Watermarking Digital Image and Video Data

A State-of-the-Art Overview

Gerhard C. Langelaar, Iwan Setyawan, and Reginald L. Lagendijk

In the past decade there has been an explosion in the use and distribution of digital multimedia data. PCs with Internet connections have taken homes by storm and have made the distribution of multimedia data and applications much easier and faster. Electronic commerce applications and on-line services are rapidly being developed. Even the analog audio and video equipment in the home is in the process of being replaced by their digital successors. As a result, we can see the digital mass recording devices for multimedia data enter the consumer market of today.

The Need for Watermarking

Although digital data has many advantages over analog data, service providers are reluctant to offer services in digital form because they fear unrestricted duplication and dissemination of copyrighted material. Because of possible copyright issues, the intellectual property of digitally recorded material must be protected [90]. The lack of such adequate protection systems for copyrighted content was the reason for the delayed introduction of the digital versatile disk (DVD) [100]. Several media companies initially refused to provide DVD material until the copy protection problem had been addressed [89], [81]. Representatives of the consumer electronics industry and the motion picture industry have agreed to seek legislation concerning digital video recording devices. Recommendations describing ways that would protect both intellectual property and consumers' rights have been submitted to the U.S. Congress [81] and resulted in the Digital Millennium Copyright Act [25], which was signed by President Clinton on October 28, 1998. The European Union is also preparing such intellectual property rights protection for digital multimedia products including CDs or DVDs [28].

To provide copy protection and copyright protection for digital audio and video data, two complementary techniques are being developed: encryption and watermarking [23]. Encryption techniques can be used to protect digital data during the transmission from the sender to the receiver [63]. After the receiver has received and decrypted the data, however, the data is identical to the original data and no longer protected. Watermarking techniques can compliment encryption by embedding a secret imperceptible signal, a watermark, directly into the original data in such a way that it always remains present. Such a watermark, for instance, can be used for the following purposes:

▲ *Copyright Protection:* For the protection of intellectual property, the data owner can embed a watermark representing copyright information in his data. This watermark can prove his ownership in court when someone has infringed on his copyrights.

▲ *Fingerprinting:* To trace the source of illegal copies, the owner can use a fingerprinting technique. In this case, the owner can embed different watermarks in the copies of the data that are supplied to different customers. Fingerprinting can be compared to embedding a serial number that is related to the customer's identity in the data. It enables the intellectual property owner to identify customers who have broken their license agreement by supplying the data to third parties.

▲ *Copy Protection:* The information stored in a watermark can directly control digital recording devices for copy protection purposes [62]. In this case, the watermark represents a copy-prohibit bit and watermark detectors in the recorder determine whether the data offered to the recorder may be stored or not.

▲ *Broadcast Monitoring:* By embedding watermarks in commercial advertisements, an automated monitoring system can verify whether advertisements are broadcasted

1053-5888/00/$10.00©2000IEEE

as contracted [3]. Not only commercials but also valuable TV products can be protected by broadcast monitoring [53]. News items can have a value of over US$100,000 per hour, which make them very vulnerable to intellectual property rights violation. A broadcast surveillance system can check all broadcast channels and charge the TV stations according to their findings.

▲ *Data Authentication:* Fragile watermarks [108] can be used to check the authenticity of the data. A fragile watermark indicates whether the data has been altered and supplies localization information as to where the data was altered.

Watermarking techniques are not only used for protection purposes. Other applications include:

▲ *Indexing:* Indexing of video mail, where comments can be embedded in the video content; indexing of movies and news items, where markers and comments can be inserted that can be used by search engines.

▲ *Medical Safety:* Embedding the date and the patient's name in medical images could be a useful safety measure [3].

▲ *Data Hiding:* Watermarking techniques can be used for the transmission of secret private messages. Since various governments restrict the use of encryption services, people may hide their messages in other data.

Some authors, for example in [11], refer to watermarking technique only when the application embeds a few bits (as few as one bit) of data for copyright notice/protection applications. Other applications are considered to fall into the category of data embedding. We prefer to use the term watermarking, however, for all these applications in this article. In our opinion, watermarking has nowadays been used for applications beyond the limits of copy protection/authentication, an example of which is Digimarc's Smart Images [1].

Watermarking Requirements

Each watermarking application has its own specific requirements. Therefore, there is no set of requirements to be met by all watermarking techniques. Nevertheless, some general directions can be given for most of the applications mentioned above:

▲ *Perceptual Transparency:* In most applications the watermarking algorithm must embed the watermark such that this does not affect the quality of the underlying host data. A watermark-embedding procedure is truly imperceptible if humans cannot distinguish the original data from the data with the inserted watermark [97]. Even the smallest modification in the host data may become apparent, however, when the original data is compared directly with the watermarked data. Since users of watermarked data normally do not have access to the original data, they cannot perform this comparison. Therefore, it may be sufficient that the modifications in the watermarked data go unnoticed as long as the data are not compared with the original data [103].

A watermark-embedding procedure is imperceptible if humans cannot distinguish the original data from the data with the inserted watermark.

▲ *Payload of the Watermark:* The amount of information that can be stored in a watermark depends on the application. For copy protection purposes, a payload of one bit is usually sufficient.

According to a recent proposal for audio watermarking technology from the International Federation for the Phonographic Industry (IFPI), the minimum payload for an audio watermark should be 20 bits per second, independently of the signal level and music type [46]. According to [75], however, this minimum is very ambitious and should be lowered to only a few bits per second.

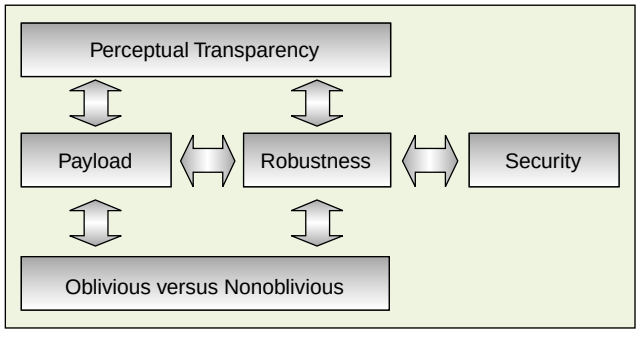
For the protection of intellectual property rights, it seems reasonable to assume that one wants to embed an amount of information similar to that used for ISBN, International Standard Book Numbering (roughly 10 digits) or better ISRC, International Standard Recording Code (roughly 12 alphanumeric letters). On top of this, one should also add the year of copyright, the permissions granted on the work, and the rating for it [59]. This means that about 60 bits [31] or 70 bits [59] of information should be embedded in the host data, the image, the video frame, or the audio fragment.

Another important concept regarding watermark payload for digital audio and video is *watermark granularity*. Watermark granularity represents how much data is needed to embed one unit of watermark information. Using the example above, one unit of watermark information consists of 60 or 70 bits. This could be embedded in a single frame of video or spread, for instance, over 100 frames of video (or similarly for audio, the watermark could be embedded in a 1-s fragment or spread for instance over 5 s of audio data). Spreading the watermark in this way may not be desirable because when someone takes just 80 frames from the watermarked video, the watermark information is no longer retrievable. For digital videos, 1 s of video is considered to be the smallest copyrighted entity. Therefore, the watermark information has to be embedded in a less than 1 s fragment of the video stream (approximately 25 frames). Again using the example above, the watermark bit rate should then be more than 70 bits/s.

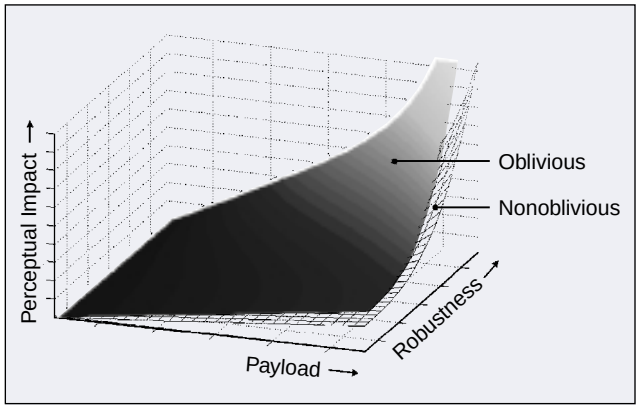
▲ *Robustness:* A fragile watermark that has to prove the authenticity of the host data does not have to be robust against processing techniques or intentional alterations of the host data, since failure to detect the watermark proves that the host data has been modified and is no longer au-

thentic. If a watermark is used for another application, however, it is desirable that the watermark always remains in the host data, even if the quality of the host data is degraded, intentionally or unintentionally. Examples of unintentional degradations are applications involving storage or transmission of data, where lossy compression techniques are applied to the data to reduce bit rates and increase efficiency. Other unintentional quality-degrading processing techniques include filtering, re-sampling, digital-analog (D/A) and analog-digital (A/D) conversion. On the other hand, a watermark can also be subjected to processing solely intended to remove the watermark [23]. In addition, when many copies of the same content exist with different watermarks, as would be the case for fingerprinting, watermark removal is possible because of collusion between several owners of copies. In general, there should be no way in which the watermark can be removed or altered without sufficient degradation of the perceptual quality of the host data so as to render it unusable.

▲ *Security:* The security of watermarking techniques can be interpreted in the same way as the security of encryption techniques. Kerckhoff's assumption states that one should assume that the method used to encrypt the data is known to an unauthorized party and that the security must lie in the choice of a key [69]. Hence a watermarking technique is truly secure if knowing the exact algorithms for embedding and extracting the watermark does not help an unauthorized party to detect the presence of the watermark or remove it [97].



▲ *1. Mutual dependencies between the basic requirements.*



▲ *2. Illustration of the relation between the basic requirements for a secure watermark.*

▲ *Oblivious versus Nonoblivious Watermarking:* In some applications, like copyright protection and data monitoring, watermark extraction algorithms can use the original unwatermarked data to find the watermark. This is called *nonoblivious* watermarking [59]. In most other applications, e.g., copy protection and indexing, the watermark-extraction algorithms do not have access to the original unwatermarked data. This renders the watermark extraction more difficult. Watermarking algorithms of this kind are referred to as *public*, *blind*, or *oblivious* watermarking algorithms.

The requirements listed above are all related to each other. For instance, a very robust watermark can be obtained by making many large modifications to the host data for each bit of the watermark. Large modifications in the host data will be noticeable, however, and many modifications per watermark bit will limit the maximum amount of watermark bits that can be stored in a data object. Hence, a tradeoff should be considered between the different requirements so that an optimal watermark for each application can be developed. The mutual dependencies between the basic requirements are shown in Fig. 1.

The relation between the basic requirements for a well-designed secure watermark is represented in Fig. 2. The perceptual impact axis represents the quality degradation of the data due to watermarking. The higher the perceptual impact, the worse the quality degradation. The payload axis represents the amount of data that could be embedded in the data. The robustness axis represents the ability of the watermarking system to resist attacks. The security of a watermark influences the robustness enormously. If a watermark is not secure, it cannot be very robust.

Scope of the Article

To embed watermark information in host data, watermark embedding techniques apply minor modifications to the host data in a perceptually invisible manner, where the modifications are related to the watermark information. The watermark information can be retrieved afterwards from the watermarked data by detecting the presence of these modifications.

A wide range of modifications in any domain can be used for watermarking techniques. Prior to embedding or extracting a watermark, the host data can be converted, for instance, to the spatial, the Fourier, the wavelet, the discrete cosine transform or even the fractal domain, where the properties of the specific transform domains can be exploited. In these domains modifications can be made, like least significant bit (LSB) modification, noise addition, coefficient re-ordering, coefficient removal, warping or morphing data parts, and block similarities enforcing. Further, the impact of the modifications can be minimized with the aid of human visual models, whereas modifications can be adapted to the anticipated

post-processing techniques or to the compression format of the host data.

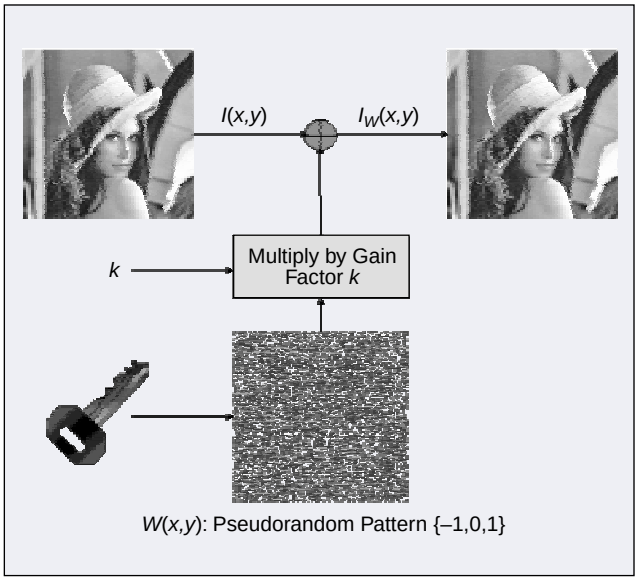
Since the most commonly used techniques use additive noise for watermark embedding and correlation techniques for watermark detection, we discuss the oblivious correlation-based techniques extensively in this article, together with all its possible variations. Other oblivious techniques are explained as well. The cryptographic security of the methods described here lies in the key that is used to generate a pseudorandom watermark pattern or to pseudorandomly select image regions or coefficients to embed the watermark. In general, the robustness of the watermark against processing techniques depends on the embedding depth and the amount of information bits of the watermark.

The article is organized as follows. First we will discuss digital watermarking techniques based on correlation in the next two sections. And then we will discuss digital watermarking techniques that are not based on correlation. The last section presents some conclusion of the article including a brief discussion of recent developments in the digital watermarking area.

Correlation-Based Watermarking Techniques

Basic Technique in the Spatial Domain

The most straightforward way to add a watermark to an image in the spatial domain is to add a pseudorandom noise pattern to the luminance values of its pixels. Many methods are based on this principle [91], [10], [76], [18], [36], [35], [77], [93], [105], [61], [106], [113], [32], [107], [108], [53]. In general, the pseudorandom noise pattern consists of the integers {–1,0,1}, however, also floating-point numbers can also be used. The pattern is generated based on a key using, for instance, seeds, linear shift registers or randomly shuffled binary images. The only constraints are that the energy in the pattern is



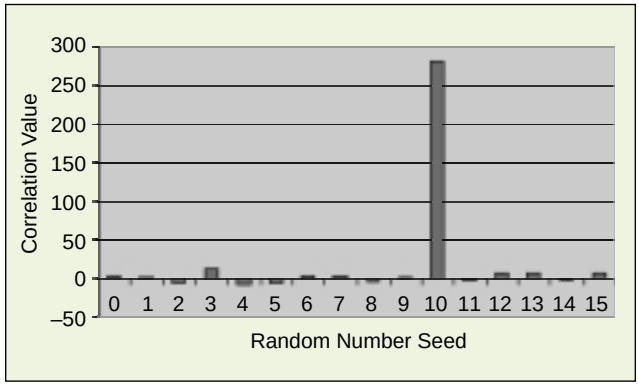
▲ *3. Watermark embedding procedure.*

To add a watermark to an image in the spatial domain, add a pseudorandom noise pattern to the luminance values of its pixels.

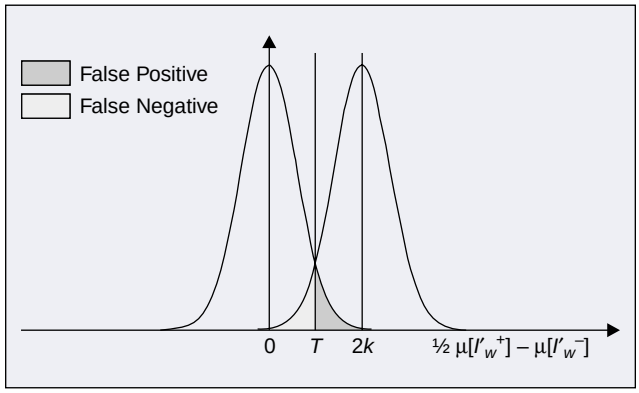
more or less uniformly distributed and that the pattern is not correlated with the host image content. To create the watermarked image $I_W(x,y)$ the pseudorandom pattern $W(x,y)$ is multiplied by a small gain factor k and added to the host image $I(x,y)$, as illustrated in Fig. 3

$$I_W(x,y) = I(x,y) + k \cdot W(x,y). \tag{1}$$

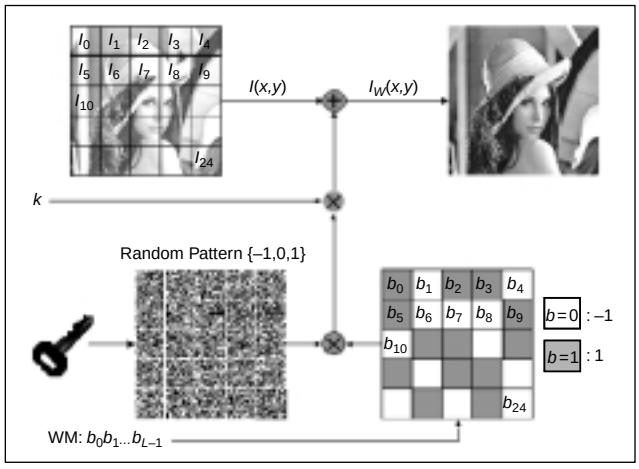
To detect a watermark in a possibly watermarked image $I'_W(x,y)$ we calculate the correlation between the image $I'_W(x,y)$ and the pseudorandom noise pattern $W(x,y)$. In general, $W(x,y)$ is normalized to a zero mean before correlation. Pseudorandom patterns generated using different keys have very low correlation with each other. Therefore, during the detection process the correlation value will be very high for a pseudorandom pattern generated with the correct key and would be very low



▲ *4. Correlation values for a pseudorandom pattern generated with seed=10 correlated with pseudorandom patterns generated with other seeds.*



▲ *5. Watermark detection procedure.*



▲ *6. Watermark bit string embedding procedure.*

otherwise. This is shown in Fig. 4. Here we have watermarked the Lena image by adding a pseudorandom pattern generated using with seed = 10 to the image. Figure 4 shows the correlation values of some pseudorandom patterns generated using seeds varying between 0 and 15 to the watermarked image. It can be seen that the correlation when the correct seed (10) is used is very high, while the correlation when the wrong seeds are used are very low.

