# A Robust Watermarking Scheme based on the JPEG2000 Standard

Julià Minguillón

Jordi Herrera-Joancomartí

David Megías

Estudis d'Informàtica i Multimèdia

Universitat Oberta de Catalunya

Av. Tibidabo 39-43, 08035 Barcelona, Spain

E-mail: jminguillona@uoc.edu

**Abstract**

In this paper an empirical evaluation of the lossy compression properties of the JPEG2000 standard for watermarking and fingerprinting purposes is performed. The JPEG2000 standard is used as a basic tool for determining both how and where the embedded watermark should be placed in the image. The original image is slightly modified in order to generate a similar image (but indistinguishable by the observer), and the mark is embedded in the pixels presenting differences between both images. The reconstruction process uses the original and the modified images to detect the embedded mark in any possible attacked image, so the watermarking scheme is non-blind. Previous experiments show that the properties of the resulting watermarking scheme depend strongly on the transformation stage characteristics of the lossy image compression system. Several parameters related to the JPEG2000 standard are tested, in addition to the compression ratio determined by the desired bit rate: the wavelet transform, block size, and the number of levels of decomposition for each block. These parameters affect not only the capacity but also the robustness of the watermarking scheme, which depends on the number of differing pixels between the original image and the slightly modified one, and the distribution of such differences. For evaluation purposes, we use the Stirmark benchmark and the classical image corpus set for lossy image compression. We compare the impact on system performance of each of the JPEG2000 standard parameters for several kinds of attacks, namely filtering (including sharpening), JPEG lossy compression, cropping, row and column removal, and a combination of rotation, cropping and scaling. False positive rate is also studied. Results show that the proposed watermarking scheme based on the JPEG2000 standard is robust against most of the classical image manipulation operations and, therefore, suitable for watermarking purposes. Furthermore, the proposed watermarking scheme can be also upgraded to be used as a fingerprinting scheme once the embedded mark is properly coded, due to the use of a collusion scheme.

**Keywords:** watermarking, fingerprinting, JPEG2000, lossy compression

# 1 Introduction

Both Internet and the massive use of high resolution digital cameras facilitate image creation and distribution for both final users and professionals. Nowadays, images are a common tool and a resource for different business areas. Photographic archives are also scanned for storing thousands of historical and classical pictures in digital format for their manipulation and sale. Several business areas, such as advertising, mass media, and multimedia content creation, have been subject to an important growth and also to a technological change in their process, due to the the increasing demand of the digital image market. Therefore, it is important to promote document rights management systems in order to guarantee the copyright of such images.

Basically, there are two kinds of copyright protection schemes, depending on whether they try to avoid forged copies, that is, the possibility of making copies; or, assuming that illegal copies will be generated, to allow the detection of the copier in order to minimize the distribution of unauthorized copies.

Electronic copyright protection schemes based on the principle of copy prevention have proven ineffective or insufficient in the last few years [1, 2]. Pragmatic approaches, like the one adopted for protecting DVD [3], combine copy prevention with copy detection. Watermarking is a well-known technique for copy detection, whereby the merchant selling the piece of information (*e.g.* an image) embeds a *mark* in the copy sold. This hidden mark can only be recovered by the owner of the image, and it may be used to prove the ownership of such digital document. Fingerprinting is a special case of watermarking where the information embedded in the digital document also includes data about the buyer, not only about the owner. In a watermarking scheme, the hidden message (the mark) is the same for all buyers, but in a fingerprinting scheme such mark depends on the buyer's identity. This subtle distinction implies important differences between both techniques. While watermarking schemes only allow proofs of ownership of the product, fingerprinting schemes are much more powerful since they allow to identify the illegal redistributor. So fingerprinting deals with problems often referred to as *traitor tracing* [4].

Every watermarking and/or fingerprinting scheme can be described in two stages: mark embedding and mark reconstruction. Since the former determines the latter, the real problem is *where* and *how* the mark should be placed into the product. Intuitively, as many marks as possible (actually, as many copies of the same mark) should be embedded in the digital document, to ensure that a copy of the mark will be found in a copy of a watermarked image even though such copy might have been manipulated and, therefore, increase the overall robustness. Nevertheless, the embedded mark should not be perceptible and, obviously, should not be easily removed by simple manipulations of the digital content. Therefore, a tradeoff between robustness and perceptual quality must be achieved.

In image watermarking schemes, the mark embedding process can be performed in different ways, since images allow multiple manipulations without affecting their perceptual quality. But, since robustness is the most important watermarking property, questions like where and how to place the mark are important issues. For instance, it is well known that the mark should not be placed in perceptually insignificant pixels of the image, because although the imperceptibility of the mark increases, robustness is drastically reduced. This problem arises because many signal and geometric processes affect these perceptual insignificant components. To determine the significance of each pixel, many watermarking schemes work in the transformed domain. Approaches based on the discrete cosine transform [5, 6, 7, 8] or the discrete wavelet transform [9, 10, 11] are robust against JPEG and JPEG2000 compression attacks, respectively, since the mark location is determined according to such compression techniques.

In this paper, we present an empirical evaluation of the lossy compression properties of the JPEG2000 standard for watermarking and fingerprinting purposes. The presented watermarking scheme is based on the ideas of a previous proposal [8] where the JPEG compression algorithm was used during the mark embedding process. In this newer proposal, we use the JPEG2000 rather than the JPEG standard, and different configurations of the JPEG2000 parameters, such as the compression ratio, the wavelet transform, the block size, and the number of levels of decomposition for each block, are tested to determine which configuration gives the best robustness results.

The organization of this paper is as follows. In Section 2, the basic concepts of JPEG2000 lossy image compression standard are defined. The parameters that determine the compression ratio are described focusing on their relationship with the watermarking scheme desired properties. Section 3 presents the method that describes the new watermarking scheme in two parts: mark embedding and mark reconstruction. The special case of fingerprinting and the necessary adaptations for color images are also addressed there. Section 4 concentrates on the properties of the resulting watermarking schemes and is focused mainly on the robustness assessment, although imperceptibility and capacity issues are also addressed here. Finally, in Section 5, conclusions and some guidelines for current and further research in this subject are outlined.

# 2 Basics of the JPEG2000 compression standard

The JPEG2000 [12] is an image compression standard that supports lossy and lossless compression of grayscale or color images. The JPEG2000 was developed by the Joint Photographic Experts Group (JPEG) in the aim of improving the properties of its predecessor, the JPEG standard [13]. The main use of both standards is lossy rather than lossless compression, that is, the image obtained after the compressed image is expanded to its original size is not exactly the original image, but a very similar approximation (under visual criteria). Although there are several lossy image compression methods that claim better performance than the JPEG standard (based on tree structured vector quantization, or fractal encoding, for example), the lack of standardization of such methods and the unavailability of useful software tools, such as the Independent JPEG Group (IJG) [14] library, for example, have been crucial for the success of the JPEG standard. The new Jasper software [15] provides the same functionalities as the Independent JPEG Group library but for reading and creating JPEG2000 images. Some of the features of the new JPEG2000 standard are:

1. progressive recovery of an image by fidelity or resolution;
2. region of interest coding, whereby different parts of an image can be coded with differing fidelity;
3. random access to particular regions of an image without needing to decode the entire code stream;
4. a flexible file format with provisions for specifying opacity information and image sequences; and
5. good error resilience.

The JPEG2000 codec follows the four stages model commonly used in different lossy compression techniques: preprocessing, transformation, quantization and coding. Each stage tries to reduce the number of bits used to represent the information in the image, by both removing imperceptible details and using efficient bit coding algorithms, with the aim of improving both perceptual quality and compression ratio, by taking advantage of the image intrinsic properties.

The preprocessing stage of the JPEG2000 ensures that its input sample data have a nominal dynamic range that is approximately centered about zero. Other operations such as cropping and scaling are also performed in this stage. Image is also tiled using a tile size which can be specified as a parameter. Each tile will be further divided in blocks which are coded independently. The JPEG2000 standard defines a complex hierarchical structure for all elements present in the coded stream in order to ensure a fast and target oriented decompression.

In the transformation phase, two main operations are performed. The first one, referred to as an intercomponent transform, deals with color properties. It operates on all of the image components together, and is aimed to reduce the correlation between components, leading to an improved coding efficiency. There are two different intercomponent transforms allowed in the JPEG2000 standard: the irreversible color transform (ICT) and the reversible color transform (RCT). The ICT is non-reversible and real-to-real in nature and corresponds to the classic RGB to YCrCb color space transform. The RCT is reversible and integer-to-integer, which is simply a integer approximation to the ICT. By "reversible" it is understood that the original values before applying the transformation can be exactly reconstructed when the inverse transformation is applied to the transformed data. The YCrCb color model separates luminance and chrominance channels in order to use a different compression setup tuned for each channel, taking advantage of the properties of the human visual system (HVS), which is less sensitive to chrominance than to luminance information. The second operation performed within the transformation stage is the intracomponent transform. In this phase, the wavelet transform is applied to each component individually. Two different wavelet transforms can be applied: 1) the 5/3 wavelet transform which is reversible, integer-to-integer and nonlinear (referred to as the "integer transform"); and 2) the 9/7 transform (real-to-real) lacking the reversible property, which can only be used for lossy compression (referred to as the "real transform"). The number of levels $k$ is a parameter of each transform which may vary from 1 up to $k$, being $2^k$ the image size. Typically, $k$ ranges from 3 to 6.

The quantization stage allows greater compression ratios, by representing the transform coefficients with only the minimal precision required to obtain the desired level of image quality. A different quantizer is employed for the coefficients of each subband, and each quantizer has only one parameter: the step size. These parameters are determined by the desired compression ratio or, analogously, the bit rate $br_C$. These quantizer step sizes used by the encoder are conveyed to the decoder via the code stream. This stage concentrates the information loss in the JPEG2000 standard and, therefore, it is not reversible by nature.

Finally, the coding stage uses the output of the previous stage and assigns a code which tries to minimize the number of bits needed to represent them, grouping coefficients in blocks. The same block size is used across all decomposition levels, so the maximum block size is determined by the coarsest level image size. This stage does not introduce any loss in the data and is completely reversible.

