ELSEVIER

Signal Processing 66 (1998) 385–403

SIGNAL PROCESSING

Robust image watermarking in the spatial domain

N. Nikolaidis[a], I. Pitas[b,*]

[a] *Department of Electrical and Computer Engineering, University of Thessaloniki, Thessaloniki 540 06, Greece*
[b] *Department of Informatics, University of Thessaloniki, Thessaloniki 540 06, Greece*

Received 22 February 1997; received in revised form 23 September 1997

Abstract

The rapid evolution of digital image manipulation and transmission techniques has created a pressing need for the protection of the intellectual property rights on images. A copyright protection method that is based on hiding an 'invisible' signal, known as *digital watermark*, in the image is presented in this paper. Watermark casting is performed in the spatial domain by slightly modifying the intensity of randomly selected image pixels. Watermark detection does not require the existence of the original image and is carried out by comparing the mean intensity value of the marked pixels against that of the pixels not marked. Statistical hypothesis testing is used for this purpose. Pixel modifications can be done in such a way that the watermark is resistant to JPEG compression and lowpass filtering. This is achieved by minimizing the energy content of the watermark signal at higher frequencies while taking into account properties of the human visual system. A variation that generates image dependent watermarks as well as a method to handle geometrical distortions are presented. An extension to color images is also pursued. Experiments on real images verify the effectiveness of the proposed techniques.

Zusammenfassung

Die schnelle Entwicklung digitaler Bildmanipulationen und -über- tragungsverfahren hat eine drängende Notwendigkeit für den Schutz intellektueller Eigentumsrechte von Bildern erzeugt. Ein Schutz des Copyrights, der auf dem Verstecken eines 'unsichtbaren' Signals beruht, ist als digitales Wasserzeichen innerhalb des Bildes bekannt und wird in dieser Arbeit vorgestellt. Der Einschluß eines Wasserzeichens wird im räumlichen Bereich durch eine geringe Modifikation der Intensität zufällig ausgewählter Bildpunkte erreicht. Die Erkennung des Wasserzeichens erfordert nicht die Vorlage des Originalbildes und wird durch Vergleich der mittleren Intensität der markierten Bildpunkte mit derjenigen der nicht markierten Punkte erreicht. Zu diesem Zweck wird ein statistischer Hypothesentest benutzt. Die Punktmodifikation kann in einer solchen Weise durchgeführt werden, daß das Wasserzeichen gegenüber einer JPEG-Kompression und Tiefpaßfilterung resistent ist. Durch die Minimierung des Energieinhaltes des Wasserzeichensignals bei höheren Frequenzen wird dies erreicht, wobei die Eigenschaften des menschlichen Gesichtssinnes berücksichtigt werden. Es werden eine Variation, die bildabhängige Wasserzeichen erzeugt, sowie eine Methode präsentiert, die geometrische Verzerrungen behandelt. Ein Erweiterung auf Farbbilder wird auch verfolgt. Experimente mit echten Bildern bestätigen die Effizienz der vorgeschlagenen Methode.

* Corresponding author. E-mail: pitas@zeus.csd.auth.gr.

0165-1684/98/$19.00
PII S0165-1684(98)00017-6

Résumé

L'évolution rapide de la manipulation des images numériques et des techniques de transmission a généré un besoin pressant de protection des droits de la propriété intellectuelle sur les images. Une méthode de protection des droits d'auteur, basée sur le camouflage d'un signal 'invisible' dans l'image, connusous le nom de filigrane numérique (digital watermark) est présentée dans cet article. Le placement du watermark est opéré dans le domaine spatial par modification légère de l'intensité de pixels de l'image choisis aléatoirement. La détection du watermark ne requiert pas l'image originale et elle s'opère par comparaison entre l'intensité moyenne des pixels marqués et celle des pixels non marqués. Un test d'hypothèse statistique est utilisé à cet effet. Les modifications des pixels peuvent être faites de telle façon que le watermark soit résistant vis-à-vis de la compression JPEG et du filtrage passe-bas. Ceci est obtenu en minimisant l'énergie du signal de watermark dans les hautes fréquences tout en tenant compte des propriétés du système visuel humain. Une variante générant des watermarks dépendant de l'image ainsi qu'une méthode de prise en compte des distorsions géométriques sont également présentées. Une extension aux images couleur est également en cours de développement. Les expériences réalisées sur des images réelles permettent de vérifier l'efficience des techniques proposées.

Keywords: Copyright protection; Watermarking; Steganography

1. Introduction

Digital media have revolutionized the way still images and image sequences are stored, manipulated and transmitted, giving rise to a wide range of new applications (digital television, digital video disc, digital image databases, electronic publishing, etc.) that are expected to have an important impact on the electronics and entertainment industry. One of the main features of digital technology is the ease with which images can be accessed and duplicated. However, this feature has an important side effect; it allows for easy unauthorized reproduction of information, i.e. data piracy. Due to this, protection of intellectual property rights, i.e. copyright protection of stored/transmitted digital images is a very important issue. One way to help protect images against illegal recordings and retransmissions is to embed an invisible signal, called *digital signature* or *copyright label* or *watermark*, that completely characterizes the person who applied it and, therefore, marks it as being his intellectual property. Obviously, the secure and unambiguous identification of the legal owner of an image requires that each individual or organization that produces, owns or transmits digital images (artists, broadcasting corporations, image database providers, etc.) uses a different, unique watermark.

Copyright protection is just one of the potential applications of embedding invisible data within images or other types of signal (e.g. audio signals), a technique usually referred to as data hiding or steganography. Other applications include authentication control, tamper-proofing (i.e. checking whether the content of an image has been altered or not) and insertion of invisible image annotations (e.g. scene/object description). For each of these applications, the embedded signal should possess a different set of properties. In this paper we would limit our discussion to digital watermarks serving the purpose of copyright protection. Digital watermarks of this type should be [14]:

- undeletable by an 'attacker';
- easily and securely detectable by their owner;
- perceptually and statistically invisible;
- resistant to lossy compression, filtering and other types of processing.

The creation of an algorithm capable of producing watermarks that fulfil all these contradicting requirements is not an easy task. A number of attempts to introduce copyright labelling techniques that comply with some or all of the above specifications have been reported lately in the literature [1,3,4,6,7,9,10,12,13,15–18,21–23,25,26,29,30]. However, research on copyright protection of images is still in its early stages and none of the existing methods is totally effective against attacks. The techniques proposed so far can be classified in two broad categories: (i) methods that embed the watermark by directly modifying the intensity of

certain pixels [1,4,22,25,15,26]. (ii) methods that act upon selected coefficients of a properly chosen transform domain (DCT domain, DFT domain, etc.) [3,7,9,13,17,18,23,30]. Watermarking techniques can be alternatively split into two distinct classes depending on whether the original image is necessary for the watermark detection or not. Although the existence of the original image facilitates to a great extent watermark detection, such a requirement is rather difficult to be met in most real life applications. It would be, e.g., totally impractical for the owner of a large image database to keep double copies of its images for authentication and copyright protection purposes. Furthermore, searching within the database for the original image that corresponds to a given watermarked image would be very time consuming. In this paper we propose a watermarking technique that belongs to the class of intensity domain techniques, i.e. it embeds copyright information by modifying the intensity of a subset of the image pixels. The proposed method is actually an extension and continuation of the method reported by the authors in [16,21,22]. The watermark casting algorithm allows for a flexible choice of the intensity modifications. This flexibility can be exploited to design watermarks that possess desirable properties like robustness against lossy compression and lowpass filtering. Watermark detection is carried out by using hypothesis testing and does not require the original image. Another important feature of the algorithm is its mathematical tractability that allows a thorough investigation of algorithm performance. Furthermore, the proposed watermarking technique can be easily combined with noise masking techniques to yield watermarks that are invisible.

The basic operating principle of the proposed algorithm presents certain similarities to the so-called Patchwork technique that has been independently developed in MIT Media Lab [1]. However, the two methods differ in the statistical approach adopted for the watermark detection. Furthermore, an extensive part of our paper is devoted to important extensions of the basic method (image dependent watermarks, robust watermark design by means of optimization techniques, handling of geometric distortions) that are not addressed in [1]. The proposed algorithm bears also certain similarities with other methods [6,9,17] developed independently at about the same time or afterwards. However, the differences between our method and the above-mentioned techniques are rather important. For example, the algorithms proposed by Cox and Ruanaidh require the original image during the watermark detection whereas our method does not. The outline of this paper is the following. The basic algorithm is described in Section 2. Design of watermarks that are robust to filtering and compression is discussed in Section 3. Section 4 deals with ways to incorporate properties of the human visual system in order to generate invisible watermarks. A method to handle geometrical distortions is presented in Section 5. Image dependent watermarks are proposed in Section 6 as a means of further robustifying the algorithm. Extension to color images is treated in Section 7. Experiments on real images are presented in Section 8.