During the detection process, it is common to set a threshold T to decide whether the watermark is detected or not. If the correlation exceeds a certain threshold T, the watermark detector determines that image $I'_W(x,y)$ contains watermark $W(x,y)$

$$\begin{aligned} R_{I'_W(x,y)W(x,y)} > T &\rightarrow W(x,y) \text{ detected} \\ < T &\rightarrow \text{No } W(x,y) \text{ detected.} \end{aligned} \tag{2}$$

If $W(x,y)$ only consists of the integers $\{-1,1\}$ and if the number of -1s equals the number of 1s, we can estimate the correlation as

$$\begin{aligned} R_{I'_W(x,y)W(x,y)} &= \frac{1}{N}\sum_{i=1}^{N} I'_{W_i}(x,y)W_i(x,y) \\ &= \frac{1}{N}\sum_{i=1}^{N/2} I'_{W_i} W_i^{+} + \frac{1}{N}\sum_{i=1}^{N/2} I'_{W_i} W_i^{-} \\ &= \frac{1}{2}\left\{\mu\left[I'^{+}_{W}(x,y)\right] - \mu\left[I'^{-}_{W}(x,y)\right]\right\}. \end{aligned} \tag{3}$$

Here N is the number of pixels in the image I'_W, and $^{+,-}$ indicates the set of pixels where the corresponding noise pattern is positive or negative, and $\mu[I'^{+}_{W}(x,y)]$ represents the average value of set pixels in $I'^{+}_{W}(x,y)$. From (3) it follows that the watermark detection problem corresponds to testing the hypothesis whether two randomly selected sets of pixels in a watermarked image have the same mean.

During the detection process, the watermark detector can make two types of errors. In the first place, it can detect the existence of a watermark, although there is none. This is called a *false positive*. In the second place, the detector can reject the existence of the watermark, even though there is one. This is called a *false negative*. The probability function for the detection process is presented in Fig. 5.

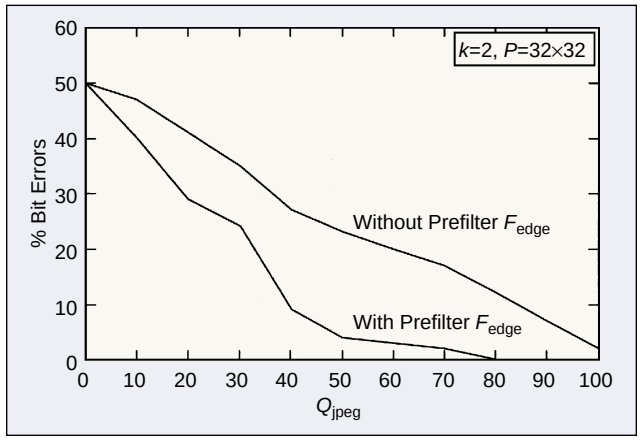
In [52] the probabilities of these two types of errors are derived based on a first-order autoregressive image model:

$$P_{fp} = \frac{1}{2}\operatorname{erfc}\left(\frac{T\sqrt{N}}{\sigma_W\sigma_I\sqrt{2}}\right) \quad \text{and}$$

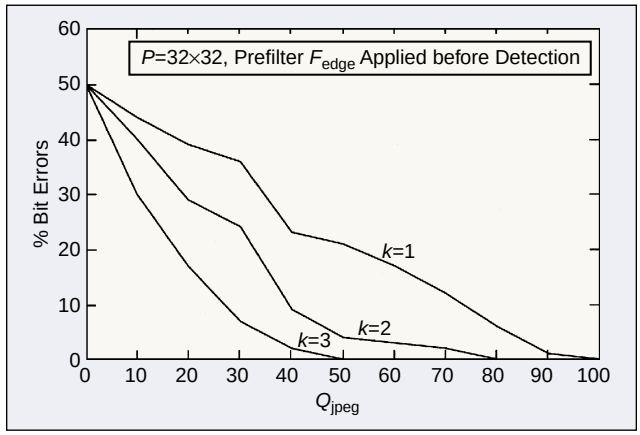
$$P_{fn} = \frac{1}{2}\operatorname{erfc}\left(\frac{(\sigma_W^2 - T)\sqrt{N}}{\sigma_W\sigma_I\sqrt{2}}\right)$$

$$\text{where} \quad \operatorname{erfc}(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-t^2/2}\,dt. \tag{4}$$

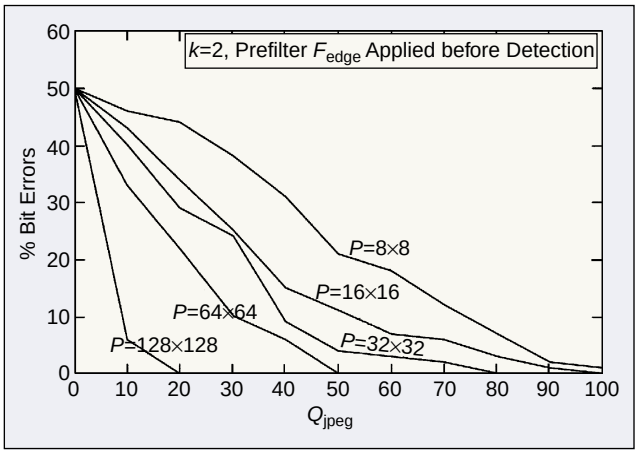
Here, P_{fp} represents the probability of false positive, P_{fn} represents the probability of false negative, σ_W^2 represents the variance of the watermark pixels and σ_I^2 denotes the variance of the image pixels. If the watermark pattern $W(x,y)$ only consists of the integers $\{-1,1\}$ and the number of -1s equals the number of 1s, the variance of the watermark σ_W^2 equals k^2. The errors P_{fp} and P_{fn} can be minimized by increasing the gain factor k. Using larger



▲ *7. Watermark detection with and without prefiltering.*



▲ *8. Influence of the gain factor k on the robustness of a watermark.*



▲ *9. Influence of the number of pixels per watermark bit P on the robustness of a watermark.*

values for the gain factor, however, decreases the visual quality of the watermarked image.

Since the image content can interfere with the watermark, especially in the low-frequency components, the reliability of the detector can be improved by applying matched filtering before correlation [26], [91], [35]. This decreases the contribution of the original image to the correlation. For instance, a simple edge-enhancing finite impulse response (FIR) filter F_{edge} can be used, where F_{edge} is given by the following convolution kernel:

$$F_{\text{edge}} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 10 & -1 \\ -1 & -1 & -1 \end{bmatrix} / 2. \tag{5}$$

The experimental results presented in the next section show that applying this filter before correlation reduces the error probability significantly, even when the visual quality of the watermarked image was affected seriously before correlation [35], [61]. In [67], the authors proposed another way to improve the robustness of the watermark. The robustness improvement is achieved by performing a spectrum equalization prior to watermark embedding.

Extensions to Embed Multiple Bits or Logos in One Image

From the watermark detector's point of view, an image I can be regarded as Gaussian noise, which distorts the watermark information W. Further, the watermarked image I_W can be seen as the output of a communication channel subject to Gaussian noise over which the watermark information is transmitted. In this case, reliable transmission of the watermark is theoretically possible if its information rate does not exceed the channel capacity, which is given by [92]

$$C = W_b \log_2\left(1 + \frac{\sigma_W^2}{\sigma_I^2}\right) \text{bit/pixel.} \tag{6}$$

Here, C is given in units of watermark information bits per image pixel and the available bandwidth W_b is equal to one cycle per pixel. For practical systems, however, a tighter empirically lower bound can be determined [93]

$$C = W_b \log_2\left(1 + \frac{\sigma_W^2}{\alpha \cdot \sigma_I^2}\right) \text{bit/pixel.} \tag{7}$$

Here, α is a small headroom factor, which is larger than one and typically around three. Since the signal-to-noise ratio σ_W^2 / σ_I^2 is significantly smaller than one, (7) can be approximated by

$$C \approx \frac{1}{\ln 2}\left(\frac{\sigma_W^2}{\alpha \cdot \sigma_I^2}\right) \text{bit/pixel.} \tag{8}$$

According to this equation, it should be possible to store much more information in an image than just 1 bit using the basic technique described in the previous section. For instance, a watermark consisting of the integers

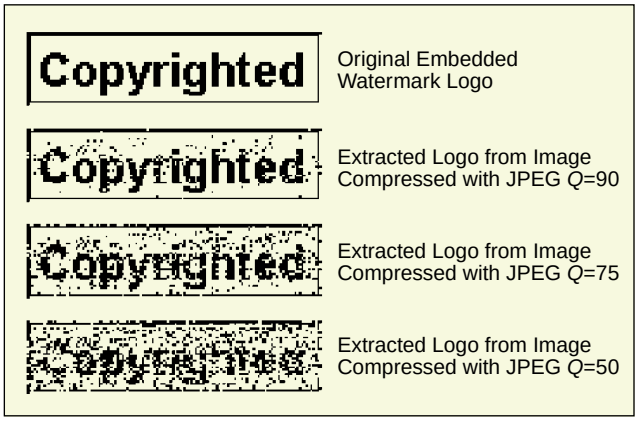
```
RP0 :-1  1  1-1-1  1-1-1  1  1-1   b0:0  →  +RP0:-1  1  1-1-1  1-1-1  1  1-1
RP1 : 1  1-1-1  1-1-1  1  1-1  1   b1:0  →  +RP1: 1  1-1-1  1-1-1  1  1-1  1
RP2 : 1-1-1  1-1-1  1  1-1  1-1    b2:1  →  −RP2:-1  1  1-1  1  1-1-1  1-1  1
RP3 :-1-1  1-1-1  1  1-1  1-1-1    b3:1  →  −RP3: 1  1-1  1  1-1-1  1-1  1  1
RP4 :-1  1-1-1  1  1-1  1-1-1  1   b4:0  →  +RP4:-1  1-1-1  1  1-1  1-1-1  1
RP5 : 1-1-1  1  1-1  1-1-1  1  1   b5:1  →  −RP5:-1  1  1-1-1  1-1  1  1-1-1
RP6 :-1-1  1  1-1  1-1-1  1  1  1  b6:0  →  +RP6:-1-1  1  1-1  1-1-1  1  1  1  +
                                                W :-3  5  1-3  1  3-7  1  3-1  3
```

▲ *10. Example of a CDMA watermark generation for 7 bits $b_0 b_1 \cdots b_7$.*

```
W  : -3   5   1 -3   1   3 -7   1   3 -1   3
I  : 98  98 97 98 97 96 97 96 95 94 94  +
IW : 95 103 98 95 98 99 90 97 98 93 97

E[(RP0 − E[RP0]) · (IW − E[IW])] = +15.6 → b0 = 0
E[(RP1 − E[RP1]) · (IW − E[IW])] = +16.4 → b1 = 0
E[(RP2 − E[RP2]) · (IW − E[IW])] = -26.4 → b2 = 1
E[(RP3 − E[RP3]) · (IW − E[IW])] = - 3.1 → b3 = 1
E[(RP4 − E[RP4]) · (IW − E[IW])] = +21.6 → b4 = 0
E[(RP5 − E[RP5]) · (IW − E[IW])] = -23.6 → b5 = 1
E[(RP6 − E[RP6]) · (IW − E[IW])] = + 0.4 → b6 = 0
```

▲ *11. Example of CDMA watermark extraction, compare to Fig. 10.*



▲ *12. Extracted watermark logos from a JPEG distorted image.*

$\{-k, k\}$ added to the 512×512 Lena image (Fig. 3) can carry approximately 50, 200, or 500 bits of information for $k = 1, 2$, or 3 respectively and for $\alpha = 3$.

There are several ways to increase the payload of the basic watermarking technique. The simplest way to embed a string of l watermark bits $b_0 b_1 \cdots b_{l-1}$ in an image is to divide the image I into l subimages $I_0 I_1 \cdots I_{l-1}$ and to add a watermark to each subimage, where each watermark represents one bit of the string [93], [35], [61]. This procedure is depicted in Fig. 6.

Using (8) we can calculate the number of pixels P required per subimage for reliable detection of a single bit in a subimage

$$P \approx \frac{\alpha \sigma_I^2 \ln 2}{\sigma_W^2} \text{ pixels.} \qquad (9)$$

The watermark bits can be represented in several ways. A pseudorandom pattern can be added if the watermark bit equals one, and the subimage can be left unaffected if the watermark bit equals zero. In this case, the detector calculates the correlation between the subimage and the pseudorandom pattern and assigns the value 1 to the watermark bit if the correlation exceeds a certain threshold T; otherwise the watermark bit is assumed to be zero.

The use of a threshold can be circumvented by adding two different pseudorandom patterns RP_0 and RP_1 for watermark bit 0 and 1. The detector now calculates the correlation between the subimage and the two patterns. The bit value corresponding with the pattern that gives the highest correlation is assigned to the watermark bit. In [93] the two patterns are chosen in such a way that they only differ in sign, $RP_0 = -RP_1$. In this case, the detector only has to calculate the correlation between the subimage and one of the patterns; the sign of the correlation determines the watermark bit value.

To investigate the effect on the robustness of the watermark of the prefilter in the detector, the gain factor k, and the number of pixels P per watermark bit, we perform the following experiments. We first add a watermark to an image with the method of [93]. Next, we compress the watermarked image with the JPEG algorithm [73], where the quality factor Q_{jpeg} of the compression algorithm is made variable. Finally, the watermark is extracted from the decompressed image and compared bit by bit with the originally embedded watermark bits. From this experiment, we find the percentages of watermark bit errors due to JPEG compression as a function of the JPEG quality factor.

The first experiment shows the effect of applying the prefilter given by (5) before detection of a watermark embedded with a gain factor $k = 2$, and $P = 32 \times 32$ pixels per watermark bit. In Fig. 7 the percentages bit errors caused by JPEG compression are plotted for a detector that uses this prefilter and for a plain detector. It can clearly be seen that prefiltering significantly increases the robustness of the watermark.

The second experiment shows the effect of increasing the gain factor k for a watermark embedded with

▲ *13. Fourier amplitude watermark. (a) Original image, (b) watermarked image, (c) difference $W(x,y) = I - I_w$ scaled for visibility, and (d) heavily marked image.*

$P = 32 \times 32$ pixels per watermark bit and detected using a prefilter. From Fig. 8 it follows that the robustness of a watermark can be improved significantly by increasing the gain factor.

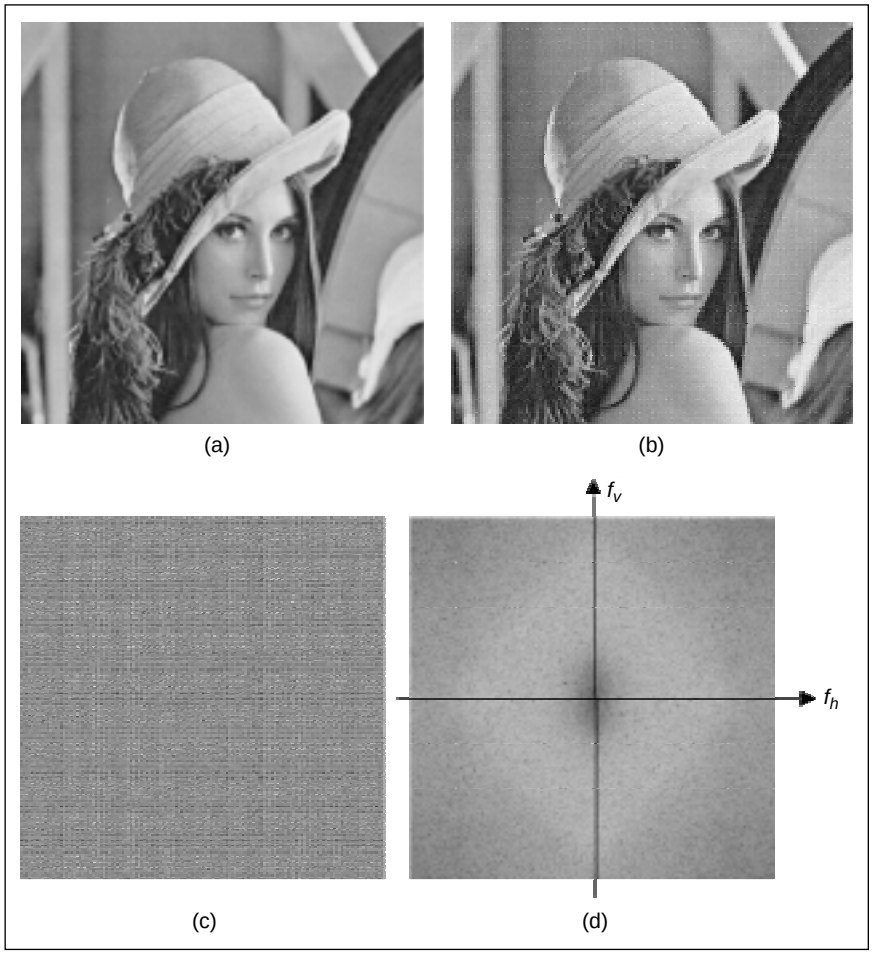
The third experiment shows the influence of the number of pixels P per watermark bit on the robustness of a watermark embedded with a gain factor $k = 2$ and detected using a prefilter. From Fig. 9 it follows that decreasing the payload of the watermark by increasing P improves the robustness significantly.

Another way to increase the payload of the basic watermarking technique is the use of direct sequence code division multiple access (DS-CDMA) spread spectrum communications [87], [88]. Here, for each bit b_j out of the watermark bit string $b_0 b_1 \cdots b_{l-1}$ a different stochastically independent pseudorandom pattern RP_i is generated that has the same size as the image. This pattern is dependent on the bit value b_j. Here we use the pattern $+RP_i$ if b_j represents a 0 and $-RP_i$ if b_j represents a 1. The summation of all l random patterns $\pm RP_i$ forms the watermark. Prior to adding the watermark to an image, we can scale the watermark by a gain factor or limit it to a certain small range. An example of the one-dimensional watermark generation is presented in Fig. 10. This example uses seven different pseudorandom patterns to embed the seven watermark bits 0011010.

Each bit b_j out of the watermark bit string $b_0 b_1 \cdots b_{l-1}$ can be extracted by calculating the correlation between the normalized image I'_W and the corresponding pseudorandom pattern RP_i. If the correlation is positive, the value 0 is assigned to the watermark bit, otherwise the watermark bit is assumed to be one. Figure 11 shows as an example the extraction of the embedded watermark bits in Fig. 10.