In Section 4, we vary different parameters from the transformation stage and the quantization process of the JPEG2000 to study how they affect the properties of a watermarking algorithm based on such a lossy compression algorithm. More precisely, in the transformation stage, we modify the number of decomposition levels, and the block size. Regardless the quantization stage, the bit rate is also studied trying to reproduce a real scenario usage for image compression. We discard the use of the integer transform since we assume that for truly lossy image compression (as needed by our watermarking scheme) it is not useful to try to recover the original image as it is obviously modified by the embedded mark.

# 3 Proposed Watermarking Algorithm

The watermarking scheme proposed in this section is non-blind in the sense that the original image is required during the reconstruction process. Although this may seem a limitation in a real scenario where thousands of images must be individually inspected, it is not a real issue because the owner is supposed to keep a copy of each sold image, and there exist several image retrieval techniques [16] that might be used to reduce the computational cost, see [17] for example, where a robust (in the watermarking sense) image hashing algorithm is developed. With such algorithm, the watermarked image can be summarized as a bit string and, therefore, images searches can be performed much more efficiently. On the other hand, the fact that the embedded mark is not needed in the reconstruction algorithm makes it suitable also for fingerprinting techniques. For watermarking purposes, it is not necessary to recover all the embedded bits exactly, but only a reasonably (high) percentage, because the probability of finding $n$ bits randomly is $2^{-n}$, which can be neglected for a big enough $n$. On the contrary, for fingerprinting purposes, the buyer must be identified with no doubt, so all the embedded bits must be recovered. In fact, the implementation used in Section 4 to test the properties of the proposed watermarking scheme uses a dual Hamming binary code as an error correcting code. This technique [18] allows to avoid the collusion of two buyers when the algorithm is used as a fingerprinting scheme, and it is fully described in Section 3.4.

As described above, any watermarking scheme can be described in two stages, namely, mark embedding and mark reconstruction. The next subsections describe these processes.

## 3.1 Mark Embedding

As we pointed out in Section 1, the mark embedding process determines *where* and *how* the mark is placed into the image. For our construction, we use the concept of *perceptual capacity*. This concept is defined [6] as the quantity of additional information which can be added without any (or with minimal) impact to the perceptual fidelity of the data. Our embedding algorithm determines a perceptual capacity value for each pixel.

A general overview of the mark embedding process is depicted in Figure 1.

To determine the mark location we use the JPEG2000 lossy compression algorithm. The original image is compared with a JPEG2000 compressed and then decompressed version of the image, that is, a close reproduction of the original image. The pixels that differ between both images determine the position of the embedded bits. Once the location of the mark is determined, the perceptual capacity of each pixel is calculated as the difference between both images. Such information will be used to embed the mark into the image.

More precisely, we represent an image as $X = \{x_i : 1 \leq i \leq n\}$, where $n = w \times h$ is the number of pixels ($w$ and $h$ are, respectively, the width and the height of the image) and $x_i$ is the gray level of the $i$-th pixel. The original image is denoted by $X$ while the watermarked image is $X''$. The mark $\varepsilon$ to be embedded into the original image is a bit stream.

To determine the position of the mark bits, a temporary JPEG2000 compressed image, referred to as $X' = JPEG2000(X)$, is obtained. Let denote by $\delta_i = x_i - x'_i$ the difference between corresponding pixels in $X$ and $X'$. The mark bit locations are fixed by the positions $i$ for which $|\delta_i| > \tau$. Usually, $\tau = 0$, so we select all modified pixels, thus increasing capacity, but we could use larger values for $\tau$ reducing capacity but ensuring that only "robust" enough pixels will be selected for hiding the mark. We take the value $\delta_i$ as the perceptual capacity of the $i$-th pixel. Nevertheless, we could use any value in the range $[1..|\delta_i|]$, as we need to obtain a slightly different image. Therefore, both $\tau$ and $\delta_i$ can be considered two parameters for fine-tuning the proposed watermarking scheme regardless the way the modified image is obtained.

Prior to the inclusion of the mark $\varepsilon$ into the image, some coding is performed to increase the robustness level of the scheme. On the one hand, an error-correcting code is applied and more redundancy is obtained by replicating the mark. On the other hand, the obtained extended mark is encrypted so that it cannot be extracted without the encryption key. The steps followed for embedding the mark are as follows:

1. Encode $\varepsilon$ using an error-correcting code (ECC) to obtain the encoded mark $E$.

2. Replicate the mark $E$ to obtain an extended mark $E'$ with as many bits as pixels in $X$ with $\delta_i \neq 0$.

3. Encrypt the extended mark using a pseudo-random bit sequence (PRBS) $\{s_i\}_{i \geq 1}$ generated by a

cryptographically sound pseudo-random generator with a secret key $k$. Then, the final value to include is $s'_j = e'_j \oplus s_j$, where $e'_j$ is the $j$-th bit of $E'$.

Henceforth, the term "mark" refers to the coded bits $s'_j$ rather than the original $\varepsilon$ value. The scheme described in Section 3 was implemented using a dual binary Hamming code $DH(31,5)$ as ECC and the pseudo-random generator is a DES cryptosystem implemented in an open feedback (OFB) mode. A 70-bit mark $\varepsilon$ (resulting in an encoded $E$ with $|E| = 434$) was included.

The mark inclusion is performed in the following way. The mark bits will be embedded into the image in the pixels $i$ where $\delta_i \neq 0$ by incrementing or decrementing the pixel as much as the perceptual capacity $\delta_i$. If we want to include a 0 in the $i$-th pixel, we compute $x''_i = x_i - \delta_i$, and we compute $x''_i = x_i + \delta_i$ if a 1 has to be embedded.

Finally, notice that the JPEG2000 algorithm block depicted in Figure 1 can be replaced with any lossy compression algorithm which produces a slightly modified image. However, previous results [19] show that the JPEG2000 standard is the most suitable lossy compression scheme and therefore, the main reason for replacing the JPEG standard used in [8] with the new JPEG2000 standard.

## 3.2 Mark Reconstruction

For mark reconstruction, since the watermarking scheme is non-blind, the original image $X$ and the secret key $k$ (used to regenerate the random sequence $\{s_i\}_{i\geq 1}$) are required. On the other hand, knowledge of the original mark $\varepsilon$ is not assumed, which allows to use the proposed algorithm for fingerprinting.

Once a redistributed image $\hat{X}$ (notice that the image $\hat{X}$ may not be equal to the watermarked image $X''$) is found, the mark can be extracted using the mark reconstruction process depicted in Figure 2.

This process consists of three main steps. The first one determines the pixels where the bit marks are hidden. The second step fixes the value found in every position in the image $\hat{X}$ that is being verified. Finally, in the third step, the decoding of the found mark is performed.

More precisely, to find the pixels where the bit marks are placed, the process performed in the embedding algorithm is repeated. Thus, the JPEG2000 algorithm is applied to the original image $X$ giving $X' = JPEG2000(X)$. The pixels where $\delta_i = x_i - x'_i \neq 0$ are determined. Notice that the JPEG2000 parameters used to obtain $X' = JPEG2000(X)$ must be exactly the same that the ones used in the embedding algorithm, otherwise the mark bits location will not be the correct ones.

Then, for each pixel where $\delta_i \neq 0$, the value $\hat{\delta}_i = \hat{x}_i - x_i$ is computed between the image $\hat{X}$ and the original one $X$. If $\hat{\delta}_i = 0$, that means such mark bit has been removed from $\hat{X}$ and the corresponding bit $\hat{s_j}$ of the reconstructed mark is set up as an erasure. Otherwise, if $\hat{\delta}_i \times \delta_i > 0$, then $\hat{s}_j = 1$ and if $\hat{\delta}_i \times \delta_i < 0$, then $\hat{s}_j = 0$. In this way, the embedded sequence $\hat{s}$ is obtained.

Finally, the decoding part determines the mark $\hat{\varepsilon}$ as follows. First of all, the extended mark $E'$ is obtained from $\hat{s}$ by subtracting the PRBS $\{s_i\}_{i \geq 1}$ added in the embedding process. Such PRBS is generated with the same pseudo-random generator and secret key $k$. Since the extended mark $E'$ is the replication of the encoded mark $E$, the $j$-th bit of the value $E$, $e_j$, is obtained using a voting scheme among the different $e_j$ values contained in $E'$. At the end, the ECC is applied to $E$ to decode $\hat{\varepsilon}$.

### 3.3 Color images

Several alternatives have been suggested to apply watermarking techniques to color images. One commonly used approach consists of marking the image luminance component only, which makes the extension of gray level techniques, such as the one described in this paper, straightforward. The obvious advantage of choosing the luminance component of the image is the fact that the mark is preserved when the color image is converted to gray level, whereas watermarking approaches which consider also the chrominance components do not satisfy this property. However, there are also some drawbacks related to luminance watermarking. The most relevant one is that luminance information is more perceptible to the HVS than chrominance. Hence, when imperceptibility is the priority, the chrominance component should be taken into account.

Some previous color watermarking schemes are briefly reviewed next. In [20], the blue channel is used to embed the mark, since it is claimed that the HVS is less sensitive to the blue band in the RGB space. In [21], the three RGB channels are used to embed the mark, but, taking into account the HVS sensibility to the spectrum, the magnitude of the mark is 10 times more significant for the blue component compared to the green one, and 5 times more significant for the blue component compared to the red one. A quite different approach is presented in [22], which embeds a sum of sinusoids into the yellow-blue channel of the opponent-color representation. [22] extends the Lab space to the S-CIELAB decomposition which includes the spatial structure of the image. A more recent result can be found in [23], where a quaternion (hypercomplex) representation is used to model the RGB space, and then the quaternion Fourier transform is applied to embed the mark in the transformed domain.

Despite its drawbacks, we have decided to embed the mark in the luminance component, since the mark will be preserved when gray level conversion is applied.