2. Basic watermarking algorithm

Consider an image I of dimensions $N \times M$:

$$x_{nm}, \quad 0 \leqslant n < N, \quad 0 \leqslant m < M. \tag{1}$$

A watermark pattern S is a binary pattern of the same size where the number of 'ones' equals the number of 'zeros':

$$s_{nm}, \quad 0 \leqslant n < N, \quad 0 \leqslant m < M,$$

$$s_{nm} \in \{0,1\}. \tag{2}$$

Using S we can split I into two subsets of equal size:

$$A = \{(n,m)|s_{nm} = 1\}, \tag{3}$$

$$B = \{(n,m)|s_{nm} = 0\}, \tag{4}$$

$$|A| = |B| = \tfrac{1}{2}|I| = \tfrac{1}{2}N \times M = P, \tag{5}$$

$$I = A \cup B. \tag{6}$$

The digital watermark is superimposed on the image as follows:

$$x^s_{nm} = x_{nm} \otimes f_{nm}, \tag{7}$$

where $\otimes$ is a superposition law (in our case addition), x^s_{nm} are the pixels of the watermarked image I^s and f_{nm} is the two-dimensional watermark signal:

$$f_{nm} = \begin{cases} k, & s_{nm} = 1, \\ 0, & s_{nm} = 0, \end{cases} \tag{8}$$

k being a positive integer that will be called embedding factor. Let us denote by $\bar{a}$, $\bar{b}$, and s_a, s_b the sample mean values and the sample standard deviations of the pixels belonging to the subsets A, B of the original image and by $\bar{c}$, s_c the sample mean value and the sample standard deviation of the pixels belonging to the subset A of the watermarked image. For example:

$$\bar{a} = \frac{1}{P} \sum_{n,m \in A} x_{nm}, \tag{9}$$

$$s_a^2 = \frac{1}{P-1} \sum_{n,m \in A} (x_{nm} - \bar{a})^2. \tag{10}$$

Watermark detection is based on the examination of the difference $\bar{w}$ of the mean values $\bar{c}$, $\bar{b}$:

$$\bar{w} = \bar{c} - \bar{b}. \tag{11}$$

If the pixels of the subsets A, B are intermixed and the image has been watermarked then $\bar{w}$ is close to k whereas in the case of an image that is not watermarked or an image bearing a watermark different from the one that we are looking for, $\bar{w}$ is approximately zero. In other words, $\bar{w}$ is a random variable whose mean is zero for an image that bears no watermark and k for an image that has been watermarked. The decision on whether the image is watermarked or not is taken using hypothesis testing. The test statistic q that has been used is based on the central limit theorem and is given by [20]

$$q = \frac{\bar{w}}{\hat{\sigma}_{\bar{w}}}, \tag{12}$$

where $\hat{\sigma}^2_{\bar{w}}$ is an estimator of the variance of $\bar{w}$:

$$\hat{\sigma}^2_{\bar{w}} = \frac{s_c^2 + s_b^2}{P}. \tag{13}$$

The Null and the Alternative Hypotheses in this case are

H_0: There is *no* watermark in the image.

H_1: There *is* a watermark in the image.

Since the number of samples in the test (i.e. the number of pixels P) is sufficiently large (e.g. for an image as small as 16×16 pixels, the value of P is 128) the test statistic q follows under the null hypothesis a zero mean, unit variance normal distribution. Under the alternative hypothesis, q follows a normal distribution having unit variance and mean equal to $k/\sigma_{\bar{w}}$. Therefore, the exact calculation of the mean value for q in this case requires knowledge of $\sigma_{\bar{w}}$. However, for a large number of samples, as in our case the estimate $\hat{\sigma}_{\bar{w}}$ gives a very good estimate of $\sigma_{\bar{w}}$.

In order to decide whether the image is watermarked or not, the value of q is tested against a threshold T. If $q > T$ we assume that the image is watermarked, otherwise we conclude that the image bears no watermark. The possible detection errors are the following:

Type I Error: Accept the existence of a watermark, although there is none.

Type II Error: Reject the existence of a watermark, although there is one.

In hypothesis testing the probability of type I error is denoted by α whereas the probability of type II error is denoted by β.

The threshold T that results in equal probabilities for errors of type I and type II is given by

$$T = \frac{k}{2\hat{\sigma}_{\bar{w}}}. \tag{14}$$

In this case the Type I Error is the shaded region of Fig. 1a and the Type II Error, appears in Fig. 1b. The requirement that errors of type I and II should have equal probabilities of occurrence ($\alpha = \beta$) is justified by the fact that both these errors are of the same importance when watermarks are used for copyright protection. Alternatively one can choose

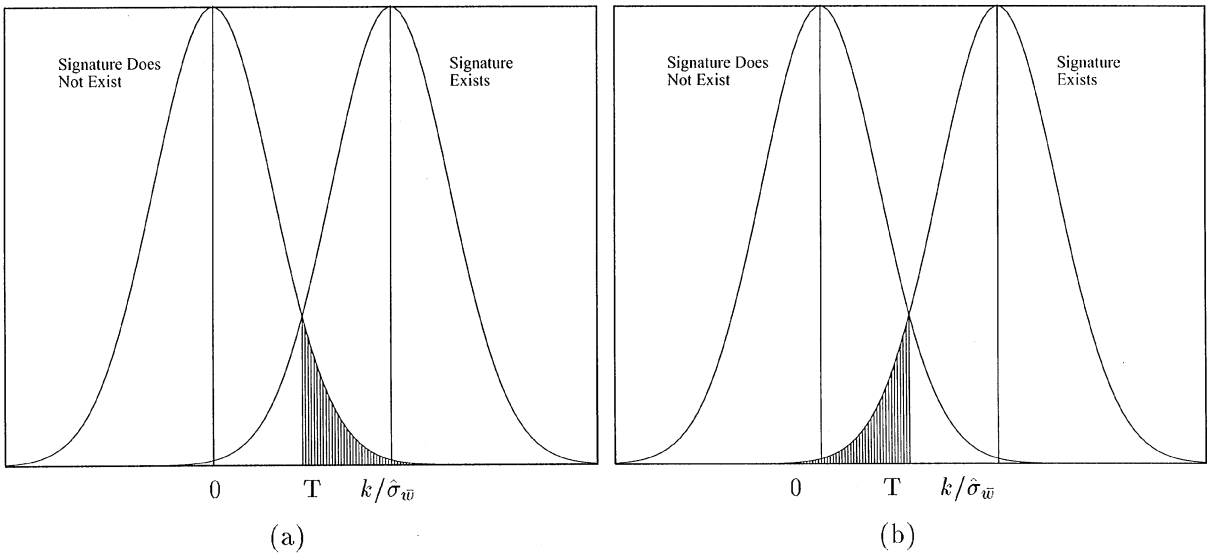

Fig. 1. Type I and Type II detection errors.

not equal probabilities for those two errors simply by selecting a threshold T that is not given by Eq. (14). Of course, if one chooses to obtain a small probability of type I error this will automatically lead to an increase of the probability of the type II error and vice versa. For a certain k and a threshold T given by Eq. (14) the probability $(1-\beta)$ of correct watermark detection in a watermarked image (certainty) would be equal to $N(T,0,1)$ where $N(x,\mu,\sigma^2)$ denotes the value at point x of the cumulative density function (cdf) of a normally distributed random variable with mean value μ and variance σ^2. Inversely, in order to generate a watermark that would be detectable with probability $1-\beta$ the threshold T should be chosen so that

$$1-\beta = N(T,0,1). \tag{15}$$

Therefore, T should be equal to the $z_{1-\beta}$ percentile of the normal distribution. This implies that the embedding factor k that should be used in this case is given by the equation:

$$k = \lceil 2\hat{\sigma}_{\bar{w}} z_{1-\beta} \rceil. \tag{16}$$

The ceiling operator in the previous equation is used because k is an integer constant (intensity value). Due to this ceiling operator, the actual probability of correct detection is bigger than $1-\beta$. In conclusion, the proposed algorithm can be summarized as follows:

Watermark casting: Generate S and calculate $\hat{\sigma}_{\bar{w}}$. Decide for the desired probability of correct detection $1-\beta$, calculate the appropriate embedding factor k using Eq. (16) and cast the watermark using Eq. (7).

Watermark detection: Generate S. Evaluate q using Eqs. (11) to (13) and compare it against $T = z_{1-\beta}$ to decide for the watermark existence.

A watermark pattern S where A, B are sufficiently intermixed can be generated by assigning to each pixel the rounded output of a random number generator that produces samples in the range $[0,\ldots,1]$. The seed that is used to initialize the random generator completely characterizes the watermark pattern S, i.e., it suffices to know this seed to recreate the watermark pattern. This number will be called in the sequel watermark key. In order to be effective for copyright protection, the watermark key should be known only to the watermark owner.