The methods to extend the watermark payload described above, namely using individual image tiles for each watermark bit and using CDMA, have their advantages and disadvantages. If each watermark bit has its own image tile, there is no interference between the bits and only a small number of multiplications are required to calculate the correlations, since each bit is completely spread over the image. If the image is cropped, however, the watermark bits located at the border are lost. If CDMA techniques are used, the probability that all bits can be recovered after cropping the image is high. The watermark bits may interfere with each other, however, and many multiplications are required to calculate the correlations.

The watermark bits embedded using the methods mentioned above can represent anything: copyright messages, serial numbers, plain text, control signals, etc. The content represented by these bits can be compressed, encrypted, and protected by error correcting codes. In some cases it may be more useful to embed a small logo instead of a bit string as a watermark. If the watermarked image is distorted, the watermark logo will also be affected. But now the sophisticated pattern-recognition capabilities of the human visual system (HVS) can be exploited to detect the logo [15], [45], [102]. For instance, we can embed a binary watermark logo with 128×32 pixels in an image with 512×512 pixels using the techniques described in this section. Each logo pixel is embedded in an image tile of 8×8 pixels by adding the pseudorandom pattern $+RP$ or $-RP$ to the image tile for a black or white logo pixel respectively. As an example in Fig. 11 the results are shown of the logos extracted after the watermarked image has been degraded with the lossy JPEG [73] compression algorithm using several quality factors. From Fig. 12 it can be seen that, although it is heavily corrupted, the logo can still be recognized.



▲ *14. An 8×8 DCT middle band image content independent watermark. (a) Watermarked image, (b) a heavily watermarked image, (c) difference $W(x,y) = I(x,y) - I_w(x,y)$, and (d) Fourier spectrum $W(u, v)$.*

Techniques for Transform Domains

The techniques described in the previous section can also be applied on transformed image data. Each transform domain has it own advantages and disadvantages. In [85] the phase of the discrete Fourier transform (DFT) is used to embed a watermark, because the phase is more important than the amplitude of the DFT values for the intelligibility of an image. Putting a watermark in the most important components of an image improves the robustness of the watermark, since tampering with these important image components to remove the watermark will severely degrade the quality of the image. The second reason to use the phase of the DFT values is that it is well known from communication theory that phase modulation often possesses superior noise immunity in comparison with amplitude modulation [85].

Many watermarking techniques use DFT amplitude modulation because of its translation or shift invariant property [40], [41], [74], [83], [86]-[88]. Because cyclic translation of the image in the spatial domain does not affect the DFT amplitude, the watermark embedded in this domain will be translation invariant. In case a CDMA watermark is used, it is even slightly resistant to cropping. Furthermore, the watermark can be embedded directly in the most important middle band frequencies, since modulation of the lowest frequency coefficients results in visible artifacts while the highest frequency coefficients are very vulnerable to noise, filtering, and lossy compression. Finally the watermark can easily be made image content dependent by modulating the DFT amplitude coefficients $|I(u,v)|$ in the following way [20]:

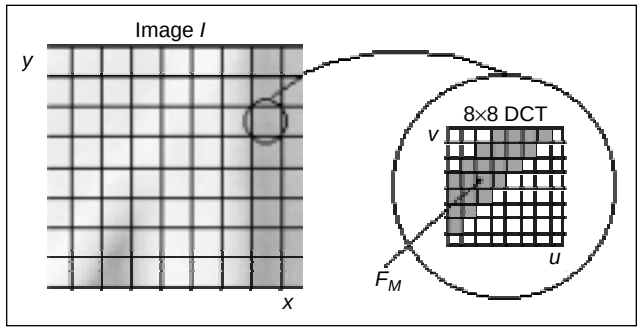
$$|I_W(u,v)| = |I(u,v)| \cdot (1 + k \cdot W(u,v)). \tag{10}$$

Here, $W(u,v)$ represents a CDMA watermark, a two-dimensional (2-D) pseudorandom pattern, and k denotes the gain factor. Now, the modification of a DFT coefficient is not fixed but proportional to the amplitude of the DFT coefficient. Small DFT coefficients are hardly affected, whereas larger DFT coefficients are affected more severely. This complies with Weber's law [50]. The HVS does not perceive equal changes in images equally, but visual sensitivity is nearly constant with respect to relative changes in an image. If ΔI is a just noticeable difference, then $\Delta I / I$ = constant. Rewriting (10) gives

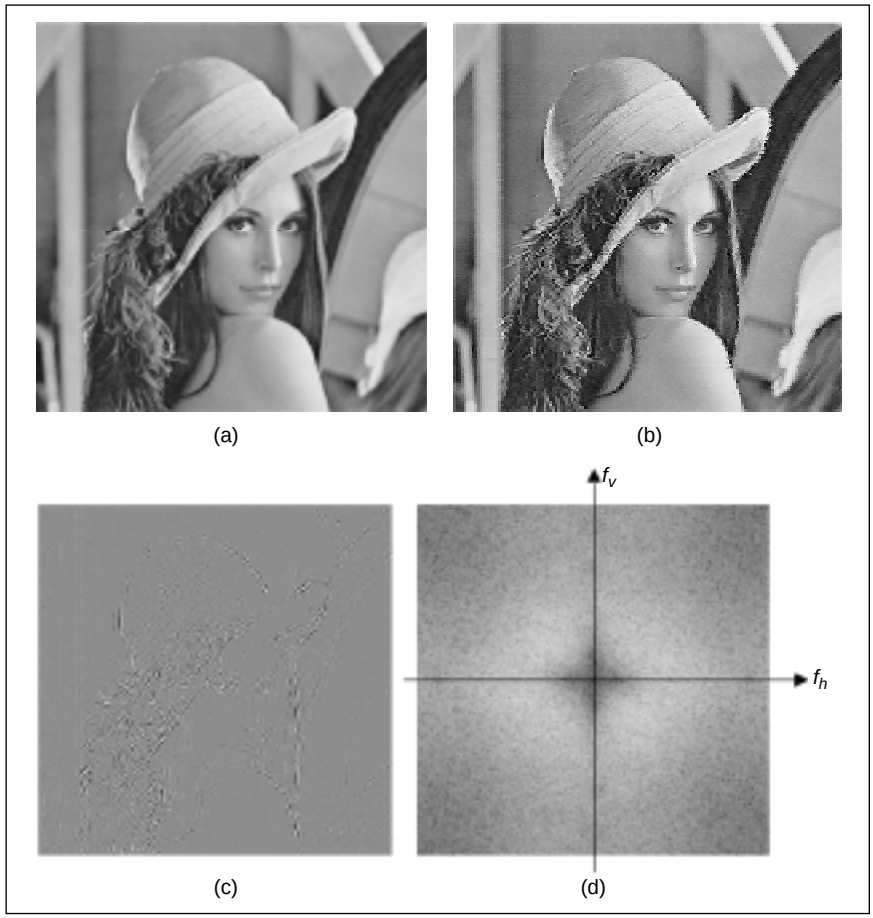
$$\frac{|I_W(u,v)| - |I(u,v)|}{|I(u,v)|} = \frac{\Delta I(u,v)}{|I(u,v)|} = k \cdot W(u,v) \cong \text{constant}. \tag{11}$$

Since the watermark here is mainly embedded in the larger DFT coefficients, i.e., the perceptually most significant components of the image, the robustness of the watermark improves.

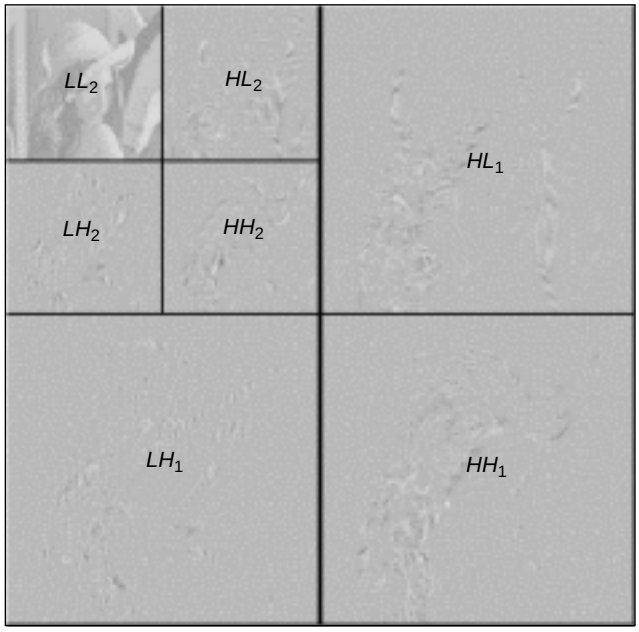
Note that the symmetry of the Fourier coefficients must be preserved to ensure that the image data is still real valued after the inverse transform to the spatial domain. If the coefficient $|I(u,v)|$ in an image with $N \times M$ pixels is modified according to (10), its counterpart



▲ *15. Definition of the middle band frequencies in a DCT block.*



▲ *16. An 8 × 8 block DCT middle band image content dependent watermark. (a) Watermarked image, (b) a heavily watermarked image, (c) difference $W(x,y) = I(x,y) - I_w(x,y)$, and (d) Fourier spectrum $W(u,v)$.*



▲ *17. DWT two-level decomposition of an image.*

$|I(N-u, M-v)|$ must be modified in the same way. In Fig. 13(b) an example is given of an image in which a watermark is embedded using all DFT amplitude coefficients according to (10) and using a relatively small gain factor k. Figure 13(c) presents the strongly amplified difference between the original image and the watermarked image. Figure 13(d) shows an image watermarked using a large value of the gain factor k.

Another commonly used domain for embedding a watermark is the discrete cosine transform (DCT) domain [12], [20]-[22], [45], [78], [79], [99], [84], [110]. Using the DCT an image can easily be split up in pseudo frequency bands, so that the watermark can conveniently be embedded in the most important middle band frequencies. Furthermore, the sensitivity of the HVS to the DCT basis images has been extensively studied, which resulted in the recommended JPEG quantization table [73]. These results can be used for predicting and minimizing the visual impact of the distortion caused by the watermark. Finally, the block-based DCT is widely used for image and video compression. By embedding a watermark in the same domain as the compression scheme used to process the image (in this case in the DCT domain) we can anticipate lossy compression because we are able to anticipate which DCT coefficients will be discarded by the compression scheme. Furthermore, we can exploit the DCT decomposition to make real-time watermark applications.

In Fig. 14(a) an example is given of an image in which a 2-D CDMA watermark W is embedded in the 8×8 block DCT middle band frequencies. The 8×8 DCT coefficients $F(u,v)$ are modulated according to the following:

$$I_{W_{x,y}}(u,v) = \begin{cases} I_{x,y}(u,v) + k \cdot W_{x,y}(u,v), & u,v \in F_M \\ I_{x,y}(u,v), & u,v \notin F_M \end{cases} \quad x,y = 1,8,16,\ldots . \tag{12}$$

Here F_M denotes the middle band frequencies, k the gain factor, (x,y) the spatial location of an 8×8 pixel block in image I, and (u,v) the DCT coefficient in the corresponding 8×8 DCT block (Fig. 15).

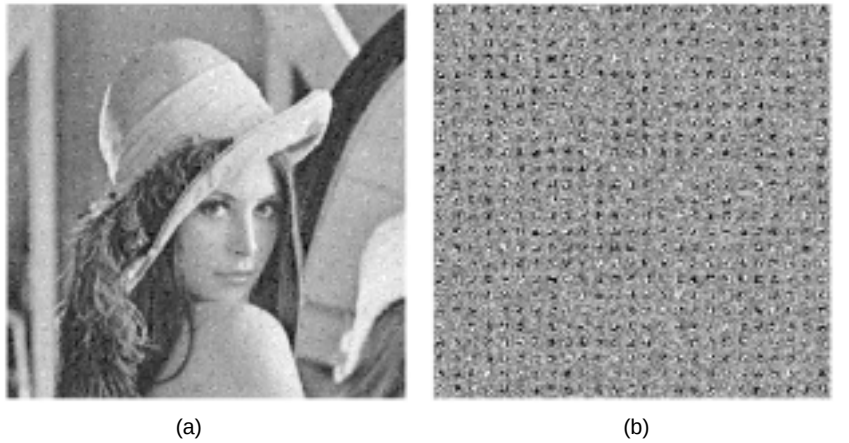
In Fig. 14(c) the strongly amplified difference between the original image and the watermarked image is presented. Figure 14(d) shows the Fourier spectrum of the watermark. Here, it can clearly be seen that watermark only affects the middle band frequencies (white regions) while leaving lower and high frequency components relatively unaffected (dark regions).

The watermark can be made image dependent by changing the modulation function to [c.f. (10)]

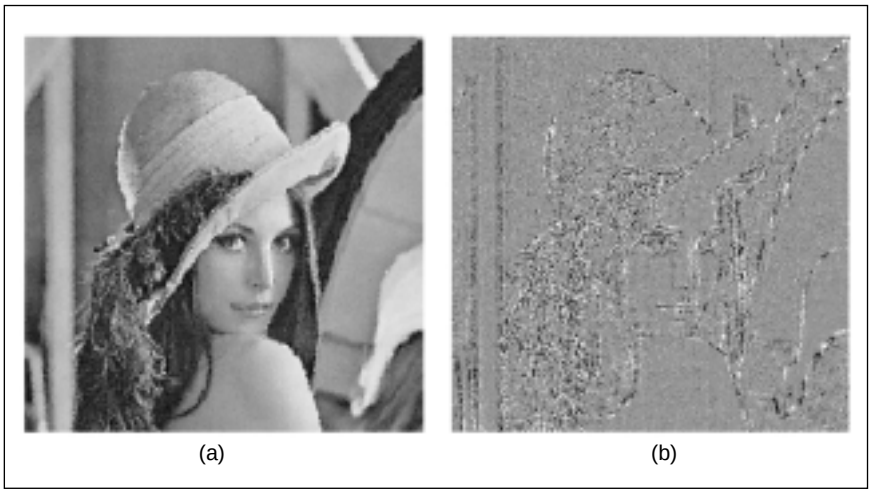
$$I_{W_{x,y}}(u,v) = \begin{cases} I_{x,y}(u,v) \cdot (1 + k \cdot W_{x,y}(u,v)), & u,v \in F_M \\ I_{x,y}(u,v), & u,v \notin F_M \end{cases} \quad x,y = 1,8,16,\ldots . \tag{13}$$

If this modulation function is applied, the results from Fig. 13 change into the results shown in Fig. 16. From Fig. 16(b) and (c) it appears that most distortion introduced by the watermark is located around the edges and in the textured areas.

Further improvements for DCT-domain correlation-based watermarking systems' performance could be achieved by using watermark detectors based on generalized Gaussian model, instead of the widely used pure Gaussian assumption [42]. By performing a theoretical analysis for DCT-domain watermarking methods for images, the authors in [42] provide analytical



▲ *18. DWT image content independent watermark. (a) A heavily watermarked image and (b) difference $W(x,y) = I(x,y) - I_w(x,y)$.*



(a) (b)

▲ *19. DWT image content dependent watermark. (a) A heavily watermarked image and (b) difference $W(x,y) = I(x,y) - I_w(x,y)$.*

expressions which can be used to measure beforehand the performance that can be expected for a certain image and to analyze the influence of the image characteristics and system parameters (e.g., watermark length) on the final performance. Furthermore, the result of this analysis can help determining the proper detection threshold T to obtain a certain false positive rate. The authors in [42] claim that by abandoning the pure Gaussian noise assumption, some substantial performance improvements could be obtained.

If watermarking techniques can exploit the characteristics of the HVS, it is possible to hide watermarks with more energy in an image, which makes watermarks more robust. From this point of view the discrete wavelet transform (DWT) is a very attractive transform, because it can be used as a computationally efficient version of the frequency models for the HVS [7]. For instance, it appears that the human eye is less sensitive to noise in high resolution DWT bands and in the DWT bands having an orientation of 45° (i.e., *HH* bands). Furthermore, DWT image and video coding, such as embedded zero-tree wavelet (EZW) coding, will be included in the upcoming image and video compression standards, such as JPEG2000 [112]. By embedding a watermark in the same domain (DWT domain) we can anticipate lossy EZW compression because we can anticipate which DWT bands is going to be affected by the compression scheme. Furthermore, we can exploit the DWT decomposition to make real-time watermark applications. Many approaches apply the basic techniques described at the beginning of this section to the high resolution DWT bands, LH_1, HH_1, and HL_1 (Fig. 17) [7], [12], [56], [84], [112].

In Fig. 18(a) an example is given of an image in which a 2-D CDMA watermark W is embedded in the LH_1, HH_1, and HL_1 DWT bands using a large gain factor k. The DWT coefficients in each of the three DWT bands are modulated as follows:

$$I_W(u,v) = I(u,v) + k \cdot W(u,v). \quad (14)$$

Figure 18(b) shows the strongly amplified difference between the original image and the watermarked image.

The DWT watermark can be made image dependent by modulating the DWT coefficients in each of the three DWT bands as follows:

$$I_W(u,v) = I(u,v) \cdot (1 + k \cdot W(u,v)). \quad (15)$$

In Fig. 19(a) an example is given of an image in which the same CDMA watermark W is embedded in the LH_1, HH_1, and HL_1 DWT bands using (15) with a large gain factor k. Figure 19(b) shows the strongly amplified difference between the original image and the watermarked image.

Watermark Energy Adaptation Based on HVS

The robustness of a watermark can be improved by increasing the energy of the watermark. Increasing the energy, however, degrades the image quality. By exploiting the properties of the HVS, the energy can be increased locally in places where the human eye will not notice it. As a result, by exploiting the HVS, one can embed perceptually invisible watermarks that have higher energy than if this energy were to be distributed evenly over the image.

If a visual signal is to be perceived, it must have a minimum amount of contrast, which depends on its mean luminance and frequency. Furthermore, a signal of a given frequency can mask a disturbing signal of a similar frequency [104], [6]. This masking effect is already used in the image-dependent DCT watermarking method described in the previous section, where the DCT-coefficients are modulated by means of (13). Here, to each sinusoid present in the image (masking signal), another sinusoid (watermark) is added, having an amplitude proportional to the masking signal. If the gain factor k is properly set, frequency masking occurs.

