of the ciphered data $X_E$ by performing the digital-to-analog conversion and the inverse TSH$_{DPV}$.

As it can be noticed, the proposed system is based on the encryption of those bit-planes that are not used for the embedding, and viceversa. Thus, the order of embedding or ciphering is not relevant. Consequently, the proposed method is compliant with the commutative property of Eq. (3).

2.2. Watermark extraction and decryption

Given an image $X_{W,E}$ that has been both watermarked and encrypted according to the procedure described in Section 2.1, we describe here the extraction of the watermark and the decryption procedures obtained by inverting the embedding and encryption schemes (see Fig. 4 for the case $N=3$):

1. Computation of $X_{W,E}$, the $n$-th order TSH$_{DPV}$ decomposition on the watermarked and encrypted image $X_{W,E}$.
2. Quantization of each coefficient of $X_{W,E}$ resulting in the bit-planes $BP_l$ with $l=1,2,...,B$.
3. Decryption with $k_{AES}$ of $BP_1$ to $BP_N$.
4. The $(B-N-1)$ bit-planes, $BP_{N+1}$ to $BP_{B-1}$, used for watermarking are converted in decimal representation. The inverse QIM is then performed to recover the
inserted watermark \( W \) according to the following formula:

\[
    u_{ij} = Q_d (c'_{ij} - k_{ij} \Delta) - (c'_{ij} - k_{ij} \Delta),
\]

\[
    W_{ij} = \begin{cases} 
        0 & \text{if } |u_{ij}| \leq \Delta/2 \\
        1 & \text{if } |u_{ij}| > \Delta/2,
    \end{cases}
\]

where \( c'_{ij} \) is the coefficient supposed to contain the watermark. The knowledge of the watermark key \( k_{ij} \) and of the quantization step \( \Delta \) are required.

5. The normalized correlation coefficient \( \rho \) between the original watermark and the extracted one is computed as follows:

\[
    \rho = \frac{\sum_i \sum_j (W - \overline{W})(W' - \overline{W}')}{\sqrt{\sum_i \sum_j (W - \overline{W})^2(W' - \overline{W}')^2}},
\]

where \( \overline{W} \) and \( \overline{W'} \) are the average values of \( W \) and \( W' \).

It is worth noticing that, besides the knowledge of the secret keys \( k_{AES} \) and \( k_{ij} \), also the DPV chosen for the definition of TSHDPV is needed to perform both the extraction and/or the decryption procedures, thus increasing the overall security of the proposed approach. Furthermore, we underline that the two procedures do not need to be simultaneous, allowing the extraction of the watermark without deciphering the data (and vice versa). The overall proposed scheme for the reconstruction of the watermarked image is shown in Fig. 8.

3. Experimental results

Simulations have been performed on the database of 25 gray-scale images of size \( 512 \times 512 \) pixels shown in Fig. 5: in the following we report the average results obtained considering the whole database. In our simulation we used the 3rd-order decomposition of TSHDPV, \( B = 12 \) bits of precision, \( k_{AES} \) is a pseudorandom scalar value drawn from a uniform distribution on the unit interval, \( \alpha = 1 \), \( \Delta = 30 \), and \( k_{ij} = 0.5 \).

Several tests have been carried out for evaluating the effectiveness of the proposed approach. First, we considered the effect of the encryption procedure \( f_C \) on the perceived quality of the image. The number \( N \) of bit-planes involved in the watermarking and ciphering procedures, has been systematically varied to determine the optimum value of \( N \) for protecting the image content from non-authorized users, while reducing the computational complexity. As stated before, the robustness of the system and the non intelligibility of the processed image are inversely proportional: encrypting more bit planes results in a smaller number of bit planes available for watermarking, thus reducing its robustness.

As already described in the introduction, the security of the selective encryption strictly depends on the application. In most of cases, as for video on demand, database retrieval, etc. ..., a low quality and non usable version of the data, still protected with a robust watermark, is desirable. This can be obtained by encrypting a smaller number of significant bit planes. On the other hand severe visual degradation is needed for sensitive data (satellite images and videos, video surveillance, etc. ...): in this case a bigger number of significant bit planes is encrypted to guarantee the non intelligibility of the content.

As analyzed by Podesser et al. in [28] it is possible to verify that, for a \( 512 \times 512 \) pixels image with 8 bpp precision, the AES encryption of the two most significant bit-planes grants the non intelligibility of the image even if the encryption of the four MSB provides a high confidentiality level. To assess the security of the bit-plane selective encryption, two different ciphertext-only attacks have been performed: the replacement attack and the reconstruction one. The results of the experiments show that, as expected, the security increases with the number of encrypted bit-planes and that, as stated before, the encryption of the MSB only is not sufficient to grant the
The cryptographic security of the system. In the case of replacement attack, it can be prevented by encrypting the four MSBs while the reconstruction attack fails when the two MSBs are encrypted.

In addition to this, the parametric nature of the Tree Structured Haar transform allows to increase the security of the method. The mutual information \( I \) between the original matrix resulted from the binary-to-decimal conversion of the \((B - N - 1)\) bit-planes used for watermarking, and the corresponding matrix computed in the extraction-decryption step is computed. For evaluating the mutual information between two binary i.i.d sequences \( X \) and \( Y \) with same length \( M \), the following formula has been applied:

\[
I(X, Y) = E_{X,Y} \left[ \log \frac{p(x,y)}{p(x)p(y)} \right] = 1 - P \log \left( \frac{1}{P} \right) \cdot (1 - P) \log \left( \frac{1}{1 - P} \right).
\]

where
\[
\hat{P} = \frac{d_H(X,Y)}{M}.
\]

Table 1 shows, for the image Lena, that if a non correct DPV is used in the extraction-decryption step, the mutual information decreases; in particular, the more different is the DPV and consequently the TSH transform, the lower the mutual information is.

Fig. 6 shows the results obtained for the image Lena: byencrypting only the most significant bit-plane \((N=1)\) (see Fig. 6(b)) it is possible to disclose the original content while the encryption of the three most significant bit-planes \((N=3)\) allows to obtain a complete visual degradation of the image (see Fig. 6(c)).

In the following the experiments obtained selecting \(N=1\) and \(N=3\) are compared.

The watermarked and the watermarked-encrypted versions of one image extracted from the database, Lena, are shown in Fig. 7 for \(N=1\) and \(N=3\). It can be noticed that in both cases the watermark insertion does not affect the perceived quality of the original image (see Fig. 7(b)–(d)).
To evaluate the invisibility of the watermarking technique $f_{W}$, the weighted peak signal to noise ratio (WPSNR) has been computed:

$$WPSNR(dB) = 10 \log_{10} \frac{\max(x)^2}{\|NVF(x' - x)\|^2},$$  \hspace{1cm} (20)$$

where $x$ and $x'$, respectively, represent the original and the test image and NVF is the Noise Visibility Function whose value is 1 in flat regions and 0 in textured regions and edges [24]. The average values obtained for the watermarked images in the database when no attacks are performed are:

- $N=1$: PSNR=38 dB and WPSNR=40 dB.
- $N=3$: PSNR=40 dB and WPSNR=45 dB.

Finally, the distortion introduced by the quantization step in the $f_{W}$ and $f_{C}$ procedures is negligible. The average PSNR values after this step is $\approx 75$ dB.

In order to demonstrate the robustness of the watermarking procedure, we report in Fig. 9 the average detector response computed on the whole database when 500 random watermarks are presented to the detector and no attacks are performed. The correlation between the inserted watermark and the extracted one is shown. It can be seen that the highest value corresponds to the original embedded watermark. Fig. 9 shows the case $N=1$ and 3.

Moreover the resistance to some of the most common watermarking attacks has been tested by the StirMark Benchmark 4.0 [25,26]. The following attacks have been considered:

- Gaussian: Gaussian noise as by Stirmark to $X_{W}$.
- Motion: approximation of the linear motion of a camera by 5 pixels, with an angle of $10^\circ$ in a counterclockwise direction through a two dimensional filter.
- Blurring: using a circular averaging filter within the square matrix of size 5.
In order to improve the robustness of the digital watermark, Error Correction Codes (ECC) can be used. Recent studies [29,30] have shown that, at the moment, Reed–Solomon (RS) codes have the highest error correction capability compared to Hamming and BCH codes: in fact, they are able to reconstruct the original message for error rates up to 5%. Therefore, in the following, the results obtained for $N=1$ and 3 without ECC are compared with the case where RS codes are adopted. It is worth noticing that the more the length of the codeword is, the more the robustness of the method is guaranteed and less capacity for the watermark is available.

![Fig. 8. Reconstruction of the watermarked image.](image1)

![Fig. 9. Average detector response computed on the whole database when 500 random watermarks are presented at the detector. Both cases $N=1$ (a) and $N=3$ (b) are shown. No ECC used.](image2)

![Fig. 10. Simulation results for the JPEG compression attack computed on the image database: average first and second highest correlation peaks when 500 random watermarks are presented at the detector are shown for the four analyzed cases.](image3)

<table>
<thead>
<tr>
<th>Attack</th>
<th>Gaussian</th>
<th>Motion</th>
<th>Blurring</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 BP no RS</td>
<td>0.78</td>
<td>0.08</td>
<td>0.11</td>
</tr>
<tr>
<td>8 BP RS</td>
<td>0.41</td>
<td>0.13</td>
<td>0.13</td>
</tr>
<tr>
<td>10 BP no RS</td>
<td>0.91</td>
<td>0.18</td>
<td>0.12</td>
</tr>
<tr>
<td>10 BP RS</td>
<td>0.97</td>
<td>0.19</td>
<td>0.14</td>
</tr>
</tbody>
</table>

![Table 2. Average peak correlation values computed on the whole database when 500 random watermarks are presented at the detector.](table1)
The average results obtained on the whole database, are summarized in Table 2. It shows the average highest correlation peaks obtained by presenting 500 random watermarks to the detector.

The experimental results have been computed considering four possible scenarios:

- Watermarking of 8 bit planes and encryption of 3 bit planes;
- Watermarking of 8 bit planes and encryption of 3 bit planes with the application of the RS error correcting codes;
- Watermarking of 10 bit planes and encryption of 1 bit plane; and
- Watermarking of 10 bit planes and encryption of 1 bit plane with the application of the RS error correcting codes.

The watermarked images were compressed using the JPEG standard with increasing quality factors from 50 to 100 with step 10. For each quality factor, the detector has been tested with 500 random watermarks: the highest correlation peak corresponding to the original watermark and the second highest correlation value are plotted. Fig. 10 shows a comparison between the four analyzed cases. In particular, for each quality factor, the plotted second highest correlation peak is the average value computed on the four considered scenarios. The second highest peaks values for each case are detailed in Table 3. The experimental results, as expected,

![Fig. 10. Simulation results for the cropping attack computed on the image database: average first and second highest correlation values when 500 random watermarks are presented at the detector are shown for the four analyzed cases.](image1)

Table 3
Average second highest correlation peaks computed on the whole database for the JPEG compression attack with quality factor increasing from 50 to 100 with step 10.

<table>
<thead>
<tr>
<th>JPEG Attack</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 BP no RS</td>
<td>0.006</td>
<td>0.007</td>
<td>0.006</td>
<td>0.006</td>
<td>0.006</td>
<td>0.007</td>
</tr>
<tr>
<td>8 BP RS</td>
<td>0.009</td>
<td>0.009</td>
<td>0.010</td>
<td>0.010</td>
<td>0.009</td>
<td>0.010</td>
</tr>
<tr>
<td>10 BP no RS</td>
<td>0.005</td>
<td>0.006</td>
<td>0.007</td>
<td>0.006</td>
<td>0.005</td>
<td>0.007</td>
</tr>
<tr>
<td>10 BP RS</td>
<td>0.008</td>
<td>0.009</td>
<td>0.009</td>
<td>0.010</td>
<td>0.009</td>
<td>0.010</td>
</tr>
</tbody>
</table>

![Fig. 11. Simulation results for the JPEG 2000 compression attack computed on the image database: average first and second highest correlation peaks when 500 random watermarks are presented at the detector are shown for the four analyzed cases.](image2)

Table 4
Average second highest correlation peaks computed on the whole database for the JPEG 2000 compression attack with quality factor increasing from 50 to 100 with step 10.

<table>
<thead>
<tr>
<th>JPEG2000 Attack</th>
<th>0.1</th>
<th>0.2</th>
<th>0.3</th>
<th>0.4</th>
<th>0.5</th>
<th>0.6</th>
<th>0.7</th>
<th>0.8</th>
<th>0.9</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 BP no RS</td>
<td>0.007</td>
<td>0.005</td>
<td>0.006</td>
<td>0.006</td>
<td>0.006</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
</tr>
<tr>
<td>8 BP RS</td>
<td>0.008</td>
<td>0.010</td>
<td>0.008</td>
<td>0.001</td>
<td>0.011</td>
<td>0.011</td>
<td>0.011</td>
<td>0.010</td>
<td>0.011</td>
<td>0.011</td>
</tr>
<tr>
<td>10 BP no RS</td>
<td>0.005</td>
<td>0.007</td>
<td>0.006</td>
<td>0.006</td>
<td>0.006</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
<td>0.007</td>
</tr>
<tr>
<td>10 BP RS</td>
<td>0.008</td>
<td>0.009</td>
<td>0.010</td>
<td>0.009</td>
<td>0.011</td>
<td>0.010</td>
<td>0.010</td>
<td>0.010</td>
<td>0.010</td>
<td>0.010</td>
</tr>
</tbody>
</table>

![Fig. 12. Simulation results for the cropping attack computed on the image database: average first and second highest correlation values when 500 random watermarks are presented at the detector are shown for the four analyzed cases.](image3)

Table 5
Average second highest correlation peaks computed on the whole database for the compression attack with percentage of the cropped area increasing from 25% to 75% with step 25.

<table>
<thead>
<tr>
<th>Cropping attack</th>
<th>25%</th>
<th>50%</th>
<th>75%</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 bit planes no RS</td>
<td>0.006</td>
<td>0.005</td>
<td>0.005</td>
</tr>
<tr>
<td>8 bit planes RS</td>
<td>0.009</td>
<td>0.009</td>
<td>0.009</td>
</tr>
<tr>
<td>10 bit planes no RS</td>
<td>0.006</td>
<td>0.005</td>
<td>0.006</td>
</tr>
<tr>
<td>10 bit planes RS</td>
<td>0.007</td>
<td>0.008</td>
<td>0.010</td>
</tr>
</tbody>
</table>
show a more robust behavior in the case when 10 bit planes are used for the watermarking and the error correcting codes are applied.

The proposed scheme has also been tested with the JPEG 2000 standard compression. In particular the compression ratio has been increased from 0.1 to 1 with step 0.1. As for JPEG standard attack, Fig. 11 shows a comparison between the four cases. The second highest peaks values, whose average is plotted, are detailed in Table 4. As expected, also in this case, the encryption of the most significant bit plane and the watermarking of 10 bit planes gives the best results in terms of robustness.

The robustness of the proposed techniques against cropping has also been analyzed. In this attack the watermarked image, of size 512 \times 512 pixels, has been cropped to different sizes; in particular the cropped area (external frame of the image) has been replaced with the corresponding original frame [27]. Three different cropping dimensions have been considered, 25%, 50%, and 75% portion of the image. The detector response has been tested with 500 random watermarks. The experimental results, averaged on the database, are shown in Fig. 12. The second highest peaks values, whose average is plotted, are detailed in Table 5.

In order to highlight the importance of the secret key DPV in the whole process, we present the results obtained by performing the extraction of the watermark and the decryption of the image using DPVs different from the one used in the watermarking/encryption phase. Fig. 13(a) and (c) show the detector response, averaged on the whole database, when 500 random DPVs are used to decompose the images in order to extract the watermark, respectively, for \(N=1\) and \(N=3\); the peak corresponds to the DPV used by the sender in the first stage of the transmission. In Fig. 13(b) and (d) an example of decryption trial of the image Lena with a random DPVs is shown. Such experiments demonstrate that if an attacker tries to use random DPVs he will get no information about the hidden watermark as well as on the original image. A brute force

\[\text{Correlation Value}\]

\[\begin{array}{c|c}
\text{Different DPVs} & \text{Correlation Value} \\
\hline
0 & 0.1 \\
50 & 0.2 \\
100 & 0.3 \\
150 & 0.4 \\
200 & 0.5 \\
250 & 0.6 \\
300 & 0.7 \\
350 & 0.8 \\
400 & 0.9 \\
450 & 1.0 \\
500 & 0.1 \\
\end{array}\]

\[\text{Different DPVs}\]

\[\begin{array}{c|c}
\text{Correlation Value} & \text{Different DPVs} \\
\hline
0.1 & 0 \\
0.2 & 50 \\
0.3 & 100 \\
0.4 & 150 \\
0.5 & 200 \\
0.6 & 250 \\
0.7 & 300 \\
0.8 & 350 \\
0.9 & 400 \\
1.0 & 450 \\
0.1 & 500 \\
\end{array}\]
force attack would be necessary in order to find the correct DPV.

4. Conclusions

In this work we have presented a commutative watermarking and encryption system, based on a layered scheme and on a key dependent transform domain. Although the proposed method presents some similarities with [12,13], there are important differences with the cited works. First of all, here we proposed to modify the same wavelet coefficients with both the AES encryption and the watermarking, achieving the commutative property in a different way. Moreover, the key dependent Tree Structured Haar transform domain improves the overall security of the system.

The proposed method grants the authenticity of the transmitted data, thanks to the watermarking technique, and the privacy, obtained through the encryption procedure. The security system is extremely flexible since the decryption and the watermark extraction can be performed simultaneously or in different stages. Experimental tests have shown the effectiveness of the proposed method.

References