At this point we should mention that Eq. (16) gives actually a lower limit for k in the sense that both the probability $1-\beta$ of correct detection in a watermarked image and the probability α of erroneous detection in an image that bears no

watermark are achieved only if the image has not been distorted. For a distorted image, both probabilities differ from the values that were used during the watermark casting phase. In such cases bigger embedding factors are required. Such an increase is essentially an increase on the energy of the watermark signal. However, this presents no difficulties because for relatively big images Eq. (16) gives a very small bound since, as the image size increases, the variance $\hat{\sigma}^2_{\bar{w}}$ of Eq. (13) of the estimator $\bar{w}$ decreases and, therefore, the embedding factor k that is required for a specific certainty becomes smaller. The final decision on the embedding factor value should be taken after experimentally checking the watermark against possible distortions. This can be done by distorting K watermarked copies of a certain image and evaluating the percentage of correctly detected watermarks (probability of correct detection). Before increasing the embedding factor we should also take into consideration that a watermark severely distorting the corresponding image is of no practical use.

The watermarking algorithm described above can generate a great number of different watermarks, all distinguishable from each other, even for small size images. For an image of size $N \times M = 2P$ the number of all possible watermarks that can be generated equals the number of ways we can select P items out of $2P$ items. As a result, the number of possible watermarks N_s is given by:

$$N_s = \binom{2P}{P} = \frac{(2P)!}{(P!)^2} \simeq \frac{2^{2P}}{\sqrt{\pi P}} \tag{17}$$

by using the Stirling formula. For example, an image of size 32×32 ($P = 512$) can host as many as 4.48×10^{306} different watermarks. Of course one might argue that two very similar watermark patterns can not be distinguishable under the previously mentioned detection algorithm, due to domain overlapping. It can be proven that the increase in the probability of error a of type I due to the existence of a watermark different than the one that we are trying to detect is limited by the following expression:

$$\alpha' - \alpha < N(z_{1-\beta}, 0, 1) - N\left(\frac{z_{1-\beta}}{\sqrt{1 + \frac{2z^2_{1-\beta}}{P}}}, 0, 1\right), \tag{18}$$

where the error probabilities α and α' denote the probability of declaring that a watermark is present when actually no watermark is present and the probability of declaring that the same watermark is present when a different watermark has been applied. The proof can be found in Appendix A.

This increase is extremely small for typical values of image size and desired probability of correct detection $1 - \beta$. For example, for an image of dimensions 128×128 ($P = 8192$) and for probability of correct detection $1 - \beta$ equal to 98% ($z_{1-\beta} = 2.0538$) the increase is smaller than 0.0051%. For more typical values i.e. $P = 32768$ (image size: 256×256) and $1 - \beta = 99.99\%$ ($z_{1-\beta} = 4$) the increase is smaller than 0.000026%.

3. Immunity to subsampling

An important issue that should be examined about the proposed watermarks is their immunity to subsampling. Only the case of mean value subsampling is considered here. In this case, if the original image I was of size $N \times M$, the subsampled image I_{sub} is $(N/2) \times (M/2)$ pixels large, and the intensity levels x'_{nm} of its pixels are given by

$$x'_{nm} = \frac{1}{4}(x_{2n,2m} + x_{2n+1,2m} + x_{2n,2m+1} + x_{2n+1,2m+1}),$$
$$0 \leqslant n < N/2,\ 0 \leqslant m < M/2. \tag{19}$$

In order to apply the detection algorithm on I_{sub}, we generate a subsampled version S' of the $N \times M$ watermark pattern S using the following method:

Let $s_1, s_2, s_3, s_4 \in S$ denote the 4 neighboring pixels to be subsampled and let $u = s_1 + s_2 + s_3 + s_4$ be their sum. The sample s, which will substitute s_1, s_2, s_3, s_4 has the following form:

1. If $u = 0$ or $u = 1$ then $s = 0$;
2. If $u = 3$ or $u = 4$ then $s = 1$;
3. If $u = 2$ then $s = 0$ or $s = 1$ with equal probabilities.

It is obvious that the subsampled watermark pattern S' contains P' pixels of value 1 and P' pixels of

value 0, where

$$P' = \frac{1}{2}\left(\frac{N}{2} \times \frac{M}{2}\right). \tag{20}$$

It can be proved that when calculating the difference $\bar{w}'$ for the subsampled image we shall have:

$$\bar{c}' = \bar{a}' + \frac{1}{8}k + \frac{4}{8}\frac{3k}{4} + \frac{3}{8}\frac{k}{2} = \bar{a}' + \frac{11}{16}k, \tag{21}$$

$$\bar{b}'' = \bar{b}' + \frac{4}{8}\frac{k}{4} + \frac{3}{8}\frac{k}{2} = \bar{b}' + \frac{5}{16}k, \tag{22}$$

$$\bar{w}' = \bar{c}' - \bar{b}'' = \bar{a}' - \bar{b}' + \frac{3}{8}k. \tag{23}$$

The proof can be found in Appendix B. We use $\bar{a}'$ and $\bar{b}'$ because they are not really the original $\bar{a}$ and $\bar{b}$, since there was an intermixing due to subsampling. Obviously $\hat{\sigma}_{\bar{w}'}$ is different from $\hat{\sigma}_{\bar{w}}$ (i.e. the corresponding quantity in the original image). If we assume that $s_c^2 = s_{c'}^2$ and $s_b^2 = s_{b''}^2$ then

$$\hat{\sigma}_{\bar{w}'}^2 = \frac{s_{c'}^2 + s_{b''}^2}{P'} = 4\hat{\sigma}_{\bar{w}}^2. \tag{24}$$

The distribution of the test statistic $q' = \bar{w}'/\hat{\sigma}_{\bar{w}'}$ under the null hypothesis is $N(q',0,1)$ whereas, under the alternative hypothesis the distribution of q' is $N(q', \frac{3/8k}{2\hat{\sigma}_{\bar{w}}}, 1)$.

Therefore, if we want the probability of correct detection for the subsampled image to be $(1-\beta')$ we should use k' given by

$$k' = \left\lceil \tfrac{16}{3} \cdot 2\hat{\sigma}_{\bar{w}} z_{1-\beta'} \right\rceil. \tag{25}$$

In this case, the probability of correct detection for the original image is

$$1 - \beta = N(k'/2\hat{\sigma}_{\bar{w}'}, 0, 1). \tag{26}$$

Better approaches can be introduced by not taking under consideration, for example, the blocks that impose the highest ambiguity, i.e. those having two signed pixels. However, such an approach reduces the amount of pixels used for parameter estimation.

4. Robust watermark design

Unfortunately the method outlined in Section 2 is not robust to compression using the well-established JPEG standard which achieves efficient image compression by combining the Discrete Cosine Transform (DCT) with appropriate DCT coefficient quantization schemes. This is due to the fact that the watermark signal f_{nm} (8) is essentially low-power, white noise. As a consequence, it is heavily distorted by JPEG. Furthermore, the watermarks can be easily deleted by other operations such as mean or median filtering. In the following we propose variations of the basic watermarking algorithm that provide immunity against lowpass operations such as lossy compression and filtering by reducing the energy content of the watermark signal in high frequencies. Other approaches to the same problem (placing the watermark information in the low-middle DCT frequencies) have been proposed in a number of other papers about watermarking [6,7,17,18].

The first way to achieve robustness against compression is by using watermarks where the marked pixels, i.e. the pixels in A, form blocks of a certain size $b_1 \times b_2$, e.g. 2×2 or 2×4. In practice, such a grouping of pixels can be implemented by selecting $P/(b_1 \cdot b_2)$ randomly positioned blocks of size $b_1 \times b_2$. It is obvious that the only difference between this variation and the basic technique described in Section 2 lies in the methodology used to select the pixels that belong to subset A. For both techniques the total number of pixels in A equals P. However, the energy of the new watermark signal is concentrated in the low frequencies and thus the watermark can endure much higher JPEG compression. The bigger the block size, the more robust to compression the watermark becomes. However, grouping the pixels in blocks along with the fact that the intensities of neighboring pixels exhibit a high degree of correlation leads to correlated samples. Thus, the statistical test described in Section 2, which is based on the assumption that samples are independent leads to erroneous results. In order to overcome this problem we evaluate $\hat{\sigma}_{\bar{w}}$, $\bar{w}$ using an image of lower resolution which we construct by substituting the pixels in each $b_1 \times b_2$ block with a pixel whose intensity is the mean intensity of all the pixels in the block. If the dimensions of the initial image are $N \times M$, the dimensions of the new image would be $(N/b_1) \times (M/b_2)$ and $P' = P/(b_1 \times b_2)$. Therefore, $\hat{\sigma}_{\bar{w}}$ given by Eq. (13)

would be bigger than that of the original image. This means that the watermarks that are designed using this variation require a bigger value of k in order to have the same certainty with the watermarks produced by the basic technique.