The HVS is less sensitive to changes in regions of high luminance. This fact can be exploited by making the watermark gain factor luminance dependent [58]. Furthermore, since the human eye is least sensitive to the blue channel, a perceptually invisible watermark embedded in the blue channel can contain more energy than a perceptually invisible watermark embedded in the luminance channel of a color image [58].

Around edges and in textured areas of an image, the HVS is less sensitive to distortions than in smooth areas. This effect is called spatial masking and can also be exploited for watermarking by increasing the watermark energy locally in these masked image areas [68]. The basic

spatial watermarking techniques described in the first two subsections of this section can be extended with spatial masking compensation, for instance, by using the following modulation function:

$$I_W(x,y) = I(x,y) + Msk(x,y) \cdot k \cdot W(x,y). \quad (16)$$

Here $W(x,y)$ represents the 2-D pseudorandom pattern of the watermark, k denotes the fixed gain factor, and $Msk(x,y)$ represents a masking image. The values of the masking image range from 0 to $k'_{\max}$ and give a measure of insensitivity to distortion for each corresponding point in the original image $I(x,y)$. In [53] the masking image Msk is generated by filtering the original image with a Laplacian high-pass filter and by taking the absolute values of the resulting filtered image.

In Fig. 20(a) a mask is shown for the Lena image [Fig. 13(a)] which is generated by a simple Prewitt edge detector [71]. Figure 20(b) shows the strongly amplified watermark modulated with this mask.

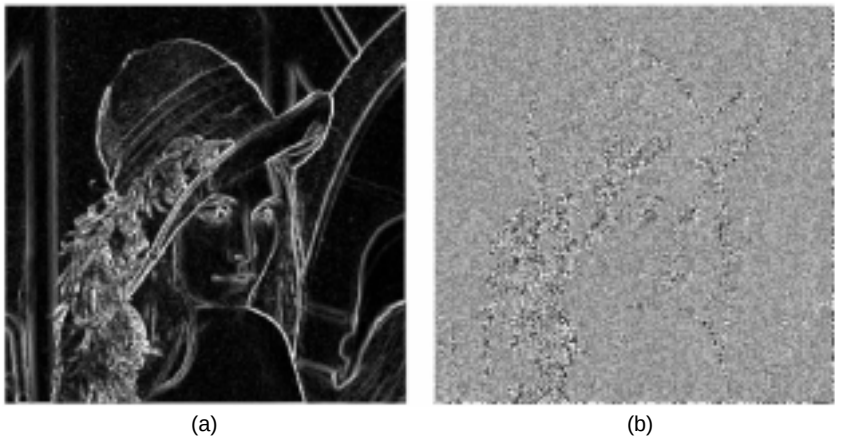
In [70] the squared sum of the 8×8 DCT AC-coefficients is used to generate a masking image. Figure 21(a) shows a mask generated using this DCT-ac energy for the Lena image. Figure 21(b) presents the strongly amplified watermark modulated with this mask.

Experiments have shown that a perceptually invisible watermark modulated with a gain factor locally adapted to such a mask can contain twice as much energy as a perceptually invisible watermark modulated with a fixed gain factor.

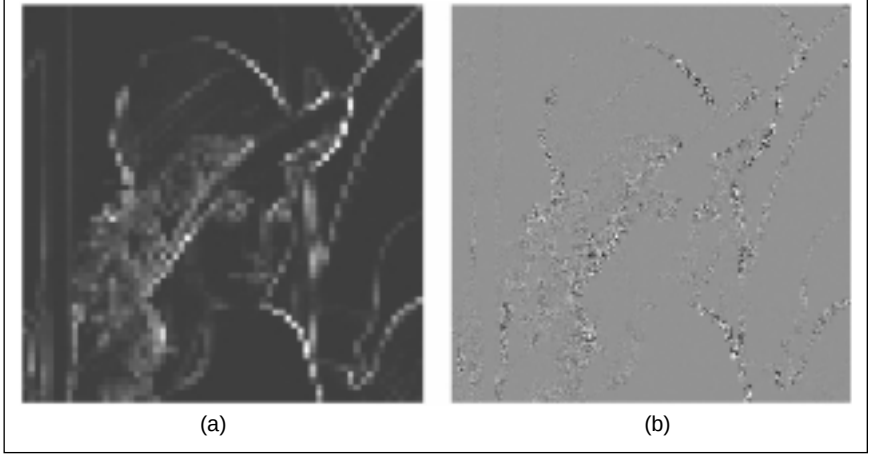
To investigate the effect of this energy doubling on the robustness of the watermark, we perform the following experiment. We add a watermark $W_{\text{fixed}}(x,y)$ to the Lena image with the "tiled" spread spectrum watermarking method described in [93] using a fixed gain factor $k=2$. Increasing this fixed gain factor causes visible artifacts in the resulting watermarked image. Next, we add a watermark $W_{\text{var}}(x,y)$ to another Lena image with the same method, but now we use a variable gain factor locally adapted to the masking image presented in Fig. 19(a). Although the watermark $W_{\text{var}}(x,y)$ contains about twice as much energy as $W_{\text{fixed}}(x,y)$ the watermark is not noticeable in the resulting watermarked image. Then we compress both watermarked images with the JPEG algorithm [73], where the quality factor Q_{jpeg} of the compression algorithm is made variable. Finally, the watermarks are extracted from the decompressed image and compared bit by bit with the originally embedded watermark bits. From this experiment, we find the percentages of watermark bit errors due to JPEG compression as a function of the JPEG quality factor. In Fig. 22 the error curves are plotted for both watermarks $W_{\text{fixed}}(x,y)$ and $W_{\text{var}}(x,y)$. It can be seen that the robustness can be slightly improved by applying a variable gain factor adapted to the HVS.

Spatial masking can also be applied if the watermark is embedded in another domain, e.g., DFT, DCT, or DWT. In this case, the nonspatial watermark is first embedded in an image I, resulting in the temporary image I_{Wt}. The watermarked image I_W is now constructed by mixing the original image I and this temporary image I_{Wt} by means of a masking image Msk as described above [6], [78]:

$$I_W(x,y) = (1 - Msk(x,y))I(x,y) + Msk(x,y) \cdot I_{Wt}(x,y). \quad (17)$$



▲ *20. Watermarking using masking image based on Prewitt operator. (a) Masking image and (b) difference $W(x,y) = I(x,y) - I_w(x,y)$.*



▲ *21. Watermarking where a masking image is used based on DCT-AC energy. (a) Masking image and (b) difference $W(x,y) = I(x,y) - I_w(x,y)$.*

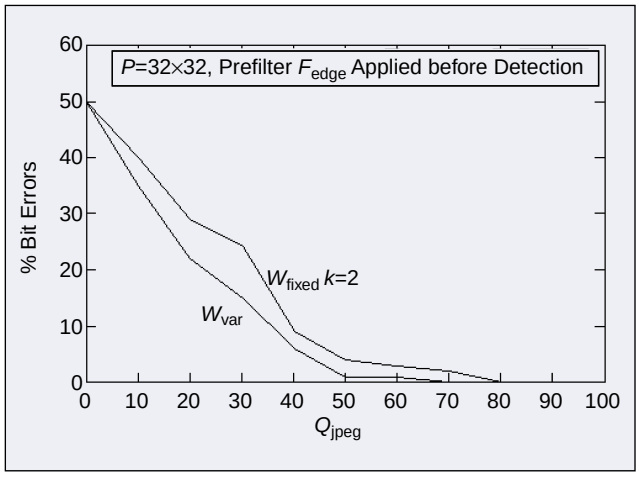
Here the masking image must be scaled to values in the range from zero to one. Watermarking methods based on more sophisticated models for the HVS can be found in [6], [7], [30], [34], [56], [78], [79], [94], [95], [109], and [110].

Extended Correlation-Based Watermarking Techniques

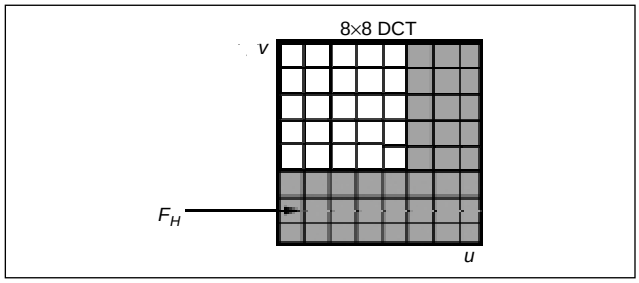
Anticipating Lossy Compression and Filtering

Watermarks that have been embedded in an image by means of the spatial watermarking techniques earlier cannot be detected reliably after the watermarked image has been highly compressed with the lossy JPEG compression algorithm. This is due to the fact that such watermarks consist essentially of low-power, high-frequency noise. Since JPEG allocates fewer bits to the higher frequency components, such watermarks can easily be distorted. Furthermore, these watermarks can also be affected severely by low-pass operations like linear or median filters.

The robustness to JPEG compression can be improved in several ways. In [93] the pseudorandom pattern W is first compressed and then decompressed using the JPEG algorithm. The energy of the resulting pattern W is increased to compensate for the energy lost through the compression. Finally, this pattern is added to the image to generate the watermarked image. The idea here is to use the compression algorithm to filter out in advance all the energy that would otherwise be lost later in the course of the compression. It is assumed that a watermark formed in this way is invariant to further JPEG compression that uses the same quality factor, except for small numerical artifacts. Other predistortion of the watermark pattern, such as filtering, can be applied to prevent other anticipated degradation of the watermarked image.



▲ *22. Influence of a variable gain factor adapted to the HVS on the robustness of a watermark.*



▲ *23. DCT bands F_H in which the watermark energy Φ is minimized.*

In [72] the energy of the watermark pattern is shifted to the lower frequencies by calculating an individual gain factor $k_{x,y}$ for each pixel of the watermark pattern instead of using the same gain factor k for all pixels. First a pseudorandom pattern $W(x,y)$ is generated consisting of the integers 0 and k. Next, the pattern is divided into 8×8 blocks, and the DCT transform $W(u,v)$ is calculated for each 8×8 block. The nonzero elements in the 8×8 blocks are now regarded as gain factors $k_{x,y}$ and are adapted in such a way that the energy Φ in the vulnerable high frequency DCT bands F_H is minimized (Fig. 23):

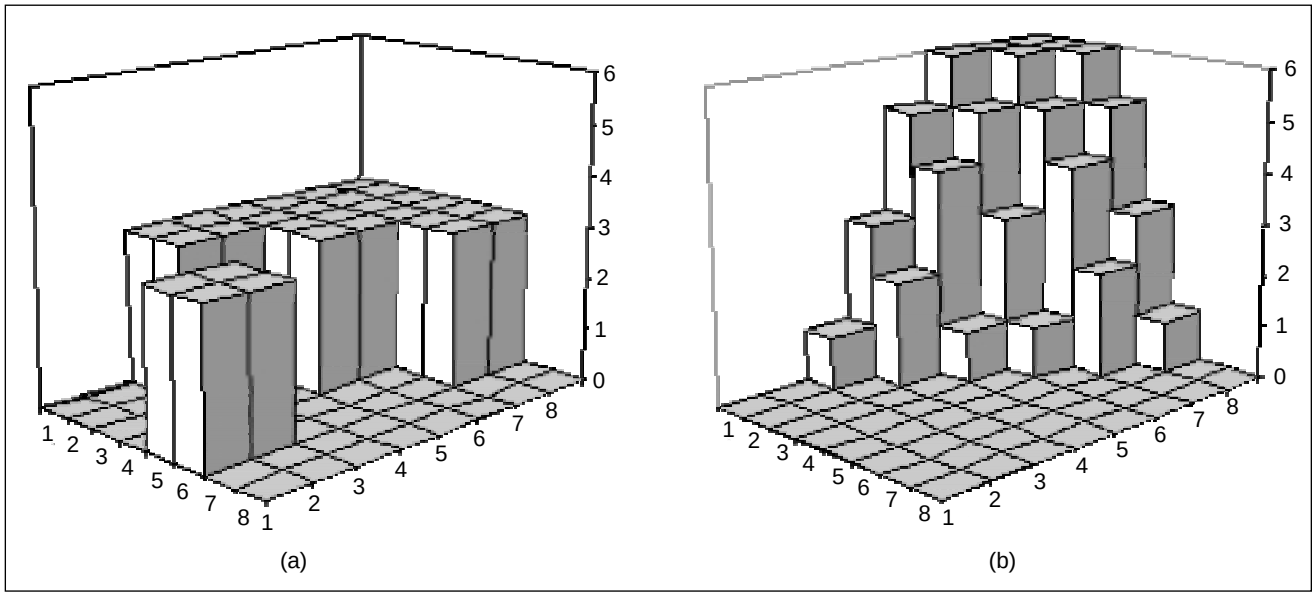
$$\Phi=\sum_{u,v\in F_H}\sum W(u,v)^2 \qquad F_H=\{u,v|5<u\le 8,5<v\le 8\}. \tag{18}$$

The energy Φ is minimized under the following constraints:

$$\sum_{x=1}^{8}\sum_{y=1}^{8}W(x,y)\cdot k=\sum_{x=1}^{8}\sum_{y=1}^{8}W(x,y)\cdot k_{x,y},\ k_{\min}\le k_{x,y}\le k_{\max}. \tag{19}$$

The effect of this high-energy minimization on the watermark pattern is illustrated in Fig. 24. Figure 24(a) shows the watermark pattern within an 8×8 block, where a constant gain factor of $k=3$ is used. After the high-energy minimization with $k_{\min}=0$ and $k_{\max}=6$, the watermark pattern fades smoothly to zero [Fig. 24(b)] although the sum of the nonzero pixels still equals the sum of the nonzero pixels in the original pattern.

In [35] and [61], JPEG compression immunity is obtained by deriving a different gain factor k for each 32×32 pixel block based on a lower quality JPEG compressed image. A 32×32 pseudorandom pattern representing a watermark bit is added to a 32×32 image tile. A copy of this watermarked image tile is degraded according to the JPEG standard for which end a relatively low quality factor is used. If the watermark bit cannot be extracted correctly from this degraded copy, the watermark pattern is added to the image by means of a higher gain factor and a new degraded copy is formed to check the bit. This procedure is repeated iteratively for each bit until all bits can be extracted reliably from the degraded copies. A watermark formed in this way is resistant to JPEG compression using a quality factor equal to or greater than the quality factor used to degrade the copies. In Fig. 25 an example of such a watermark is shown, amplified for visibility purposes.



▲ *24. (a) Original watermark block and (b) low frequency watermark block.*

Anticipating Geometrical Transforms

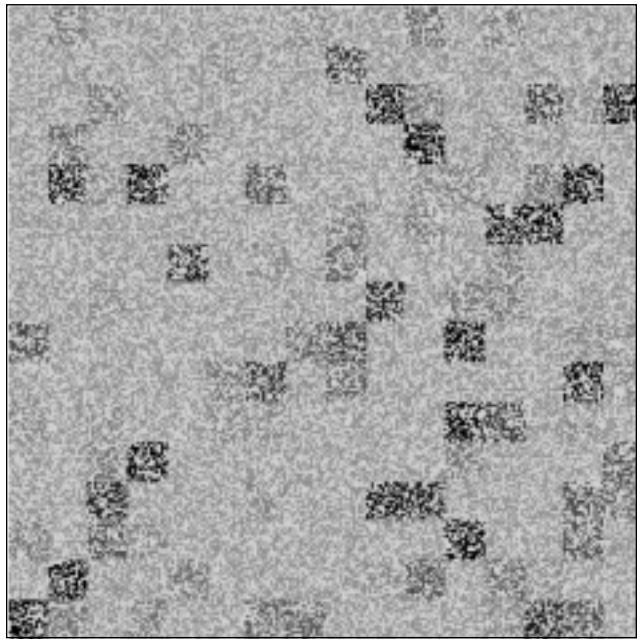
A watermark should not only be robust to lossy compression techniques, but also to geometrical transformations such as shifting, scaling, cropping, rotation, etc. Geometrical transforms hardly affect the image quality, but they do make most of the watermarks that have been embedded by means of the techniques described in the previous sections undetectable for the watermark detectors. Since geometrical transforms typically affect the synchronization between the pseudorandom pattern of the watermark and the watermarked image, the synchronization must be retrieved before the detector performs the correlation calculations.

The most obvious way to achieve shift invariance is using the DFT amplitude modulation technique. If, for some reason, another watermarking embedding domain is preferred and shift invariance is required, a marker can be added in the spatial domain to determine the translation. This marker can be a pseudorandom pattern like the watermark itself. The detector first determines the spatial position of this marker by shifting the marker over all possible locations in the image and calculating the correlation between the marker and the corresponding image part. The translation with the highest correlation defines the spatial position of the marker. Finally, the image is shifted back to its original position and the normal watermarking detection procedure is applied.

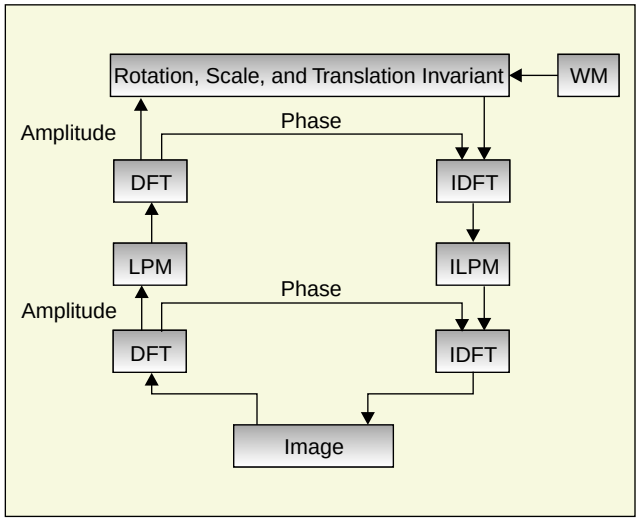
An exhaustive search for a marker is computationally quite demanding. Therefore, in [53] a different approach is proposed: adding a pseudorandom pattern twice, but at different locations in the image. The content of the watermark, i.e., the watermark bits, is embedded here in the relative positions of the two watermark patterns. To detect the watermark, the detector computes the phase correlation between the image and the watermark pattern using the fast Fourier transform (FFT) and it detects the two correlation peaks of the two patterns. The content of the watermark is derived from relative position of the peaks. If the whole image is shifted before detection, the absolute positions of the correlation peaks will change, but the relative positions will remain unchanged, leaving the watermark bits readable for the detector.