## 3.4 Fingerprinting

As stated previously, there are two basic kinds of marks: *fingerprints* and *watermarks*. One may think of a fingerprint as a serial number (*i.e.* something identifying the buyer of the copy) while a watermark is an embedded copyright message similar to the © symbol that appears on books.

Upgrading a watermarking scheme to a fingerprinting scheme may seem an easy task: the merchant embeds on the image different information for each different buyer. However this simple approach does not always yield a useful fingerprinting scheme since two main problems can arise:

1. Need for the mark in the mark reconstruction process.

2. Security against collusion attack.

The first property depends on the mark reconstruction process of the watermarking scheme. If the mark reconstruction process needs the embedded mark as an input, once the merchant finds a redistributed copy he does not know a priori who the original buyer is, and he should try all the buyers' marks which could be unfeasible for large number of users.

On the other hand, in the fingerprinting scenario, schemes need extra properties derived from the fact that different buyers get different marks. In particular, security against collusion attacks is needed. In a collusion attack, two or more different buyers can detect the mark bits by comparison of their copies of the same information item (*marking assumption*, [24]). After detection, there are two attacking strategies:

**Mark bit deletion** The mark bits that differ between buyers are deleted. At mark reconstruction time, $M$ knows where mark bits should be, so he will try to restore deleted mark bits. The probability of correctly restoring a deleted mark bit is 1/2.

**Mark bit tweaking** Colluders mix their marked copies in an attempt to tweak some mark bits. This attack is worse than deletion, as $M$ has no way to detect that the bit was tweaked.

Encoding the mark using an error-correcting code (ECC) prior to embedding is an obvious alternative to increase robustness against mark bit deletion and tweaking. However, if the number of bits tweaked (or incorrectly restored from deletion) is greater than the maximum number of errors that the ECC can correct, then mark reconstruction will fail.

In [24], collusion-secure fingerprinting was introduced. Rather than using standard ECCs, a new type of codes called $c$-secure codes were introduced in that proposal to resist collusions of up to $c$ buyers. $c$-secure codes have the property that intersection of any set of $c$ words can determine, at least, one of the words of the set. In this way, an attack by comparing $c$ or less copies of the marked image will produce a marked image that identifies, at least, one of the colluders. Later on, in [18] dual binary Hamming codes where presented as a 2-secure codes. Despite those codes only solve collusion of two buyers, they have a shorter wordlength than those proposed in [24, 25], and then capacity of the fingerprinting scheme is increased.

The effectiveness of a two-buyer collusion depends on the distance between codewords of the ECC used in the mark embedding algorithm. The larger the distance, the more bits will differ between the codewords of colluders, *i.e.* the easier to tweak a codeword bit. On the other hand, the smaller the distance, the less error-correcting capacity will be obtained from the ECC. Dual binary Hamming codes offer a good tradeoff, because the distance between any two codewords is fixed to half their length. More precisely, a dual binary Hamming code of length $N = 2^n - 1$ consists of $N$ codewords (excluding the all zeroes codeword) such that the distance between any two different codewords is $2^{n-1}$. Then, as proved in [18], when every buyer is identified for a codeword, an innocent buyer will never be declared guilty (*2-frameproofness*) and the probability that a participant in a two-buyer collusion can be identified is

$$1 - \left( \left( \frac{1}{2} \right)^{2^{n-1}} \cdot 2^n \right).$$

### 3.4.1 Configuration for fingerprinting enhancement

We enhance the watermarking scheme proposed in the previous sections to a fingerprinting scheme in the following way. Since the mark reconstruction process of the presented scheme does not require the original mark, the possible drawback stated before related with the need for the mark embedded does not hold.

On the other hand, in order to obtain a 2-collusion secure fingerprinting scheme we code the mark using a dual binary Hamming code $DH(31, 5)$ as 2-secure code, before the mark is embedded into the image.

## 4 Experimental Results

Three main measures [26] are commonly used to assess the performance of information hiding schemes, which are a general class including watermarking schemes:

**Imperceptibility:** the extent to which the embedding process leaves undamaged the perceptual quality of the marked image.

**Capacity:** the amount of information that may be embedded and then recovered.

**Robustness:** the resistance to accidental removal of the embedded bits.

To determine the reliability of the system, false positive rate is also measured. Such measure shows the probability of finding a correct mark in an image which has not been watermarked. As pointed out in [27], there are two subtly different ways to define the false positive rate. On one hand, we can measure the probability of, given a fixed image and a randomly selected mark, the mark reconstruction process confirms the selected mark is embedded into the given image. On the other hand, we can fix the mark and measure the probability of finding such mark into randomly-selected images. We use this second definition, since it is more suitable for watermarking schemes where proof of ownership is the main goal. In particular, our randomly-selected images are the non-marked original images distorted with different Stirmark attacks.

In this section, we test the properties of the proposed scheme presented in Section 3, depending on the parameters used in the JPEG2000 standard for obtaining the modified image $X'$. The proposed watermarking scheme has been implemented by tuning different parameters of the JPEG2000 standard. We discuss the performance of every resulting algorithm derived from a different parameter configuration, using the bit rate as the variable for fine-tuning and analyzing the proposed watermarking scheme, extending the work described in [28]. A comparison of the different results for each one of the three properties mentioned above has been carried out.

## 4.1 Experiment setup

To determine the experiment setup, two different stages have been defined. In the first stage, several configurations for the parameters of the JPEG2000 standard are considered in order to measure the impact of those parameters on imperceptibility and capacity properties. In the second stage, we only deal with the configurations selected in the first stage as the best ones, and the robustness of the proposed watermarking scheme with such configurations is studied.

For the first stage, we evaluate the imperceptibility and capacity properties of the JPEG2000 standard using the following configurations: block size $B$ ranges from 8 up to 64, using always square blocks, and the number of levels of decomposition of the wavelet transform $L$ ranges from 3 to 6. The compression bit rates (denoted by $br_C$) used are 1, 0.75, 0.5, 0.375, 0.25, 0.1875, 0.125, 0.09375, 0.0625, 0.046875, 0.03125 and 0.0234375 bpp, thus a total of 12 experiments have been carried out for each parameter configuration. Due to the fact that we need to modify the original image to embed the mark, it is useless to apply the reversible wavelet transform, so we have used only the irreversible (real) transform. This makes a total of 192 experiments (in the first stage) to determine the best configurations for the JPEG2000 standard parameters. No other properties of the JPEG2000 standard such as tiling are considered, since the main objective is to obtain a similar image with a high PSNR with respect to the original one. This setup is much richer than the previously described in [8] (which is based on the JPEG standard), as such setup only allowed the modification of the quality factor. Finally, for the second stage, only the best 12 configurations (the highest 12 PSNR values) will be selected from the first stage to test the robustness property of the proposed watermarking scheme.

The following ten images[1] were used as the corpus set for the experiments: *Lenna*, *Balloons*, *Barbara1*, *Barbara2*, *Board*, *Boats*, *Girl*, *Goldhill*, *Hotel* and *Zelda*. These images are the standard corpus set as defined in [29]. All of these images were manipulated to be $512 \times 512$ pixels performing a central cropping of the original copy. This means that a total of 1920 tests have been carried out for the first stage, and 120 for the second stage.

## 4.2 Imperceptibility

The imperceptibility property determines how much the watermarked image $X''$ differs from the original one $X$. That is, how much the inclusion of the mark distorts the original image. In the image compression field, such a property is referred to as image quality and different measures can be performed.

[1] `http://sipi.usc.edu/services/database/Database.html`

Image quality is usually measured using the peak signal-to-noise ratio (PSNR), although this measure lacks of a direct relationship with the quality perceived by the observer, specially when geometrical distortions are applied [30]. Several image quality measures such as weighted PSNR (wPSNR) and others which include information related to perceptual issues [31] have been studied in order to overcome this problem, although there is no consensus for determining which one should be adopted as the standard. Such problem has also been addressed in the watermarking community [32]. Like other authors in the watermarking field, we will use PSNR in our experiments as the basic tool for measuring image qualities.

The proposed watermarking scheme allows to determine such property (indirectly) *a priori* since, in the embedding algorithm, the PSNR of the watermarked image with respect to the original image is very similar to the PSNR of the modified image after the lossy compression stage. The imperceptibility can be set up as much adjusted as desired, but such value determines the capacity of the watermarking scheme and also affects the robustness. A tradeoff between imperceptibility and capacity must thus be achieved. The quality of the watermarked image strongly depends on the first stage of the proposed watermarking scheme, showing the direct relationship between lossy compression and image watermarking. Unfortunately, unlike for the JPEG standard (see [33]), it is not possible to specify a target PSNR for the compressed image, so several parameter configurations must be tried to achieve the desired PSNR. Nevertheless, the JPEG2000 standard generates compressed images within a PSNR range which is more dynamic than the JPEG standard, and visible artifacts are also more imperceptible, specially for low bit rates, as stated in [12]. Our results show that, for the same capacity, PSNR is slightly above for the JPEG2000 standard.

Table 1 shows the relative importance of each parameter of the JPEG2000 standard with respect to the obtained PSNR. Although it is possible to predict such importance using intuitive criteria, it is easy to prove it using a three-way contingency table model and studying the relationships found between the measured PSNR and the configuration parameters. Table 1 is computed using the number of times that a combination of block size and number of decomposition levels yields a worse PSNR than the other combinations of block size $B$ and number of decomposition levels $L$ for all images and for all bit rates $br_C$. This computed score is called the parameter configuration index. The smaller this score is, the more influent the parameter combination is and, therefore, preferred. Obviously, the compression bit rate determines the PSNR for a given configuration of block size and number of decomposition levels. The larger the compression bit rate, the higher the PSNR achieved. Regarding the other parameters, the block size is more important than the number of decomposition levels for large block sizes. Surprisingly, five decomposition levels seem to be better than the default JPEG2000 standard value, six, although this is probably related to image size; for larger images, six decomposition levels would probably have obtained better results. Therefore, in the light of these results, the 12 configurations with highest PSNR and a reasonable capacity are those which combine a block size of 32 or 64, a number of levels of decomposition of 5 or 6, and, obviously, the three highest compression bit rates, 2.0, 1.0 and 0.75 bpp, as expected. Table 1 shows the ranking of these 12 configurations according to the obtained results. Once again, this selection has been done having in mind that our main goal is to obtain similar images (suitable for the first stage of the proposed watermarking scheme), and not low or very low bit rate images. From left to right, Figure 3 shows the original *Lenna* image, the slightly perturbed version using $br_C = 0.75$, $B = 64$ and $L = 6$, and the watermarked image, respectively. The modified image has a PSNR of 39 dB with respect to the original one, with a 84.9 % of modified pixels (labelled as % NMP). The watermarked image has also a PSNR of 39 dB. The three images are visually indistinguishable, as desired. As expected, the JPEG2000 standard achieves a better image quality and compression ratio than the JPEG standard, so imperceptibility is improved with respect to [8].