The second method for designing robust watermarks exploits the fact that the decision for the existence of the watermark is based on the examination of the difference $\bar{w}$ of the mean values $\bar{c},\bar{b}$. Therefore, the detection algorithm will give the same results if, instead of modifying the intensity of all pixels in A by the same constant k, we use a different value k_{nm} for each pixel in A, provided that the sum of all k_{nm} is equal to Pk:

$$\sum_{n,m\in A}\sum k_{nm} = Pk. \tag{27}$$

This versatility in the choice of k_{nm} enables us to design watermarks that are of low energy content in the higher DCT frequencies. The approach used to design such a signal is the following. First, the watermark pattern S is generated and an appropriate embedding factor k is chosen. The watermark signal f_{nm} is then divided into blocks of size 8×8 in the same way as in the DCT algorithm and the optimum values k_{nm} are separately calculated for each block. Suppose that within a certain block D f^D_{nm}, k^D_{mn} $(n,m = 0,\ldots,7)$, denote the watermark signal in the spatial domain and F^D_{uv} $(u,v = 0,\ldots,7)$ the watermark signal in the DCT domain. F^D_{uv} is given by [27]

$$F^D_{uv} = \frac{1}{4}C(u)C(v) \times\left[\sum_{n=0}^{7}\sum_{m=0}^{7} f^D_{nm}\cos\frac{(2n+1)u\pi}{16}\cos\frac{(2m+1)v\pi}{16}\right], \tag{28}$$

where $C(u)$ are appropriate scale coefficients. Suppose also that r denotes the number of pixels of block D that belong to the subset A. Although the total number of image pixels that belong to A is $(N\times M)/2$, the number of pixels within a certain block that belong to A is not exactly $(8\times 8)/2$ but may vary, i.e. r is a random variable with mean equal to 32. The design of a low frequency watermark signal can be achieved by minimizing the following energy function with respect to k^D_{nm}:

$$\Phi = \sum_{u,v\in H}\sum (F^D_{uv})^2, \tag{29}$$

where H is some user-selected set of higher frequencies, e.g.:

$$H = \{u,v\,|\,4 < u \leqslant 7 \text{ or } 4 < v \leqslant 7\}. \tag{30}$$

Better results can be expected if the DCT coefficients are ordered in a zig-zag fashion and H is chosen to include a certain number R of the higher rank coefficients:

$$H = \{u,v\,|\,f(u,v) > 64 - R\}, \tag{31}$$

where $f(u,v)$ is the function that gives the rank of the (u,v) zig-zag ordered coefficient:

$$f(u,v) = (u+v)(u+v+1) + u. \tag{32}$$

The choice of DCT coefficients where the energy minimization will take place depends on the amount of compression that the image is expected to undergo. As the expected compression increases, H should be chosen so as to contain more mid-high range coefficients, i.e., a bigger R should be used in Eq. (31). However, a direct relation between JPEG compression and the appropriate choice of coefficients in H is very difficult to be established due to the complex nature of JPEG.

Minimization of Φ should take into account the following constraints:

1. k^D_{nm} should be chosen so that Eq. (27) holds for the entire image. This can be achieved by enforcing the following constraint within the block D:

$$\sum_{n=0}^{7}\sum_{m=0}^{7} f^D_{nm} = rk; \tag{33}$$

2. k^D_{nm} should lie within a certain range so that the watermark remains perceptually unnoticeable:

$$k^{\min} \leqslant k^D_{nm} \leqslant k^{\max}. \tag{34}$$

For example if $k = 3$ then a suitable choice for $k^{\min}$,$k^{\max}$ could be 0,6, respectively.

As can be seen from Eqs. (28) and (29), Φ is quadratic in k_{nm}^D. Therefore, finding the values of k_{mn} that minimize Eq. (29) under the previous constraints is a quadratic minimization problem that has been solved using quadratic programming techniques [24,28]. The minimum value $\Phi_{\min}$ achieved for the objective function depends on the limits (34). The smaller the changes we allow to k_{nm}^D the more difficult it is to achieve a satisfactory minimum for Φ. It should also be noted, that the optimization procedure is image independent and needs to be performed only once for a specific watermark. The two methods proposed in this section can be easily combined to achieve robustness to even bigger compression ratios.

The watermarks produced using the optimization technique consist of smooth geometrical shapes instead of impulses. The effect of this algorithm on the watermark signal is illustrated in Figs. 2 and 3. Fig. 2 shows the watermark signal within a 8×8 block when a constant value of $k = 3$ is used and the watermark pixels are chosen so as to form 2×2 blocks. The number of marked pixels in the 8×8 block for this particular case is $r = 28$. The optimized watermark using the proposed algorithm with $k^{\min} = 0$, $k^{\max} = 6$ and H given by Eq. (30) can be seen in Fig. 3. The new watermark 'fades' smoothly to zero and, therefore, its energy content in the higher DCT frequencies is limited.

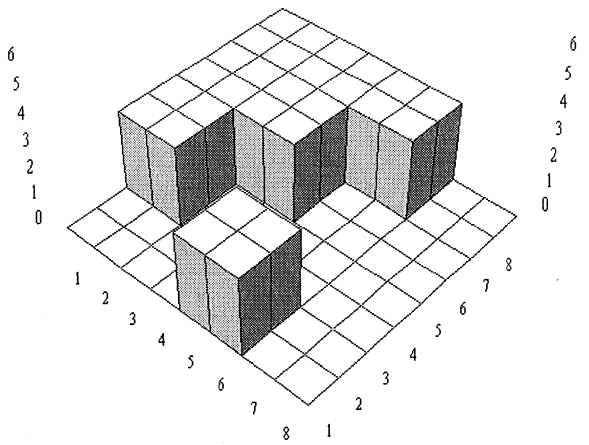

Fig. 2. The watermark signal within an 8×8 block when a constant value $k = 3$ is used.

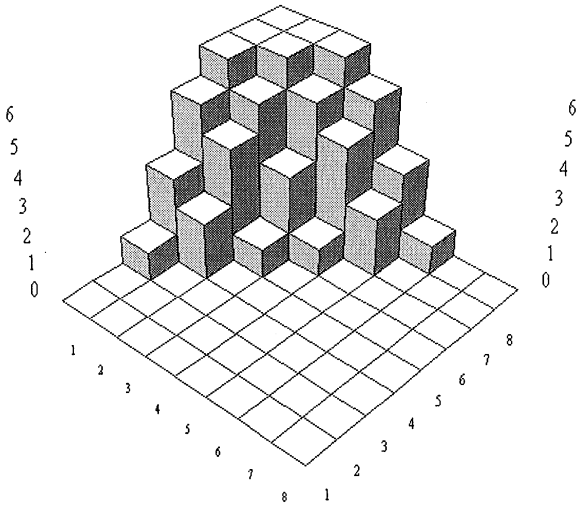

Fig. 3. The same signal when the signal is optimized for low energy content in the higher frequencies.

Note that although some of the watermark pixels have been set to zero the sum of the watermark signal values is the same with that of the signal in Fig. 2.

A slightly different approach to the watermark casting methodology is the following. Instead of using a watermark signal f_{nm} given by Eq. (8) (i.e. a watermark constructed by increasing the intensity of all pixels in A by a suitable quantity k_{nm} and leaving the intensity of the pixels in B unchanged) we can use a watermark signal that is nonzero in all image pixels, provided that it satisfies the following constraint:

$$\sum_{n,m\in A}\sum f_{nm} - \sum_{n,m\in B}\sum f_{nm} = Pk. \tag{35}$$

It is obvious that the watermark in Eq. (27) is a special case of Eq. (35) for $f_{nm} = 0$, $n,m \in B$. By using Eq. (35) instead of Eq. (27) we loosen the constraints that we impose on the watermark signal and therefore we can reach a better solution (a smaller $\Phi_{\min}$) during the optimization. However, a watermark designed to satisfy Eq. (35) affects the intensity of all image pixels and therefore causes more prominent distortions than a watermark that complies to Eq. (27).

5. Invisible watermarking using properties of the human visual system

Since watermarks are essentially additive noise superimposed on the image, they should be invisible so as not to affect the image quality to a great extent. Another important motivation for generating invisible watermarks is the fact that such watermarks would be difficult to be detected (and destroyed) by visual inspection. From this point of view, watermarking exhibits significant resemblance to lossy image compression [18], an application where invisible distortions are also highly desirable. During the last few years, a considerable amount of research in the field of image compression is directed towards algorithms that are trying to exploit the properties of the human visual system (HVS) [11] in order to achieve compression distortions that are totally invisible or just noticeable. A fundamental principle in such techniques is that of *noise masking* or *distortion masking* which refers to the decrease in the perceived intensity of a visual stimulus when this is superimposed over another stimulus. This phenomenon has been extensively studied and modelled through psychovisual tests. Noise masking has been incorporated in a couple of watermarking schemes in order to achieve invisible watermarks. In this paper we have tried to achieve noise masking by using *Just Noticeable Distortion* (JND) [11] which refers to the biggest possible invisible distortion that can be made on the signal. The evaluation of JND for a certain image is equivalent to the evaluation of a distortion map $jnd(m,n)$, called JND profile. Each element of the JND profile gives the absolute value of the biggest invisible distortion that we can cause on the corresponding image pixel. The JND profile of an image can be readily incorporated in the robust watermark design algorithm described in Section 4 to help specify in an optimal way the limits (34) that we impose on k_{nm}. In other words, instead of using the same limits $k^{\min},k^{\max}$ for all image pixels, one can use the corresponding value of the JND profile for this purpose.

$$-jnd(m,n) < k_{mn} < jnd(m,n). \tag{36}$$

We have experimented with two different JND evaluation methods that have been proposed in [5,8]. Unfortunately, both JND evaluation methods did not give satisfactory results (i.e. the predicted invisible distortions were rather noticeable). The cause of this failure lies in the fact that both these models depend on a number of parameters (image–observer distance, background luminance, etc.) whose fine-tuning requires extensive experimentation thus making their successful application rather difficult. We are currently experimenting with other JND models.