In [30] a method is proposed to add a grid to an image that can be used to scale, rotate, and shift an image back to its original size and orientation. The grid is represented by a sum of sinusoidal signals, which appear as peaks in the FFT frequency domain. These peaks are used to determine the geometrical distortion.

In [59] a method is proposed which embeds a pseudorandom pattern multiple times at different loca-



▲ *25. Watermark where the local gain factor per block is based on a lower quality image.*



▲ *26. Rotation, scale, and translation invariant watermarking scheme.*

tions in the spatial domain of an image. The detector estimates the watermark W' by applying a high pass filter F_{HP} to the watermarked image

$$W' = I_W \otimes F_{HP}$$

$$F_{HP} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & -1 & -1 & 12 & -1 & -1 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} / 12. \tag{20}$$

Next, the autocorrelation function of the estimated watermark W' is calculated. This function will have peak values at the center and the positions of the multiple embedded watermarks. If the image has undergone a geometrical transformation, the peaks in the autocorrelation function will reflect the same transformation and hence provide a grid that can be used to transform the image back to its original size and orientation.

In [40], [41], [86], [74], [87], and [88] a method is proposed that embeds the watermark in a rotation, scale, and translation invariant domain using a combination of DFT and a log polar map (LPM). Figure 26 presents a scheme of this watermarking method.

First the amplitude of the DFT is calculated to obtain a translation invariant domain. Next, for every point (u, v) of the DFT amplitude a corresponding point in the LPM (μ, θ) is determined:

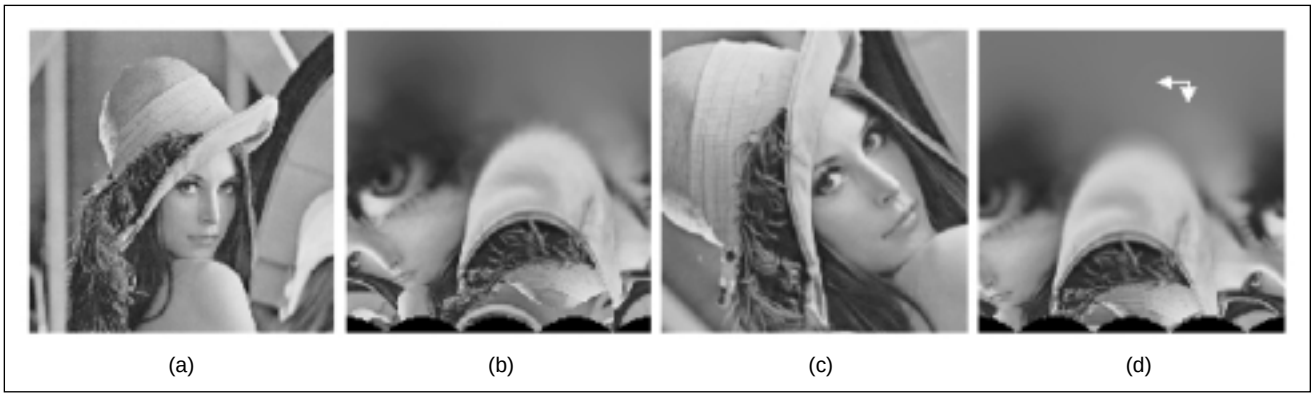
$$u = e^{\mu} \cos(\theta) \qquad v = e^{\mu} \sin(\theta). \tag{21}$$

This coordinate system of the LPM converts rotation and scaling into translation along the horizontal and vertical axis. By taking the amplitude of the DFT of this LPM, we obtain a rotation, scale, and translation invariant domain. In this domain a CDMA watermark can be added, for instance by modulating the coefficients using (10).

Figure 27 demonstrates an example of the properties of the LPM. Part (b) shows the LPM of the Lena image (a). Part (c) depicts a rotated and scaled version of the Lena image, and (d) shows its corresponding LPM. It can clearly be seen that the rotation and scaling in the original spatial domain are converted into translations in the LPM domain.

In practice implementing the watermarking scheme illustrated in Fig. 26 has been proven to be difficult. The authors therefore propose a different approach, where a CDMA watermark is embedded in the translation invariant amplitude DFT domain. To make the watermark scale and rotation invariant, they embed a second watermark, a template, in this domain. To extract the watermark, they first determine the scale and orientation of the watermarked image by using the template in the following way:

- The DFT of the watermarked image is calculated.
- The LPM of the DFT amplitudes and the template pattern is calculated.
- The horizontal and vertical offsets between the two LPMs are calculated using exhaustive search and cross-correlation techniques, resulting in a scale and rotation factor.



▲ *27. Example of the properties of the LPM. (a) Original image, (b) LPM of (a), (c) scaled and rotated, and (d) LPM of (c).*

Next, the image is transformed back to its original size and orientation, and the information-carrying watermark is extracted.

Correlation-Based Watermarking Techniques for MPEG

In real-time watermarking applications, robustness is not the only factor that plays an important role. The other factor that plays a very important role is computational complexity. In general, image or video data is transmitted in JPEG or MPEG compressed form. Real-time watermark embedding must take into account this compressed form, because first decompressing the data, adding a watermark and then recompressing the data is computationally too demanding. Therefore, it is desirable to develop watermarking techniques that can operate directly on the compressed bit stream, the code words, or the DCT transformed coefficients because then it is not necessary to fully decompress and recompress the data. In this section we discuss two such methods for MPEG video streams. Other methods that also operate on code words and DCT coefficients are discussed in upcoming sections.

In [111] a method is proposed that adds a DCT transformed pseudorandom pattern directly to the DC-DCT coefficients of an MPEG compressed video stream. The watermarking process only takes the luminance values of the I-frames into account. To embed a watermark the following procedure is performed: First a pseudorandom pattern consisting of the integers {–1,1} is generated based on a secret a key. This pattern has the same dimensions as the I-frames. Next, the pattern is modulated by a watermark bit string and multiplied by a gain factor. Finally, the 8×8 block DCT transform is applied on the modulated pattern and the resulting DC-coefficients are added to the corresponding DC-values of each I-frame. The watermark can be detected using correlation techniques in the DCT domain or in the spatial domain as described earlier.

The authors report that the algorithm decreases the visual quality of the video stream drastically. Therefore, the gain factor of the watermark has to be chosen to be very low (<1) and the number of pixels per watermark bit has to be chosen to be extremely high ($>>$ 100,000) to maintain reasonable visual quality for the resulting video stream. This is mainly due to the fact that the watermark pattern is embedded in just one of the 64 DCT coefficients, the DC-component. Furthermore, the pattern consists only of low frequency components to which the human eye is quite sensitive. For comparison, the algorithm used to embed multiple bits using the correlation technique described earlier uses a gain factor of two and about 1000 pixels per watermark bit.

In [36]-[39] and [115] a more sophisticated watermarking algorithm is proposed that embeds a watermark not only in the DC-coefficients, but also in the AC-coefficients of each I-, P-, and B-frame. The watermark here is also a pseudorandom pattern consisting of the integers {–1,1} generated based on a secret key. This pattern has the same dimensions as the video frames. The pattern is modulated by a watermark bit string and multiplied by a gain factor k.

To embed the watermark, the watermark pattern $W(x,y)$ is divided into 8×8 blocks. These blocks are transformed to the DCT domain and denoted by $W_{x,y}(u,v)$, where $x,y = 0,8,16,...$ and $u,v = 0,...,7$. Next, the 2-D blocks $W_{x,y}(u,v)$ are reordered in a zig-zag scan fashion and become arrays $W_{x,y}(i)$, where $i = 0,...,63$. $W_{x,y}(0)$ represents the DC-coefficient and $W_{x,y}(63)$ denotes the highest frequency AC-coefficient of a 8×8 watermark block. Since the corresponding MPEG encoded 8×8 video content blocks are encoded in the same way as $I_{x,y}(i)$, these arrays can directly be used to add the watermark. For each video block $I_{x,y}(i)$ out of an I-, P-, or B-frame the following steps are performed:

1. The DC-coefficient is modulated as follows:

$$I_{W_{x,y}}(0) = I_{x,y}(0) + W_{x,y}(0) \tag{22}$$

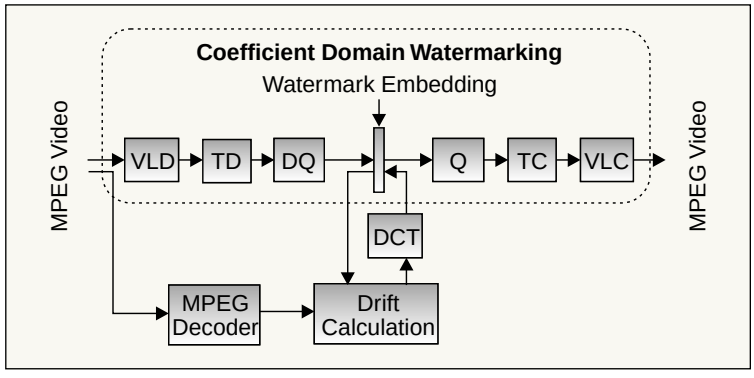
which means that the average value of the watermark block is added to the average value of the video block.

2. To modulate the AC-coefficients the bit stream of the encoded video block is searched VLC-by-VLC for the next VLC code word, representing the next nonzero DCT coefficient. The run and level of this code word are decoded to determine its position i along the zig-zag scan and its amplitude $I_{x,y}(i)$.

A candidate DCT coefficient for the watermarked video block is generated, which is defined as

$$I_{W_{x,y}}(i) = I_{x,y}(i) + W_{x,y}(i), \qquad i \neq 0. \tag{23}$$

Now the constraint that the video bit rate may not be increased comes into play. The size Sz_I of the VLC needed to encode $I_{x,y}(i)$ and the size Sz_{I_W} of the VLC needed to encode $I_{W_{x,y}}(i)$ are determined using the VLC-Tables B.14 and B.15 of the MPEG-2 standard [47]. If the size of VLC encoding the candidate DCT coefficient is equal or smaller than the size of the existing VLC, the existing VLC is replaced. Otherwise the VLC is



▲ *28. Increase of complexity due to drift compensation.*

The robustness of a watermark can be improved by increasing the energy of the watermark.

left unaffected. This means that the DCT coefficient $I_{x,y}(i)$ is modulated in the following way:

$$\begin{aligned} &\text{If } Sz_{I_W} \le Sz_I \quad \text{then} \quad I_{W_{x,y}}(i) = I_{x,y}(i) + W_{x,y}(i) \\ &\qquad\qquad\quad \text{else} \quad I_{W_{x,y}}(i) = I_{x,y}(i). \end{aligned} \tag{24}$$

This procedure is repeated until all AC-coefficients of the encoded video block are processed.

To extract the watermark information, the MPEG encoded video stream is first fully decoded and the watermark bits are retrieved by correlating the decoded frames with the watermark pattern $W(x,y)$ in the spatial domain using the standard techniques.

A major problem of directly modifying DCT-coefficients in an MPEG encoded video stream is drift or error accumulation. In an MPEG encoded video stream predictions from previous frames are used to reconstruct the actual frame, which itself may serve as a reference for future predictions. The degradation caused by the watermarking process may propagate in time and may even spatially spread. Since all video frames are watermarked, watermarks from previous frames and from the current frame may accumulate and result in visual artifacts. Therefore, a drift compensation signal *Dr* must be added. This signal must be equal to the difference of the (motion compensated) predictions from the unwatermarked bit stream and the watermarked bit stream. Equation (23) changes for a drift compensated watermarking scheme into

$$I_{W_{x,y}}(i) = I_{x,y}(i) + W_{x,y}(i) + Dr_{x,y}(i). \tag{25}$$

A disadvantage of this drift signal is that the complexity of the watermark embedding algorithm increases substantially, since an additional DCT operation and a complete MPEG decoding step are required to calculate the drift compensation signal. The increase in complexity compared to the coefficient domain methods is illustrated in Fig. 28.

Due to the bit-rate constraint, only around 10-20% of the DCT coefficients are altered by the watermark embedding process, depending on the video content and the coarseness of the MPEG quantizer. In some cases, especially for very low bit-rate video, only the DC-coefficients are modified. This means that only a fraction of the watermark pattern $W(x,y)$ can be embedded, typically around 0.5 ... 3% [115]. Since only existing (nonzero) DCT coefficients of the video stream are watermarked, the embedded watermark is video content dependent. In areas with only low-frequency content, the watermark automatically consists of only low frequency components. This complies with the HVS. The watermark energy is mainly embedded in areas containing a lot of video content energy.

The authors in [115] report that the complexity of the watermark embedding process is much lower than the complexity of a decoding process followed by watermarking in the spatial domain and re-encoding. The complexity is somewhat higher than the complexity of a full MPEG decoding operation. Typical parameter settings for the embedding are $k = 1,\ldots,5$ for the gain factor of the watermark and $P = 500{,}000,\ldots,1{,}000{,}000$ for the number of pixels per watermark bit, yielding watermark label bit rates of only a few bytes per second. The authors claim that the watermark is not visible, except in direct comparison to the unwatermarked video, and that the watermark is robust against linear and nonlinear operations like filtering, noise addition and quantization in the spatial or frequency domain.

Noncorrelation-Based Watermarking Techniques

Least Significant Bit Modification

The simplest example of a spatial domain watermarking technique that is not based on correlation is the LSB modification method. If each pixel in a gray level image is represented by an 8-bit value, the image can be sliced up in eight bit planes. In Fig. 29 these eight bit planes are represented for the Lena image, where the upper left image represents the most significant bit plane and the lower right image represents the LSB plane.

Since the least significant bit plane does not contain visually significant information, it can easily be replaced by an enormous amount of watermark bits. More sophisticated watermarking algorithms that make use of LSB modifications can be found in [91], [4], [5], [43], and [33]. These watermarking techniques are not very secure and not very robust to processing techniques because the LSB plane can easily be replaced by random bits, effectively removing the watermark bits.

MPEG Video Watermarking by Parity Bit Modification

In a compressed bit stream we have direct access to the code words used in the compression algorithm. Similar to the LSB technique described above, we can embed watermark in the stream by modifying these code words, yielding a computationally efficient watermarking method with a high payload [62], [35].

The technique is described as follows. A watermark consisting of l label bits b_j $(j = 0,1,2,\ldots,l-1)$ is embedded in the MPEG-stream by selecting suitable VLCs and forcing the LSB of their *quantized* level to the value of b_j. To ensure that the change in the VLC is perceptually invisible after decoding and that the MPEG-bit stream keeps its original size, we select only those VLCs for which another VLC exists with:

- the same run length,

- a quantized level difference of one,
- the same code word length.

A VLC that meets this requirement is called a label-bit-carrying-VLC (lc-VLC). According to Tables B.14 and B.15 of the MPEG-2 standard [47], an abundance of such lc-VLCs exists. Furthermore, all fixed-length-coded DCT-coefficients following an Escape-code meet the requirement. Some examples of lc-VLCs are listed in Table 1, where the symbol s represents the sign-bit. This sign-bit represents the sign of the DCT coefficient level.

The VLCs in the intra- and intercoded macro blocks can be used in the watermarking process. The DC coefficients are not used, because they are predicted from other DC coefficients and coded with a different set of VLCs and Escape-codes. Furthermore, replacing each DC coefficient in intra- and intercoded frames can result in visible artifacts due to drift. By only taking the AC coefficients into account the watermark will adapt itself more to the video content and the drift will be limited.

To add the label bit stream L to an MPEG-video bit stream, the VLCs in each macro block are tested. If an lc-VLC is found and the LSB of its level is unequal to the label bit b_j ($j = 0,1,2,...,l-1$), this VLC is replaced by another one, whose LSB-level represents the label bit. If the LSB of its level equals the label bit b_j the VLC is not changed. The procedure is repeated until all label bits are embedded. In Fig. 30 an example is given of the watermarking process, where three label bits are embedded in the MPEG video stream.

To extract the label bit stream L the VLCs in each macro blocks are tested. If an lc-VLC is found, the value represented by its LSB is assigned to the label bit b_j. The procedure is repeated for $j = 0,1,2,...,l-1$ until no lc-VLCs can be found anymore.

This technique gives a high payload (up to 29 kbit/s) without significant perceptible quality degradation [65]. The watermark embedded with this method can easily be removed by decoding and reencoding the video stream or by relabeling the stream using another randomly generated watermark pattern. This technique can be extended to make it resistant to relabeling [65], as follows. The watermark label bits b_i are now not directly stored in the LSBs of the VLCs, but a one-dimensional pseudorandom watermark pattern $W(x)$ is generated consisting of the integers $\{-1,1\}$ based on a secret key, which is modulated with the label bits b_i. The procedure to add this modulated pattern to the video stream is similar to the procedure described above.

However, we now select only those VLCs for which two other VLCs exist, with the same run length and the same codeword length. One VLC must have a level difference of $+\delta$ and the other VLC must have a level difference of $-\delta$. Most lc-VLCs meet these requirements for a relative small δ (e.g., $\delta = 1,2,3$). For notational simplicity we call these pattern-carrying-VLCs (*pc-VLCs*).

To embed a watermark in a video stream, we add the modulated watermark pattern to the levels of the *pc-VLCs*. To extract the watermark, we collect the *pc-VLCs* in an array. The watermark label bits can now be retrieved by calculating the correlation between this array of *pc-VLCs* and the secret watermark pattern $W(x)$. In Fig. 31 an example is given of the watermark embedding process. About 1,000,...,10,000 *pc-VLCs* are now required to encode one watermark label bit b_i and thus drastically reduce the payload of the watermark. However, several watermark label bit strings can be added without interfering with each other, if independent pseudorandom patterns are used to form the basic pattern $W(x)$.

▲ *29. Bit planes for the Lena image.*

DCT Coefficient Ordering

In [55], [114], [54], and [17] a watermarking method is proposed that adds a watermark bit string in the 8×8 block DCT domain. To watermark an image, the image is divided into 8×8 blocks. From these 8×8 blocks the DCT transform is calculated and two or three DCT coefficients are selected in each block in the middle band frequencies F_M (Fig. 32). The selected coefficients are quantized using the default JPEG quantization table [73] and a relatively low JPEG quality factor. The selected coefficients are then adapted in such a way that their magnitudes form a certain relationship. The relationships among the selected coefficients compose eight patterns (combinations), which are divided into three groups. Two groups are used to represent the watermark bits "1" or "0," and the third group represents invalid patterns. If the modifications which are needed to hold a desired pattern become too large, the block is marked as invalid. For example, if a watermark bit with value 1 must be embedded in a block, the third coefficient should have a lower value than the two other coefficients. The embedding process and the list of patterns are represented in Fig. 32.