Even though the mark embedding process modifies the original image by adding or subtracting $|\delta_i|$, it is worth to remark that such changes are still imperceptible (for reasonable PSNR values) because most $|\delta_i|$ values are small. For example, for all the 12 selected configurations and the 10 images in the corpus set, the percentage of modified pixels with $|\delta_i| \leq 10$ is 99.3 %. On the other hand, for low or high luminance ranges, the $|\delta_i|$ values are clipped in order to avoid false spots (white or black pixels).

Experiments also show that the ratio $\delta_i/x_i$ is small for most pixel values. Finally, the quantization noise introduced by the embedding mark process can be considered imperceptible: suppose $X$ is a random variable describing the quantization noise generated by the lossy compression process, and that $Y$ is a binary random variable taking values $-1$ or $+1$, which describes the mark embedding process (adding or subtracting $|\delta_i|$). Obviously, $X$ and $Y$ are uncorrelated. Therefore, the resulting random variable of multiplying $X$ and $Y$ is a random variable with the same distribution of $X$ (if $X$ is considered to be symmetrical). Therefore, the mark (that is, the modified quantization noise) added to the original image) can be considered also quantization noise, which is perceptually imperceptible for reasonable PSNR values.

## 4.3 Capacity

The capacity of the watermarking scheme is also determined by the parameters of the JPEG2000 standard used in the embedding process. Figure 4 show the relationship between the PSNR of the modified image $X'$ with respect to the original one $X$ and the percentage of pixels which are changed (namely $C$) when different parameters of the JPEG2000 are applied to the *Lenna*, *Balloons* and *Barbara2* images, that is, the percentage of pixels with a perceptual capacity $|\delta_i| > 0$. As we pointed out in Section 3, every pixel which is modified allows one bit to be embedded.

Notice that, as expected, the number of bits which can be hidden in the original image decreases as the PSNR increases, since both images $X$ and $X'$ become more similar. However, for reasonable PSNR (above 36 dB, for example), the number of bits that can be hidden is still really large (over 85% of the original pixels are modified). For example, for the *Lenna* image, which is $512 \times 512$ pixels, this means that about 220000 bits may be hidden. As we need 434 bits for each mark embedded in the image (see section 3.1), this means that the mark can be replicated more than 500 times. For the other images in the corpus set, similar results are obtained. For the selected configurations, results show that the best results in average should be obtained with $B = 32$ and $L = 5$, followed by $B = 64, L = 5$, then

$B = 32, L = 6$ and, finally, $B = 64, L = 6$, that is, number of decomposition levels is more important than block size for capacity purposes. Obviously, the larger the compression bit rate $br_C$ is, the lower the capacity is. We will arrange the results in the robustness assessment section according to this ordering.

Although the percentage of modified pixels depends on the image intrinsic characteristics, the results for all the images in the corpus set and for each set of parameters of the JPEG2000 standard used in this paper show that, for PSNR values up to 45 dB, more that 70% of the pixels are modified. It is interesting to remark that although the obtained PSNR also strongly depends on the image intrinsic characteristics, the relationship between capacity and PSNR shows a strong logarithmic correlation for all the images in the corpus set, and for any configuration of block size, number of decomposition levels and compression bit rate. The percentage of modified pixels is slightly above (around 2 %) the results obtained in [8] where the standard JPEG was used (instead of the new JPEG2000 standard), specially for large compression bit rates, which are the preferred configurations.

## 4.4 Robustness Assessment

The robustness of the watermarking scheme proposed in this paper has been evaluated using the base test of the Stirmark 3.1 benchmark [1, 34]. We have compared the performance of each of the experimental set parameters of the JPEG2000 for several kinds of attacks, namely filtering (including median, Gaussian, Frequency Mode Laplacian Removal (FMLR) and sharpening), JPEG compression, cropping and row and column removal. We also use the JPEG2000 standard as another attack based on lossy compression. It is worth pointing out that we have used the base test of Stirmark 3.1 benchmark as it is provided, accepting all the implemented attacks. However, some of the attacks that cannot be resisted for the proposed watermarking scheme produce images which have poor perceptual quality. For instance, the JPEG compression attack with small quality factors (10, 15 or 20, for instance) produce versions of the *Lenna* image with PSNR lower than 33 dB and with a high level of visual degradation, including block artifacts. Other attacks cause evident image deformations, but the PSNR is not able to capture such visual degradation. Thus, perceptual based image quality measures should be used instead, see [30] for example, although the lack of standardization of such measures and their restricted use make them virtually useless for the sake of comparison.

Due to the design of the embedding algorithm, robustness has a strong relationship with the perceptual capacity of each modified pixel. Figure 5 shows the relationship between PSNR and the perceptual capacity length average $|\delta_i|$ for the *Lenna*, *Balloons* and *Barbara2* images and the different parameter configurations. Remember that $\delta_i$ is the quantity added or subtracted to each modified pixel in order to embed a single bit. The smaller $\delta_i$ is, the more likely such bit is to be erased when the watermarked image is attacked. On the other hand, the imperceptibility property can be lost for several image areas with large $\delta_i$, so once again a tradeoff between PSNR and capacity must be established. Once again, the perceptual capacity length average is slightly better than the results obtained with the setup described in [8] where the JPEG standard was used.

Robustness has been assessed using a correlation measure between the embedded mark $W$ and the identified mark $W'$. Let $W_i$ and $W'_i$ be, respectively, the $i$-th bit of $W$ and $W'$, and let

$$\beta_i = \begin{cases} 1, & \text{if}\, W_i = W'_i, \\ -1, & \text{if}\, W_i \neq W'_i. \end{cases}$$

Now, the correlation $c$ is computed, taking into account $\beta_i$ for all $|W|$ bits (70 in our case) of the mark, as follows:

$$c = \frac{1}{|W|}\sum_{i=1}^{|W|}\beta_i.$$

This measure is 1 when all $|W|$ bits are correctly recovered ($W = W'$) and $-1$ when all $|W|$ bits are misidentified. A value of about 0 is expected when 50% of the bits are correctly recovered, as it would occur if the mark bits were reconstructed randomly. In this paper, we consider that the watermarking scheme survives an attack **if the correlation is greater than or equal to 0.8**, that is, if at least 90% of the mark bits are correctly recovered. If we set this threshold to 100% (all the bits must be recovered for considering that an attack is not successful), the presented watermarking scheme could be also used for fingerprinting, once the mark is coded using an appropriate method like those proposed in [24, 18]. In this paper, we explore both possibilities.

When no attacks are performed, the method is robust (because it is completely deterministic) and the embedded mark is always recovered, with a correlation of $c = 1$, as expected. In the following sections we describe the results obtained for each family of attacks.

### 4.4.1 Filtering

There are six filtering attacks: median filter (with window sizes of $2 \times 2$, $3 \times 3$ and $4 \times 4$), frequency mode Laplacian removal [35], Gaussian filter and sharpening, hence a total of $6 \times 10 \times 12 = 720$ experiments have been performed.

The results show that the $2 \times 2$ and the $3 \times 3$ median filters, the Gaussian filter and the sharpening filter attacks are always survived, for any image and for any configuration parameter. For the other two filtering attacks, the $4 \times 4$ median filter is always survived but for one image (*Balloons*, the smoothest one in the corpus set) and one configuration. In this case, the measured correlation is $c = 0.69$, which is below the threshold specified (0.8) and, therefore, it is considered a successful attack. Nevertheless, relaxing the correlation threshold this attack could also be considered survived. Notice that the PNSR of the images filtered using the $4 \times 4$ median filter is sometimes below 30 dB, which is totally unacceptable in most cases. Finally, regarding the FMLR filter, it is clearly the most problematic attack, as shown in Table 2. Only a few marks can be correctly recovered, being the compression bit rate the parameter more important to determine robustness. Once again, the smoothest images are the most likely to be successfully attacked with this kind of filter. Obviously, as lossy compression performance is determined by image intrinsic characteristics (smoothness among others), robustness is somehow dependant on such characteristics.

Regarding the robustness of the proposed watermarking scheme for fingerprinting purposes, the attacks based on the $2 \times 2$ and the $3 \times 3$ median filters, the Gaussian filter and the sharpening filter attacks are always survived. However, only one configuration ($br_C = 0.75$, $B = 32$ and $L = 6$) resists the $4 \times 4$ median filter attack. Finally, the FMLR attack performs similarly to the watermarking scenario, it is the worst attack and no configuration resists it for all the images in the corpus set.

### 4.4.2 JPEG compression

This attack uses the JPEG standard varying the quality factor $Q$: 10, 15, 20, 25, 30, 35, 40, 50, 60, 70, 80 and 90. As expected, the number of attacks survived increases as the compression bit rate decreases, since more pixels are modified, increasing capacity. More precisely, for quality factors above 50, all attacks are survived for any image and any parameter configuration. For a quality factor of 50, only two attacks are successful, but measured correlations for both attacks are 0.75 and 0.77, so the proposed watermarking scheme could also be considered very robust for such value. For quality factors of 40 and 35, similar results are obtained, as only the smoothest images in the corpus set are successfully attacked. Nevertheless, 25 is the minimum acceptable quality factor without causing strong visible artifacts in the attacked image. Attacks using quality factors of 25 and below produce a low PSNR score with respect to the original one, with visible quality degradation and block artifacts. Table 3 shows the results for this family of attacks (several values of $Q$ have been omitted to avoid repeats).