Noise masking properties have been studied not only for the intensity domain but also in other image representation domains, e.g., in the DCT domain. The JPEG quantization table $\boldsymbol{Q}$ [2] i.e., the matrix whose element Q_{uv} represents the quantization step for the F_{uv} DCT coefficient, is based on such psychovisual experiments. The degree of compression (and consequently the quality of the compressed image) obtained by the JPEG algorithm can be controlled by multiplying the elements of $\boldsymbol{Q}$ by a scaling factor g that is usually called quality factor. A quality factor equal to 0.5 produces a compressed image that is almost indistinguishable from the original image, whereas, for bigger values of g, distortions become gradually visible. The distortion $e_{uv} = \Delta F_{uv}$ caused on the coefficient F_{uv}, when this coefficient is uniformly quantized by a quality factor g, lies in the range:

$$-\frac{gQ_{uv}}{2} < e_{uv} < \frac{gQ_{uv}}{2}. \tag{37}$$

The fact that Q_{uv} were selected by psychovisual experiments ensures that, if the distortions e_{uv} on the DCT coefficients of an image lie within the limits (37), then the image distortion is in principle less visible than any other distortion of the same power.

Let us now denote by F_{uv}, F^{I}_{uv} the DCT coefficients of the watermarked and the original image, respectively, and by F^{W}_{uv} the DCT coefficients of the watermark signal. Taking into account the linear nature of the DCT transform and Eq. (7), we can easily deduce the following relation:

$$F_{uv} = F^{\mathrm{I}}_{uv} + F^{\mathrm{W}}_{uv}. \tag{38}$$

Eq. (38) can be rewritten as follows:

$$\Delta F^{\mathrm{I}}_{uv} = F_{uv} - F^{\mathrm{I}}_{uv} = F^{\mathrm{W}}_{uv}. \tag{39}$$

Eq. (39) states that the distortion caused on a DCT coefficient of an image by watermarking is equal to the corresponding DCT coefficient of the watermark signal. Therefore, according to the preceding discussion, if we restrict the DCT coefficients of the watermark signal within limits of the form:

$$-\frac{gQ_{uv}}{2} < F_{uv}^{\mathrm{W}} < \frac{gQ_{uv}}{2} \tag{40}$$

the distortions $\Delta F_{uv}^{\mathrm{I}}$ that are caused by the watermark on the image would be, in general, less visible than any other distortion of the same power. Inequality (40) can be used instead of Eq. (34) in the watermark optimization procedure described in Section 4. Alternatively one can use both Eqs. (40) and (34) in the optimization procedure. By doing so, better results (i.e. less visible watermarks) can be expected because the ad hoc limits $k^{\min}$, $k^{\max}$ are replaced by limits that have been selected using psychovisual criteria. Using a single quantization table for all images and all blocks within an image is not the optimal solution. However, it is a practical solution that usually leads to satisfactory results. It should be noted here that constraints of the form (40) cannot be imposed on the DC coefficient of the watermark, i.e. on the coefficient F_{00}^{W}, whose value, according to Eqs. (27) and (28), is fixed and equal to $Pk/8$.

The elements of the quantization matrix can be also exploited to introduce weights on the terms of the objective function Φ (29). Such weights should reflect the fact that the minimization of the energy accumulated on higher order DCT coefficients (which are more severely distorted from the JPEG algorithm) is more important than the minimization of the energy of the lower rank coefficients. Therefore, Q_{uv} which denote the quantization step for the corresponding coefficient can be used to form a weighted objective function Φ_{w} of the following type:

$$\Phi_{\mathrm{w}} = \sum_{u,v\in H}\sum (Q_{uv}F_{uv}^{D})^2. \tag{41}$$

6. Handling geometrical distortions

A weak point of the watermarking methods described in the previous sections is that the pixels that form the subset A are specified only in terms of their spatial location and therefore, any geometrical distortion (e.g. line removal) can 'fool' the detection algorithm. However, this type of distortion can be treated using a correlation-based preprocessing module. The output of this module is an index $d(n)$ that gives the translation that line n has undergone. $d(n)$ can be fed into the detection algorithm to correct the distortions before proceeding to the evaluation of the statistic q. The procedure that we used was the following. We construct the signal of horizontal differences $\Delta x_{nm}^{\mathrm{s}}$ for the distorted, watermarked image in the following way:

$$\Delta x_{nm}^{\mathrm{s}} = x_{nm}^{\mathrm{s}} - x_{nm-1}^{\mathrm{s}} = \Delta x_{nm} + (f_{nm} - f_{nm-1}),$$
$$0 \leqslant n < N, \quad 0 < m < M. \tag{42}$$

Since neighboring pixels are usually highly correlated, Δx_{nm} is a low energy, noise-like, zero mean signal. We construct also the signal Δf_{nm} of horizontal differences for the watermark signal:

$$\Delta f_{nm} = f_{nm} - f_{nm-1},$$
$$0 \leqslant n < N, \quad 0 < m < M. \tag{43}$$

Then, for each image line n, we evaluate the correlation C_{nl} of $\Delta x_{nm}^{\mathrm{s}}$ with a certain number of lines of Δf_{nm} in the neighborhood of n:

$$C_{nl} = \sum_{m} \Delta f_{n+lm}\Delta x_{nm}^{\mathrm{s}}, \quad -L < l < L. \tag{44}$$

The correct location of line n is then pronounced to be $n + d(n)$ where $d(n) = \arg\max_l\{C_{nl}\}$. A similar procedure can be used to find the correct position of image columns in an image where some of its columns have been removed, or to find the exact position of a cropped image within the original image. In all these cases, distortion detection is carried out using the distorted watermarked image and the watermark signal (which is of course known), i.e., without using the original watermarked image. It should be noted here that erroneous estimates of $d(n)$ for a line n or a set of lines, do not affect watermark detection on the rest of the lines, i.e. the lines whose translation has been correctly estimated. This implies that even with a certain number of errors the detection algorithm should

reach the correct decision. Of course each erroneously estimated line translation $d(n)$ decreases the probability that the detection algorithm will succeed in correctly deciding upon the presence or absence of a watermark.

7. Image dependent watermarks

Using a single watermark on all images of equal size whose copyright is owned by the same individual is very convenient in terms of implementation simplicity and speed. However, such a practice is not a safe one. If, for example, someone can manage to obtain both the original and the watermarked version of the same image, he can easily recover the watermark signal by performing a simple subtraction of the two images and then eliminate the watermark from all images in which it exists. Another way to recover and subsequently destroy a watermark embedded in a set of images is the following. Let us suppose that a set of K images $u^1_{mn},\dots,u^K_{mn}$ have been marked with the same watermark f_{mn} and a certain embedding factor k to produce the watermarked images $v^1_{mn},\dots,v^K_{mn}$. We subtract the mean value from each watermarked image and average the zero mean images to obtain $\bar{v}_{mn}$

$$\bar{v}_{mn}=\frac{1}{K}\sum_{i=1}^{K}(v^i_{mn}-m'_i), \tag{45}$$

or equivalently

$$\bar{v}_{mn}=f_{mn}-\frac{k}{2K}+\frac{1}{K}\sum_{i=1}^{K}(u^i_{mn}-m_i), \tag{46}$$

where m'_i,m_i are the mean intensities of v_{mn}, u_{mn}, respectively, which are related by the following formula:

$$m'_i=m_i+k/2. \tag{47}$$

By averaging a large number of images the sum in the right-hand side of Eq. (46) would tend to zero and thus $\bar{v}_{mn}$ would give a noisy version of the watermark signal that can be used to eliminate the watermark from all images where it exists.

In order to robustify the watermarking technique against this type of attack we have devised a variation of the basic method that generates image dependent watermarks, i.e. a technique that, for the same watermark key, leads to different watermark patterns when applied on different images. This is achieved using the following methodology: First, we construct the watermark pattern S. Then we split the pixels in A in two subsets A_c,A_m containing hP and $(1-h)P$ pixels respectively ($0\leqslant h\leqslant 1$). The pixels of the subset A_c form the fixed part of the watermark pattern. The position of these pixels does not change when the watermark is applied on different images. On the other hand, the spatial location of the pixels that form the subset A_m changes when the watermark is applied on different images. Each pixel in A_m is translated $(\Delta n,\Delta m)$ pixels away from its original position. The actual value of this translation $(\Delta n,\Delta m)$ depends on the intensity value of some other pixel that belongs to A_c. The whole procedure of moving a pixel that belongs to A_m into a new position should be insensitive to image distortions. Otherwise, the detection algorithm would not be able to calculate the positions of these pixels. In our case we used the following procedure. We split the intensity range $[0\dots255]$ into C intervals $G_1,\dots,G_C$ and assign to each interval G_i a different translation vector $(\Delta n_i,\Delta m_i)$. Then for each pixel in A_c we find the corresponding interval G_i and translate one or more of the pixels in A_m by $(\Delta n_i,\Delta m_i)$. The procedure of determining $(\Delta n_i,\Delta m_i)$ is robust to distortions since even if the image is distorted by compression, filtering etc., the intensity interval of a pixel is not likely to change, especially if the intensity range has been coarsely quantized, i.e. if C is small (e.g. $C=4$ or $C=6$). Intervals G_i are selected so that each of them includes the same number of pixels. This ensures that equal numbers of each translation will be performed and, therefore, the distribution of the marked pixels on the image will continue to be random. Such a segmentation can be done by splitting the intensity range so that the area under the image histogram curve is the same for all intervals. For the algorithm implemented in this paper we selected $h=1/3$ of the pixels in A to be of fixed position, i.e. pixels that belong to A_c. Also, the intensity range was split into $C=4$ intervals in the way described above. The intensity of each pixel (m,n) in A_c controlled the movements of two pixels

(m', n'), (m'', n'') of A_m because the number of pixels in A_m is twice the number of pixels in A_c:

$$|A_m| = \tfrac{2}{3}|A| = 2|A_c|. \quad (48)$$

The selection of the pixels (m',n'), (m'',n'') whose translation depends on the intensity of (m,n) was done through appropriate coordinate mapping functions $f_1(m)$, $g_1(n)$, $f_2(m)$, $g_2(n)$ in a way that ensured that all pixels in A_m are visited. A simple method to do this mapping is to split the pixels in A in triplets $(\mathscr{A}_1,\mathscr{B}_1,\mathscr{C}_1),(\mathscr{A}_2,\mathscr{B}_2,\mathscr{C}_2),\ldots$ and use pixel $\mathscr{A}_i$ to control the translation of pixels $\mathscr{B}_i,\mathscr{C}_i$.

A similar procedure is followed during the detection phase. S is generated, the pixels in A_c as well as the C intervals of the intensity range are being selected and then the positions of pixels in A_m are evaluated. When the determination of the pixels that belong to A has been completed, we proceed to the watermark detection as described in Section 2. It is obvious that the image dependent watermark design method can be easily combined with the robust watermark design methods proposed in the previous sections.

The above procedure is also robust to histogram equalization, logarithmic grayscale transform and in general all grayscale transforms that are done through an increasing function. The proof can be found in Appendix C.

8. Watermarks for color images

The watermarking techniques presented in the previous sections can be easily extended to handle color images. In this case, watermark casting is done by generating three different watermark patterns S_R,S_G,S_B, one for each RGB channel, and modifying, for each channel, the intensity of the pixels that belong to the corresponding sets A_R,A_G,A_B. The watermark casting and detection procedures for color images are exactly the same as for the corresponding procedures for grayscale images. Sets A, B are now considered to be the union of the sets A_R,A_G,A_B and B_R,B_G,B_B, respectively. Therefore, in a color image, the number of samples P that are used in the evaluation of the test statistic q is three times the number of samples in a grayscale image of the same size. This implies that the variance $\hat{\sigma}_{\bar{w}}$ of $\bar{w}$ will be smaller and so the embedding factor k that is required for a specific probability of correct detection $1-\beta$ (without considering image distortions) is smaller than the one that we should use to obtain the same certainty in a grayscale image of the same dimensions. However, if we want to generate watermarks that are robust to image distortions this value might be too small to ensure the desired distortion immunity level. In such cases we should proceed to an experimental evaluation of the appropriate k value, as described in Section 4.

Instead of marking the three R, G, B components one can choose to mark the luminance and the chrominance components of the image. The watermark casting and detection methodology is exactly the same.

9. Experimental results

The resistance of the proposed watermarking algorithms to various distortions was studied in a series of experiments on grayscale images. The first set of experiments dealt with the resistance of the various techniques to JPEG compression. As it was mentioned in Section 4, when an image is subject to a certain distortion (in our case compression), both the probability $1-\beta$ of correct detection in a watermarked image and the probability α of erroneous watermark detection in an image that bears no watermark change. The new values for $1-\beta$ and α cannot be theoretically evaluated. Therefore, we resort to the following experimental evaluation procedure. We create K copies of an image, each marked with a different watermark, i.e. a watermark generated by a different watermark key. Then, we compress the watermarked images using the JPEG algorithm and a certain quality factor g and we try to detect the K watermarks. The percentage of the successfully detected watermarks expresses the probability of correct detection for this compression. By repeating the above experiment for a range of different quality factors we can generate plots that depict the change of certainty $1-\beta$ with respect to compression. Three such plots for three different watermarking techniques can be seen in Fig. 4.

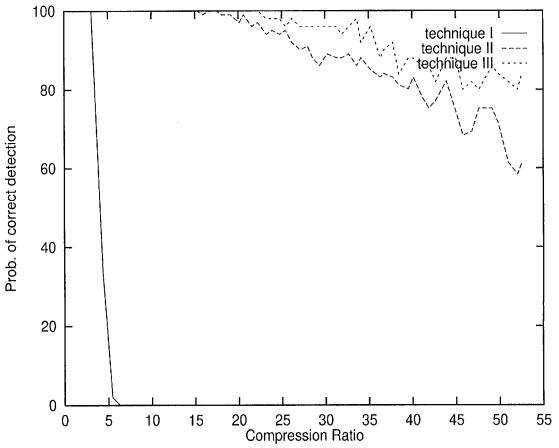

Fig. 4. Plots of probability of correct detection $1-\beta$ versus the compression ratio for the three watermarking techniques presented in Section 9.

Fig. 5. Test image.

The watermarking techniques that were compared were the following:

1. The basic algorithm described in Section 2.
2. The first technique described in Section 4 with a block size of 2×4.
3. A combination of the technique (2) with the technique for the generation of optimal watermarks described in Section 4. The objective function in this specific implementation was that of Eq. (41) whereas the coefficients in the set H were the 43 zig-zag ordered coefficients of the higher rank ($R = 43$ in Eq. (31)). A combination of constraints (34) ($k^{\min} = 0$, $k^{\max} = 4$) and (40) ($g = 1.5$) was used in the optimization procedure.

The embedding factor used in all three methods was chosen so that the SNR for the watermarked images was 35 db. The probability of correct detection (no distortions considered) for the above embedding factor was practically 100% (100% for technique (1) and 99.99% for techniques (2) and (3)). In all three techniques we have incorporated the image dependent watermarking method presented in Section 7. The original test image is presented in Fig. 5. The test image (dimensions 960×960 pixels) watermarked by the techniques (1)–(3) described above can be seen in Figs. 6–8. The watermark signal is almost invisible.

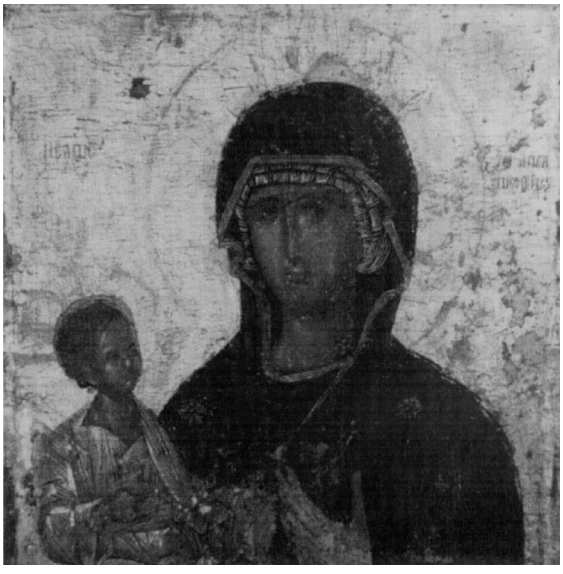

Fig. 6. Test image watermarked with technique (1) described in Section 9.

Fig. 4 suggests that the basic watermarking technique generates watermarks that are extremely vulnerable to compression. On the other hand, techniques (2) and (3) exhibit a highly satisfactory

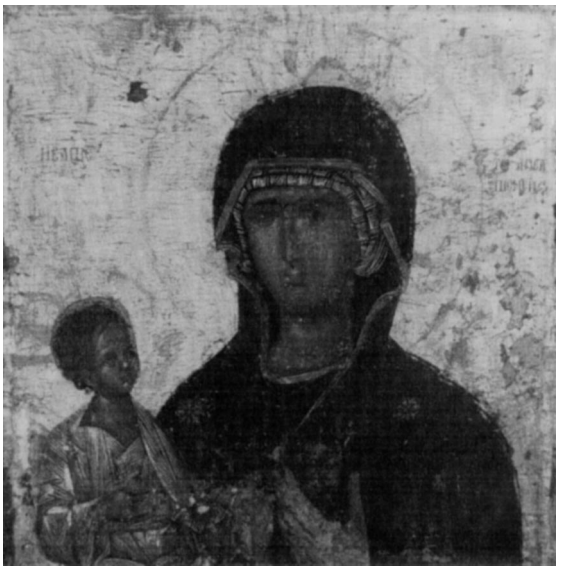

Fig. 7. Test image watermarked with technique (2) described in Section 9.

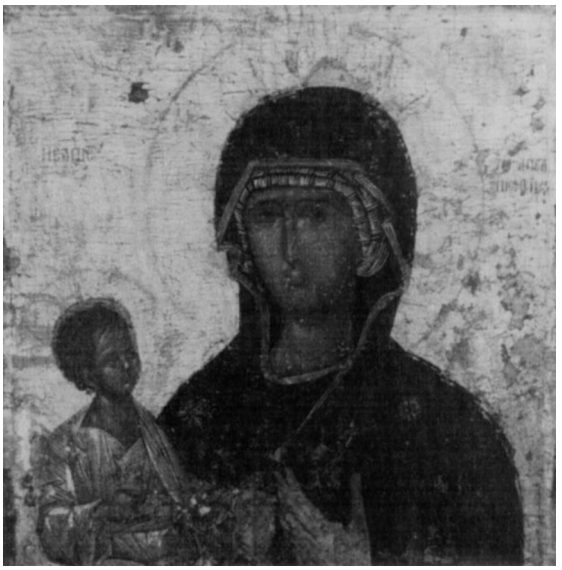

Fig. 8. Test image watermarked with technique (3) described in Section 9.

robustness against JPEG-caused distortions. For instance, watermarks generated by technique (3) are detected with a certainty of 100% even when the image is compressed down to 1 : 23. In contrast to what happens to the probability of correct detection $1 - \beta$, the probability α of erroneous detection of an image that is not watermarked is not affected by the JPEG compression, i.e. it remains practically zero.

Table 1
Probability of correct detection for the watermarks generated using the techniques (1)–(3) presented in Section 9 when the watermarked images have been filtered by 3×3 mean and median filters

Method	3×3 median	3×3 mean
(1)	0	0
(2)	97	92
(3)	100	99

In the second set of experiments we tested the robustness of the proposed watermarking techniques against 3×3 mean and median filtering. The probability of correct detection for the three techniques was experimentally calculated by filtering 100 watermarked copies of the test image and evaluating the percentage of correctly detected watermarks. Results can be seen in Table 1 . The robustness of techniques (2) and (3) to both operations is very satisfactory. As expected, the watermark optimization method (technique (3)) gave the best results.

The effectiveness of the correlation-based module described in Section 6 was tested in the third set of experiments. 100 copies of the test image were watermarked using the technique (2) and then three lines were removed from each image. The detection algorithm preceded by the correlation-based module succeeded in detecting the watermark in all distorted images thus indicating that the proposed technique can successfully handle geometrically distorted images.

In the last experiment we tested the resistance of the proposed techniques to noise. For this purpose we generated 100 copies of the test image, watermarked them by method (2) and then added zero mean Gaussian noise having standard deviation $\sigma = 15$. A sample noisy image can be seen in Fig. 9. The percentage of successfully detected watermarks on the noisy images was 99%, a strong indication that the proposed watermarking techniques are extremely robust to noise.

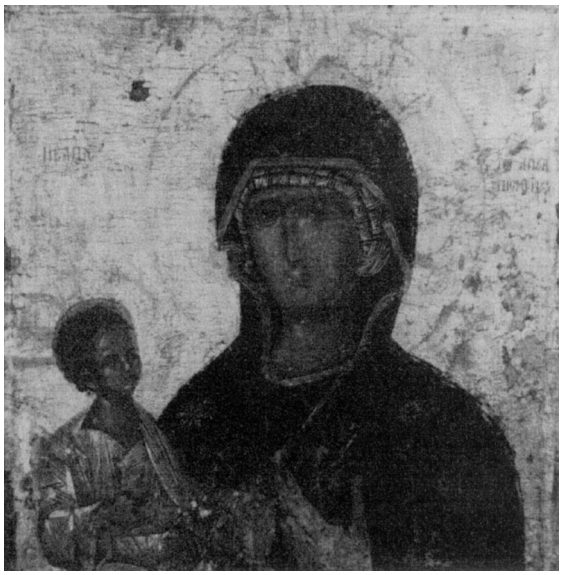

Fig. 9. Test image watermarked and distorted by Gaussian noise of standard deviation $\sigma = 15$.

10. Conclusions

A new method for the copyright protection of images using digital watermarks has been proposed in this paper. The proposed method produces watermarks that are not detectable by visual inspection but at the same time are robust to JPEG compression and lowpass filtering. Variations that produce watermarks that are immune to geometric transformations and also image-dependent were also presented. An extension to color images was also presented.

Appendix A.

Let us consider the case when two watermarks have X out of P pixels in common (partial overlap). When we try to detect one of these watermarks, while this image was watermarked by using the other, we will get the following:

$$\bar{c}' = \bar{a} + \frac{X}{P}k, \tag{A.1}$$

$$\bar{b}' = \bar{b} + \left(1 - \frac{X}{P}\right)k, \tag{A.2}$$

$$\bar{w}' = \bar{c}' - \bar{b}' = (\bar{a} - \bar{b}) + \left(2\frac{X}{P} - 1\right)k. \tag{A.3}$$

In order to have a wrong answer, the following inequality must hold:

$$\frac{(\bar{a} - \bar{b}) + (2\frac{X}{P} - 1)k}{\hat{\sigma}_{\bar{w}'}} > z_{1-\beta}, \tag{A.4}$$

where $\hat{\sigma}_{\bar{w}'}$ is given by

$$\hat{\sigma}^2_{\bar{w}'} = \frac{s^2_{\bar{c}'} + s^2_{\bar{b}'}}{P}. \tag{A.5}$$

Therefore, the probability of a wrong answer is given by

$$\text{Prob}(p + h > z_{1-\beta}), \tag{A.6}$$

where p and h are given by the following equations:

$$p = \frac{\bar{a} - \bar{b}}{\hat{\sigma}_{\bar{w}'}}, \tag{A.7}$$

$$h = \frac{(2\frac{X}{P} - 1)\ k}{\hat{\sigma}_{\bar{w}'}}. \tag{A.8}$$

In order to proceed, we shall assume that $\hat{\sigma}_{\bar{w}'} = \sigma_{\bar{w}'}$ which is a reasonable assumption since the number P of samples is usually very large. Under the previous assumption p and h are independent random variables and thus the density function of their sum is given by the convolution of their density functions [19]. p can be rewritten as follows:

$$p = \frac{(\bar{a} - \bar{b})}{\sigma_{\bar{w}}} \cdot \frac{\sigma_{\bar{w}}}{\sigma_{\bar{w}'}} = q\frac{\sigma_{\bar{w}}}{\sigma_{\bar{w}'}}, \tag{A.9}$$

where q is the test statistic we would get if we examined the clear image. Since the cumulative density function of q is given by $N(q,0,1)$ the cumulative density function of p is $N(p,0,\sigma^2_{\bar{w}}/\sigma^2_{\bar{w}'})$.

We shall now try to find the distribution of h. The first watermark divides the set of image pixels into two equally sized subsets, namely subset A with P pixels having $s_{nm} = 1$ and subset B with P pixels as well, having $s_{nm} = 0$. The probability

that the second watermark will have exactly X pixels in A and $P-X$ pixels in B is given by

$$\frac{\binom{P}{X}\binom{P}{P-X}}{\binom{2P}{P}} = \frac{\binom{P}{X}^2}{\binom{2P}{P}}. \tag{A.10}$$

X follows a hypergeometric distribution with mean value $P/2$ and variance $P^2/(8P-4)$. A good approximation of the distribution of X can be achieved through a normal distribution $N(X,P/2,P/8)$ having mean value $P/2$ and variance $P/8$. As a result, according to Eq. (A.8), the cumulative density function of h is $N(h,0,k^2/2P\hat{\sigma}^2_{\bar{w}'})$, or, due to the assumption about $\hat{\sigma}_{\bar{w}'}$, $N(h,0,k^2/2P\sigma^2_{\bar{w}'})$.

The convolution of two normal distributions is also a normal distribution, having mean value the sum of means and variance the sum of variances [20]. Therefore, if we denote:

$$t = p + h \tag{A.11}$$

it follows that the distribution of t is $N(t,0,\frac{\sigma^2_{\bar{w}}}{\sigma^2_{\bar{w}'}} + \frac{k^2}{2P\sigma^2_{\bar{w}'}})$. Using Eq. (16) the variance σ_t^2 of the random variable t can be rewritten as follows:

$$\begin{aligned}\sigma_t^2 &= \frac{\sigma^2_{\bar{w}}}{\sigma^2_{\bar{w}'}} + \frac{k^2}{2P\sigma^2_{\bar{w}'}},\\ &= \frac{\sigma^2_{\bar{w}}}{\sigma^2_{\bar{w}'}} + \frac{\sigma^2_{\bar{w}} 2z^2_{1-\beta}}{\sigma^2_{\bar{w}'}\ P}.\end{aligned} \tag{A.12}$$

Since the watermark signal has the nature of additive noise, the overall variance of the image is increased, which means that $\sigma_{\bar{w}} < \sigma_{\bar{w}'}$. Therefore,

$$\sigma_t^2 < 1 + \frac{2z^2_{1-\beta}}{P}. \tag{A.13}$$

The error probabilities α and α' are given by the following expressions:

$$\alpha = 1 - N(z_{1-\beta},0,1), \tag{A.14}$$

$$\alpha' = 1 - N(z_{1-\beta},0,\sigma_t^2). \tag{A.15}$$

Therefore, the increase in the probability of error a of type (I) due to the existence of a watermark different from the one that we are trying to detect is given by

$$\begin{aligned}\alpha' - \alpha &= N(z_{1-\beta},0,1) - N(z_{1-\beta},0,\sigma_t^2)\\ &= N(z_{1-\beta},0,1) - N\left(\frac{z_{1-\beta}}{\sigma_t},0,1\right)\\ &< N(z_{1-\beta},0,1) - N\left(\frac{z_{1-\beta}}{\sqrt{1+\frac{2z^2_{1-\beta}}{P}}},0,1\right).\ \square\end{aligned} \tag{A.16}$$

Appendix B.

The pixels that belong to the subset A' of the subsampled image, i.e. the pixels x'_{nm} for which $s'_{nm}=1$ would be the result of averaging Eq. (19) within 2×2 blocks containing 2, 3 or 4 marked pixels. One can easily find out that within a 2×2 block, there are 4 different ways to have 3 marked pixels, 3 ways to have 2 marked pixels and, obviously, only one way to have 4 marked pixels. In other words, there are 8 different types of 2×2 blocks in I that result in a marked pixel in I_{sub}. Therefore, in 1/8 of the pixels in A', k' would be equal to k, in 3/8 of the pixels in A', k' would be equal to $k/2$ and in 4/8 of the pixels in A', k' would be equal to $3k/4$.

The pixels that belong to the subset B' of the subsampled image, i.e. the pixels x'_{nm} for which $s'_{nm}=0$ would be the result of averaging Eq. (19) within 2×2 blocks containing 0, 1 or 2 marked pixels. Within a 2×2 block, there are 4 different ways to have 1 marked pixel, 3 ways to have 2 marked pixels and only one way to have 0 marked pixels. In other words, there are 8 different types of 2×2 blocks in I that result in an unmarked pixel in I_{sub} . Therefore, in 1/8 of the pixels in B', k' would be equal to 0, in 3/8 of the pixels in B', k' would be equal to $k/2$ and in 4/8 of the pixels in B', k' would be equal to $k/4$. Therefore, when calculating $\bar{w}'$, Eqs. (21) to (23) result. $\square$

Appendix C.

Let us suppose that the intensity range [0,255] is being split into C intervals $G_1,\ldots,G_C$ whose limits

are denoted by $L_0, \ldots, L_C$ ($L_0 = 0$, $L_C = 255$). Let us also suppose that a pixel x_{nm} belongs to the ith interval C_i:

$$L_{i-1} < x_{nm} < L_i$$

Finally, let $L'_0, \ldots, L'_C$, x'_{nm} be the mappings of $L_0, \ldots, L_C$, x_{nm} in the transformed image (e.g. $x'_{nm} = f(x_{nm})$) and $G'_1 \ldots, G'_C$ the intervals selected by $L'_0, \ldots, L'_C$. If the transformation function f is an increasing one, which is the case for histogram equalization and logarithmic grayscale, then we will have

$$L'_{i-1} < x'_{nm} < L'_i,$$

i.e. x'_{nm} will still belong to the ith interval G'_i in the transformed image. Similarly, all pixels of the original image whose intensity belonged to the ith interval would belong to the G'_i interval in the transformed image. Therefore, the intervals $G'_1, \ldots, G'_C$ would contain equal numbers of pixels which implies that $G'_1, \ldots, G'_C$ will be the intervals selected by the detection algorithm. Due to this fact a pixel that has been classified to the ith interval during watermark casting would be classified in the same interval in the transformed image by the detection algorithm and so the algorithm will not be 'fooled' by the transform. □

References

[1] W. Bender, D. Gruhl, N. Morimoto, Techniques for data hiding, Proc. SPIE 2420 (1995) 40.

[2] V. Bhaskaran, K. Konstantinides, Image and video compression standards: algorithms and architectures, Kluwer, Dordrecht, 1995.

[3] A. Bors, I. Pitas, Image watermarking using DCT domain constraints, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 231–234.

[4] O. Bruyndonckx, J.-J. Quisquater, B. Macq, Spatial method for copyright labeling of digital images, Proc. IEEE Workshop on Nonlinear Signal and Image Processing, Neos Marmaras, Greece, 20–22 June 1995, pp. 456–459.

[5] C. Chou, Y. Li, A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile, IEEE Trans. Circuits Systems Video Technol. 5 (6) (December 1995) 467–476.

[6] I.J. Cox, J. Kilian, T. Leighton, T. Shamoon, Secure spread spectrum watermarking for multimedia, NEC Research Institute, Technical Report 95–10, 4 December 1995.

[7] I. Cox, J. Kilian, T. Leighton, T. Shamoon, Secure spread spectrum watermarking for images, audio and video, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 243–246.

[8] B. Girod, The information theoretical significance of spatial and temporal masking in video signals, SPIE, Human Vision, Visual Processing, and Digital Display 1077 (1989) 178–187.

[9] F. Hartung, B. Girod, Digital watermarking of raw and compressed video, Proc. SPIE 2952, Digital Compression Technologies and Systems for Video Communications, October 1996, pp. 205–213.

[10] C. Hsu, J. Wu, Hidden-signatures in images, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 223–226.

[11] N. Jayant, J. Johnston, R. Safranek, Signal compression based on models of human perception, Proc. IEEE 81 (10) (October 1993) 1385–1422.

[12] H. Kinoshita, An image digital signature system with ZKIP for the graph isomorphism, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 247–250.

[13] E. Koch, J. Zhao, Towards robust and hidden image copyright labelling, Proc. IEEE Workshop on Nonlinear Signal and Image Processing, Neos Marmaras, Greece, 20–22 June 1995, pp. 452–455.

[14] B.M. Macq, J.J. Quisquater, Cryptology for digital TV broadcasting, Proc. IEEE, June 1995, 944–957.

[15] K. Matsui, K. Tanaka, Video-steganography: how to secretly embed a signature in a picture, IMA Intellectual Property Project Proc. 1 (1) (January 1994) 187–206.

[16] N. Nikolaidis, I. Pitas, Copyright protection of images using robust digital signatures, IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP-96), vol. 4, May 1996, pp. 2168–2171.

[17] J. O'Ruanaidh, W. Dowling, F. Boland, Phase watermarking of digital images, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 239–242.

[18] J. O'Ruanaidh, W. Dowling, F. Boland, Watermarking digital images for copyright protection, IEEE Proc. on Vision, Image and Signal Processing 143 (4), August 1996, pp. 250–256.

[19] A. Papoulis, Probability, Random Variables and Stochastic Processes, McGraw-Hill, New York, 1965.

[20] A. Papoulis, Probability & Statistics, Prentice Hall, Englewood Cliffs, NJ, 1990.

[21] I. Pitas, T.H. Kaskalis, Applying signatures on digital images, IEEE Workshop on Nonlinear Image and Signal Processing, Neos Marmaras, Greece, June 1995, pp. 460–463.

[22] I. Pitas, A method for signature casting on digital images, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, pp. 215–218.

[23] M. Swanson, B. Zhu, A. Tewfik, Transparent robust image watermarking, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 211–214.

[24] R.J. Vanderbei, LOQO: an interior point code for quadratic programming, Technical Report SOR 94-15, Princeton University, 1994.

[25] R.G. van Schyndel, A.Z. Tirkel, C.F. Osborne, A digital watermark, Proc. IEEE Int. Conf. on Image Processing, Austin, TX, 13–16 November 1994, pp. 86–90.
[26] G. Voyatzis, I. Pitas, Applications of toral automorphisms in image watermarking, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. II, 1996, pp. 237–240.
[27] G.K. Wallace, The JPEG still picture compression standard, IEEE Trans. Consumer Electron., 38 (February 1992), pp. 18–34.
[28] G.R. Walsh, Methods of Optimization, Wiley, New York, 1975.
[29] R. Wolfgang, E. Delp, A watermark for digital images, Proc. 1996 IEEE Int. Conf. on Image Processing (ICIP 96), vol. III, 1996, pp. 219–222.
[30] J. Zhao, E. Koch, Embedding robust labels into images for copyright protection, Technical Report, Fraunhofer Institute for Computer graphics, Darmstadt, Germany, 1994.