In Fig. 33 the heavily amplified difference between the original Lena image and the watermarked version is shown. In [13] and [14] a similar watermarking method is proposed, but here the DCT coefficients are modified in such a way that they fulfill a linear or circular constraint imposed by the watermark code.

We note that the techniques described above are similar to the DEW method for real-time MPEG video watermarking described in the next section.

MPEG Video Watermarking Using the DEW Algorithm

The DEW method is based on *selectively discarding* high frequency DCT coefficients in the compressed data stream. The information bits of the data identifier (label) are encoded in the pattern of DCT blocks in which high frequency DCT coefficients are removed, i.e., in a pattern of energy differences between DCT blocks. For this reason, the technique is called a differential energy watermark (DEW).

The technique is described as follows. The information that we wish to embed into the image or video frame is represented by the label bit string L consisting of label bits $L_j (j = 0,2,...,l-1)$. This label bit string is embedded bit-by-bit in a set of n 8×8 DCT blocks taken from a JPEG compressed still image or from an I-frame of an MPEG compressed video stream. For the purpose of simplicity of the discussion, we will refer to still images and MPEG I-frames as "image."

To obtain sufficient robustness, typically n takes on values between 16 and 64, which means that a single label bit is embedded in a *region* of the image. Before the label bits are embedded, however, the positions of the 8×8 DCT blocks in the image are shuffled randomly as illustrated in Fig. 34. This shuffling operation, on the one hand, forms the secret key of the labeling algorithm, while on the other hand it spatially randomizes the statistics of DCT blocks.

Each bit of the label bit string is embedded in its private label bit-carrying-region, or *lc-region* for short, in a shuffled image. For instance, in Fig. 33 the first bit is located in the top-left-corner of the image in an lc-region of $n = 16$ DCT blocks. The value of the label bit is encoded by introducing an energy difference between the high frequency DCT-coefficients of the top half of the lc-region (denoted by *lc-subregion A*) containing in this case $n/2 = 8$ DCT blocks, and the bottom half (denoted by *lc-subregion B*) also containing $n/2 = 8$ DCT blocks. If the *lc-subregion A* contains more high frequency energy than the *lc-subregion B*, the label bit value 0 has been embedded into the data, and vice versa.

To make the determination of "high frequency" energy easy for images or video frames that are JPEG or MPEG compressed, we compute energies over a subset of zigzag scanned DCT-coefficients indicated by $S(c)$

$$S(c) = \{ i \in \{0,63\} \,|\, (i > c) \}. \tag{26}$$

The zigzag scanned DCT coefficients are numbered according to Fig. 35. The index $i = 0$ refers to the DC-coefficient of a DCT block. The subset of DCT coefficients $S(c)$ over which energies are computed is defined by the *cut-off index c*. The selection of a suitable cut-off index c for an lc-region is essential for the robustness and the visibility of the label bit. The larger the cut-off index is chosen, the less degradation the label embedding will introduce. Here we assume that we have available a suitable cut-off in-

Table 1. Example of lc-VLCs in Table B.14 of the MPEG-2 Standard.

Variable Length Code	VLC size	Run	Level	LSB of Level
0010 0110 s	8 + 1	0	5	1
0010 0001 s	8 + 1	0	6	0
0000 0001 1101 s	12 + 1	0	8	0
0000 0001 1000 s	12 + 1	0	9	1
0000 0000 1101 0 s	13 + 1	0	12	0
0000 0000 1100 1 s	13 + 1	0	13	1
0000 0000 0111 11 s	14 + 1	0	16	0
0000 0000 0111 10 s	14 + 1	0	17	1
0000 0000 0011 101 s	15 + 1	1	10	0
0000 0000 0011 100 s	15 + 1	1	11	1
0000 0000 0001 0011 s	16 + 1	1	15	1
0000 0000 0001 0010 s	16 + 1	1	16	0

dex c for each lc-region [66]. Note that different lc-regions may have different cut-off indexes depending on their spatial contents.

The (DCT high frequency) *energy* E_A in lc-subregion A is now defined as follows:

$$E_A\left(c,n,Q_{\text{jpeg}}\right)=\sum_{b=0}^{n/2-1}\sum_{i\in S(c)}\left([\theta_{i,b}]_{Q_{\text{jpeg}}}\right)^2. \tag{27}$$

Here $\theta_{i,b}$ denotes the non-weighted DCT coefficient with index i in the bth DCT block of the lc-subregion A under consideration. Prior to the calculation of E_A, the notation $[]_{Q_{\text{jpeg}}}$ indicates that, the DCT-coefficients are re- or prequantized, in our case using the standard JPEG quantization procedure [73] with quality factor Q_{jpeg}. For embedding label bits into MPEG compressed I-frames a similar approach can be followed, but here, we confine ourselves to the JPEG notation without loss of generality. The prequantization is done only in determining the cut-off indexes and the calculation of (26), but is *not* applied to the actual image data upon embedding the label. The energy in lc-subregion B, denoted by E_B, is defined similarly.

We now define the energy difference D between the lc-subregions A and B as follows:

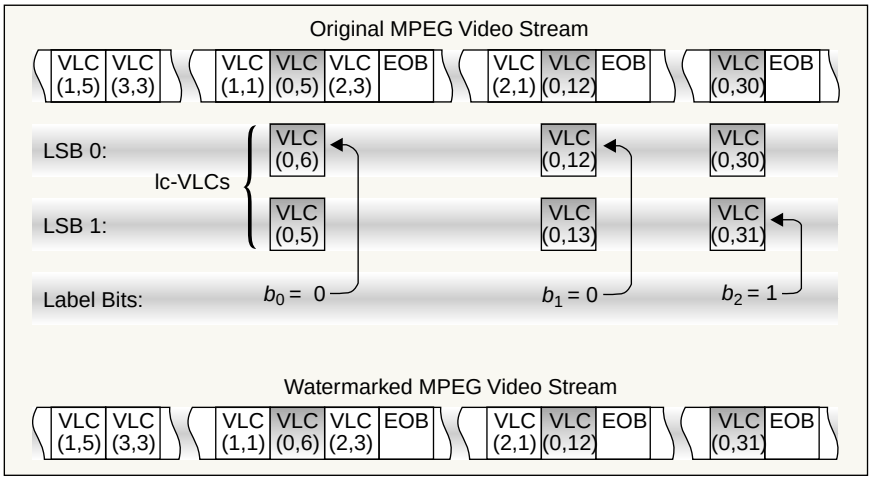
$$D\left(c,n,Q_{\text{jpeg}}\right)=E_A\left(c,n,Q_{\text{jpeg}}\right)-E_B\left(c,n,Q_{\text{jpeg}}\right). \tag{28}$$

The value of a label bit is encoded as the sign of the energy difference D. Label bit 0 is defined as $D>0$ and label bit 1 as $D<0$. The label embedding procedure must therefore adapt E_A and E_B to manipulate the energy difference D. If label bit 0 must be embedded, all energy after the cut-off index c in the DCT-blocks of lc-subregion B is eliminated by setting the corresponding DCT-coefficients to zero, yielding

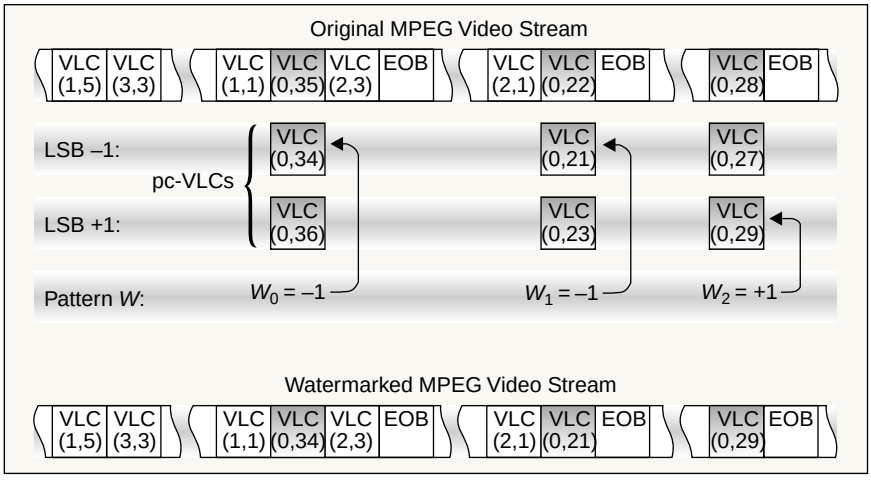
$$D=E_A-E_B=E_A-0=+\boldsymbol{E}_A. \tag{29}$$

If label bit 1 must be embedded, all energy after the cut-off c index in the DCT-blocks of lc-subregion A is eliminated, yielding $D=-E_B$. Since the watermark is embedded in the compressed bit stream, the DCT coefficients can easily be forced to zero without re-encoding the bit stream by shifting the *end of block marker* (EOB) of 8 × 8 DCT blocks in one of the two lc-subregions towards the DC-coefficient, up to the selected cut-off index.

In Fig. 35 the complete procedure to calculate the energy difference D in an lc-region is illustrated for $n=16$ nonshuffled 8 × 8 DCT blocks. The white triangularly shaped areas illustrate the subsets over which the energies are calculated for a particular choice of the cut-off index $c=27$. At the right a blow-up of one 8 × 8 DCT block is presented. In Fig. 34(c), the difference between the original and watermarked image is shown, illustrating that the DEW algorithm embeds information bits in those regions of the image that contain many details. Because of the prequantization with (JPEG) quality Q_{jpeg} in the calculation of the energy of the (high-frequency) DCT coefficients in (26), the DEW algorithm effectively embeds the label bits in perceptually important image details that are not significantly affected by JPEG/MPEG compression. Consequently, removing the DEW watermark is not possible without strongly affecting the perceptual image quality.



▲ *30. Example of the LSB watermarking process. The (x, y) pairs represent the (zero run, level) pairs used in the MPEG VLC encoding.*

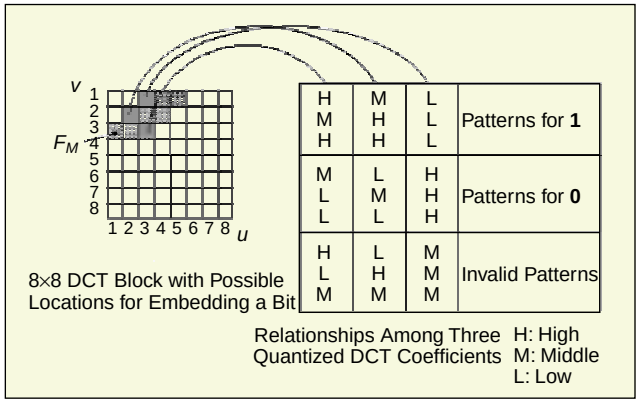


▲ *31. Example of the relabeling resistant watermarking method.*

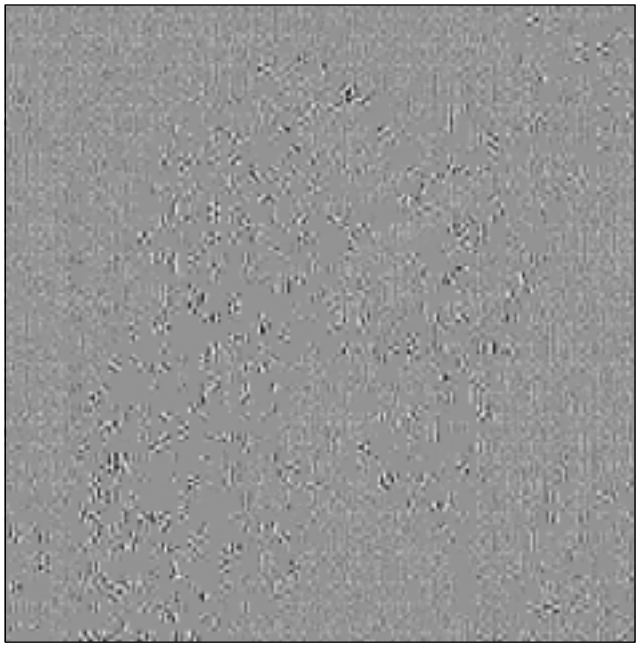
Salient-Point Modification

In [82] a watermarking method is proposed that is based on the modification of salient points in an image. Salient points are defined as isolated points in an image for which a given saliency function is maximal. These points could be corners in an image or locations of high energy, for example.

To embed a watermark we extract the set of pixels with highest saliency S from the image. Next, a binary pseudorandom pattern $W(x,y)$ with the same dimensions as the image is generated. This can be a line or block pattern as represented in Fig. 36. If this pattern is sufficiently random and covers 50% of all the image pixels, 50% of all salient points in set S will be located on the pattern and 50% off the pattern $W(x,y)$. Finally, the salient points in set S are adapted in such a way that a statistically significant high percentage of them lies on the watermark pattern (i.e., the black pixels in the pattern). There are two ways to adapt the salient points:



▲ *32. Watermarking based on adapting relationship between three coefficients.*



▲ *33. Watermark $W(x,y) = I(x,y) - IW(x,y)$ created by adapting relationships between DCT coefficients.*

▲ The location of the salient points can be changed by warping the points towards the watermark pattern. In this case small, local geometrical changes are introduced in the image.

▲ The saliency of the points can be decreased or increased by adding well-chosen pixel patterns to the neighborhood of a salient point.

To detect the watermark we extract the set of pixels with highest saliency S from the image and compare the percentages of the salient points on the watermark pattern and off the pattern. If both percentages are about 50% no watermark is detected. If there is a statistically significant high percentage of salient points on the pattern, the watermark is detected. The payload of this watermark is 1 bit.

Fractal-Based Watermarking

Several watermark embedding algorithms based on fractal compression techniques have been proposed [24], [80], [8], [9]. They mainly use block-based local iterated function system coding [49]. We first briefly describe the basic principles of this fractal compression algorithm here. An image is partitioned at two different resolution levels. On the first level, the image is partitioned in range blocks of size $n \times n$. On the second level the image is partitioned in domain blocks of size $2n \times 2n$. For each range block, a transformed domain block is searched for which the mean square error between the two blocks is minimal. Before the range blocks are matched on the domain blocks, the following transformations are performed on the domain blocks.

First, the domain blocks are subsampled by a factor of two to get the same dimensions as the range blocks. Subsequently, the eight isometries of the domain blocks are determined (the original block and its mirrored version rotated over 0, 90, 180, and 270°). Finally, the scale factor and the offset for the luminance values is adapted. The image is now completely described by a set of relations for each range block, by the index number of the best fitting domain block, its orientation, the luminance scaling, and the luminance offset. Using this set of relations, an image decoder can reconstruct the image by taking any initial random image and calculating the content of each range block from its associated domain block using the appropriate geometric and luminance transformations. Taking the resulting image as initial image one repeats this process iteratively until the original image content is approximated closely enough.

In [80] a watermarking technique is proposed which embeds a watermark of 32 bits $b_0 b_1 \cdots b_{l-1}$ in an image. The embedding procedure consists of the full fractal encoding and decoding process as described above, where the watermark embedding takes place in the fractal encoding process. First, the image $I(x,y)$ is split in two regions $A(x,y)$ and $B(x,y)$. For each watermark bit b_j U range blocks are pseudorandomly chosen from $I(x,y)$. If

b_j equals one, the domain blocks to code the U range blocks are searched in region $A(x, y)$. If b_j equals zero, the domain blocks to code the U range blocks are searched in region $B(x, y)$. For range blocks which are not involved in the embedding process, domain blocks are searched in regions $A(x, y)$ and $B(x, y)$. To extract the watermark information, we must select and re-encode the U range blocks for each bit b_j. If most of the best fitting domain blocks are found in region $A(x, y)$, the value 1 is assigned to bit b_j, otherwise the bit is assumed to be zero.

In [8] and [9] a watermark is embedded by forcing range blocks to map exactly on specific domain blocks. The watermark pattern here consists of this specific mapping. This mapping is enforced by adding artificial local similarities to the image. The size of the range blocks may be chosen to be equal to the size of the domain blocks. In Fig. 37 an example is given of this process.

The left image illustrates how a fractal encoder would map the range block Rb_{18} on domain block Db_0 in an unwatermarked image. To embed the watermark, this mapping $Db_0 \rightarrow Rb_{18}$ must for instance be changed to $Db_0 \rightarrow Rb_{21}$. To force the mapping to this form, a block Rb'_{21} is generated from block Db_0 by changing its luminance values. By adding block Rb' to the image, we change the optimal fractal mapping to its desired form $Db_0 \rightarrow Rb_{21}$, because the quadratic error between Db_0, corrected for luminance scale and offset and Rb_{21} is now smaller than the error between Db_0 and Rb_{18}.

To detect the watermark we calculate the optimal fractal mapping between the range blocks and the domain blocks. If a statistically significant high percentage of the mappings between range blocks and domain blocks match the predefined mappings of the watermark pattern, the watermark is detected.

Concluding Remarks

This article has given a state-of-the-art overview of common watermarking techniques. New watermarking techniques are invented regularly. Some of the watermarking techniques are designed for specific applications, while the others are not well established yet but have a great potential. For the purpose of completeness we briefly list the principles of these watermarking techniques below:

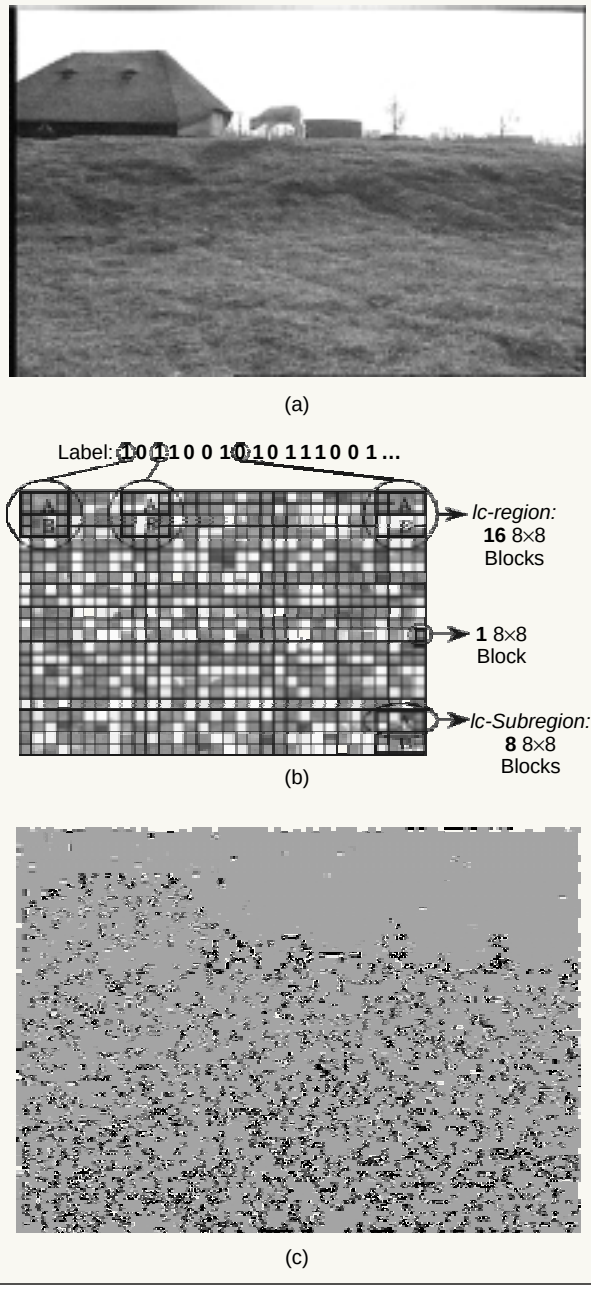
▲ For printed images dithering patterns can be adapted to hide watermark information [98], [19].

▲ Instead of the pixel values, the histogram of an image can be modified to embed a watermark [116].

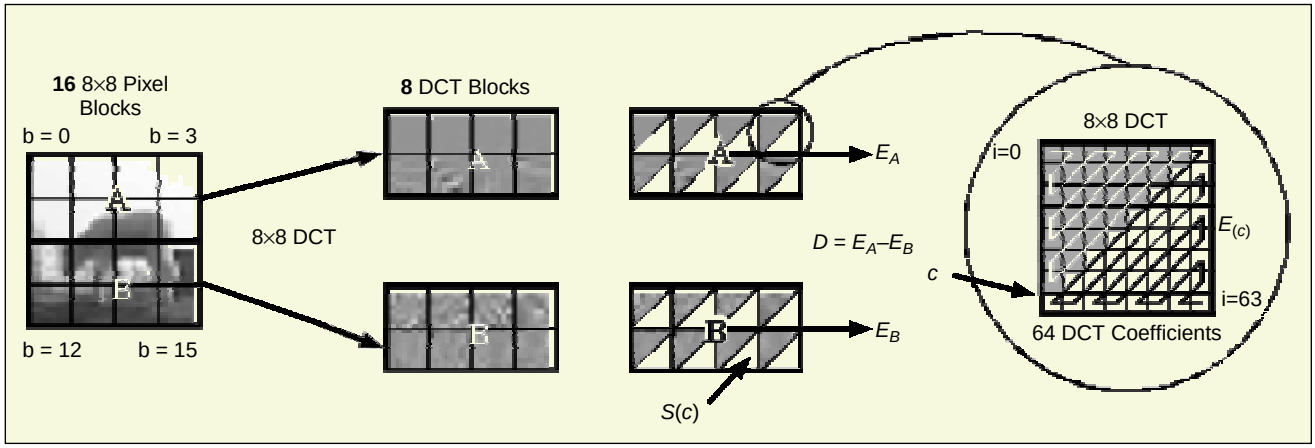
▲ The method proposed in [16] embeds a watermark by modifying the mean value of the pixels of randomly selected blocks in an image.

▲ The authors in [10] proposed the so-called "texture block coding" in which the watermark is embedded by copying one image texture block to another area in the image with a similar texture. Recovering the watermark is achieved by computing the autocorrelation function. This method offers high robustness to any kind of distortion because both image areas are distorted in a similar way. This means that the watermark recovery by autocorrelation will still work.

▲ Quantization can be exploited to hide a watermark. In [85] a method is proposed in which the pixel values of an image are first coarsely quantized, before some small adaptations are made to the image. To detect these adaptations the watermarked image is subtracted from its coarsely quantized version. In [57] selected wavelet coefficients are quantized using different quantizers for watermark bits 0 and 1.



▲ *34. (a) Sample I-frame; (b) block-based randomly shuffled I-frame showing the label-carrying (lc) regions and lc-subregions; (c) difference between the original and watermarked image showing that the DEW algorithm put the watermark in regions with a lot of spatial details.*



▲ *35. Illustration of the calculation of the energy carried by high-frequency DCT coefficients by (26).*

▲ Watermarks can also be embedded by using projection-based techniques [96], [2]. In these techniques, the original data (divided into blocks) are projected into another direction/subspace. The data here can be the transform coefficients of the original image. The projection direction could be random or image dependent. The authors in [2] also show that their proposed technique could resist rotation and scaling to some extent.

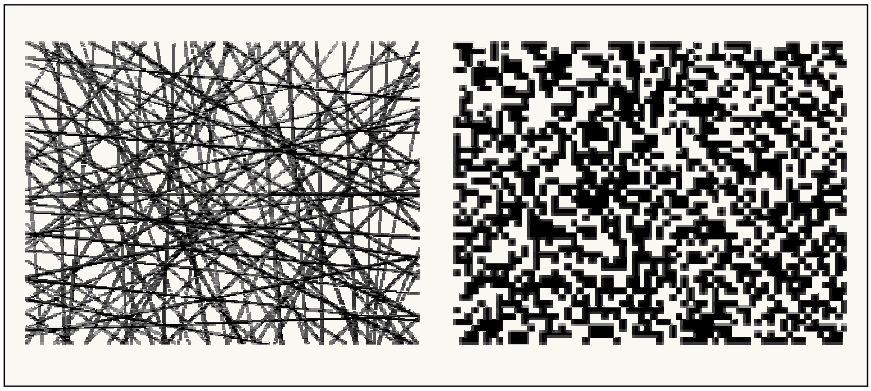
▲ The concept of self-embedding [101], that is embedding important parts of an image (for example, the eyes of a person) onto the image itself, is important to detect (and if possible recover from) a tampering attack in which a portion of the image has been altered. In [101] the authors proposed a high capacity watermarking technique that is capable of detecting tampering and to some extent recover from it.

In this article we have discussed the most important classes of watermarking techniques. The first class comprises the correlation-based methods. Here a watermark is embedded by adding pseudorandom noise to image components and detected by correlating the pseudorandom noise with these image components. The second class comprises the noncorrelation-based techniques. This class of watermarking methods can roughly be divided into two groups: the group based on LSB modification and group based on geometrical relations.

Digital watermarking is still a very active research area and by far a mature field. We discuss three ongoing research efforts in this area briefly as follows.

In the first place, watermarking algorithms that are more content dependent are being investigated. Such systems are needed to combat attacks such as the copy attack [60] that could copy a watermark from one image to another without knowledge of embedding system or cryptographic keys used. A watermark that is dependent on the content of the data being watermarked could resist this attack because the watermarks of images with different content will be distinctly different, and the watermark of one image can not be derived from the watermark of the other image.

Second, in the digital video area many research activities are now directed to low bit-rate video watermarking for applications such as video over the Internet. Low bit-rate video presents challenges because obviously there is not much room left to embed additional information. Such additional information will unquestionably have a big impact on the quality of the compressed video. One solution is to utilize the temporal dimension to spread the watermark to embed enough information for the watermark. Care must be taken, however, to satisfy the watermark granularity requirement. Also, some research activities are now being conducted for watermarking systems that watermarks both the video and audio parts of an audio visual data to protect against alteration of one of the components without interfering with the other (watermarked) part [27]. For example, a watermarked video may not be altered but its accompanying dialog could be altered without disturbing the watermark embedded in the video stream. The watermarks of the video and the audio could be independent or could be designed to be dependent on each other to increase the robustness of the complete watermark.

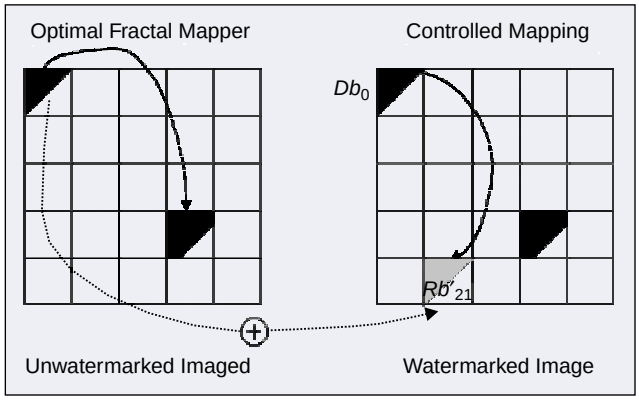


▲ *36. Examples of watermark patterns for salient-point modification.*

Finally, to provide potential users of watermarking systems information about a watermarking system's performance, research activities are being conducted to develop an internationally recognized watermark benchmarking system [29]. This benchmarking system is needed to give a fair comparison between the available watermarking systems. Such a comparison can help potential user to decide which system to use and also help watermarking system developers in improving their system. Closely related to the development of a watermark benchmarking system is the question of whether watermarking technique must be standardized. The formulation of future standards such as MPEG-4 and MPEG-7 is also trying to address the issue of protection of intellectual property rights (IPR) [44], [48]. Watermarking technology can play an important part in the implementation of such IPR protection.

Gerhard Cornelis Langelaar was born in Leersum, The Netherlands, on August 28, 1971. In 1989 he received his VWO-β (Gymnasium) diploma from the Revius Lyceum in doorn, and he received his M.Sc. in electrical engineering from Delft University of Technology in 1995. He received his Ph.D. from TU Delft in 2000. From September 1995 until October 1999 he worked at the Information Theory Group on the ACTS SMASH project, sponsored by the European Union. Since November 1999 he has been working for the Philips Advanced Systems & Applications Laboratory in Eindhoven where he continues working in the field of copy protection systems.

Iwan Setyawan received his B.Sc. degree in electrical engineering (majoring in computer engineering) from the Bandung Institute of Technology, Indonesia, in April 1996. He received his M.Sc. degree in information systems in 1999 from the same institution. He is currently pursuing his Ph.D. degree at the Delft University of Technology, The Netherlands, where he is a Research Assistant in the Information and Communication Theory Group. He is doing research work on digital watermarking techniques, with emphasis on low bit rate video watermarking. This work is performed as part of the European Community Project CERTIMARK.



▲ *37. Modifying the mapping between range and domain blocks.*

In real-time watermarking applications, robustness and computational complexity play important roles.

Reginald L. Lagendijk received the M.Sc. and Ph.D. degrees in electrical engineering from the Technical University (TU) of Delft in 1985 and 1990, respectively. He has been with TU of Delt since 1987 as an Assistant Professor, Associate Professor and, currently, Full Professor and Head of the Information and Communication Theory Group. He was a Visiting Scientist in the Electronic Image Processing Laboratories of Eastman Kodak Research in Rochester, NY, in 1991 and a Visiting Researcher at Microsoft Research, Beijing, in 2000. He is the author of *Iterative Identification and Restoration of Images* (Kluwer, 1991), and co-author of *Motion Analysis and Image Sequence Processing* (Kluwer, 1993) and *Image and Video Databases: Restoration, Retrieval, and Watermarking* (Elsevier, 2000). He is an Associate Editor of the *IEEE Transactions on Image Processing* and is currently Region Editor of Eurasip's Signal Processing: Image Communications. He is a member of the IEEE SP Society Technical Committee on Image and Multidimensional Signal Processing. His research interests include image and video compression, object-based compression, image quality measures, watermarking, image and video libraries, wireless multimedia communications, and image sequence restoration and enhancement. He is the Program Leader of the Delft University of Technology multidisciplinary research programme "Ubiquitous Communications (UbiCom)."

References

[1] A.M. Alattar, "Smart images using Digimarc's watermarking technology," in *Proc. SPIE Electronic Imaging'00, Security and Watermarking of Multimedia Content II*, vol. 3971, San Jose, CA, Jan. 2000, pp. 264-273.

[2] M. Alghoniemy and A.H. Tewfik, "Progressive quantized projection watermarking scheme," in *Proc. ACM Multimedia'99*, Orlando, FL, Oct. 1999, pp. 295-298.

[3] R.J. Anderson and F.A.P. Petitcolas, "On the limits of steganography," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 474-481, May 1998.

[4] T. Aura, "Invisible communication," in *Proc. HUT Seminar on Network Security '95*, Espoo, Finland, Nov. 6, 1995.

[5] T. Aura, "Practical invisibility in digital communication," in *Proc. Workshop Information Hiding* (Lecture Notes in Computer Science, vol. 1174), Cambridge, U.K., May 1996.

[6] F. Bartolini, M. Barni, V. Cappellini, and A. Piva, "Mask building for perceptually hiding frequency embedded watermarks," in *Proc. 5th IEEE Int. Conf. Image Processing ICIP'98*, vol. I, Chicago, IL, Oct. 4-7, 1998, pp. 450-454.

[7] M. Barni, F. Bartolini, V. Cappellini, A. Lippi, and A. Piva, "A DWT-based technique for spatio-frequency masking of digital signatures,"

in *Proc. SPIE/IS&T Int. Conf. Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, Jan. 25-27, 1999, pp. 31-39.

[8] P. Bas, J-M. Chassery, and F. Davoine, "Self-similarity based image watermarking," in *Proc. IX European Signal Processing Conf.*, Island of Rhodes, Greece, Sept. 8-11, 1998, pp. 2277-2280.

[9] P. Bas, J-M. Chassery, and F. Davoine, "A geometrical and frequential watermarking scheme using similarities," in *Proc. SPIE Electronic Imaging'99, Security and Watermarking of Multimedia Contents*, San Jose, CA, Jan. 1999, pp. 264-272.

[10] W. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," in *Proc. SPIE, Storage and Retrieval for Image and Video Databases III*, vol. 2420, San Jose, CA, Feb. 9-10, 1995, pp. 165-173.

[11] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Syst. J.*, vol. 35, no. 3 & 4, pp. 313-336, 1996.

[12] F.M. Boland, J.J.K. Ó Ruanaidh, and C. Dautzenberg, "Watermarking digital images for copyright protection," in *Proc. IEE Int. Conf. on Image Processing and Its Applications*, Edinburgh, U.K., July 1995, pp. 326-330.

[13] A.G. Bors and I. Pitas, "Embedding parametric digital signatures in images," in *Proc. EUSIPCO-96,* vol. III, Trieste, Italy, Sept. 1996, pp. 1701-1704.

[14] A.G. Bors and I. Pitas, "Image watermarking using DCT domain constraints," in *Proc. IEEE Int. Conf. Image Processing (ICIP'96)*, vol. III, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 231-234.

[15] G.W. Braudaway, "Protecting publicly-available images with an invisible watermark," in *Proc. ICIP 97, IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 524-531.

[16] O. Bruyndonckx, J.J. Quisquater, and B. Macq, "Spatial method for copyright labeling of digital images," in *Proc. IEEE Workshop on Nonlinear Signal and Image Processing*, Halkidiki, Greece, June 1995.

[17] S. Burgett, E. Koch, and J. Zhao, "Copyright labeling of digitized image data," *IEEE Commun. Mag.*, pp. 94-100, Mar. 1998.

[18] G. Caronni, "Assuring ownership rights for digital images," in *Proc. Reliable IT Systems, VIS '95,* Germany, 1995, pp. 251-263.

[19] B. Chen and G.W. Wornell, "An information-theoretic approach to the design of robust digital watermarking systems," in *Proc. ICASSP'99*, vol. 4, Phoenix, AZ, Mar. 15-19, 1999 [CD-ROM].

[20] I.J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," NEC Res. Insti., Princeton, NJ, Tech. Rep. 95-10, 1995.

[21] I.J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "A secure, robust watermark for multimedia," in *Preproc. Information Hiding*, Univ. of Cambridge, U.K., May 1996.

[22] I.J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for images, audio and video," in *Proc. 1996 Int. Conf. Image Processing*, vol. 3, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 243-246.

[23] I.J. Cox and M.L. Miller, "A review of watermarking and the importance of perceptual modeling," in *Proc. SPIE Electronic Imaging '97, Storage and Retrieval for Image and Video Databases V,* San Jose, CA, Feb. 1997.

[24] P. Davern and M. Scott, "Fractal-based image steganography," in *PreProc. Information Hiding*, Univ. of Cambridge, U.K., May 1996, pp. 245-256.

[25] U.S. Copyright Office Summary, "The Digital Millennium Copyright Act of 1998," Dec. 1998 [Online]. Available: http://lcweb.loc.gov/copyright/legislation/dmca.pdf

[26] G. Depovere, T. Kalker, and J.-P. Linnartz, "Improved watermark detection using filtering before correlation," in *Proc. 5th IEEE Int. Conf. Image Processing ICIP'98*, vol. I, Chicago, IL, Oct. 4-7, 1998, pp. 430-434.

[27] J. Dittmann, T. Fiebig, R. Steinmetz, S. Fischer, and I. Rimac, "Combined video and audio watermarking: Embedding content information in multimedia data," in *Proc. SPIE Electronic Imaging'00, Security and Watermarking of Multimedia Content II,* vol. 3971, San Jose, CA, Jan. 2000, pp. 455-464.

[28] Commission of the European Communities, "Amended proposal for a European parliament and council directive on the harmonisation of certain aspects of copyright and related rights in the information society" [Online]. Available: europa.eu.int/comm/internal_market/en/intprop intprop/copy2en.pdf

[29] European Union Community Research & Development Information Service (CORDIS), "Project CERTIMARK IST-1999-10987: Certification for watermarking techniques" [Online]. Available: www.cordis.lu/ist/projects /99-10987.htm.

[30] D.J. Fleet and D.J. Heeger, "Embedding invisible information in color images," in *Proc. ICIP 97, IEEE Int. Conf. Image Processing,* Santa Barbara, CA, Oct. 1997.

[31] J. Fridrich and M. Goljan, "Comparing robustness of watermarking techniques," in *Proc. Electronic Imaging '99, The International Society for Optical Engineering, Security and Watermarking of Multimedia Contents,* vol. 3657, San Jose, CA, Jan. 25-27, 1999, pp. 214-225.

[32] J. Fridrich, "Robust bit extraction from images," in *Proc. IEEE ICMCS'99 Conf.*, Florence, Italy, June 7-11, 1999.

[33] J. Fridrich and M. Goljan, "Protection of digital images using self embedding," in *Proc. Symp. Content Security and Data Hiding in Digital Media*, New Jersey Institute of Technology, Mar. 16, 1999.

[34] F. Goffin, J.-F. Delaigle, C. De Vleeschouwer, B. Macq, and J-.J. Quisquater, "A low cost perceptive digital picture watermarking method," in *Proc. SPIE Electronic Imaging '97, Storage and Retrieval for Image and Video Databases V,* San Jose, CA, Feb. 1997, pp. 264-277.

[35] A. Hanjalic, G.C. Langelaar, P.M.B. van Roosmalen, J. Biemond, and R.L. Lagendijk, *Image and Video Databases: Restoration, Watermarking and Retrieval* (Advances in Image Communications, vol. 8). New York: Elsevier Science, 2000.

[36] F. Hartung and B. Girod, "Digital watermarking of raw and compressed video," in *Proc. SPIE 2952: Digital Compression Technologies and Systems for Video Communication*, Oct. 1996, pp. 205-213.

[37] F. Hartung and B. Girod, "Watermarking of MPEG-2 encoded video without decoding and re-encoding," in *Proc. Multimedia Computing and Networking 1997 (MMCN 97)*, San Jose, CA, Feb. 1997.

[38] F. Hartung and B. Girod, "Digital watermarking of MPEG-2 coded video in the bitstream domain," in *Proc. ICASSP 97*, vol. 4, Munich, Germany, Apr. 21-24, 1997, pp. 2621-2624.

[39] F. Hartung and B. Girod, "Copyright protection in video delivery networks by watermarking of pre-compressed video," in *Multimedia Applications, Services and Techniques-ECMAST'97* (Springer Lecture Notes in Computer Science, vol. 1242), S. Fdida and M. Morganti, Eds. Heidelberg, Germany: Springer, 1997, pp. 423-436.

[40] A. Herrigel, H. Petersen, J. Ó Ruanaidh, T. Pun, and P. Shelby, "Copyright techniques for digital images based on asymmetric cryptographic techniques," presented at Workshop on Information Hiding, Portland, Oregon, USA, Apr. 1998.

[41] A. Herrigel, J.J.K. Ó Ruanaidh, H. Petersen, S. Pereira, and T. Pun, "Secure copyright protection techniques for digital images," in *Information Hiding* (Lecture Notes in Computer Science, vol. 1525), D. Aucsmith, Ed. Berlin, Germany: Springer, 1998, pp. 169-190.

[42] J.R. Hernández, M. Amado, and F. Pérez-Gonzalez, "DCT-Domain watermarking techniques for still images: Detector performance analysis and a new structure," *IEEE Trans. Image Processing*, vol. 9, pp. 55-68, Jan. 2000.

[43] K. Hirotsugu, "An image digital signature system with ZKIP for the graph isomorphism," in *Proc. ICIP-96*, *IEEE Int. Conf. Image Processing*, vol. III, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 247-250.

[44] K. Hill, "A perspective: The role of identifiers in managing and protecting intellectual property in the digital age," *Proc. IEEE*, vol. 87, pp. 1228-1238, July 1999.

[45] C.-T. Hsu and J.-L. Wu, "Hidden signatures in images," in *Proc. ICIP-96*, *IEEE Int. Conf. Image Processing*, vol. III, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 223-226.

[46] International Federation of the Phonographic Industry, "Embedded signalling systems issue 1.0," Request for Proposals, June 1997.

[47] ISO/IEC13818-2:1996(E), *Information Technology-Generic Coding of Moving Pictures and Associated Audio Information,* Video International Standard, 1996.

[48] ISO/IEC JTC1/SC29 WG11, *Coding of Moving Pictures and Audio: Overview of the MPEG-7 Standard,* ISO, Mar. 2000.

[49] A.E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. Image Processing*, vol. 2, pp. 18-30, Jan. 1992.

[50] A.K. Jain, "Image data compression: A review," *Proc. IEEE*, vol. 69, pp. 349-389, Mar. 1981.

[51] D. Kahn, *The Codebreakers.* New York: MacMillan, 1967.

[52] T. Kalker, J-P. Linnartz, G. Depovere, and M. Maes, "On the reliability of detecting electronic watermarks in digital images," in *Proc. IX European Signal Processing Conf.*, vol. 1, Island of Rhodes, Greece, Sept. 8-11, 1998, pp. 13-16.

[53] T. Kalker, G. Depovere, J. Haitsma, and M. Maes, "A video watermarking system for broadcast monitoring," in *Proc. SPIE Electronic Imaging '99, Security and Watermarking of Multimedia Contents,* San Jose, CA, Jan. 1999, pp. 103-112.

[54] E. Koch, J. Rindfrey, and J. Zhao, "Copyright protection for multimedia data," in *Proc. Int. Conf. Digital Media and Electronic Publishing*, Leeds, U.K., Dec. 6-8, 1994.

[55] E. Koch and J. Zhao, "Towards robust and hidden image copyright labeling," in *Proc. IEEE Workshop Non-Linear Signal and Image Processing*, Neos Marmaras, Thessaloniki, Greece, June 1995, pp. 452-455.

[56] D. Kundur and D. Hatzinakos, "A robust digital image watermarking scheme using wavelet-based fusion," in *Proc. ICIP 97*, *IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 544-547.

[57] D. Kundur and D. Hatzinakos, "Digital watermarking using multi resolution wavelet decomposition," in *Proc. IEEE Int. Conf. Acoustics*, *Speech and Signal Processing*, vol. 5, Seattle, WA, May 1998, pp. 2969-2972.

[58] M. Kutter, F. Jordan, and F. Bossen, "Digital signature of color images using amplitude modulation," in *Proc. SPIE Electronic Imaging '97, Storage and Retrieval for Image and Video Databases V,* San Jose, CA, Feb. 1997, pp. 518-526.

[59] M. Kutter and F.A.P. Petitcolas, "A fair benchmark for image watermarking systems," in *Proc. Electronic Imaging '99, Security and Watermarking of Multimedia Contents,* vol 3657, San Jose, CA, Jan. 25-27, 1999, pp. 226-239.

[60] M. Kutter, S. Voloshynovskiy, and A. Herrigel, "The watermark copy attack," in *Proc. SPIE Electronic Imaging '00, Security and Watermarking of Multimedia Content II,* vol. 3971, San Jose, CA, Jan. 2000, pp. 371-380.

[61] G.C. Langelaar, J.C.A. van der Lubbe, and R.L. Lagendijk, "Robust labeling methods for copy protection of images," in *Proc. SPIE Electronic Imaging '97, Storage and Retrieval for Image and Video Databases V*, San Jose, CA, Feb. 1997, pp. 298-309.

[62] G.C. Langelaar, R.L. Lagendijk, and J. Biemond, "Real-time labeling of MPEG-2 compressed video," *J. Visual Commun. Image Representation*, vol. 9, no. 4, pp. 256-270, Dec. 1998.

[63] G.C. Langelaar, "Conditional access to television service," in *Wireless Communication, the Interactive Multimedia CD-ROM*, 3rd ed. Amsterdam, The Netherlands: Baltzer Science, 1999.

[64] G.C. Langelaar, R.L. Lagendijk, and J.Biemond, "Watermarking by DCT coefficient removal: A statistical approach to optimal parameter settings," in *Proc. SPIE Electronic Imaging '99, Security and Watermarking of Multimedia Contents,* San Jose, CA, Jan. 1999, pp. 2-13.

[65] G.C. Langelaar, "Real-time watermarking techniques for compressed video data," Ph.D. dissertation, Delft University of Technology, The Netherlands, Jan. 2000.

[66] G.C. Langelaar and R.L. Lagendijk, "Optimal differential energy watermarking (DEW) of DCT encoded images and video," *IEEE Trans. Image Processing*, to be published.

[67] T. Liang and J. Rodriguez, "Improved watermarking robustness via spectrum equalization," in *Proc. IEEE ICASSP 2000*, Istanbul, Turkey, June 5-9, 2000.

[68] B.M. Macq and J.-J. Quisquater, "Cryptology for digital TV broadcasting," *Proc. IEEE*, vol. 83, pp. 944-957, June 1995.

[69] J.L. Massey, "Contemporary cryptology: An introduction," in *Contemporary Cryptology: The Science of Information Integrity*, G.J. Simmons, Ed. New York: IEEE Press, 1992, pp. 3-39.

[70] K.S. Ng and L.M. Cheng, "Selective block assignment approach for robust digital image watermarking," in *Proc.SPIE/IS&T Int. Conf. Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, Jan. 25-27, 1999, pp. 14-17.

[71] W. Niblack, *An Introduction to Digital Image Processing*. London: Prentice-Hall Int., 1986.

[72] N. Nikolaidis and I. Pitas, "Copyright protection of images using robust digital signatures," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP-96),* vol. 4, Atlanta, GA, May 1996, pp. 2168-2171.

[73] W.B. Pennebakker and J.L. Mitchell, *The JPEG Still Image Data Compression Standard*. New York: Van Nostrand, 1993.

[74] S. Pereira, J.J.K. Ó Ruanaidh, F. Deguillaume, G. Csurka, and T. Pun, "Template based recovery of Fourier-based watermarks using log-polar and log-log maps," in *Proc. IEEE Multimedia Systems 99, Int. Conf. Multimedia Computing and Systems,* Florence, Italy, June 7-11, 1999.

[75] F.A.P. Petitcolas and R.J. Anderson, "Weaknesses of copyright marking systems," presented at Multimedia and Security Workshop at ACM Multimedia '98. Bristol, U.K., Sept. 1998.

[76] I. Pitas and T.H. Kaskalis, "Applying signatures on digital images," in *Proc. IEEE Workshop on Nonlinear Signal and Image Processing*, Neos Marmaras, Thessaloniki, Greece, June 20-22, 1995, pp. 460-463.

[77] I. Pitas, "A method for signature casting on digital images," in *Proc. ICIP-96*, *IEEE Int. Conf. Image Processing*, vol. III, Lausanne, Switzerland, Sept. 15-17, 1996, pp. 215-218.

[78] A. Piva, M. Barni, F. Bartolini, and V. Cappellini, "DCT-based watermark recovering without resorting to the uncorrupted original image," in *Proc. ICIP 97, IEEE Int. Conf. Image Processing,*, Santa Barbara, CA, Oct. 1997, pp. 520-527.

[79] C. Podilchuk and W. Zeng, "Perceptual watermarking of still images," in *Proc. 1997 IEEE 1st Workshop Multimedia Signal Processing*, Princeton, NJ, June 23-25, 1997, pp. 363-368.

[80] J. Puate and F. Jordan, "Using fractal compression scheme to embed a digital signature into an image," in *Proc. SPIE Photonics East Symp.*, Boston, MA, Nov. 18-22, 1996.

[81] J.-L. Renaud, "PC industry could delay DVD," *Advanced Television Markets*, issue 47, May 1996.

[82] P.M.J. Rongen, M.J.J.J.B. Maes, and C.W.A.M. van Overveld, "Digital image watermarking by salient point modification," in *Proc. SPIE Electronic Imaging '99, Security and Watermarking of Multimedia Contents,* San Jose, CA, Jan. 1999, pp. 273-282.

[83] J.J.K. Ó Ruanaidh, F.M. Boland, and O. Sinnen, "Watermarking digital images for copyright protection," in *Proc. Electronic Imaging and the Visual Arts 1996*, Florence, Italy, Feb. 1996.

[84] J.J.K. Ó Ruanaidh, W.J. Dowling, and F.M. Boland, "Watermarking digital images for copyright protection," *Proc. Inst. Elec. Eng. Vision, Image, and Signal Processing*, vol. 143, no. 4, pp. 250-256, Aug. 1996.

[85] J.J.K. Ó Ruanaidh, W.J. Dowling, and F.M. Boland, "Phase watermarking of digital images," in *Proc. IEEE Int. Conf. Image Processing*, vol. III, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 239-242.

[86] J.J.K. Ó Ruanaidh and T. Pun, "Rotation, scale and translation invariant digital image watermarking," in *Proc. ICIP 97, IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 536-539.

[87] J.J.K. Ó Ruanaidh and S. Pereira, "A secure robust digital image watermark" *Electronic Imaging: Processing, Printing and Publishing in Colour*, *SPIE Proceedings*, Zürich, Switzerland, May 1998.

[88] J.J.K. Ó Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Signal Processing*, vol. 66, no. 3, pp. 303-317, May 1998.

[89] S. Rupley, "What's holding up DVD?" *PC Mag.*, vol. 15, no. 20, pp. 34, Nov. 19, 1996.

[90] P. Samuelson, "Legally speaking: Digital media and the law," *Commun. ACM*, vol. 34, no. 10, pp. 23-28, Oct. 1991.

[91] R.G. van Schyndel, A.Z. Tirkel, and C.F. Osborne, "A digital watermark," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Austin, TX, Nov. 1994, pp. 86-90.

[92] C.E. Shannon and W.W. Weaver, *The Mathematical Theory of Communications.* Urbana, IL: Univ. of Illinois Press, 1949.

[93] J.R. Smith and B.O. Comiskey, "Modulation and information hiding in images," in *Preproc. Information Hiding*, University of Cambridge, U.K., May 1996.

[94] M.D. Swanson, B. Zhu, and A.H. Tewfik, "Transparant robust image watermarking," in *Proc. IEEE Int. Conf. Image Processing*, vol. III, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 211-214.

[95] M.D. Swanson, B. Zhu, and A.H. Tewfik, "Robust data hiding for images," in *Proc. 7th IEEE Digital Signal Processing Workshop*, Loen, Norway, Sept. 1996, pp. 37-40.

[96] M.D. Swanson, B. Zhu, and A.H. Tewfik, "Data hiding for video-in-video," in *Proc. ICIP-97*, *IEEE Int. Conf. Image Processing*, Santa Barbara, CA , Oct. 26-29, 1997.

[97] M.D. Swanson, M. Kobayashi, and A.H. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proc. IEEE*, vol. 86, pp. 1064-1087, June 1998.

[98] K. Tanaka, Y. Nakamura, and K. Matsui, "Embedding secret information into a dithered multi-level image," in *Proc. 1990 IEEE Military Communications Conf.*, Sept. 1990, pp. 216-220.

[99] B. Tao and B. Dickinson, "Adaptive watermarking in the DCT domain," *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Apr. 1997.

[100] J. Taylor, *DVD Demistified: The Guidebook for DVD-Video and DVD-ROM*. New York: McGraw-Hill, 1997.

[101] T.-H. Lan and A.H. Tewfik, "Fraud detection and self embedding," in *Proc. ACM Multimedia '99 (Part 2)*, Orlando, FL, Oct. 1999, pp. 33-36.

[102] G. Voyatzis and I. Pitas, "Applications of Toral automorphisms in image watermarking," in *Proc. ICIP-96*, *IEEE Int. Conf. Image Processing*, vol. II, Lausanne, Switzerland, Sept. 16-19, 1996, pp. 237-240.

[103] G. Voyatzis, N. Nikolaides, and I. Pitas, "Digital watermarking: An overview," in *Proc. IX European Signal Processing Conference (EUSIPCO)*, Island of Rhodes, Greece, Sept. 8-11, 1998, pp. 13-16.

[104] B. A. Wandell, "Foundations of vision," Sinauer Associates, Inc., Sunderland, MA, 1995.

[105] R.B. Wolfgang and E.J. Delp, "A watermark for digital images," in *Proc. IEEE Int. Conf. Image Processing*, vol. III, Sept. 16-19, 1996, Lausanne, Switzerland, pp. 219-222.

[106] R.B. Wolfgang and E.J. Delp, "A watermarking technique for digital imagery: Further studies," in *Proc. Int. Conf. Imaging Science, Systems, and Technology,* Las Vegas, NV, June 30-July 3, 1997.

[107] R.B. Wolfgang and E.J. Delp, "Overview of image security techniques with applications in multimedia systems," in *Proc. SPIE Conf. Multimedia Networks: Security, Displays, Terminals, and Gateways,* vol. 3228, Dallas, TX, Nov. 2-5, 1997, pp. 297-308.

[108] R.B. Wolfgang and E.J. Delp, "Fragile watermarking using the VW2D watermark" in *Proc. Electronic Imaging '99*, vol. 3657, San Jose, CA, Jan. 25-27, 1999, pp. 204-213.

[109] R.B. Wolfgang, C.I. Podilchuk, and E.J. Delp, "Perceptual watermarks for digital images and video," in *Proc. SPIE/IS&T Int. Conf. Security and Watermarking of Multimedia Contents*, vol. 3657, San Jose, CA, Jan. 25-27, 1999.

[110] R.B. Wolfgang, C.I. Podilchuk, and E.J. Delp, "Perceptual watermarks for digital images and video," *Proc. IEEE*, vol. 87, pp. 1108-1126, July 1999.

[111] T.L. Wu and S.F. Wu, "Selective encryption and watermarking of MPEG video," in *Proc. Int. Conf. Image Science*, *Systems*, *and Technology*, *CISST'97*, June 1997.

[112] X.-G. Xia, C.G. Boncelet, and G.R. Arce, "A multiresolution watermark for digital images," in *Proc. ICIP 97*, *IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 548-551.

[113] W. Zeng and B. Liu, "On resolving rightful ownerships of digital images by invisible watermarks," in *Proc. ICIP 97*, *IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, pp. 552-555.

[114] J. Zhao and E. Koch, "Embedding robust labels into images for copyright protection," *Proc. Int. Congr. Intellectual Property Rights for Specialized Information, Knowledge and New Technologies,* Vienna, Austria, Aug. 21-25 1995.

[115] F. Hartung and B. Girod, "Watermarking of uncompressed and compressed video," *Signal Processing*, vol. 66, no. 3, pp. 283-301, May 1998.

[116] D. Coltuc and P. Bolon, "Watermarking by histogram specification," in *Proc. SPIE Electronic Imaging '99, Security and Watermarking of Multimedia Contents,* San Jose, CA, Jan. 1999, pp. 252-263.