In this case, fingerprinting results are very similar. For quality factors of 50 and above, only the smoothest images in the corpus set are successfully attacked. The proposed watermarking scheme can be considered robust in this scenario, because the attacked images have low PSNR values with visible artifacts.

### 4.4.3 JPEG2000 compression

This attack uses the JPEG2000 standard varying the attack bit rate (denoted by $br_A$, which must not be confounded with the compression bit rate defined in Section 4.1): 2.0, 1.0, 0.75, 0.5, 0.25, 0.125 and 0.0625 bpp, while maintaining $B = 64$ and $L = 6$, the default values of the Jasper [15] implementation which are considered to be optimal for most images. Once again, the number of attacks survived increases as the compression bit rate decreases. More precisely, for attack bit rates above or equal to 0.75 bpp, all attacks are survived for any image and any parameter configuration. For an attack bit rate equal to 0.5 bpp, only one attack cannot be survived, but the measured correlation is 0.77, so it could also be considered almost robust against such attack. On the contrary, for the two lowest attack bit rates (0.125 and 0.0625 bpp), all attacks are successful and the mark cannot be recovered. It is easy to see that robustness against this kind of attack is directly related to the compression bit rate used for generating the modified image as the first step in the proposed watermarking scheme, and that the proposed scheme is robust beyond the compression bit rate used. Table 4 shows the results for this family of attacks (several attack bit rate values have been omitted to avoid repeats). Surprisingly enough, robustness drastically drops about 0.25 bpp, so it seems to be a transition between 0.5 and 0.25 bpp which may be directly related to the bit parameter configuration used for obtaining the modified image. Therefore, fine-tuning the bit rate parameter becomes a crucial issue when combining compression and watermarking in a single process, thus it is not a clear relationship between the bit rate used for generating the modified image $br_C$ and the bit rate of the lossy compression attack $br_A$.

Once again, fingerprinting results are very good except for the highest compression bit rate ($br_C =$ 2.0), where even for reasonable compression ratios (16:1) the attack is successful for the smoothest images in the corpus set. Once again, this value is directly related to the bit rate used for obtaining the modified image, so robustness can be increased by using a smaller bit rate, thus increasing capacity (and therefore robustness) but reducing image quality.

### 4.4.4 Cropping

This attack performs a centered cropping with a factor of 1%, 2%, 5%, 10%, 15%, 20%, 25%, 50% and 75%. Notice that the proposed method uses synchronization to find a possible cropping windows, as all images (original, modified and possibly attacked) must have the same dimensions. Synchronization, which is independent of the watermarking scheme as it tries to undo all the changes made by the attack, can be a very expensive operation, as thousands of parameter combinations must be tested. The simplest synchronization mechanism tries to find the cropped image position within the original image by testing all possibilities (column and row shifts), which can lead to thousands of operations for high cropping rates, when the image sizes are very different.

We propose to use a more complicated synchronization mechanism based on partial histograms for determining the best possible position for the cropped image. The basic idea is to perform the reconstruction process only on those image regions with a similar histogram, instead of trying all the possible combinations determined by shifting the cropped image within the original one. Let $d_x$ and $d_y$ be the difference (in pixels) of images widths and heights respectively. A classical synchronization mechanism tries to find the embedded mark for each possible cropping window, that is, a total of $d_x \times d_y$ times, which is a very expensive operation. We propose to compute the histogram of the cropped image (using a reduced number of bins, 16 for example, instead of 256), and then to use a variable shift size $s$ to compute the histogram for each window located at $(i \times s, j \times s)$. When both histograms are similar (using a mean square error criterion), the mark reconstruction process is then performed. If the embedded mark cannot be found, $s$ is divided by two and the process is repeated again, avoiding the possible repeats. In the worst case, $s$ will be 1, so all the possible cropping windows will be tested and the mechanism is exactly as the sequential case. Nevertheless, the cost of computing the histogram of the moving window and discarding the mark reconstruction process is very fast, thus reducing the total computational cost. Although this synchronization mechanism is still under development, the experiments carried out in this paper show that it is possible to reduce the total amount of time in a factor of 10 or even 100, depending on image characteristics.

Table 5 shows the results for this family of attacks (several bit rate values have been omitted to avoid repeats). Notice that both watermarking and fingerprinting scores are identical. This is because the robustness is achieved via synchronization, although the large number of times the mark has been embedded in the image is also an important reason, as robustness is directly related with capacity. Thus, even for the highest cropping rates, it is possible to reconstruct the embedded mark. There are a few cases where the cropping attack is successful, specially for highest cropping rates (20%, 50% and 75%), because the cropped images are very smooth regions which are almost unmodified during the lossy compression stage of the mark embedding process and, therefore, only a few pixels are available to embed the mark in such region, causing a local low capacity condition which does not allow the reconstructing process to recover the embedded mark.

### 4.4.5 Row and column removal

This attack removes several rows and columns of the watermarked image. More precisely, the attacks have the following configurations: (1,1), (1,5), (5,1), (5,17) and (17,5), where the first component denotes the number of rows and the second one the number of columns removed. Once again, synchronization is used to determine the best coincidence between the watermarked and the attacked image, as in the case of a cropping attack.

In this case, the results shown in Table 6 are very surprising. Once again, and because of synchronization, watermarking and fingerprinting results are identical. When the number of columns and rows is small (five or less), the proposed watermarking scheme is always able to successfully reconstruct the mark mainly due to the voting scheme and the properties of the error correcting code. On the other hand, when the number of columns or rows removed increases, robustness is completely lost, as all attacks are successful. This is one of the weakest points derived from the raster scan followed by the proposed embedding mark scheme. To overcome this drawback, different strategies can be adopted during the embedding process. For instance, all the pixels in the image where the mark is embedded in could be sorted following Peano or Hilbert space-filling curves, instead of the sequential raster described in Section 3.1.

### 4.4.6 Rotation, cropping and scale

This attack performs a combination of attacks: a rotation followed by a cropping and a scaling, to ensure the resulting image has the same size than the original one. More precisely, the attacks have the following configurations (that is, the rotation angle): ±0.25, ±0.5, ±0.75, ±1, ±2, 5, 10, 15, 30, 45 and 90 degrees. No synchronization is performed because the attacked image maintains the original image size. Nevertheless, it could be possible to determine the parameters of the attack and try to reverse the rotation angle and scaling prior to the mark reconstruction.

The results show that this is, by far, the most successful attack among those presented in this paper. Only two attacks are occasionally survived (±0.25 degrees), while the embedded mark is never detected for higher rotation angles, as shown in Table 7. This is caused again by the spatial sequence followed by the watermarking scheme for embedding the mark, which is probably its main weakness against this kind of attacks.

## 4.5 False positive rate

False positive rate is defined as the probability of finding out a mark from an image which has not been watermarked, that is, when a false positive occurs. It should be computed for all possible images and for all possible embedded marks, but this is clearly unfeasible. As pointed out in Section 4, for a given mark (which is randomly generated, although we use the same mark for all the experiments), we try to find it in as many modified non-watermarked images as possible, expecting that the mark reconstruction process will be unable to recover such mark from such images.

Due to the intrinsic design of the proposed watermarking scheme, two different scenarios must be studied. In the first case, if an original image is slightly modified (as determined by the mark embedding process) and then such image (namely $X'$) is used as a possibly watermarked image (namely $\hat{X}$) in the mark reconstruction process without any further manipulation, it is obvious that if both images ($X'$ and $\hat{X}$) are the same, the set of possible locations of the embedded mark is empty, and thus the mark cannot be recovered, which is the expected ideal behavior.

In the second case, if the modified image is manipulated (i.e. attacked), then it is possible to recover a mark because there will be a large enough set of modified pixels which can be considered to be part of the mark. Nevertheless, the majority voting scheme used for mark reconstruction makes it very unlikely to recover the mark in such a situation, because most bit positions are left as asterisks (erasures), because there is no a clear majority of zeros or ones. Although the use of error correction codes may be a problem for fingerprinting purposes, as a mark is always generated, the large number of erasures makes the correction code completely useless, so an embedded mark cannot be recovered. For example, for all the 6480 images generated in all the attacks, the error correcting code is always unable to generate a valid mark due to the large percentage of erasures present in the codewords reconstructed by the majority voting scheme.

## 4.6 Summary

Table 8 shows the total number of attacks survived by the proposed watermarking scheme for each parameter configuration. As expected, the bit rate is the most important parameter determining the overall robustness of the scheme. The best results in terms of number of attacks survived are achieved for the lowest bit rates (0.75 bpp), whereas the capacity is the largest. On the other hand, results show that the combination $B = 32$ and $L = 5$ yields the best results in average, although $B = 64$ and $L \in \{5, 6\}$ can be also considered to perform similarly.

Although Table 8 summarizes the number of attacks survived over the total number of attacks performed, such information only gives an average idea of the correctness of the proposed scheme in order to allow to compare somehow the robustness with other existing watermarking algorithms. However, there is no standard or common measure that gives an idea of how good a watermarking algorithm can be in view of practical applicability. In fact, such measurement will depend on the specific application and scenario where the watermarking scheme should be used. There have been attempts [36, 37, 38] to define a watermarking benchmark focused on the application framework. The idea is to identify only those relevant attacks for this particular scenario, that means attacks that make sense for the selected scenario and attacks that produce an attacked image with still some value for those particular scenario. Nevertheless, those scenarios and those benchmark profiles are still to be clearly defined.

Notice that the JPEG2000 standard defines a default values of $B = 64$ and $L = 6$, which can be considered also useful for watermarking and fingerprinting purposes. Nevertheless, both parameters performance (for lossy compression purposes) are directly related to image size, so there is a need to develop a mathematical model of JPEG2000 image quality involving image size, bit rate, number of decomposition levels and block size. Furthermore, these parameters also directly determine capacity and, therefore, the amount of modified pixels suitable for hiding the mark and the perceptual capacity, so a more detailed analysis for discovering the relationships between these parameters and the intrinsic image characteristics is also necessary. Finally, in the light of the results obtained in [19], the new watermarking scheme using the JPEG2000 standard is always more robust or at least as robust as the basic scheme using the JPEG standard proposed in [8], mainly because capacity is increased when the same PSNR is used for the modified image for both watermarking schemes.

# 5 Conclusions

In this paper, an empirical evaluation of the lossy compression properties of the JPEG2000 standard for watermarking purposes has been described. The JPEG2000 standard is used as a basic tool for determining both how and where the embedded watermark should be placed in the image. The original image is slightly modified in order to generate a similar image (indistinguishable by the observer) and the mark is embedded in the pixels presenting differences between both images.

To increase the robustness level of the scheme, some coding is performed before the inclusion of the mark into the image. On the one hand, an error-correcting code is applied and even more redundancy is obtained by replicating the mark. On the other hand, the mark is encrypted (in order to randomize its distribution) so it cannot be obtained without the encryption key. The watermarking scheme proposed in this paper is non-blind in the sense that the original image is required during the reconstruction process. But the fact that the embedded mark is not needed in the reconstruction algorithm makes it suitable for fingerprinting techniques. In fact, the implementation used in this paper uses a dual Hamming binary code as a error correcting code. This technique allows the watermarking scheme to avoid collusion of two buyers when the algorithm is used as a fingerprinting scheme.

The experiments show that the properties of the resulting watermarking scheme depend on the transformation stage characteristics of the lossy image compression system. Several parameters related to the JPEG2000 standard are tested (appart from the compression ratio), namely the block size and the number of levels of decomposition for each block. These parameters have a large impact in the compression ratio and the image quality of the compressed image, and are supposed to determine two important measures directly related to the robustness of the watermarking scheme: the number of different pixels between the original image and the slightly modified one, and the distribution of such variations.

Regarding the impact of each one of the configuration parameters in the watermarked image, the bit rate is obviously the most important one, since it determines capacity and, therefore, system performance. On the other hand, the number of decomposition levels and block size do not seem to have a large impact on performance, but slightly improvements could be achieved by fine-tuning both parameters. On the other hand, regarding the robustness property, we have used the Stirmark benchmark for evaluation purposes. We have compared the impact on the system performance of each of the JPEG2000 standard parameters for several kinds of attacks, namely filtering (including sharpening), JPEG and JPEG2000 compression, cropping, row and column removal and a combination of rotation, cropping and a scaling attack. The proposed watermarking scheme can be considered robust in front of most of these attacks (for reasonable attacked image qualities), but it cannot withstand locally random geometric distortion attacks.

The experiments show that the different parameters tested perform similarly, yielding similar results. However, the parameter configuration using $br_C = 0.75$, $B = 32$ and $L = 5$ achieves the best results in average. More precisely, taking the watermarking scheme presented in Section 3 with the JPEG2000 using the real wavelet transform with 5 level decomposition, with blocks of size 32 and with a bit rate of 0.75, we obtain a watermarking algorithm that survives the following attacks (for example, for the *Lenna* image, with a PSNR of 38.9 dB): median filter ($2 \times 2$, $3 \times 3$ and $4 \times 4$); frequency mode Laplacian removal; Gaussian filter; sharpening attacks; JPEG standard lossy compression with quality factors 20, 25, 30, 35, 40, 50, 60, 70, 80 and 90 (that is, all the tested quality factors but the lowest ones, 10 and 15, which produce images which cannot be considered useful); JPEG2000 lossy compression at 2.0, 1.0, 0.75 and 0.5 bpp; centered cropping with a factor of 1%, 2%, 5%, 10%, 15%, 20% and 25%; rows and columns removal (1,1), (1,5), and (5,1) and rotation followed by a cropping and a scaling only at $\pm 0.25$ degrees. Furthermore, with such configuration all the attacks listed above but the $4 \times 4$ median filter (which produces a poor quality image, about 23.8 dB) are survived recovering every bit of the embedded mark. This property makes the watermarking scheme suitable for fingerprinting once the mark is coded using an appropriate method, since the mark embedded is not needed in the reconstruction process and the complete mark can be recovered after the attack.

The experiments performed in this paper show that the most efficient attacks are those which destroy the implicit mark sequence defined by the embedding mark process. Even though the use of synchronization mechanisms might be useful to face this kind of attacks, the proposed watermarking scheme could be improved by using different strategies for the mark embedding process, namely by breaking the sequentiality of the embedded mark using Peano or Hilbert curves, and including a randomization factor based on a secret key, which would make sequence manipulations not possible, thus increasing the overall performance in a robustness sense of the watermarking scheme.

Current and future research in this subject include a fine-tuning analysis of the intrinsic parameters of the proposed watermarked scheme, that is, the threshold $\tau$ used for selecting those pixels where to embed the mark, and $\delta_i$, the amount added or subtracted for embedding a bit in a pixel. As mentioned above, the use of space-filling curves for modifying the mark embedding sequence is also an interesting issue, due to the possibility of increasing the overall robustness by means of the inclusion of a randomization scheme of such curve based on a secret key, for example.

## Acknowledgements

This work is partially supported by the Spanish MCYT and the FEDER funds under grant no. TIC2003-08604-C04-04 MULTIMARK and by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

## Disclaimer

The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

## References

[1] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn. Attacks on copyright marking systems. In *2nd Workshop on Information Hiding*, LNCS 1525, pages 219–239. Springer-Verlag, 1998.

[2] F.A.P. Petitcolas and R.J. Anderson. Evaluation of copyright marking systems. In *Proceedings of IEEE Multimedia Systems'99*, pages 574–579, 1999.

[3] A.E. Bell. The dynamic digital disk. *IEEE Spectrum*, 36(10):28–35, October 1999.

[4] B. Chor, A. Fiat, and M. Naor. Tracing traitors. In *Advances in Cryptology- CRYPTO'94*, LNCS 839, pages 257–270. Springer-Verlag, 1994.

[5] E. Koch and J. Zhao. Towards robust and hidden image copyright labeling. In *Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing*, pages 452–455, Halkidiki, Greece, June 1995.

[6] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal G. Shamoon. Secure spread spectrum watermarking for multimedia. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '97*, volume 6, pages 1673 – 1687, Santa Barbara, California, USA, October 1997.

[7] H.J. Lee, J.H. Park, and Y. Zheng. Digital watermarking robust against JPEG compression. In M. Mambo and Y. Zheng, editors, *Second International Information Security Workshop, ISW'99*, LNCS 1729, pages 167–177. Springer-Verlag, November 1999.

[8] J. Domingo-Ferrer and J. Herrera-Joancomartí. Simple collusion-secure fingerprinting schemes for images. In *Proceedings of the Information Technology: Coding and Computing ITCC'2000*, pages 128–132. IEEE Computer Society, 2000.

[9] Liehua Xie and Gonzalo R. Arce. Joint wavelet compression and authentication watermarking. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '98*, volume 2, pages 427–431, Chicago, IL, USA, 1998.

[10] Rakesh Dugad, Krishna Ratakonda, and Narendra Ahuja. A new wavelet-based scheme for watermarking images. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '98*, volume 2, pages 419–423, Chicago, IL, USA, October 1998.

[11] Shelby Pereira, Sviatoslav Voloshynovskiy, and Thierry Pun. Optimized wavelet domain watermark embedding strategy using linear programming. In Harold H. Szu and Martin Vetterli, editors, *Wavelet Applications VII (part of SPIE AeroSense 2000)*, Orlando, Florida USA, April 26–28 2000.

[12] David Taubman and Michael Marcellin. *JPEG2000: Image compression fundamentals, standards and practice*. Kluwer Academic Publishers, 2002.

[13] G.K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, April 1991.

[14] Thomas G. Lane et al. Independent JPEG Group library v6.1a, 1997. `http://www.ijg.org`.

[15] Image Power Inc. Jasper JPEG-2000 transcoder v1.701.0, 2003. `http://www.imagepower.com`.

[16] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[17] R. Venkatesan, S.M. Koon, M.H. Jakubowski, and P. Moulin. Robust image hashing. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 664–666, Vancouver, BC, Canada, 2000.

[18] J. Domingo-Ferrer and J. Herrera-Joancomartí. Short collusion-secure fingerprinting based on dual binary hamming codes. *Electronics Letters*, 36(20):1697–1699, September 2000.

[19] J. Herrera-Joancomartí, J. Minguillón, and D. Megías. A family of image watermarking schemes based on lossy image compression. In *Proceedings of the Information Technology: Coding and Computing ITCC'2003*, pages 559–563. IEEE Computer Society, 2003.

[20] M. Kutter, F. Jordan, and F. Bossen. Digital signatures of color images using amplitude modulation. In *Proceedings of the SPIE Electronic Imaging 97*, pages 518–526, San Jose, California, USA, 1997.

[21] A. Piva, M. Barni, F. Bartolini, and V. Cappellini. Exploiting the cross-correlation of RGB channels for robust watermarking of color images. In *Proceedings of the IEEE-ICIP 99*, pages 306–310, Kobe, Japan, 1999.

[22] D. Fleet and D. Heeger. Embedding invisible information in color images. In *Proceedings of the IEEE-ICIP 97*, pages 532–535, Santa Barbara, California, USA, 1997.

[23] P. Bass, N. Le Bihan, and J.-M. Chassery. Color image watermarking using quaternion Fourier transform. In *Proceedings of the ICASSP 2003*, Hong Kong, China, 2003.

[24] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. In *Advances in Cryptology-CRYPTO'95*, LNCS 963, pages 452–465. Springer-Verlag, 1995.

[25] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *IEEE Transactions on Information Theory*, 44(5):1897–1905, 1998.

[26] Joshua R. Smith and Barrett O. Comiskey. Modulation and information hiding in images. In *Workshop on Information Hiding*, pages 207–226, Isaac Newton Institute, University of Cambridge, UK, May 1996. Springer-Verlag. Lecture Notes in Computer Science Volume 1174.

[27] I.J. Cox, M. T. Miller, and J.A. Bloom. Watermarking applications and their properties. In *Proceedings of the Information Technology: Coding and Computing ITCC'2000*, pages 6–10. IEEE Computer Society, 2000.

[28] Julià Minguillón, J. Herrera-Joancomartí, and David Megías. Empirical evaluation of a JPEG2000 standard-based robust watermarking scheme. In *Proceedings of the IS&T/SPIE's 15th annual symposium on Electronic Imaging*, volume 5020, pages 717–727, Santa Clara, CA, January 2003.

[29] William B. Pennebaker and Joan L. Mitchell. *JPEG still image data compression standard.* Van Nostrand Reinhold, 1993.

[30] I. Setyawana, D. Delannayb, B. Macqb, and R. L. Lagendijka. Perceptual quality evaluation of geometrically distorted images using relevant geometric transformation modeling. In *Proceedings of the IS&T/SPIE's 15th Annual Symposium on Electronic Imaging*, volume 5020, Santa Clara, CA, US, January 2003.

[31] Andrew B. Watson, editor. *Digital Images and Human Vision.* The MIT Press, 1993.

[32] Sviatoslav Voloshynovskiy, Shelby Pereira, Victor Iquise, and Thierry Pun. Attack modelling: Towards a second generation benchmark. *Signal Processing*, 81(6):1177–1214, June 2001. Special Issue: Information Theoretic Issues in Digital Watermarking, 2001. V. Cappellini, M. Barni, F. Bartolini, Eds.

[33] Julià Minguillón and Jaume Pujol. JPEG standard uniform quantization error modeling with applications to sequential and progressive operation modes. *Journal of Electronic Imaging*, 10(2):475–485, apr 2001.

[34] Fabien A. P. Petitcolas. Watermarking schemes evaluation. In *IEEE Signal Processing*, volume 17, pages 58–64. IEEE, September 2000.

[35] R. Barnett and D. E. Pearson. Frequency mode L.R. attack operator for digitally watermarked images. *Electronics Letters*, 34(19):1837–1839, September 1998.

[36] IST project CERTIMARK (IST-1999-10987). Watermarking applications and requirements for benchmarking. Deliverable 2. l, 2004.

[37] Andreas Lang and Jana Dittmann. Stirmark and profiles: from high end up to preview scenarios. In *Proceedings of International Workshop for Technology, Economy, Social and Legal Aspects of Virtual Goods*, Ilmenau, Germany, 2004.

[38] Andreas Lang and Jana Dittmann. Application-oriented audio watermark benchmark service. In *Proceedings of the IS&T/SPIE's 17th annual symposium on Electronic Imaging*, volume 5681, San Jose, CA, January 2005.

# Tables

| $L$ / $B$ | 8 | 16 | 32 | 64 |
|---|---|---|---|---|
| 3 | 16 | 14 | 11 | 10 |
| 4 | 15 | 9 | 6 | 5 |
| 5 | 13 | 8 | 3 | 1 |
| 6 | 12 | 7 | 4 | 2 |

Table 1: Relative importance of each JPEG2000 standard configuration parameter with respect to the measured PSNR, ranked according to the parameter importance index.

| Attack | $2 \times 2$, $3 \times 3$ Median Filter, Gaussian Filter, Sharpening | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| Attack | $4 \times 4$ Median Filter | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 9 | 10 / 9 | 9 / 9 | 10 / 7 | 10 / 7 | 10 / 8 |
| $L = 6$ | 10 / 10 | 10 / 7 | 9 / 8 | 10 / 9 | 10 / 7 | 10 / 7 |
| Attack | Frequency Mode Laplacian Removal Filter | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 7 / 5 | 7 / 5 | 5 / 4 | 5 / 4 | 4 / 3 | 4 / 3 |
| $L = 6$ | 6 / 5 | 7 / 5 | 5 / 4 | 5 / 4 | 3 / 3 | 3 / 3 |

Table 2: Number of correctly recovered marks (watermarking / fingerprinting) for the filtering attack. The maximum number is ten, meaning that the mark could be recovered for all images in the corpus set.

| Attack | $Q = 80$, $Q = 90$ | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| Attack | $Q = 50$ | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 9 | 10 / 9 | 9 / 9 | 10 / 9 | 10 / 8 | 10 / 8 |
| $L = 6$ | 10 / 9 | 10 / 10 | 10 / 9 | 10 / 9 | 10 / 8 | 9 / 9 |
| Attack | $Q = 25$ | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 8 / 8 | 8 / 8 | 8 / 6 | 8 / 5 | 6 / 2 | 6 / 2 |
| $L = 6$ | 9 / 7 | 9 / 7 | 8 / 5 | 8 / 7 | 1 / 0 | 5 / 3 |

Table 3: Number of correctly recovered marks (watermarking / fingerprinting) for the JPEG attack (out of 10).

| Attack | $br_A = 2.0$, $br_A = 1.0$, $br_A = 0.75$ | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| Attack | $br_A = 0.5$ | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 9 | 10 / 10 | 10 / 9 | 10 / 5 | 9 / 5 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 9 | 10 / 10 | 10 / 5 | 10 / 5 |
| Attack | $br_A = 0.25$ | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 1 / 1 | 3 / 0 | 1 / 0 | 1 / 0 | 0 / 0 | 0 / 0 |
| $L = 6$ | 2 / 0 | 3 / 0 | 1 / 0 | 2 / 1 | 0 / 0 | 0 / 0 |

Table 4: Number of correctly recovered marks (watermarking / fingerprinting) for the JPEG2000 attack (out of 10).

| Attack | 1%, 2%, 5% Cropping | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| Attack | 25%, 50% Cropping | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 9 / 9 | 9 / 9 | 9 / 9 | 9 / 9 | 10 / 10 | 10 / 10 |
| $L = 6$ | 9 / 9 | 9 / 9 | 9 / 9 | 9 / 9 | 10 / 10 | 10 / 10 |
| Attack | 75% Cropping | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 8 / 8 | 8 / 8 | 9 / 9 | 9 / 9 | 10 / 10 | 10 / 10 |
| $L = 6$ | 8 / 8 | 8 / 8 | 9 / 9 | 9 / 9 | 10 / 10 | 10 / 10 |

Table 5: Number of correctly recovered marks (watermarking / fingerprinting) for the cropping attack (out of 10).

| Attack | (1,1), (5,1), (1,5) Removal | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| $L = 6$ | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 | 10 / 10 |
| Attack | (17,5), (5, 17) Removal | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 |
| $L = 6$ | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 |

Table 6: Number of correctly recovered marks (watermarking / fingerprinting) for the removal attack (out of 10).

| Attack | $-0.25$ R-C-S | | | | | |
|---|---|---|---|---|---|---|
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 9 / 7 | 9 / 9 | 10 / 8 | 10 / 9 | 10 / 9 | 9 / 4 |
| $L = 6$ | 9 / 8 | 9 / 7 | 10 / 9 | 9 / 9 | 10 / 4 | 10 / 7 |
| Attack | $+0.25$ R-C-S | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 9 / 8 | 8 / 8 | 10 / 8 | 10 / 8 | 10 / 9 | 10 / 7 |
| $L = 6$ | 9 / 8 | 10 / 9 | 10 / 9 | 10 / 9 | 10 / 7 | 9 / 4 |
| Attack | $\pm0.5$, $\pm0.75$, $\pm1$, $\pm2$, 5, 10, 15, 30, 45, 90 R-C-S | | | | | |
| | $br_C = 0.75$ | | $br_C = 1.0$ | | $br_C = 2.0$ | |
| | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 |
| $L = 6$ | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 | 0 / 0 |

Table 7: Number of correctly recovered marks (watermarking / fingerprinting) for the R-C-S attack (out of 10).

|  | $br_C = 0.75$ |  | $br_C = 1.0$ |  | $br_C = 2.0$ |  |
|---|---|---|---|---|---|---|
|  | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ | $B = 32$ | $B = 64$ |
| $L = 5$ | 345 / 329 | 344 / 327 | 337 / 319 | 339 / 317 | 327 / 295 | 322 / 289 |
| $L = 6$ | 342 / 324 | 345 / 326 | 341 / 320 | 339 / 326 | 320 / 291 | 322 / 292 |

Table 8: Number of correctly recovered marks (watermarking / fingerprinting) for each parameter configuration (out of 550 total experiments).

# Figures

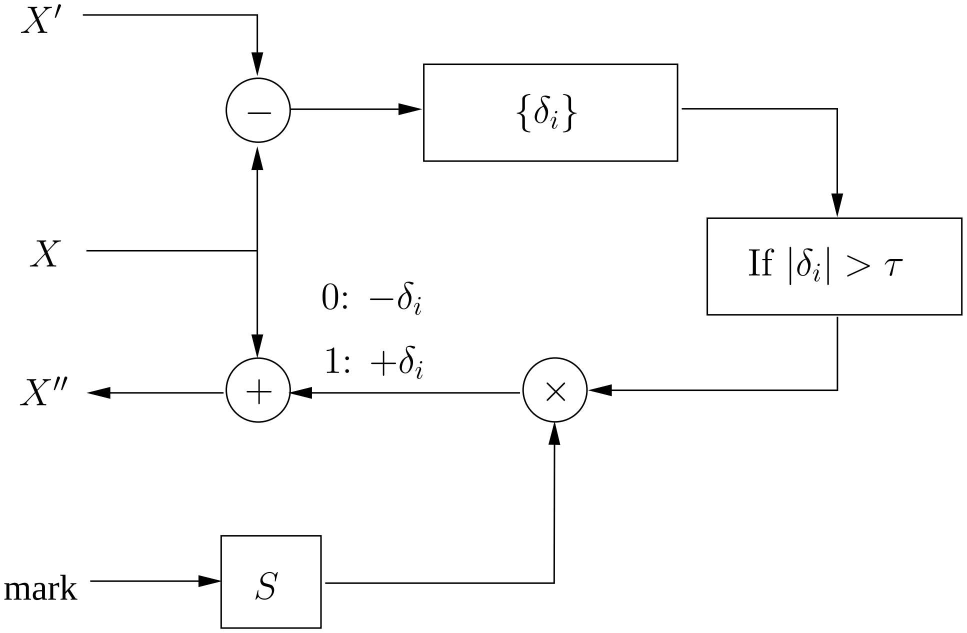


Figure 1:

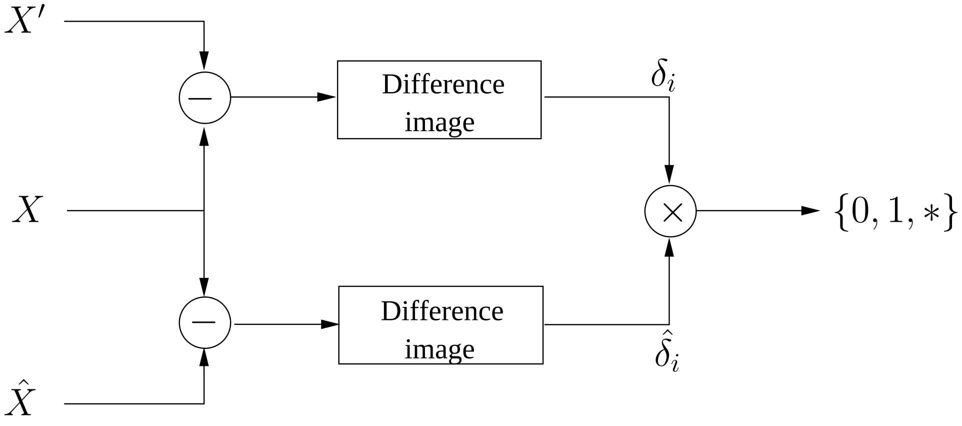


Figure 2:

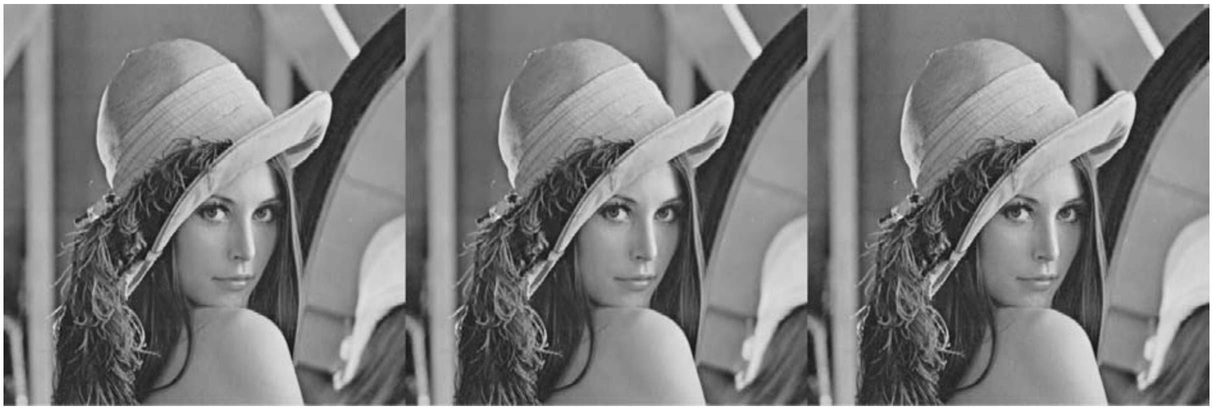


Figure 3:

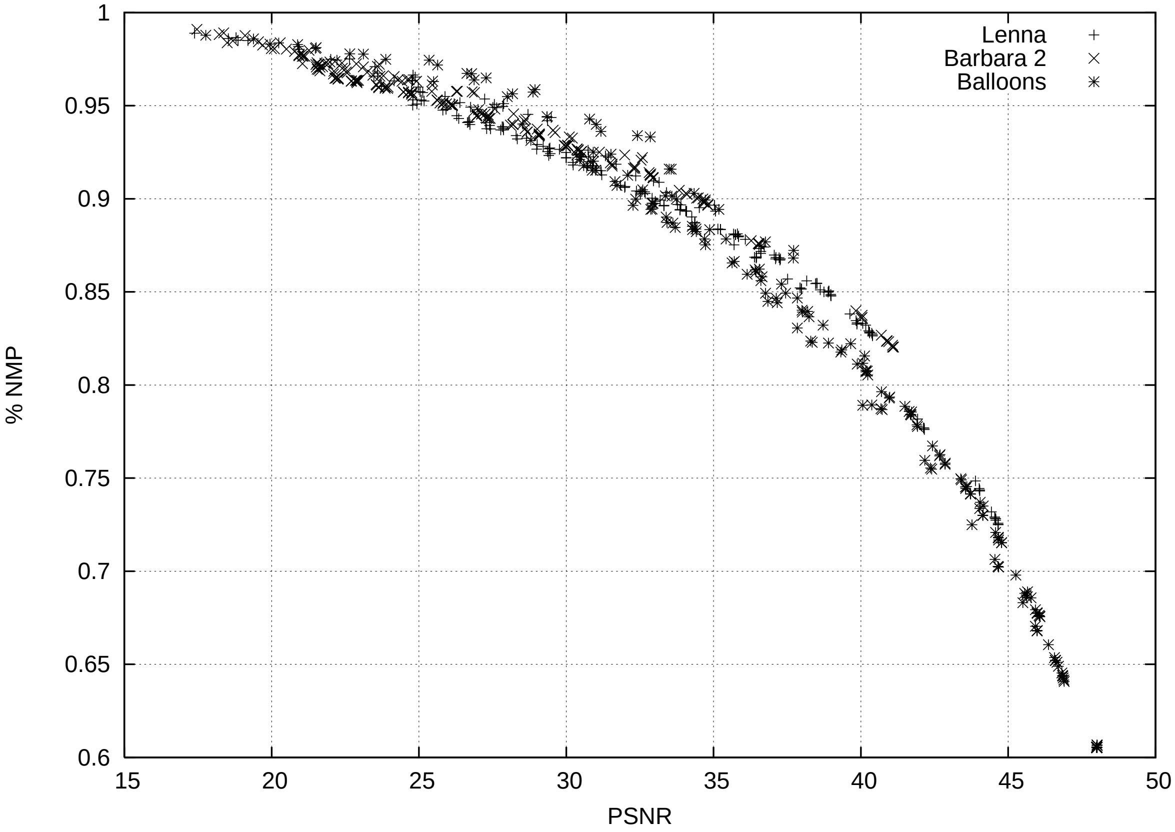
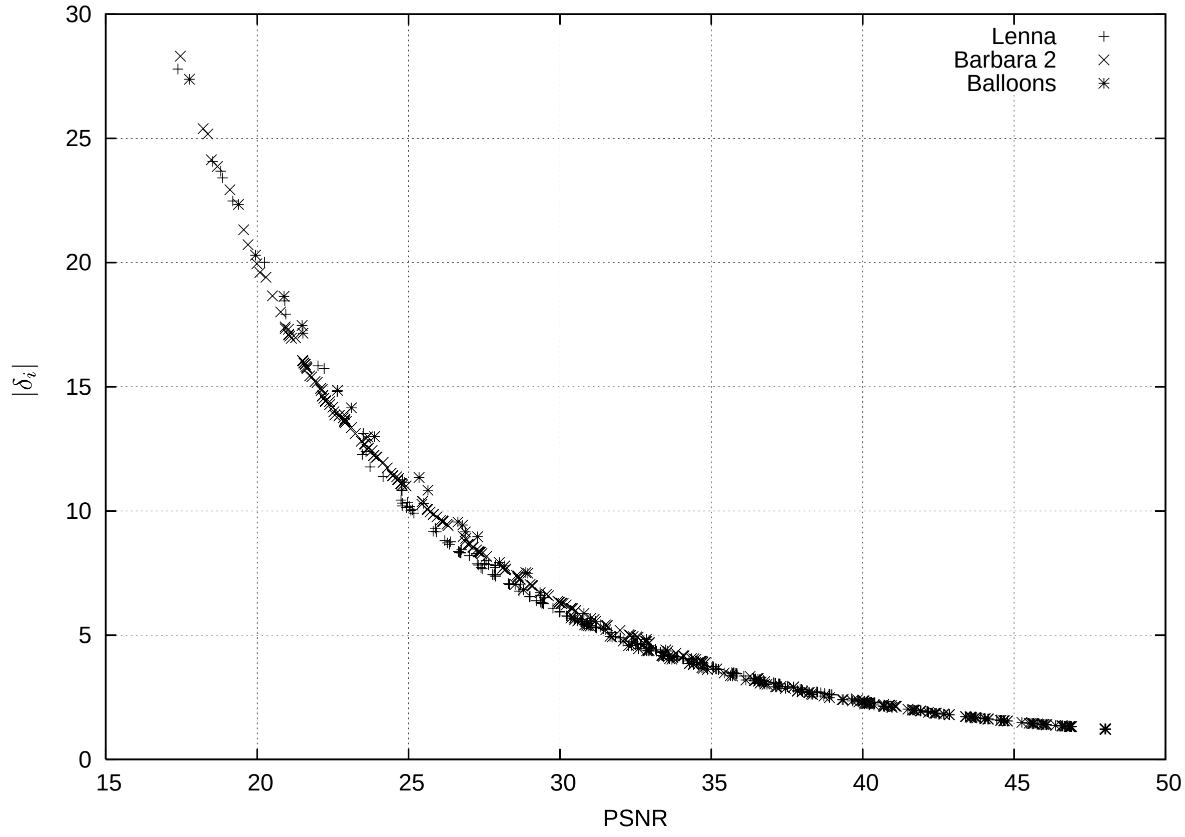

1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
% NMP
15
20
25
30
35
40
45
50
PSNR
Lenna +
Barbara 2 ×
Balloons ✱

Figure 4:



30
25
20
15
10
5
0
15
20
25
30
35
40
45
50
$|\delta_i|$
PSNR
Lenna +
Barbara 2 ×
Balloons ✳

Figure 5: