9.5

Watermarking Techniques for Image Authentication and Copyright Protection

Anastasios Tefas, Nikos Nikolaidis, and Ioannis Pitas
Aristotle University of Thessaloniki

1 Introduction

Digital watermarking is a relatively new research area that attracted the interest of numerous researchers both in the academia and the industry and became one of the hottest research topics in the multimedia signal processing community. Although the term watermarking has slightly different meanings in the literature, one definition that seems to prevail is the following [1]: Watermarking is the practice of imperceptibly altering a piece of data in order to embed information about the data. The above definition reveals two important characteristics of watermarking. First, information embedding should not cause perceptible changes to the host medium (sometimes called cover medium or cover data). Second, the message should be related to the host medium. In this sense, the watermarking techniques form a subset of information hiding techniques. The latter ones also include cases where the hidden information is not related to the host medium (e.g., in covert communications). However, certain authors use the term watermarking with a meaning equivalent to that of information hiding in the general sense.

A watermarking system should consist of two distinct modules: A module that embeds the information in the host data and a module that detects if a given piece of data hosts a watermark and subsequently retrieves the conveyed information. Depending on the type, the amount and the properties of the embedded information (e.g., robustness to host signal alterations), as well as on the type of host data, watermarking can serve a multitude of applications as will be described in Section 2.

The first handful of papers on digital watermarking appeared in the late 1980s—early 1990s but very soon the area witnessed a tremendous growth and an explosion in the number of published papers, mainly due to the fact that people believed, at that stage, that watermarking could be a significant weapon in the battle against the continuously increasing digital media piracy. During the early days, researchers focused mainly on a limited range of host data, that is, digital image, video and audio data. Later on, watermarking techniques that are applicable to other media types appeared in the corresponding literature. Such media types include but are not limited to voxel-based 3D images, 3D models represented as polygonal meshes or parametric surfaces (e.g., NURBS surfaces), vector graphics, GIS data (e.g., isosurface contours), animation parameters, object-based video representations (e.g., MPEG-4 video objects),

symbolic description of audio (e.g., MIDI files), text (either in ASCII format, or as a binary image), software source code, binary executables, Java byte code and numeric data sets (stock market data, scientific data). This chapter will focus on still image watermarking. However, most of the principles and techniques that will be presented are readily applicable to other media types.

Although, in its first steps, watermarking was dominated by heuristic approaches without significant theoretical background and justification, soon researchers recognized that solid theoretical foundations had to be set and worked towards this direction by adopting and utilizing successfully techniques, principles and theoretical results from several scientific areas like communications (detection theory, error correction codes, spread spectrum communications), information theory (channel capacity), signal processing (signal transforms, compression techniques) and cryptography. Today, although the optimism of the first years is over, watermarking is still a very active research area, despite the failure of the currently available watermarking technology to serve the needs of the industry (as made clear by Secure Digital Music Initiative (SDMI) case [2]). Researchers are now very well aware that devising effective watermarking schemes, especially for the so-called security oriented applications (e.g., copyright protection, copy control etc), is an extremely difficult task. However, the introduction of new application scenarios and business models along with the small but steady steps towards solid theoretical foundations of this discipline and the combination of watermarking with other technologies like cryptography and perceptual hashing are expected to keep the interest in this new area alive [3,4]. For a thorough review of existing schemes and a detailed discussion on the main requirements of a watermarking scheme, the interested reader may consult some monographs [1,5–7] and several review papers and journal special issues [8–12].

This chapter is organized as follows. The main application domains of watermarking are reviewed in Section 2. Properties and classification schemes of watermarking techniques are presented in Section 3, whereas Section 4 presents the basic functional modules of a watermarking scheme. Finally sections 5 and 6 delve in more detail into principles and techniques devised for two major application areas namely copyright protection and authentication.

2 Applications of Watermarking Techniques

Watermarking can be the enabling technology for a number of important applications [13–15]. Obviously, each application imposes different requirements on the watermarking system. As a consequence, watermarking algorithms targeting different applications might be very different in nature. Furthermore, devising an efficient watermarking scheme might be much more difficult for certain applications. In the remainder of this section, we will briefly review some of the main application domains of watermarking.

- *Owner identification and proof of ownership.* This class of applications was the first to be considered in the watermarking literature. In this case the embedded data can carry information about the legal owner or distributor or any rights holder of a digital item and be used for notifying/warning a user that the item is copyrighted, for tracking illegal copies of the item or for possibly proving the ownership of the item in the case of a legal dispute.
- *Broadcast monitoring.* In this case, the embedded information is utilized for various functions related to digital media (audio, video) broadcasting. The embedded data can be used to verify whether the actual broadcasting of commercials took place as scheduled, i.e., whether proper airtime allocation occurred, for devising an automated royalty collection scheme for copyrighted material (songs, movies) aired by broadcasting operators or in order to collect information about the number of people that watched/listened to a certain broadcast (audience metering). Broadcast monitoring is usually performed by automated monitoring stations and is one of the watermarking applications that has found its way towards successful commercialization.
- *Transaction tracking.* In this application, each copy of a digital item that is distributed as part of a transaction bears a different watermark. The aim of this watermark is not only to carry information about the legal owner/distributor of the digital item but also to mark the specific transaction copy. As a consequence, the embedded information can be used for the identification of entities that illegally distributed the digital item or did not adopt efficient security measures for preventing the item from being copied or distributed and for deterring such actions. Identification of movie theaters where illegal recording of a movie with a handheld camera took place is a scenario that belongs to this category of applications. The watermarks used in such cases are often termed fingerprints and the corresponding application fingerprinting. However, the same term is used for the class of techniques that try to extract a unique descriptor (fingerprint) for each digital item, which is invariant to content manipulation [16,17]. Obviously these techniques (which are sometimes called perceptual or robust hashing techniques) are totally different from the watermark-based fingerprinting, since they do not embed any data on the digital item, i.e., they are passive techniques.
- *Usage control.* In contrast to the applications mentioned above, where watermarking is used to deter intellectual rights infringement or to help in identifying such infringements, in usage control applications, the

watermarking plays an active protection role by controlling the terms of use of the digital content. The embedded information can be used in conjunction with appropriate compliant devices, to prohibit unauthorized recording of a digital item (copy control), or playback of unauthorized copies (playback control). The DVD copy and playback control using watermarking complemented by content scrambling is such a case [15,18].

- *Authentication and tamper-proofing.* In this case, the role of the watermark is to verify the authenticity and integrity of a digital item for the benefit of either the owner/distributor or the user. Example applications include the authentication of surveillance videos in case their integrity is disputed [19], the authentication of critical documents (e.g., passports) and the authentication of news photos distributed by a news agency. In this context, the watermarking techniques can either signal an authentication violation even when the digital item is slightly altered or tolerate certain manipulations (e.g., valid mainstream lossy content compression) and declare an item as non-authentic only when "significant" alterations, have occurred (e.g., content editing). Certain watermarking methods used for authentication can provide tampered region localization, e.g., can detect the image regions that have been modified/edited.
- *Persistent item identification.* According to this concept, watermarking is used for associating an identifier with a digital item in a way that resists to certain content alterations. This identifier can be used, in conjunction with appropriate databases, to convey various information about the digital item. Depending on the related information, persistent identification can be the vehicle for some of the applications presented above, e.g., owner identification, or usage control. Furthermore, the attached information can be used both for carrying copyright information and for enhancing the host data functionalities e.g., by providing access to free services and products, thus, implicitly, discouraging the user from removing the watermark or illegally distributing the item as this would imply that he/she would loose the added value provided by the watermark. Persistent association is dealt with in the MPEG-21 standard.
- *Enhancement of legacy systems.* Data embedded through watermarking can be used for the enhancement of information or functionalities carried/provided by legacy systems while ensuring backwards compatibility. For example, using techniques capable of generating watermarks that are robust to analog to digital and digital to analog conversion one can embed in a digital image URLs that are related to the depicted objects. When such an image is printed (e.g., in a magazine) and then scanned by a reader, the embedded URL can be used for connecting him automatically to the corresponding webpage [20]. Digital data embedding in conventional analog PAL/SECAM signals is another application in this category. In a more "futuristic" scenario, one can envision that information capable of enabling stereoscopic viewing to stereo-enabled receivers could be embedded through watermarking in conventional digital TV broadcasts. Using such an approach, conventional TV receivers would continue to receive the conventional signal with non-perceptible degradations.

3 Classification of Watermarking Algorithms

Various types of watermarking techniques, each with its own distinct properties and characteristics can be found in the watermarking literature. In the following, we will review the basic categories of watermarking schemes and provide descriptions for the properties that distinguish each class from the rest.

A first classification of watermarking schemes can be organized on the basis of their resistance to host medium modifications. Such modifications can be either the result of common signal processing operations (e.g., lossy compression) or be specifically devised and applied in order to render the watermark undetectable or affect the credibility and reliability of a watermarking system in other ways. Such modifications are usually referred to as *attacks*. Attacks for intellectual property rights (IPR) protection watermarking systems will be discussed in Section 5.2. The degree of resistance of a watermarking method to host medium modifications is usually called robustness. Depending on the level of robustness offered, one can distinguish between the following categories of watermarking techniques:

- *Robust.* In this class the watermarks are designed so as to resist host signal manipulations and are usually employed in intellectual property rights protection applications. Obviously, no watermarking scheme can resist all types of modifications, regardless of their severity. As a consequence, robustness refers to a subset of all possible manipulations and up to a certain degree of host signal degradation.
- *Fragile.* In this case, the watermarks are designed to be vulnerable to all modifications, i.e., they become undetectable by even the slightest host data modification. Fragile watermarks are more easy to devise than robust ones and are usually applied in authentication scenarios.
- *Semi-fragile.* This class of watermarks provides selective robustness to a certain set of manipulations which are considered as legitimate and allowable, while being vulnerable (fragile) to others. Such watermarks can also be used in authentication cases instead of the fragile

ones. In practice, all robust watermarks are essentially semi-fragile but, in the former case, the selective robustness is not a requirement imposed by the system designer but rather something that cannot be avoided.

In order to achieve a sufficient level of security, watermark embedding and detection are usually controlled by a (usually secret) key K (see Section 4). In a way analogous to cryptographic systems, the watermarking schemes can be distinguished in two classes on the basis of whether the same key is used during embedding and detection:

- *Symmetric or private-key.* In such schemes, both watermark embedding and detection are performed using the same key K.
- *Asymmetric or public-key.* In contrast to the previous class, these watermarks can be detected with a key that is different than the one that was used in the embedding stage [21,22]. Actually, a pair of keys is used in this case: a private key to generate the watermark for embedding, and a public one for detection. For each private key, many public keys may be produced. Despite their advantages over their symmetric counterparts, asymmetric schemes are much more difficult to devise.

In terms of the information taken into account during embedding, the watermarking methods can be broadly classified in two categories:

- *Blind embedding schemes.* Schemes belonging to this category consider the host data as noise or interference. Therefore, these techniques essentially treat watermarking like the classic communications problem of signal transmission over a noisy channel, the only difference being that, in the case of watermarking, restrictions on the amount of distortions imposed on the channel (i.e., the host medium) by the signal (the watermark) should be taken into consideration. In most cases, these methods rely implicitly or explicitly on a certain degree of knowledge of the host signal statistics, thus leading to the subclass of "known host statistics" methods. Essentially all methods developed in the first years of watermarking research belong to this category, most of them revolving around the spread spectrum principle where the watermark signal consists of a pseudorandom sequence embedded, usually in an additive way, in the host signal.
- *Informed coding/embedding schemes.* These schemes emerged after the work of Cox [23] and exploit the fact that during embedding, not only the statistics of the host data but also the actual host data themselves are known. Knowledge of the host data can be utilized in order to improve watermark detection performance through interference cancellation. These methods are also known as known host state methods and treat watermarking as a problem of communication with side information at the transmitter. Many of these schemes make use of the quantization index modulation (QIM) principle [24] for message coding where embedding is achieved by quantizing the host signal or certain derived features using appropriately selected quantizers. Quantizer selection is controlled by the signal to be embedded and aims at minimizing host signal interference. Perceptual masking, i.e., utilization of the host signal along with principles of human perception in order to modify the watermark in a way that renders it imperceptible is another form of informed embedding. Both informed coding/embedding and perceptual masking will be reviewed later on in this chapter.

With respect to the information conveyed by the watermark, watermarking systems can be classified to one of the following two classes:

- *Zero bit systems*: Watermarking systems of this type can only check whether the data under consideration host a watermark generated by a specific key K, i.e., verify whether the data are watermarked or not. Certain authors use the term single-bit when referring to systems of this category, implying that the existence or absence of a specific watermark in the data essentially conveys one bit of information. The term watermark detection is used in this chapter to denote the procedure used to declare the presence of a watermark when one is indeed present in the host data and come up with a "no watermark present" answer when applied to data hosting no watermark or hosting a different watermark than the one under investigation.
- *Multiple bit systems*: These systems are capable of encoding a multiple bit message in the host data. For systems of this type, we make the distinction between watermark detection and message decoding. The data under investigation are first tested to verify whether they host a watermark or not. This procedure is identical to the detection procedure described above for zero bit watermarks. As soon as the detection algorithm declares that the data are indeed watermarked, the embedded message is decoded. Thus, for multiple bit systems, watermark detection and message decoding should be considered as two distinct steps that are performed in cascade, the message decoding step taking place only if a watermark has been found to reside in the data.

When it comes to watermark detection, watermarking methods can be categorized into two main classes:

- Techniques that require that the original signal is available during the detection phase. These schemes

are referred to as private, non-blind or non-oblivious schemes (see for example [25,26]). Non-blind schemes can be considered as the extremum of a more general category, that of informed detection schemes (e.g., [27]), which include methods that require that some sort of information related to the host signal (e.g., its original dimensions or a feature vector derived from the host signal) is available at the detector.

- Techniques that do not require the original signal (or other information about it) for watermark detection. These techniques are called oblivious or blind. Due to their wider scope of application, blind techniques received much more attention among researchers. Obviously, the lack of knowledge on the original host signal makes blind detection a much more difficult task than non-blind detection. Correlation based detection, where the decision on the watermark presence is obtained by evaluating the correlation between the watermark and the signal under investigation is an approach that belongs in this category. Correlation detection schemes implicitly assume that the host signal is Gaussian. Due to their simplicity, they have been very popular in the early days of watermarking (see for example [28–30]). Later on, a number of researchers tried to devise optimal detectors for a number of situations, where the Gaussianity assumption does not hold [31–35]. Both correlation and optimal detectors will be reviewed later on in this chapter.

With respect to the output of the watermark detection procedure, systems are categorized as follows:

- *Hard decision detectors* generate a binary output (watermark detected, watermark not detected).
- *Soft decision detectors* provide along with the binary output a real number which is essentially the value of the test statistic used for detection (e.g., the value of the correlation between the signal under investigation and the watermark) and is related to detection reliability. In this case, the binary decision is obtained by internally thresholding this number using an appropriately selected threshold.

4 Watermark Embedding, Detection, and Decoding

Having described the main categories of watermarking algorithms along with their characteristic properties we can now proceed in providing more formal definitions of the watermark embedding, detection and decoding procedures.

Watermark embedding can be performed in the spatial/temporal domain [28,36,37], by modulating the intensity of preselected samples, or by modifying the magnitude of selected coefficients in an appropriate transform domain, e.g., the DCT [25,38,39], DFT [30,40] or wavelet transform [29,41] domain. Watermark embedding can be considered as a function that involves the host medium $\boldsymbol{f}_o$, the embedding key K (usually an integer), a set of parameters U that control the embedding procedure, and, in the case of multiple-bit schemes, the message $\boldsymbol{m}$ that is to be embedded in the data. The message can be a character string, a number or even multimedia data (audio, images). However, at this stage it suffices to consider the message as a sequence of bits. The set of parameters U can contain, among other things, the so-called watermark embedding factor, i.e., a parameter that controls the amount of degradation that will be inflicted to the host signal by the watermark. The output of the watermark embedding function consists of the watermarked data $\boldsymbol{f}_w$. Thus, for multiple bit schemes, the watermark embedding function is of the following form:

$$\boldsymbol{f}_w = E(\boldsymbol{f}_o, K, \boldsymbol{m}, U) \tag{1}$$

whereas for zero-bit schemes $\boldsymbol{m}$ is not an input parameter of the function.

In certain cases, it is much more intuitive to view watermark embedding as a two-step procedure, i.e., a watermark generation step that results in the watermark signal $\boldsymbol{w}$, followed by a watermark embedding step, that aims at actually embedding $\boldsymbol{w}$ in the host data. For an informed embedding multiple-bit watermarking scheme, these two functions are of the following form:

$$\boldsymbol{w} = E_1(\boldsymbol{f}_o, K, \boldsymbol{m}, U) \tag{2}$$

$$\boldsymbol{f}_w = E_2(\boldsymbol{f}_o, \boldsymbol{w}, U) \tag{3}$$

Watermark detection, in the way that is defined in this chapter can be considered as a function that receives as input the data $\boldsymbol{f}'$ under investigation, a key K' (which, depending on whether the system is a symmetric or an asymmetric one, can be the same as the embedding key, or a different, public key) and, in case of non-blind schemes, the original data $\boldsymbol{f}_o$. The output of this function is a binary digit d (0: watermark has been detected, 1: watermark has not been detected), complemented, in the case of soft decision detectors, by a value r (usually in the range $[0, 1]$) that corresponds to the detection reliability. Therefore, the detection function takes the following form in the case of blind and non-blind schemes, respectively:

$$\{d, r\} = D(\boldsymbol{f}', K') \tag{4}$$

$$\{d, r\} = D(\boldsymbol{f}', \boldsymbol{f}_o, K') \tag{5}$$

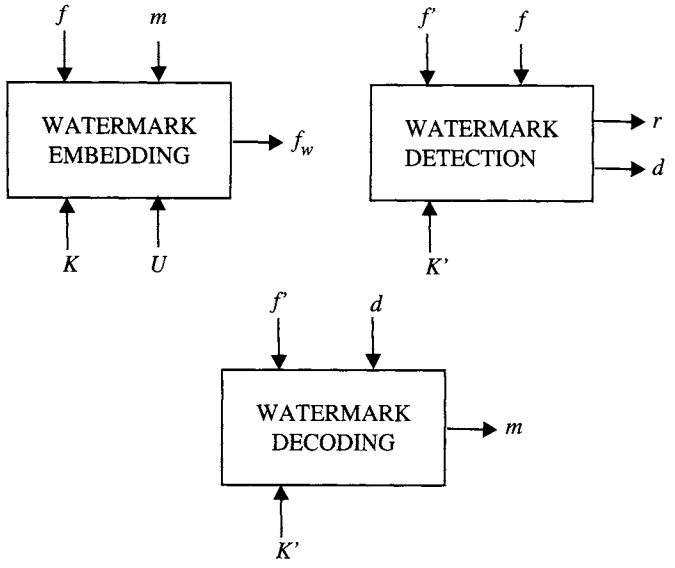

FIGURE 1 Modules of a watermarking system. Dashed lines indicate optional inputs or outputs.

where, as mentioned before, the reliability value r is available only in the case of soft decision detectors.

In case of multiple bit watermarking systems, whenever $D()$ declares that the data are watermarked, message decoding takes place. In the case of blind schemes, this operation can be expressed as follows:

$$\boldsymbol{m} = Dec(d, \boldsymbol{f}', K') \tag{6}$$

The detection output d has been included among the arguments of function *Dec* to denote the fact that *Dec* is called only if $d = 1$.

A schematic representation of the above procedures can be seen in Fig. 1.

5 Copyright Protection Watermarking

Copyright protection or more generally digital rights protection and management is a major application domain of watermarking. It is actually, an umbrella of applications that encompasses owner identification, proof of ownership, transaction tracking, copy & playback control, automated collection of royalties etc. Watermarking techniques aiming at copyright protection have to face some very difficult challenges, since they have to cope with attempts to infringe digital rights through illegal copying, distribution or playback of multimedia data. Copyright protection applications belong to the class of security-related applications, which are generally considered as the most tough for a watermarking method. In this Section we will present the basic requirements of a copyright protection watermarking scheme, namely, imperceptibility, cryptographic security and robustness, briefly describe methods, metrics and benchmarking platforms for measuring its performance, review the main categories of attacks against such a scheme and then proceed with providing information about the most important approaches that have been proposed in the literature.

5.1 Requirements and Metrics

5.1.1 Imperceptibility—Visual Quality

By definition, host data alterations imposed by watermarks should be imperceptible. Thus, imperceptibility is a requirement that is important in all watermarking applications and not only in copyright protection applications. In practice, the requirement of imperceptibility implies that the perceptual quality of the watermarked data, in our case digital images, should be kept high. Perceptual quality can be characterized either in terms of absolute quality (or simply quality) of watermarked images, i.e., without reference to the originals, or in terms of the relative quality of the watermarked images with respect to the originals, which is usually referred to as fidelity of the watermarked images. Normally, viewers of watermarked images do not have access to the originals. Thus, for those watermarking applications, quality is more important than fidelity. In order to measure quality or fidelity one needs to quantify the degree of distortion introduced to an image due to watermarking and, if possible, indicate whether this distortion is visible or not. The most effective way to conduct such measurements is by subjective evaluation procedures. Many different subjective testing methodologies exist: two alternative, forced choice (2AFC) tests [1], double stimulus continuous quality scale tests [42], double stimulus impairment scale tests [42], etc.

The perceptual quality of the watermarked images can be also measured in a quantitative way by using image quality metrics like the signal-to-noise ratio (SNR) or the peak signal-to-noise ratio (PSNR), considering the watermark as noise and the host image as signal. However, these metrics exhibit poor correlation with the visual quality as perceived by humans. Other quantitative metrics that correlate better with the perceptual image quality can be used. Weighted PSNR [43,44] which equals PSNR weighted at each image pixel by the local noise visibility function (local signal activity) could be such a metric. However, no globally agreeable and effective visual quality metric currently exists.

Obviously, the notion of "high quality" is application-dependent. For example, a certain amount of distortion on a movie might be perfectly acceptable for a television broadcast but unacceptable if the movie is to be displayed in cinema. It should be also noted, that certain applications require that the alterations imposed on the image are not only perceptually insignificant but also very small in a numeric sense, i.e., they require that watermarking preserves the

"numeric" quality of the data. Watermarking of medical images that are to be used as input in diagnosis procedures whose performance critically depends on the pixel intensities, is such an example.

5.1.2 Cryptographic Security

In compliance with one of the basic principles of cryptography, namely Kerckhoff's principle, the security of a copyright protection watermarking system should be based on the secrecy of keys that are used to embed/detect the watermark rather than on the secrecy of the algorithms. This means that designers of a watermarking system should assume that the embedding and detection algorithm (and perhaps their software implementations) will be available to users of this system and the fact that these users cannot detect or remove the watermark should be based solely on their lack of knowledge of the correct keys. An obvious implication of this property is that the cardinality of the keyspace should be large enough to make exhaustive search through this space practically infeasible.

5.1.3 Robustness

As already mentioned in Section 3, robustness can be defined as the degree of resistance of a watermarking method to modifications of the host signal due to either common signal processing operations or operations devised specifically in order to render the watermark undetectable. Watermarking systems aiming at copyright protection should ideally exhibit high resistance to all attacks that might occur in the host data in a specific application. This means that the detection performance of the system, i.e., its ability to declare correctly the presence/absence of a watermark in an image, and, in the case of multiple-bit systems, its decoding performance, i.e., its ability to retrieve successfully the hidden message bits, should not degrade significantly when data are altered due to intentional or unintentional attacks. Naturally, the set of manipulations that the watermark should be able to withstand as well as the severity of degradations that should be handled successfully depend on the target application. For example, a watermarking method designed to protect a database of high quality/resolution images that are to be used in desktop publishing need not be able to withstand high compression as such a manipulation would make the images practically unusable and, thus, is not very likely to occur. In order to measure the robustness of a watermarking method, one should be able to measure the detection performance of the algorithm, usually in relation to the severity of the degradation imposed by a certain attack. Furthermore, in the case of multiple bit algorithms, the decoding performance should be quantified [7].

Watermark Detection Performance

Watermark detection can be considered as a hypothesis testing problem, the two hypotheses (events) being:

- H_0 : the image under test hosts the watermark under investigation.
- H_1 : the image under test does not host the watermark under investigation.

Hypothesis H_1 can be further divided into two sub-hypotheses:

- H_{1a} : the image under test is not watermarked.
- H_{1b} : the image under test hosts a watermark different than the one under investigation.

Thus, the detection performance can be characterized by the false alarm (or false positive) error and its corresponding probability P_{fa}, i.e., the probability to detect a watermark in an image that is not watermarked or is watermarked with a different watermark than the one under investigation, and the false rejection (or false negative) error, described by the false rejection probability P_{fr}, i.e., the probability of not detecting a watermark in an image that is indeed watermarked with the watermark under investigation. Depending on the application, these two types of errors might have different importance. P_{fa} can be evaluated using detection trials with erroneous watermarks (hypothesis H_{1b}) or detection trials on non-watermarked images (hypothesis H_{1a}). The former might sometimes be preferable, since it usually corresponds to the worst case scenario. Furthermore, false alarm probability evaluated on images watermarked by a different key than the one used for detection provides an indication on whether the keys in the algorithm "keyspace" are able to generate distinct "non-overlapping" watermarks, and, thus, lead to estimates of the "effective keyspace". One can distinguish between three types of false alarms and false rejections [1]: those evaluated on a single image using multiple keys, those evaluated on multiple images using a single key and those evaluated on multiple images using multiple keys.

In the case of soft decision detectors (see Section 3), one can derive the empirical probability distribution functions (histograms) of the detection test statistic for both hypotheses H_0 and H_{1b} (or H_{1a}). By utilizing these empirical distributions the probabilities of false alarm $P_{fa}(T_k)$ and false rejection $P_{fr}(T_k)$ as a function of the detection threshold T can be extracted. Using $P_{fa}(T_k), P_{fr}(T_k)$ we can plot the *receiver operating characteristic* (ROC) curve, i.e., the plot of P_{fa} versus P_{fr} (Fig. 2). The ROC curve provides an overall view of the watermark detection performance in various operating conditions. Using the ROC curve, one can select the threshold value that gives a (P_{fa}, P_{fr}) pair satisfying the application requirements. Furthermore, the ROC curve can be used for the evaluation of other performance metrics, like the P_{fa} for a fixed, user-defined P_{fr}, the P_{fr} for a fixed, user-defined P_{fa}

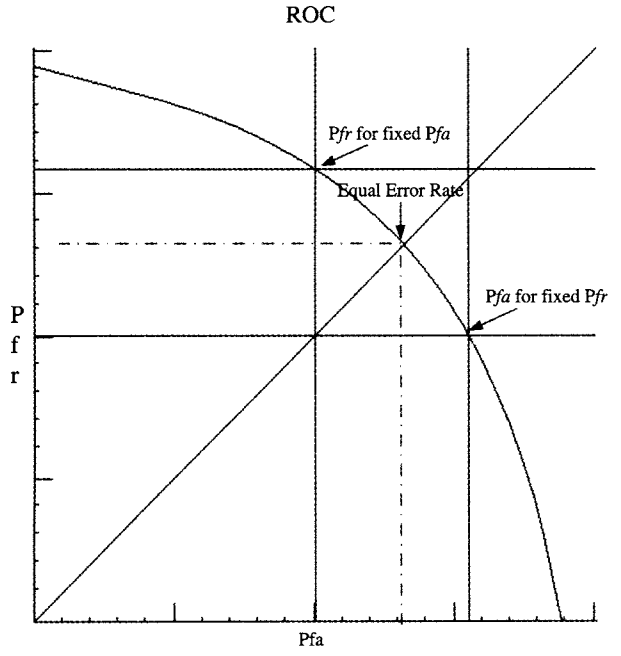

FIGURE 2 Receiver operating characteristic (ROC) curve.

and the equal error rate (EER), i.e, the point on the ROC where $P_{fa} = P_{fr}$ (Fig. 2).

Message Decoding Performance. The decoding performance of a watermarking method that supports message encoding can be characterized by the bit error rate (BER), i.e., the probability of erroneously decoding one message bit. Since message decoding is assumed to take place only in the case of successful detection, there is a close relation between the decoding and detection performance. As a consequence, a BER value should only be referenced along with the corresponding detection error probabilities, i.e., the probabilities of false alarm and false rejection.

Another metric that is related to the decoding performance of a watermarking algorithm is its payload, which can be defined as the maximum number of bits that can be encoded in a fixed amount of host data and decoded with a pre-specified BER or alternatively as the amount of data required to host a fixed number of bits, so that they can be decoded with a pre-specified BER. Essentially, the payload expresses the number of information bits that can be embedded per host image pixel.

As an example, suppose that we want to hide a message $\boldsymbol{m}$ comprised of M information bits in the image $f(\boldsymbol{x})$ that has N pixels. The performance of the watermark decoder can be also measured by the probability of error P_e, defined as the probability of getting a wrong estimate of the hidden message:

$$P_e \triangleq Pr\{\hat{\boldsymbol{m}} \neq \boldsymbol{m}\} \tag{7}$$

The goal of the watermark decoder is to obtain an estimate $\hat{\boldsymbol{m}}$ of the hidden message $\boldsymbol{m}$ such that P_e is minimized. By conditioning the decoding error probability P_e associated with the specific decoder and the minimum acceptable BER, we can estimate the maximum number of message bits that can be encrypted in the image. In the case of image watermarking the maximum acceptable error probability of the watermark detector is conditioned by the threshold T of the watermark detection ratio. In other words, an image is considered watermarked if the watermark detection ratio is greater than the predefined threshold. Thus, the minimum number of samples needed for the encoding of the hidden message for a specific decoding error probability is constrained by the detection threshold used in the watermark detection. Let us denote by q the probability of getting a wrong estimate of a message bit during watermark decoding:

$$q \triangleq Pr\{\hat{\boldsymbol{m}}(i) \neq \boldsymbol{m}(i)\} \tag{8}$$

Error correction codes can be used for correcting the watermark detection errors, resulting at a correct decoding of the hidden message. We suppose that we use an error correction code (e.g., BCH) that can correct R errors in M bits. The objective is to find the number of information bits that can be encrypted in the codeword for a specific message decoding error probability. The probability of getting a correct estimate of the message by using an error correction code that corrects R bit errors in M bits when the BER is q is given by [45]:

$$P_d = \sum_{i=R}^{M} \frac{M!}{i!(M-i)!} q^i (1-q)^{M-i} \tag{9}$$

5.2 Attacks Against Copyright Protection Watermarking Systems

As mentioned in the previous sections, a copyright protection watermarking system should exhibit a significant degree of robustness to attacks. The most obvious effect of an attack in a watermarking system is to render the watermark undetectable. Such attacks can be actually classified in two categories [44]: removal attacks and desynchronization (or geometric) attacks. As implied by their name, removal attacks result in the removal of the watermark from the host image or in a significant decrease of its energy relatively to the energy of the host signal. In most cases, removal attacks affect the amplitude of the watermarked signal, i.e., in the case of images, the pixel intensity or color. Removal attacks include linear or non-linear filtering (e.g., arithmetic mean, median, Gaussian, Wiener filtering), sharpening, contrast

enhancement (e.g., through histogram equalization), gamma correction, color quantization or color subsampling (e.g., due to format conversion), lossy compression (JPEG, JPEG2000, etc.) and other common image processing operations. Additive or multiplicative noise (Gaussian, uniform, salt and pepper noise), insertion of multiple watermarks on a single image or image printing and rescanning (essentially a D/A-A/D conversion) are some additional examples of removal attacks. Finally, intentional removal attacks, i.e., attacks that have been devised with the intention to remove the watermark include, among others, the *averaging attack* where N instances of the same image, each hosting a different watermark, are averaged in order to obtain a watermark-free image, and the *collusion attack* where N images hosting the same watermark are averaged to obtain a (noisy) version of the watermark signal. This watermark estimate can be subsequently subtracted from each of the images to obtain watermark-free images.

Contrary to removal attacks, desynchronization attacks do not remove the watermark but cause a loss of of synchronization (usually loss of the image coordinates) between the watermark signal embedded in the host signal and the watermark signal used for the correlation evaluation (see Section 5.4 for an example illustrating such a case). In other words, the watermark signal is still embedded in the host signal (with its energy almost intact) but cannot be detected. Desynchronization attacks usually involve global geometric distortions like translation, rotation, mirroring, scaling and shearing (i.e., general affine transformations), cropping, line or column removal, projective distortions, etc. Local geometric distortions like the random bending attack [46] that includes local shifting, rotation and scaling along with noise addition can also be very effective in inducing loss of synchronization. The mosaic attack that implies cutting an image into non-overlapping pieces can also be considered as a desynchronization attack. The small image tiles can be easily assembled and displayed so as to be perceptually identical to the original image using appropriate commands on the display software (e.g., the Web browser). However, a detector applied on each image tile separately will fail to detect the watermark due to cropping. The template removal attacks is another category of desynchronization attacks that are only applicable to systems using a synchronization template (see Section 5.4) to regain synchronization in case of geometric distortions. Such attacks first estimate and remove the synchronization template from an image and then apply a geometric distortion to render the watermark undetectable.

Apart from the two attack categories described above, which are the most studied in the watermarking literature, other attacks can be devised that do not aim at making the watermark undetectable but try to harm a watermarking system or render the watermarking concept unreliable by other means [1]. Such attacks include unauthorized embedding attacks and unauthorized detection or decoding attacks. The copy attack [47] is an attack that illustrates the concept of unauthorized embedding. According to this attack, an attacker that is in possession of a method that can estimate the watermark that is embedded in an image or a set of images (e.g., through the collusion attack mentioned above) can subsequently embed this watermark in other watermark-free images. Thus, a claim from a copyright owner that images bearing his watermark are his property can be confronted by the attacker, who can show that this watermark exists in images that are not his, i.e., in the fake watermarked images that the attacker has created. The SWICO (single watermarked image counterfeit original) and TWICO attacks [48] also belong to this category. In short, the SWICO attack involves the creation of a fake original image $\boldsymbol{f}$ by subtracting a watermark $\boldsymbol{w}$ from an image $\boldsymbol{f}_w$ watermarked by another person. The attacker can then claim that he has both the original image $\boldsymbol{f} = \boldsymbol{f}_w - \boldsymbol{w}$ and an image $\boldsymbol{f}_w = \boldsymbol{f} + \boldsymbol{w}$ watermarked with his own watermark, thus causing an ownership dispute.

Unauthorized detection attacks include attacks that aim at providing the attacker with information on whether an image is watermarked and perhaps reveal the encoded message (if any). Unauthorized detection does not consist a threat for all copyright protection applications. An example of an unauthorized detection attack is a brute force, exhaustive search approach where an attacker in possession of the detection algorithm checks successively all keys in the key space in order to find whether an image is watermarked.

In order to measure the effect of a certain attack on the detection or decoding performance of an algorithm, plots of an appropriate performance metric (e.g., BER or probability of false alarm) versus the attack severity can be constructed. For attacks whose impact on the host image varies monotonically with respect to a certain parameter, it might be sufficient for the user to know only the most severe attack that the algorithm can withstand [7]. For a chosen performance metric, the "breakdown point" of the algorithm for this attack can be evaluated by increasing the attack severity (e.g., decreasing the JPEG quality factor) in appropriately selected steps until the detector output does not satisfy the chosen performance criterion. The strongest attack, for which the algorithm performance is above the selected threshold, is the algorithm break down point for this attack.

5.3 Benchmarking of Copyright Protection Image Watermarking Algorithms

A benchmarking tool for image watermarking methods should be able to pinpoint the advantages and disadvantages of such methods and enable the user to perform efficient comparison of methods. Unfortunately, benchmarking of image watermarking algorithms is not an easy task since

it requires the cross-examination of a set of dependent performance indicators like algorithmic complexity, decoding/detection performance and perceptual quality of watermarked images. As a consequence, one cannot derive a single figure of merit but should deal with a set of performance indicators. An efficient benchmarking system should be able to quantify and present in an intuitive way the relations among the various performance indicators, e.g., the relation between watermark detection performance and perceptual quality. A small number of attempts to create benchmarking systems took place over the last years but this field is still in need of more efficient methodologies and actual implementations. Three of the most well known benchmarking systems are presented below.

5.3.1 Stirmark

Stirmark [46,49] is the first benchmarking tool that has been developed. A new version is currently under development. The source code of the benchmark is available to the public, and thus users can programme their own attacks in addition to those provided by the benchmark (sharpening, JPEG compression, noise addition, filtering, scaling, cropping, shearing, rotation, column and line removal, flipping and "Stirmark" attack). The user should provide, apart from the embedding and detection algorithms, appropriate command files (evaluation profiles) that define the tests or the attacks that will be performed. Currently, one can perform tests for measuring how the embedding strength influences the PSNR of the watermarked image, tests for the evaluation of the time required to perform embedding and tests for measuring the influence of attacks on the detection and decoding performance. In this last category of tests, Stirmark performs for each attack parameter within a certain range embedding and detection with a random key and message and measures the detection certainty or the BER. In the future, the benchmark will include tests for measuring the probability of false alarm. A first version of a Web-based client-server architecture that implements the Stirmark benchmark has been also developed.

5.3.2 Checkmark

Checkmark [50] is essentially a successor of the previous Stirmark version (namely, Stirmark version 3.1). In addition to the attacks implemented in Stirmark, Checkmark provides a number of new attacks that include wavelet compression, projective transformations, modelling of video distortions, image warping, copy attack, template removal attack, denoising, non-linear line removal, collage attack, down/up sampling, dithering and thresholding. The developers of Checkmark provide the Matlab source code of the application and thus one can add new attacks to the existing ones. The benchmark provides a number of "application templates" which are essentially lists of attacks related to a specific application. In addition, Checkmark incorporates two new objective quality metrics: the weighted PSNR and the so-called Watson metric. Despite the major improvements that have been introduced, the basic principles of Checkmark are very similar with those of Stirmark 3.1. In both cases, the user should provide the benchmark with a set of watermarked images and a detection routine along with a user-defined detection rule. The attacks that are included in the application template that has been selected by the user are applied in every watermarked image and the detection routine is being used to provide the detection result.

5.3.3 Optimark

Optimark [51] is a benchmarking platform that provides a graphical user interface and incorporates the same attacks with Stirmark 3.1. These attacks can be performed either one at a time or as a cascade. The user should supply embedding and detection/decoding routines in the form of executable files. Optimark supports hard and soft decision detectors. The user selects the set of test images, the set of keys and messages that will be used in the trials and the attacks that will be performed on the watermarked images. Furthermore, she provides the set of PSNR values for the watermarked images, along with the embedding factors that the embedding software should use in order to achieve these PSNR values. Optimark performs in an automated way multiple trials using the selected images, embedding strengths, attacks, keys and messages. Detection using both correct and erroneous keys (which are necessary for the evaluation of the probability of false alarms) is performed. Message decoding performance is evaluated separately from watermark detection. The "raw" results are processed by the benchmark in order to provide the user with a number of performance metrics and plots, depending on the type of the algorithm under test. For example, when testing a multiple-bit algorithm that employs a soft decision detector the user can obtain the following metrics: ROC, equal error rate, probability of false alarm for a user defined probability of false rejection, probability of false rejection for a user defined probability of false alarm, plots of bit error rate and percentage of perfectly decoded messages versus the detection threshold (for a specific message length), plot of payload versus the detection threshold (for a specific BER). The software evaluates various complexity metrics like average embedding, detection and decoding time and provides an option to evaluate the algorithm breakdown point for a given attack. Finally, it can summarize the results in various ways, e.g., provide average results for a set of images and a specific attack or average results over a number of different attacks for a specific image. The next Optimark version, which is currently under development and testing, will give users the ability to provide their own attacks.

A thorough treatment of the subject of performance evaluation of watermarking algorithms can be found in [1,7].

5.4 Spread Spectrum Watermarking

Spread spectrum watermarking draws its name from spread spectrum communication techniques [52] that are used to achieve secure signal transmission in the presence of noise and/or interception attacks that generate an appropriate jamming signal to interfere with the transmission. In such a situation, one can spread the energy of a symbol to be transmitted either in the time domain, by multiplying it by a pseudorandom sequence, or in the frequency domain by spreading its energy over a large part of the signal spectrum.

5.4.1 Blind Additive Embedding with Correlation Detection

In this section, a simple zero-bit spread spectrum watermarking system that consists of a blind additive embedder and a blind correlation detector will be presented. Despite its simplicity, this methodology has been utilized extensively, in many variations, in the early days of watermarking [53,54]. Means of improving or creating variants of the basic algorithm will also be presented in this section.

The embedding procedure of this system employs the addition of a white, zero-mean pseudorandom signal $\boldsymbol{w}$ (generated by using a secret key K in conjunction with the appropriate generation function) on the host signal $\boldsymbol{f}_o$:

$$\boldsymbol{f}_w = \boldsymbol{f}_o + p\boldsymbol{w} \tag{10}$$

where $\boldsymbol{f}_w$ is the watermarked signal and $p > 0$ is a constant that controls the watermark embedding energy (watermark embedding factor). Obviously, p is closely related to the watermark perceptibility. On a per-sample basis, the above equation can be stated as follows:

$$\boldsymbol{f}_w(n) = \boldsymbol{f}_o(n) + p\boldsymbol{w}(n), \quad n = 0, \ldots, N-1 \tag{11}$$

where N denotes the signal length. In the following, we will assume that equations (10), (11) refer to the spatial domain. In case of image watermarking, the watermark modifies the intensity or color of the image pixels and $\boldsymbol{f}_o$, $\boldsymbol{w}$ and $\boldsymbol{f}_w$ are two-dimensional signals.

As has already been mentioned, the watermark detection aims at verifying whether a given watermark $\boldsymbol{w}_d$ is embedded in the test signal $\boldsymbol{f}_t$ or not. During detection, $\boldsymbol{f}_t$ can be represented in the following form:

$$\boldsymbol{f}_t = \boldsymbol{f}_o + p\boldsymbol{w}_e \tag{12}$$

This equation can summarize all three possible detection hypotheses, namely:

- The watermark $\boldsymbol{w}_d$ is indeed embedded in the signal (event H_0) which corresponds to $p \neq 0$ and $\boldsymbol{w}_e = \boldsymbol{w}_D$.
- The watermark $\boldsymbol{w}_d$ is not embedded in the signal (event H_1) which can imply either that no watermark is present (event H_{1a}), or that the signal bears a different watermark than the one under investigation (event H_{1b}). In the equation above, event H_{1a} corresponds to $p = 0$, whereas event H_{1b} corresponds to $\boldsymbol{w}_e \neq \boldsymbol{w}_d$.

In order to decide which event holds, i.e., which is the valid hypothesis, the correlation between the signal under investigation and the watermark is evaluated:

$$c = \frac{1}{N}\sum_{n=0}^{N-1} f_t[n]w_d[n] = \frac{1}{N}\sum_{n=0}^{N-1}\left(f_o[n]w_d[n] + pw_e[n]w_d[n]\right) \tag{13}$$

Such a detection scheme is usually called a *correlation detector* (also known as matched filter). By assuming statistical independence between the host signal $\boldsymbol{f}_o$ and both watermarks $\boldsymbol{w}_e$, $\boldsymbol{w}_d$, an expression for the mean of the correlation c can be derived in a straightforward manner [55]:

$$\mu_c = E[c] = E\left[\frac{1}{N}\sum_{n=0}^{N-1}\left(f_o[n]w_d[n] + pw_e[n]w_d[n]\right)\right]$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} E[f_o[n]]E[w_d[n]] + \frac{1}{N}p\sum_{n=0}^{N-1} E[w_e[n]w_d[n]] \tag{14}$$

Since the watermark has been chosen to be a zero mean random signal, the first term of the expression will be zero and, therefore, μ_c will depend only on the second term. When the signal bears no watermark, i.e., when $p = 0$ the second term is also zero and the mean value of the correlation is zero. Furthermore, when the signal bears a different watermark than the one under investigation ($\boldsymbol{w}_e \neq \boldsymbol{w}_d$) the second term will obtain a small value, close to zero, as two watermarks generated using two different keys are expected to be almost orthogonal to each other. When the signal hosts the watermark under investigation, i.e., when $p \neq 0$ and $\boldsymbol{w}_e = \boldsymbol{w}_d$, the mean value of c can be easily shown to be equal to $p\sigma_w^2$ where σ_w^2 is the variance of the watermark signal. Thus, the conditional probability distributions $p_{c|H_0}$, $p_{c|H_1}$ of the correlation value c under the two hypotheses H_0 and H_1 will be centered around $p\sigma_w^2$ and 0, respectively (Fig. 3). Furthermore, for the case under study, these distributions will be approximately Gaussian. For suitable values of p, σ_w^2 and by assuming that the variances $\sigma_{c|H_0}^2$, $\sigma_{c|H_1}^2$ of c under the two hypotheses are reasonably small, a decision on the valid hypothesis can be obtained by comparing c against

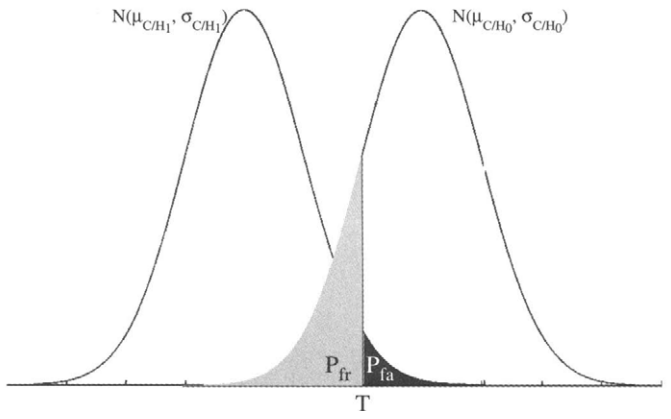

FIGURE 3 Conditional pdfs of the correlation value c under hypotheses H_0, H_1.

a suitably selected threshold $T > 0$ that lies between 0 and $p\sigma_w^2$. More specifically, a decision to accept hypothesis H_0 or H_1 is taken when $c > T$ and $c < T$, respectively.

For a given threshold, the probabilities of false alarm $P_{fa}(T)$, and false rejection $P_{fr}(T)$ which characterize the performance of this system can be evaluated as follows:

$$P_{fa}(T) = Prob\{c > T|H_1\} = \int_T^{\infty} p_{c|H_1}(t)dt \tag{15}$$

$$P_{fr}(T) = Prob\{c < T|H_0\} = \int_{-\infty}^{T} p_{c|H_0}(t)dt \tag{16}$$

Obviously, these two probabilities depend on $\mu_{c|H_0}$, $\mu_{c|H_1}$, $\sigma^2_{c|H_0}$, $\sigma^2_{c|H_1}$. By observing Fig. 3, one can conclude that the system performance improves (i.e., the probabilities of false alarm and false rejection for a certain threshold decrease) as the two distributions come further apart, i.e., as the difference $\mu_{c|H_0} - \mu_{c|H_1}$ increases. Furthermore the performance improves as the variances of the two distributions $\sigma^2_{c|H_0}$, $\sigma^2_{c|H_1}$ decrease.

Provided that the additive embedding model (10) has been used and under the assumptions that no attacks have been applied on the signal and that the host signal $\boldsymbol{f}_o$ is Gaussian, the detection theory states that the correlation detector described above is optimal with respect to the Neyman-Pearson criterion, i.e., it minimizes the probability of false rejection P_{fr} subject to a fixed probability of false alarm P_{fa}.

A variant of the above algorithm that employs non-blind detection can be easily devised by subtracting the original signal $\boldsymbol{f}_o$ from the signal under investigation before evaluating the correlation c. It can be proven that such a substraction drastically improves the performance of the algorithm by reducing the variance of the correlation distribution. Instead of the correlation (13) one can also use the normalized correlation, i.e., the correlation normalized by the magnitudes of the watermark and the watermarked signal:

$$c = \frac{\sum_{n=0}^{N-1} f_t[n]w_d[n]}{\sqrt{\sum_{n=0}^{N-1} f_t^2[n] \sum_{n=0}^{N-1} w_d^2[n]}} \tag{17}$$

Normalized correlation can grand the system robustness to operations such as increase or decrease of the overall image intensity.

The zero-bit system presented above can be easily extended to a system capable of embedding one bit of information. In such a system, symbol 1 is embedded by using a positive value of p whereas symbol 0 is embedded by using $-p$. Watermark detection can be performed by comparing $|c|$ against T, i.e., a watermark presence is declared when $|c| > T$. In case of a positive detection, the embedded bit can be decoded by comparing c against T and $-T$, i.e., 0 is decoded if $c < -T$ and 1 if $c > T$.

Another popular approach for embedding the watermark in the host signal is multiplicative embedding:

$$f_w(n) = f_o(n) + pf_o(n)w(n) \tag{18}$$

Using such an embedding law, the embedded watermark $pf_o(k)w(k)$ becomes image-dependent, thus providing an additional degree of robustness, e.g., against the collusion attack. Furthermore, by modifying the magnitude of a watermark sample proportionally to the magnitude of the corresponding signal sample (be it pixel intensity or magnitude of a transform coefficient), i.e., by imposing larger modifications to large amplitude signal samples, a form of elementary perceptual masking can be achieved.

The spectral characteristics and the spatial structure of the watermark play a very important role to robustness against several attacks. These characteristics can be controlled in the watermark generation procedure and affect the more general characteristics of the watermarking system, like robustness and perceptual invisibility. In the following Section we will see the basic categories of watermarks as they are derived by the various existing watermark generation techniques.

5.4.2 Chaotic Watermarks

Chaotic watermarks have been introduced, as a promising alternative to pseudorandom signals [56–58]. An overview of chaotic watermarking techniques can be found in [59]. Sequences generated by chaotic maps constitute an efficient alternative to pseudorandom watermark sequences. A chaotic discrete-time signal $x[n]$ can be generated by

a chaotic system with a single state variable by applying the recursion:

$$x[n] = \mathcal{T}(x[n-1]) = \mathcal{T}^n(x[0]) = \underbrace{\mathcal{T}(\mathcal{T}(\ldots(\mathcal{T}}_{n \text{ times}}(x[0]))\ldots)) \tag{19}$$

where $\mathcal{T}(\cdot)$ is a nonlinear transformation that maps scalars to scalars and $x[0]$ is the system initial condition. The notation $\mathcal{T}^n(x[0])$ is used to denote the n-th application of the map. It is obvious that a chaotic sequence $\boldsymbol{x}$ is fully described by the map $\mathcal{T}(\cdot)$ and the initial condition $x[0]$. By imposing certain constraints on the map or the initial condition, chaotic sequences of infinite period can be obtained.

A performance analysis of watermarking systems that use sequences generated by piecewise-linear Markov maps and correlation detection is presented in [60]. One property of these sequences is that their spectral characteristics are controlled by the parameters of the map. That is, watermark sequences having uniform distribution and controllable spectral characteristics can be generated using piecewise-linear Markov maps. An example of a piecewise-linear Markov map is the *skew tent map* given by:

$$\mathcal{T} : [0,1] \rightarrow [0,1]$$

$$\text{where} \quad \mathcal{T}(x) = \begin{cases} \frac{1}{\alpha}x, & 0 \le x \le \alpha \\ \frac{1}{\alpha-1}x + \frac{1}{1-\alpha}, & \alpha < x \le 1 \end{cases}, \quad \alpha \in (0,1) \tag{20}$$

The autocorrelation function of skew tent sequences depends only on the parameter α of the skew tent map. Thus, by controlling the parameter α we can generate sequences having any desirable exponential autocorrelation function. The power spectral density of the skew tent map can be easily derived [60]:

$$S_t(\omega) = \frac{1-(2\alpha-1)^2}{12(1+(2\alpha-1)^2 - 2(2\alpha-1)\cos\omega)} \tag{21}$$

By varying the parameter α, either highpass ($\alpha < 0.5$), or lowpass ($\alpha > 0.5$) sequences can be produced. For $\alpha = 0.5$ the symmetric tent map is obtained. Sequences generated by the symmetric tent map possess white spectrum, since the autocorrelation function becomes the Dirac delta function. The control over the spectral properties is very useful in watermarking applications, since the spectral characteristics of the watermark sequence are directly related to watermark robustness against attacks, such as filtering and compression.

The statistical analysis of chaotic watermarking systems that use a correlation detector was undertaken leading to a number of important observations on the watermarking system detection performance [55]. Highpass chaotic watermarks proved to perform better than white ones, whereas lowpass watermarks have the worst performance when no distortion is inflicted on the watermarked signal. The controllable spectral/correlation properties of Markov chaotic watermarks prove to be very important for the overall system performance. Moreover, Markov maps that have appropriate second and third order correlation statistics, like the skew tent map, perform better than sequences with the same spectral properties generated by either Bernoulli or pseudorandom number generators [55].

The simple watermarking systems presented above using either pseudorandom or chaotic generators and either additive or multiplicative embedding would not be robust to geometric transformations e.g., a slight image rotation or cropping, as such attacks would cause a "loss of synchronization" (see Section 5.2) between the watermark signal embedded in the host image and the watermark signal used for the correlation evaluation. This happens because the success of the correlation detection method relies on our ability to correlate the watermarked signal $\boldsymbol{f}_t$ with the watermark $\boldsymbol{w}_d$ in a way that ensures that the n-th sample $w_d(n)$ of the watermark signal will be multiplied in equation (13) with the watermarked signal sample $f_t(n)$ that hosts the same sample of the watermark. In the case of geometric distortions, this "synchronization" will be lost and chances are that the correlation c will be below T, i.e., a false rejection will occur.

A brute force approach could involve the evaluation of the correlation between the watermarked signal and all transformed versions of the watermark. For example, if the image has been subject to rotation by an unknown angle, one can evaluate its correlation with all rotated versions of the watermark and decide that the image is watermarked if the correlation of one of these versions with the signal is above the threshold T. Obviously, this approach has extremely large computational complexity, especially when the image has been subject to a cascade of transforms (e.g., rotation and scaling). Multiple remedies to this problem have been proposed that will be presented in detail in the following sections.

5.4.3 Transformed Watermarks

The idea of transformed watermarks is to construct watermarks transformed in a specific domain whose detection performance is invariant to the geometric distortions of the watermarked image. For example, it is well known that the amplitude of the Fourier transform is translation invariant:

$$f(x_1 + a, x_2 + b) \leftrightarrow F(k_1, k_2)e^{-i(ak_1 + bk_2)} \tag{22}$$

Therefore, if the watermark is embedded in the amplitude of the Fourier transform, it will be insensitive to a spatial shift of the image. The transform space of Mellin-Fourier is one such invariant space. It has been proposed for watermark embedding because, when the watermark is applied to the amplitude of the Fourier transform, it is invariant to translation, rotation and scale of the watermarked image [26]. In order to become invariant to translation, the image is transformed in the Fourier domain and the amplitude of the Fourier is transformed using the log-polar mapping (LPM) defined as follows:

$$x = e^{\rho} \cos\theta \tag{23}$$

$$y = e^{\rho} \sin\theta \tag{24}$$

with $\rho \in \mathcal{R}$ and $\theta \in [0, 2\pi]$. Any rotation in the spatial domain will cause rotation of the Fourier amplitude and translation in the polar coordinate system. Similarly, a scaling of the spatial domain will result in a translation in the logarithmic coordinate system. That is, both rotation and scaling in the spatial domain are mapped to translation in the LPM domain. Invariance to these translations is achieved by taking again the amplitude of the Fourier of the LPM. Taking the Fourier of a log-polar map is equivalent to computing the Mellin-Fourier transform. Combining the DFT and the Mellin-Fourier transform results in rotation, scale and translation (RST) transformation invariance.

The major drawback of the method above is that the various transforms decrease the embedded watermark power. That is, the interpolation applied during the various transforms constitutes an attack to the watermark, thus making it usually undetectable even without any further distortion of the watermarked image. Indeed, in the watermark embedding procedure, the watermark undergoes two inverse DFTs and one inverse LPM with the corresponding interpolations needed. In the detection procedure two DFTs and one LPM are needed as well. Thus, the watermark should be very strong in order to resist all these transforms. Of course, stronger embedding means possible visually perceptible watermarks i.e., quality reduction for the host image. To overcome all these problems, iterative embedding has been proposed in [61]. The watermark is embedded iteratively until it can be reliably detected after the transforms needed in the detection procedure. Even in that case, reliable watermark detection demands very strong embedding and the results are not very promising.

Another reason for which the transform domains have been proposed for watermark embedding is that they also provide robustness against other intentional or unintentional attacks, such as filtering and compression. In such a case, the watermark affects the value of certain transform coefficients and the watermarked signal is obtained by applying the inverse transform on the watermarked coefficients. Transform domain watermarking allows system designers to exploit the transform properties for the benefit of the system. For this purpose, embedding, e.g., in the DFT, DWT, DCT domains has been proposed. For example, one can embed the watermark signal in the low-to-middle frequency coefficients of the DCT transform applied on small image blocks. By doing so one can ensure that the watermark will remain essentially intact by lowpass operations, e.g., JPEG lossy compression or lowpass filtering, since these operations suppress mainly the higher frequencies. Moreover, the distortions imposed on the signal due to watermarking can be held at a reasonably low level as the lower frequencies, whose alterations are known to cause visible distortions, will be kept intact. Embedding in the DWT transform domain has been proposed for increased robustness against JPEG2000 compression. The two-dimensional Radon Wigner transform has been used for watermark embedding in order to obtain robustness against geometric attacks in [62].

5.4.4 Template Watermarks

Another class of watermarks that have been proposed to cope with the problem of geometric transformations is the *template watermarks* class. A template is a structured pattern that is embedded in the image and usually conveys geometric information. Basically, it is an additional signal that is used as a tool for recovering possible geometric transformations of the watermarked image. The template is usually a set of peaks in the DFT domain [63–65]. The peaks are embedded in specific locations so as to define a certain structure that can be easily recovered in the detection procedure. Templates that can be used for watermarking applications are shown in Fig. 4.

As an example, we shall describe in more detail the template proposed in [63]. The template peaks are distributed uniformly along two lines in the DFT domain at certain angles. The angles and radii are chosen pseudo-randomly by using a secret key. The strength of the template can be determined adaptively. Inserting points at a strength equal to the local average value of DFT points plus two standard

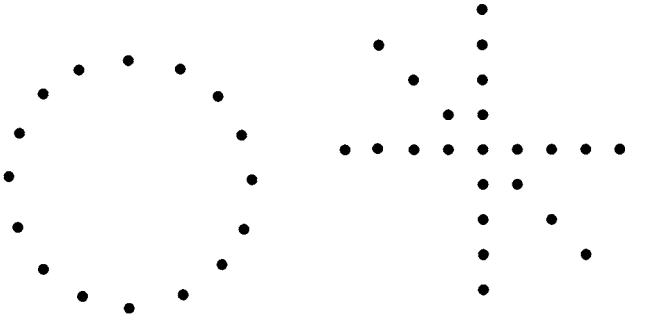

FIGURE 4 Templates proposed for watermarking applications.

deviations yields a good compromise between visibility and robustness during decoding. Peaks in the high frequencies are constructed to be less strong since, in these regions, the average spectra power is usually lower than that of the low frequencies. This type of template is applicable for all images. If someone uses more than two peak lines to construct the template, the cost of the detection algorithm is increased. However, at least two peak lines are required in order to resolve ambiguities arising from the symmetry of the magnitude of the DFT. In particular, after a rotation in the spatial domain, an ambiguity will exist as to whether the rotation was clockwise or counter-clockwise. Depending on features of the specific watermark technology, there are different strategies for the template generation.

The watermark detection process consists of two phases. First the affine transformation (if any) undergone by the watermarked image is determined, then the transformation is inverted and the watermark is detected. For detecting the template, some approaches transform the template matching problem into a point-matching problem. After this problem has been solved, the best candidates for the template points are identified. If an affine transformation has been applied, the identified template points will differ from the original ones. This change is exploited to estimate the applied affine transformation. The corresponding inverse affine transformation is then applied for a better synchronization of the watermark.

The major drawback of the template watermarking is its vulnerability against the template removal attack [66]. The main goal of this attack is to destroy, without any key knowledge, the synchronization pattern of the watermark in order to fool the detection process after an affine transformation of the image. A counterfeiter does not need to know how the specific template in a domain is constructed, since the template applied will always generate some peaks in the target domain used for the template.

The attack can be easily applied by a counterfeiter. In the first phase of the attack, the watermarked image $\boldsymbol{f}_w$ is filtered using a Wiener or median filter and an estimate of the original image $\hat{\boldsymbol{f}}_o$ is derived. Then, the watermark estimate $\hat{\boldsymbol{w}}$ is obtained by subtracting the image $\hat{\boldsymbol{f}}_o$ from the watermarked image. Using the estimate of the watermark, the peaks of the template are extracted in the appropriate transform domain (e.g., DFT). The amplitude of the extracted peaks is modified by replacing the specific amplitude of the watermarked image with the average amplitude value of the neighbors within a certain window. In general, the attacker can apply the same procedure as the template detector in order to extract the template and then he can remove it from the watermarked image. Once the template is removed the watermark is vulnerable to any geometric attack.

5.4.5 Special Structure Watermarks

To solve the problems of the template-based watermarking a different approach has been proposed that involves watermarks whose spatial structure can provide either invariance to certain transforms or a significant reduction in the size of the parameter search space (e.g., the space of possible rotation angles) that has to be searched during detection in order to re-establish synchronization. In this case, self-reference watermarks are mostly used in practice. The self-reference watermarks do not use any additional template to cope with the geometric transforms. Instead, the watermark itself is constructed so as to attain a special spatial structure with desired statistical properties. The most often used watermarks within this approach have self-similar features, i.e., repetition of the same watermark in many spatial directions depending on the final goal and the targeting attack. Spatial self-similarity of the watermarks reduces the search space parameters in case of an affine transformation of the watermarked image [58,67].

An example of self-similar watermarks are the so-called *circularly-symmetric* watermarks. The are defined by [67]:

$$\mathcal{W}(r,\theta) = \begin{cases} 0, & r < r_{\min} \quad \text{or} \quad r > r_{\max} \\ \pm b, & r_{\min} < r < r_{\max} \end{cases} \tag{25}$$

where (n_1, n_2) represent the spatial coordinates and $r = \sqrt{n_1^2 + n_2^2}$, $\theta = \arctan(\frac{n_2}{n_1})$. $r_{\min}$ and $r_{\max}$ are the minimum and maximum radii that define the watermark circular or ring-like support region. b is an integer representing the embedding level or watermark strength. In order for $\mathcal{W}(r,\theta)$ to attain sufficient lowpass characteristics and, thus, be more robust to compression or lowpass filtering, its cyclic or ring-like support domain (25) is additionally divided to a number of s sectors having an extend of $\frac{s}{360}$ degrees. All watermark samples inside a sector for a constant radius are set equal to b or $-b$ according to a pseudorandom number generator initialized with a random key.

A circularly-symmetric watermark has the advantage of robustness against rotation with angles less than $\frac{s}{360}$ degrees. In this case detection is possible without rotating the watermark. When rotation angles are bigger, detection is performed faster since watermark rotation is only needed for multiples of $\frac{s}{360}$ degrees. Spatial self-similarity with respect to the cartesian grid is accomplished by repeating the basic circularly-symmetric watermark at different positions in the image. Additionally, the shifted versions of the basic watermark can be also scaled versions in order to cope with scaling attacks [68] or rotated versions in order to cope with rotation attacks [36].

Another type of self-similar watermarks has been proposed in [69]. The basic watermark is replicated in the image in order to create 4 repetitions of the same watermark. This enables to

have 9 peaks in the autocorrelation function (ACF) that are used in order to recover geometric transformations. The descending character of the ACF peaks shaped by a triangular envelope reduces the robustness of this approach to the geometric attacks accompanied by a lossy compression. The need for computing two discrete Fourier transforms (DFT) of double image size to estimate the ACF creates also some problems for fast embedding/detection in the case of large images.

The known fact that periodic signals have a power spectrum containing peaks can be used to obtain a regular grid of reference points that can easily be employed for recovering from general affine transformation attacks. The existence of many peaks in the magnitude spectrum of the periodically repeated watermark increases the probability to detect geometric transforms even after lossy compression [65]. This fact indicates the enhanced robustness of these watermarks. Furthermore, it is more difficult to remove the peaks in the magnitude spectrum based on a local interpolation in comparison with a template scheme. Such an attack would create considerable visible distortions in the attacked image. A practical algorithm based on the magnitude spectrum of the periodical watermarks is described in [65] for spatial, wavelet or any transform domain. First, the magnitude spectrum is computed from the estimated watermark. Due to the periodicity of the embedded information, the estimated watermark spectrum possesses a discrete structure. Assuming that the watermark is white noise within a block, the power spectrum of the watermark will be uniformly distributed. Therefore, the magnitude spectrum shows aligned and regularly spaced peaks. If an affine distortion was applied to the host image, the peaks layout will be rescaled, rotated and/or sheared, but alignments will be preserved. Therefore, it is easy to estimate any affine geometric distortion from these peaks by fitting alignments and estimating periods of the peaks.

Of course, as in the case of using a watermark template, the counterfeiter may try to estimate the watermark, i.e., to find the peaks on the magnitude spectrum and then remove them by interpolation. Another possible attack is to perform an affine transformation and, afterwards, to embed a periodical signal that will create another regular grid of peaks that may deceive the detector. Finally, a critical issue in this approach is to find the minimum number of watermark samples needed in order to have reliable detection.

5.4.6 Watermarking Systems Involving Optimal Detectors

As mentioned in the beginning of this section, the correlation detector is optimal only in the case of additive watermarks and host signals following a Gaussian distribution. In the case of watermarks embedded in the signal in a multiplicative way or when the host signal follows a different distribution, detectors that are optimal in a certain sense can be constructed using the statistical detection and estimation theory. According to this theory, a decision over the two hypotheses H_0, H_1 can be obtained by evaluating the so-called likelihood ratio $L(\boldsymbol{f}_t)$:

$$L(\boldsymbol{f}_t) = \frac{p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_0)}{p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_1)} \tag{26}$$

In the previous formula $p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_0)$, $p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_1)$ are the probability density functions of the random vector $\boldsymbol{f}_t$, i.e., the watermarked signal, conditioned on the hypotheses H_0, H_1, respectively. A decision is obtained by comparing $L(\boldsymbol{f}_t)$ against a properly selected threshold T. More specifically, a decision to accept hypotheses H_0 or H_1 is taken when $L(\boldsymbol{f}_t) > T$ and $L(\boldsymbol{f}_t) < T$, respectively. The appropriate value of T depends on the performance criterion that we wish to optimize. If an optimal detector with respect to the Neyman-Pearson criterion (i.e., a detector that minimizes the probability of false rejection P_{fr} subject to a fixed probability of false alarm P_{fa}) is to be designed, the threshold value that achieves this minimization is the one evaluated by solving the following equation for T, assuming a fixed, user-provided probability of false alarm $P_{fa} = e$:

$$P_{fa} = e = Prob\{L(\boldsymbol{f}_t) > T|H_1\} = \int_T^{+\infty} p_L(L|H_1)dL \tag{27}$$

In the previous equation $p_L(L|H_1)$ is the pdf of the likelihood ratio L conditioned on the hypothesis H_1.

Various optimal detection schemes, for different situations, have been proposed in the literature. Obviously, in order to obtain an analytic expression for the likelihood ratio (26) and evaluate the threshold T using (27), one has to obtain analytic expressions for the probability density functions that appear in these expressions. To proceed with such derivations, certain assumptions about the statistics of the involved quantities should be adopted. Optimal detectors for watermarks that have been embedded with a multiplicative rule on the magnitude of the DFT coefficients are derived in [35]. The watermark samples $w(n)$ are assumed to be bi-valued, $\{-1, 1\}$, each value having equal probability 0.5. The assumptions adopted in this paper are that the host signal is an ergodic, wide-sense stationary process that follows a first order seperable autocorrelation function and that the DFT coefficients are independent Gaussian random variables, each with a different mean and variance. The authors verify the Gaussianity assumption through a Kolmogorov-Smirnov test. Based on these assumptions, the authors proceed in deriving an analytic expression for the probability distribution of the magnitude of the DFT coefficients. Using this expression, expressions for $p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_0)$, $p_{\boldsymbol{f}_t}(\boldsymbol{f}_t|H_1)$ are derived

and exploited to obtain the following analytic formula for $L(f_t)$:

$$L(f_t) = \prod_{k=1}^{N-1} \frac{\frac{2}{(1+w(k)p)^2} I_0\left(0, \frac{\sigma_I^2(k)-\sigma_R^2}{4\sigma_R^2\sigma_I^2(k)} \frac{f_t(k)^2}{(1+w(k)p)^2}\right)}{\frac{1}{(1+p)^2} \exp\left(-\frac{\sigma_R^2(k)+\sigma_I^2(k)}{4\sigma_R^2(k)\sigma_I^2(k)} \frac{2p(w(k)-1)f_t(k)^2}{(1+w(k)p)^2(1+p)^2}\right) I_0\left(0, \frac{\sigma_I^2(k)-\sigma_R^2(k)}{4\sigma_R^2(k)\sigma_I^2(k)} \frac{f_t(k)^2}{(1+p)^2}\right) + \frac{1}{(1-p)^2} \exp\left(-\frac{\sigma_R^2(k)+\sigma_I^2(k)}{4\sigma_R^2(k)\sigma_I^2(k)} \frac{2p(w(k)+1)f_t(k)^2}{(1+w(k)p)^2(1-p)^2}\right) I_0\left(0, \frac{\sigma_I^2(k)-\sigma_R^2(k)}{4\sigma_R^2(k)\sigma_I^2(k)} \frac{f_t(k)^2}{(1-p)^2}\right)} \quad (28)$$

In the previous expression, p is the embedding factor, $I_0()$ is the modified Bessel function and $\sigma_I^2(k)$, $\sigma_R^2(k)$ are the variances of the imaginary and the real part of the k-th DFT coefficient for which analytic expressions have been also derived in [35]. It should be also noted that f_t is the host signal, which, in this case, consists of the magnitudes of the DFT coefficients of the host signal. In order to evaluate the threshold in (27) for a certain $P_{fa} = e$, the pdf $p_L(L|H_1)$ of the likelihood ratio L conditioned on the hypotheses H_1 needs to be derived. The authors assume that L attains a Gaussian distribution and proceed in an experimental evaluation of its mean $\widehat{\mu}$ and variance $\widehat{\sigma}$. Using these quantities, the threshold T can be found to be:

$$T = \widehat{\mu} - \widehat{\sigma}\sqrt{2}\ erf^{-1}(2\,P_{fa} - 0.5) \quad (29)$$

Similar approaches have been proposed by others in order to derive optimal detectors under other optimality criteria or embedding domains with different distributions [31–34].

5.5 Watermarking with Side Information

5.5.1 Informed Embedding Watermarking

A watermarking system that uses the information of the host signal (also called cover signal) in the coding/embedding procedure is called *informed coding/embedding watermarking* system and is described in Fig. 5. An informed embedding scheme exploits the information conveyed in the host signal $\boldsymbol{f}$ for generating a robust watermark $\boldsymbol{w}$ in the spatial domain. The watermark generation can be seen as an optimization problem with respect to a detection statistic or to a robustness measure as it will described in the following.

The simplest way to construct an informed embedding watermarking system is the so-called *precancellation,* which has been proposed in communication systems [70]. In such a system the interference of the host signal is completely removed by setting the watermark to be embedded as $\boldsymbol{w}_a = \boldsymbol{w} - \boldsymbol{f}_o$. Thus, after embedding in the host signal the watermarked signal to be transmitted is equal to $\boldsymbol{f}_w = \boldsymbol{f}_o + \boldsymbol{w}_a = \boldsymbol{w}$. That is, the host signal is completely replaced by the watermark. It is obvious that this watermarking system is optimal with respect to host signal cancellation but it is unacceptable for real applications, since the fidelity constraint is violated. That is, the host signal is not preserved at all. In most cases where fidelity constraints are imposed in the watermark generation and embedding procedure, the problem of generating an informed embedding watermarking system is dealt with as an optimization problem. The problem can be defined in two alternative ways:

- Construct a watermark to be embedded in the given host signal that maximizes watermark robustness, under the constraint that the perceptual distortion will be inside a prescribed limit.
- Construct a watermark to be embedded in the given host signal that minimizes the perceptual distortion, under the constraint that the robustness of the resulting watermark will be inside a prescribed limit.

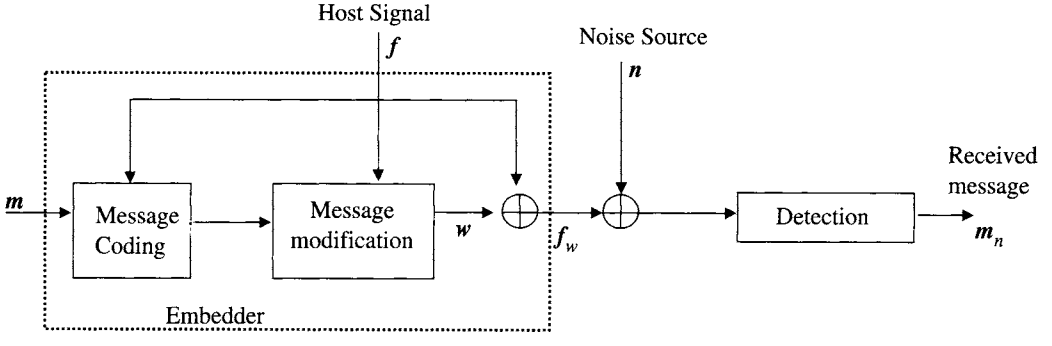

FIGURE 5 Watermarking with informed coding and informed embedding.

More complicated optimization problems can be set where both perceptual distortion and robustness are optimized inside prescribed limits. It is obvious that, in order to solve the above optimization problems, someone has to define measures of robustness and perceptual distortion. This task is not trivial and many researchers proposed such measures under rather simplistic assumptions. For example, a detection statistic can be assumed to measure the robustness of a watermarking system. This assumption is not valid in most cases, since the robustness should be mostly determined for specific attacks.

If someone considers a watermarking system that uses linear correlation as detection statistic, then it can be assumed that increased robustness means higher correlation value between the watermarked signal and the watermark. Thus, imposing a constraint on robustness can be considered as generating watermarks that have constant correlation value with the watermarked signal (i.e., $\boldsymbol{f}_w \boldsymbol{w}^T = c$). This can be interpreted as moving all the signals to be watermarked to a hyperplane in the high dimensional space of the host signal that is perpendicular to the watermark vector. This hyperplane corresponds to constant correlation with the watermark signal. In such a watermarking scheme, the more distant the hyperplane from the detection threshold is, the more robust is the watermarking scheme. Having constant linear correlation between the watermark and the watermarked signal corresponds to constant robustness and, thus, the constrained optimization problem is reduced to finding a watermarked signal, i.e., a vector on this hyperplane that is less perceptually distorted from the host signal. The optimal watermarked signal is obviously the one that has the minimum Euclidean distance from the host signal, if the MSE is used as a fidelity measure.

However, if someone wants to use normalized instead of linear correlation as watermark detection statistic, then the above solution is not optimal. Indeed, for certain host signals the normalized correlation statistic may be lower than the detection threshold even if the linear correlation is constant. This is valid since constant normalized correlation corresponds to a hypercone in the high-dimensional space of the host signal and not to a hyperplane. Thus, if someone wants to utilize normalized correlation as detection statistic then he should construct the watermark in a way that all watermarked signals lie on a hypercone. The corresponding optimization problem (i.e., minimize perceptual distortion having constant robustness) is solved by finding the vector on the hypercone that has minimum Euclidean distance from the host signal and thus minimizes the MSE.

In the above analysis, the robustness measure that has been used was the detection statistic. However, this assumption is rather simplistic and, in most cases, optimizing the detection statistic prior to an attack does not guarantee optimal behavior against this attack. Thus, someone may design the informed embedding watermarking algorithm having as robustness measure an estimate of the robustness against a specific attack. One simple attack that can be used for this purpose is the additive white Gaussian noise (AWGN), that has been widely used in the literature to model the attacks incurred in a transmission channel. In this case, the optimization problem can be described as the minimization of the perceptual distortion after watermark embedding having constant robustness against AWGN. Assuming that the detection statistic is the normalized correlation, it can be easily proven that the optimal solution lies on a hyperboloid inside the hypercone of the normalized correlation [23]. Finding the optimal watermarked vector corresponds to finding the vector that lies on the hyperboloid that has minimum distance from the host signal. The analytic solution of this problem involves the solution of a quartic equation and, thus, for simplicity reasons the solution can be found by exhaustive search.

From the above analysis it is obvious that the watermarking scheme designer that wants to use informed embedding, has to decide on the robustness and perceptual distortion measures that will be used and on the appropriate detection statistic. Unfortunately, no robustness measure can be defined for the majority of the common attacks. Furthermore, if someone can define a robustness measure against one attack, it is most probable that this measure will not be valid for other attacks. Another problem is that effective perceptual distortion measures are a research subject themselves since, in case of images, measuring the distortion involves the modelling of the human visual system (HVS). For the above reasons, the informed embedding watermarking systems cannot outperform the practical, often heuristically designed, watermarking systems under common watermark attacks such as image rotation, scaling and cropping, despite their solid mathematic foundation.

5.5.2 Informed Coding Watermarking

Informed embedding algorithms yield better performance than their blind competitors. However, the previously described algorithms do not fully exploit the information of the host signal in that they do not use this information during the message coding procedure. That is, when multiple bit watermarks are considered, the message coding procedure can use the information of the host signal in order to select an appropriate code vector for the specific host signal that minimizes the perceptual distortion of the watermarked signal and maximizes the robustness in terms of appropriate measures. These algorithms that take advantage of the side information during the message encoding procedure are called *informed coding* watermarking algorithms. It is possible to combine informed embedding and informed coding in order to achieve significantly better performance.

Informed coding has been described based on a theoretical result obtained by Costa [71]. This result implies that the

capacity of a watermarking system might not depend on the distribution of the host signal but rather on the distortions the watermark must survive (i.e., the desired level of robustness). Costa introduced the idea of *"writing on dirty paper"* which can be described as follows: *Having a piece of paper covered with independent dirty spots of normally distributed intensity, someone has to write a message using a limited amount of ink. The dirty paper, with the message on it, is then transmitted to someone else, and acquires more normally distributed dirt on the way. If the receiver cannot distinguish between the ink and the dirt, how much information can be reliably send?* Costa presented this problem as an analogy to the communication channel shown in Fig. 6. The channel has two independent white Gaussian noise sources. The first one represents the host signal and the second one represents the attacks that the watermarked signal may face. Before the transmitter chooses a signal to send, it is informed that the host signal is $\boldsymbol{f}_o$. The transmitter should send a signal that is limited by a power constraint that corresponds to the limited ink on the dirty paper example or to the perceptual distortions constraint in a real watermarking system. Costa showed that, for the above communication scheme, the first noise source has no effect on the channel capacity. That is, if a watermarking scheme behaves enough like this dirty paper channel, its maximum message payload should not depend on the host media.

Unfortunately, a real watermarking scheme has substantial differences from the dirty paper communication scheme:

- The two noise sources, especially the host signal, are rarely Gaussian.
- The distortions implied by the channel (i.e., the second AWGN source) are dependent on the watermarked signal.
- The fidelity constraint (i.e., the constraint on perceptual distortions) cannot be represented by the constrained power, since this is analogous to the MSE, which is a poor perceptual distortion estimate.

For the above reasons it is not straightforward to prove that the capacity of watermarking is independent of the distribution of the host signal. Other researchers studied channels that model better a real watermarking scheme [72]. They have shown that, even if the second noise source is not AWGN but a hostile adversary that intentionally tries to remove the watermark, the dirty paper theory result still holds. They have also found that, even when the unwatermarked signal is not Gaussian, the result is approximately true.

The basic idea of applying Costa's theory to watermarking is using several alternative code words for each message instead of a single one. The code word that will be used for transmitting the message is the one that minimizes the interference of the host signal. Let us consider a simple channel in which only two different host signals, $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$, can be transmitted. That is, the first AWGN source can generate only these two signals, while the second AWGN source can generate any noise signal. Suppose that the transmitter should sent one of the M possible messages $\boldsymbol{m}$. If the host signal to be transmitted is $\boldsymbol{f}_1$ then the optimal solution for encoding the message $\boldsymbol{m}$ is to select $\boldsymbol{w} = \boldsymbol{m} - \boldsymbol{f}_1$ as the signal to be transmitted, since it is known to the transmitter that the interference will be $\boldsymbol{f}_1$. The constraint of the limited power of the watermark $\boldsymbol{w}$ defines a hypersphere in the multidimensional space of the host signal that is centered in $\boldsymbol{f}_1$. All the watermarked versions of $\boldsymbol{f}_1$ containing the different encoded messages should lie inside the hypersphere and should be as far as possible from each other. If the host signal to be transmitted is the signal $\boldsymbol{f}_2$ then the optimal solution is to construct the watermark as $\boldsymbol{w} = \boldsymbol{m} - \boldsymbol{f}_2$. Again all the encoded messages should lie inside a hypersphere centered in $\boldsymbol{f}_2$. It is obvious that, if the two host signals are sufficiently different from each other, the two hyperspheres do not overlap and two different sets of code vectors can be used for encoding the messages depending on the host signal. The receiver uses also two sets of code vectors and selects the code vector that is closer, in terms of Euclidean distance, to the received signal. This communication scheme is reliable even though the receiver does not have any side information about the host signal.

The measures that indicate the robustness of the described scheme are firstly the radius of the hyperspheres which allows for many different messages to be encoded (i.e., be included and sufficiently scattered inside the hypersphere), and secondly, the distance between the host signals which

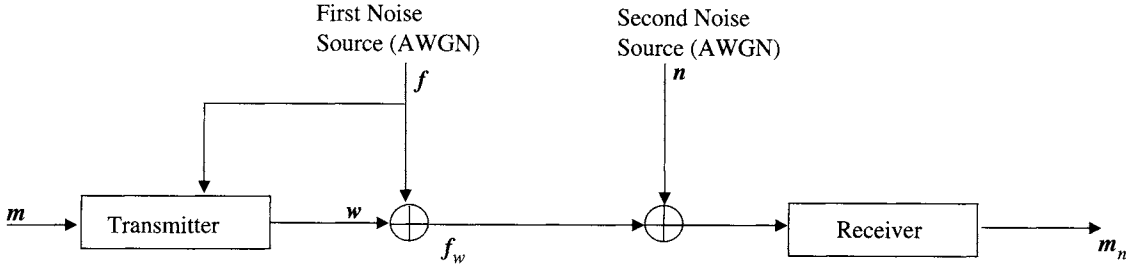

FIGURE 6 Communication channel model proposed by Costa.

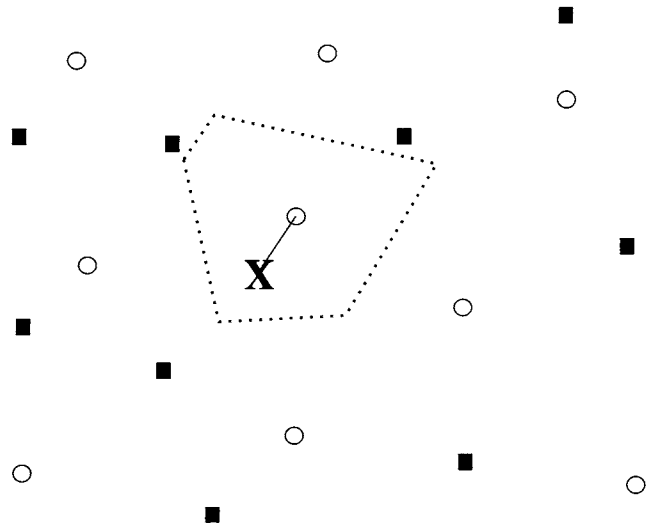

FIGURE 7 Quantization index modulation for information embedding. Two different quantizers are depicted. The input signal is quantized with the O-quantizer that corresponds to the first of the two possible messages.

indicates the amount of noise that can be dealt with without causing shifting to a set of code vectors different from the one used for encoding.

In a more realistic situation the hyperspheres of the host signals will overlap and more generally, the number of different host signals will be unlimited and thus generating a set of code vectors for each host signal will be infeasible. In order to solve these problems and design a scheme that can deal with the constraints of a real watermarking system the *quantization index modulation* (QIM) method has been proposed [24]. QIM refers to embedding information by first modulating an index or a sequence of indices with the embedded information and then quantizing the host signal with the associated quantizer or sequence of quantizers. A simple example with two quantizers that are used to encode two different messages is shown in Fig. 7. The size of the quantization cells, one of which is shown in this figure, determine the distortion that can be tolerated. In QIM, each message is encoded using a different quantizer that covers the entire multidimensional space of the host signal, in a structured manner. During the encoding procedure, the host signal is quantized using the quantizer that corresponds to the message to be embedded. The distortion induced by the quantization step is associated with the fidelity constraint whereas the minimum distance between the nodes of two different quantizers is associated with the maximum perturbation that can be tolerated by the watermarking scheme. A practical implementation of QIM is the *dither modulation*, where the so-called dither quantizers are used during message encoding. The property of these quantizers that renders them appropriate for QIM is that the quantization cells and reconstruction points of any given quantizer are shifted versions of the quantization cells and the reconstruction points of any other quantizer. This property is exploited in the message encoding procedure, where each possible message maps uniquely onto a different dither vector (i.e., quantizer) and the host signal is quantized using the resulting quantizer. Many practical realizations of QIM have been proposed to deal with the problem of computational complexity [73]. Regular lattices are used as realization of the quantizers. The drawback of these lattice codes is that they are not robust against valumetric scaling such as contrast changes in images. A small change on the contrast of a watermarked image may result to errors in message decoding.

5.5.3 Perceptual Masking

As we have previously seen, one of the watermark characteristics is imperceptibility which can be defined using two different terms: *fidelity* and *quality*. Fidelity is a measure of the similarity between the original host signal and the watermarked signal and, thus, should be as high as possible. Image quality is a measure of the viewer's satisfaction when viewing an image. A high quality image looks good (i.e., it has no obvious processing artifacts). Thus, fidelity is a two argument measure, whereas quality is a single argument one.

It is obvious from the above definitions it is obvious that image quality may be reduced after watermark embedding, only if the quality of the original host signal is high. However, there are many cases where the quality of the original host signal is medium or low (for example, surveillance images). In these cases, the objective is to maintain the image quality. Preserving the quality after watermark embedding is achieved when the fidelity is high, since in this case, the watermarked signal is indistinguishable from the original host signal. High fidelity of the watermarked image assures quality preservation, whereas high quality of the watermarked image does not assure high fidelity.

In the design of watermarking schemes using informed embedding or coding, we have seen that the MSE is the more simple measure of the perceptual distortion implied by the watermark. Constructing watermarks resulting in constant MSE between the watermarked and the original signal was considered as a way of keeping constant fidelity. However, in a real situation two different watermarked signals that have the same MSE when compared to their originals are not perceived as having the same fidelity, because MSE does not correlate well with subjective fidelity. Instead of the MSE, other more complex models have been proposed for measuring visual fidelity that are closer to the way HVS works.

The properties of the HVS that are taken into account when constructing a watermarking scheme are the following:

- *Sensitivity*, which is related to the eye response to direct stimuli. For example, a common sensitivity measure is the minimum brightness required for an eye to perceive certain spatial or temporal image frequencies.

- *Masking*, that is a measure of the observer's response to one stimulus (watermark), when a second "masking" stimulus (host image) is also present. Thus, by using visual masking, we can hide more information in a textured image region than in a uniform region since texture provides a better mask.
- *Pooling*, which is a single estimate of the overall change in the visual appearance of an image that is caused by combining the perception of separate distortions.

The important issue is the adaptation of the watermark to the properties of the HVS, i.e., content-adaptive watermarking. Assuming we are given a masking function constructed according to the HVS properties, we wish to embed the watermark into the host image by keeping it under the threshold of visual imperceptibility. The perceptually adaptive watermarking was mainly inspired by the achievements in the perceptual-based image/video compression, especially on the early stages of watermarking technology development. Therefore, the perceptual masking was mainly addressed in the transform domain.

One of the first approaches to content-adaptive digital watermarking was proposed in [74]. This approach was based on the just noticeable difference (JND), which was computed in the DCT domain using a mask developed for lossy JPEG compression by Watson [75]. Another sophisticated approach takes into account the luminance and contrast sensitivity of the human visual system, and differentiates between edge and texture regions visibility. The resulting mask was used in the spatial image.

The next generation of content-adaptive digital watermarking methods utilizes the idea that the perceptual masking should be performed directly in the transform domain (mostly wavelet domain) to be matched with JPEG2000 image compression standard. The perceptual masking performed in the transform domain has a number of advantages. First, the watermark embedding process is accomplished in the same domain as image/video compression that renders watermarking-on-the-fly possible. Second, it allows JPEG2000 compliance that ensures additional robustness of the watermarking algorithms to lossy JPEG2000 compression. It should be noted that the mask and watermark energy allocation should be simultaneously adapted to the other types of lossy compression algorithms such as DCT-based JPEG.

The factors that should be taken into account for the mask design matching HVS properties are the following:

- Background luminance sensitivity
- Contrast sensitivity depending on image subband or resolution
- Orientation sensitivity (anisotropy)
- Edge and pattern (texture) masking

The background luminance sensitivity is described according to Weber's law: the eye is less sensitive to noise in bright regions. For the subband sensitivity, experiments have proven that the eye is differently stimulated depending on the frequency, orientation and luminance of the image content. Finally, the higher the texture energy of an image region is, the lower is the visual sensitivity with respect to content changes in this region. The combination of high frequency edges and low frequency luminance variance are used to calculate the texture masking factor. The edge proximity factor refers to the fact that the higher the luminance difference of an edge is, the less is the visual sensitivity at the vicinity of the edge. However, as we get away from the edge, the visual sensitivity is reestablished.

To establish a bridge between the rate-distortion theory and the content-adaptive watermarking, it was proposed to use a stochastic texture perceptual mask based on a noise visibility function (NVF) developed earlier only for the spatial image domain [76]. This result was also extended to the wavelet domain aiming at including the multi-resolution paradigm in the stochastic framework to take into account a modulation transfer function (MTF) of the HVS and to match the proposed watermarking algorithm with the image compression standard JPEG2000. This practically means that the assigned watermark strength depends on the image sub-band. Such a modification leads to a non-white watermark spectrum matched with the MTF that was not a case for the spatial domain based version of the NVF. The second reason to use wavelet domain embedding is the desire to incorporate the HVS anisotropy to different spatial directions in the perceptual mask. The spatial domain version of the NVF uses the isotropic image decomposition based on the extraction of a local mean from the original image or its high-pass filter output. In the wavelet domain, the image coefficients in 3 basic spatial directions, i.e., vertical, horizontal and diagonal ones, are received as a result of the decomposition that reflects the HVS anisotropy properties better. As a result, the watermark strength varies with orientations in the proposed mask.

6 Image Content Integrity and Authentication Watermarking

In this section, we focus on how watermarks can be used to assist in maintaining and verifying image content integrity. Image editing is in many cases legitimate and desirable. However, for many applications modifications introduced by image editing may be malicious or may affect image interpretation, thus resulting in image tampering. For example, tampering with images acquired for medical applications may result to misdiagnosis. Changes in images that are used as evidence in a criminal trial can result in a wrong conviction. Image content authentication mainly focuses on the development of *fragile* or *semi-fragile* watermarks.

In a real world application scenario, the image owner, embeds a watermark so that either she or the user, is able to check if the image has been manipulated by someone. Sometimes, the owner can become subsequently the user as well. The information conveyed by the watermark is related to the authenticity and/or integrity of the product. In certain cases, it is interesting to detect which image regions have been altered. In general, the attacker intends to alter the content of the product without distorting the watermark. This goal is the primary distinction between watermarking for image authentication and for copyright protection. Alternatively, the attacker may try to copy the watermark from a watermarked image to another image. This is the *forgery attack.* The watermark should generally be destroyed by any authentication attack on the image. In some cases, it is desirable that the watermark withstands certain attacks that do not degrade image quality (e.g., high quality lossy compression). A usual application occurs when photographs or movies are sent by news agencies (owners) to newspapers or TV channels (users). The news broadcaster wants to be assured that the images are authentic before broadcasting. Thus, the authenticity check is a procedure that concerns primarily the user of the image. If someone wants to use an image authentication algorithm in front of a court, then the user is the judge who needs to be assured for the image authenticity.

The purpose of an image authentication algorithm is to provide the user with all the information needed about image authenticity. Image authentication algorithms usually produce an authenticity measure in the range [0,1]. If required that the image data should not change at all, any image alteration is prohibited and the threshold for accepting an image as authentic should be very high (or close to 1). Thus, the use of an authenticity measure is enough to assure authenticity. This type of authentication is called *complete* or *exact* or *hard* authentication. However, in certain cases of image content authentication (e.g., in the case of stored surveillance images), legitimate distortions such as high quality image compression, are allowed, i.e., an image is considered authentic even after high quality lossy compression. In such cases, the image authenticity measure is lower than 1 even for authentic images, where the image content remains unaltered. Thus, the use of the authenticity measure alone is not adequate for deciding about the image authenticity. In this case, the image authentication algorithm should have the ability of tamper proofing, i.e., to detect the tampered image regions that change the image content. This type of authentication is called *content* or *selective* or *soft* authentication. We shall use only the terms complete and content authentication for the rest of this section.

It is obvious that the requirements of an image authentication scheme are different, depending on whether we are considering complete or content authentication. In complete authentication, the watermark should be destroyed even if a single bit has been changed in the watermarked image. Localization of the tampered regions is a desirable feature in complete authentication. In content authentication schemes, the basic requirement is that the watermark should be destroyed whenever an illegitimate distortion is applied to the watermarked image and should not be destroyed when the distortions are legitimate. It is obvious that the critical point is the characterization of a distortion as legitimate or illegitimate. This issue cannot be dealt with universally but it is specific for each particular application. For example, uniform geometric distortions applied on an image may be legitimate for a press agency but illegitimate for a medical application. The distortions that are usually considered legitimate for many applications are those that do not change in any manner the content of the image. Such distortions are the high quality lossy compression, mild noise filtering, geometric transformations etc. In any case, the content authentication should have the ability of tampering localization. That is, the image regions that have been subject to any illegitimate distortion should be detected.

In the extreme case of content authentication, only the semantic meaning of an image should be preserved for an image to be considered as authentic. The semantic content of the image can be represented in the form of a feature vector. In this case, content authentication means that the feature vector of the image should always be extracted unaltered. In this case, the image is considered authentic even if heavy distortions have significantly reduced the image quality. Another characteristic of some content authentication algorithms is the property of self-restoration. Self-restoration is ability of the algorithm to restore the image to its original content even if the content has been distorted by illegitimate manipulations. To this end, *self-embedding* has been proposed in [77]. In this approach, a low quality version of the image is embedded in the original image using LSB modulation. The low quality information of each image block is embedded in the LSBs of a different image block so as to assure that either the original image block or the embedded one will be available at the restoration phase.

A general watermarking framework used for image authentication was presented in [8]. This framework is comprised of three steps: the watermark generation, embedding and detection procedures. Let $\mathcal{R}$ and $\mathcal{Z}$ denote the set of real and integer numbers, respectively, and $\mathcal{U} = \{0, 1, 2\}$. Given an image $f(\mathrm{x}) : \mathcal{D} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{Z}$ and a watermark key $k \in \mathcal{Z}$, a ternary watermark $w(\mathrm{x}) : \mathcal{G} \subseteq \mathcal{Z}^2 \rightarrow \mathcal{U}$ can be produced by the watermark generation procedure, where $\boldsymbol{x}$ denotes the coordinate vector of a pixel. In the watermark generation either chaotic techniques or pseudorandom number generators can be used for producing the watermark $w(\boldsymbol{x})$. The watermark embedding procedure of the watermark $w(\mathrm{x})$ in the image $f(\mathrm{x})$ is given by:

$$f_w(\boldsymbol{x}) = f(\boldsymbol{x}) \otimes w(\boldsymbol{x}) \tag{30}$$

where $f_w(\boldsymbol{x})$ is the watermarked image and $\otimes$ is a generalized superposition operator which includes appropriate data truncation and quantization, if needed. The watermark embedding procedure can be applied either in the spatial domain or in the other domains (e.g., in the DFT, DWT, DCT domains). The watermark detection produces a binary decision on image authenticity and the watermark detection credibility measured by the watermark detection ratio. Additionally, the watermark detection produces an image denoting the unaltered (authentic) regions of the watermarked image.

In the following, a brief description of the most important schemes proposed for image authentication is presented. The first class of authentication methods is based on using a separate header (*digital signature*) which must be known to the receiver that will perform the authenticity check. These schemes, although not based on watermarking will be briefly reviewed since they appeared first in the literature. The development of a "trustworthy digital camera" was proposed in [78]. A digital image is captured by a camera and then it is passed through a hash function. The output of the hash function is encrypted by the photographer's private key and a separate authentication signal is created. In order to ensure image authentication, the encrypted signal is decrypted by the photographer's public key and the hashed version of the original image is compared to that of the received image. A compression tolerant method for image authentication is proposed in [79]. The proposed scheme is based on the extraction of feature points that are almost unaffected by lossy compression. The major drawbacks of this method are the need of a separate header for storing the digital signature and the low accuracy in the detection of the tampered regions.

An authentication method that gives a distortion measurement instead of a binary decision on image authenticity is proposed in [80]. It does not require a separate signature file or header for image authentication, but it can not detect the image regions that are authentic, if selective modifications to fine image details have been made. A method for image authentication that is based on changing the least significant bit (LSB) of the image pixels is proposed in [81]. The method can detect alterations that are made in several image regions. However, it is not robust to high quality lossy compression.

Another class of authentication methods uses a transform domain such as DFT, DWT, DCT etc., for watermark embedding. These methods are usually more robust against attacks such as filtering and compression [82]. A method in which a look-up table is needed for image authentication is proposed in [83]. The watermark is embedded in the DCT domain and attains robustness to lossy compression. This method is based on [84], where the LUT is used for watermark embedding in the spatial domain. However, it is not robust against lossy compression and, furthermore, is very sensitive to the forgery attack presented in [85]. A method for wavelet transform domain authentication watermarking is proposed in [86]. The detection of the tampered image regions is not addressed and the compression of the watermarked image is not supported. Instead, the authentication algorithm is integrated within the SPIHT codec. This means that the authentication algorithm protects the already compressed image. An image watermarking method for tamper proofing and authentication is proposed in [87]. The watermark is embedded in the discrete wavelet transform domain by quantizing the corresponding wavelet coefficients. The method is robust against compression and succeeds in detecting alterations of the host image, whether they are produced by changing fine image details or by high quality image compression. The major drawback of the proposed method is that, when combined attacks are made to the watermarked image, the authenticity check is based on the low DWT levels, i.e., levels 4 and 5. These levels contain a small number of coefficients and a decision about image authenticity based on the watermark detection at these levels is unreliable, since each coefficient represents a region of 16×16 or 32×32 image pixels. Robustness against combined attacks is not supported.

An attack that has been developed for image authentication algorithms is presented in [85] and succeeds in copying the watermark from a watermarked image to another image that is not watermarked. The attack has been successfully applied to the methods presented in [81,88,89]. Knowledge of the logo to be embedded is assumed in order to forge the watermark. The main disadvantage that renders these methods vulnerable to the attack is that they are block-wise independent. That is, the watermark in a certain block depends neither on the entire image nor on other watermarked blocks. Another fact that helps the attacker to estimate the inserted watermark statistically, is that critical watermark parameters, such as the positions of the watermarked pixels and the embedded logo are known to the attacker.

Another technique for image authentication has been proposed in [90,91]. It is based on a modification of an established watermarking technique for copyright protection [8]. The image authentication algorithm generates a watermark according to the owner's private key. Subsequently, the watermark is imperceptibly embedded in the image. In the authentication detection procedure, the watermark is extracted from the image and a measure of tampering is produced for the entire image. The algorithm detects the regions of the image that are altered/unaltered and, thus, are considered non-authentic/authentic. The alterations that are produced by a relatively mild compression and do not change significantly the quality of the image are also detected. An example of an image authentication procedure using the image "Opera of Lyon" (*http://www.petitcolas.net/fabien/watermarking/image_database/index.html*), which has been used as a reference image for watermark benchmarking, is depicted in Fig. 8.

FIGURE 8 (a) Original watermarked image. (b) Tampered watermarked image. (c) Tampered regions.

Differentiating between malicious and incidental manipulations in content authentication remains an open issue. Exploitation of robust watermarks with self-restoration capabilities for image authentication is another research topic. The authentication of certain regions instead of the whole image when only some regions are tampered has also attracted the attention of the watermarking community.

Acknowledgment

The authoring of this chapter has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

References

[1] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, 2002.

[2] S. Craver and J. Stern, "Lessons learnd from sdmi," in *IEEE Workshop on Multimedia Signal Processing, MMSP 01*, Cannes, France, 213–218, October 2001.

[3] M. Barni, "What is the future for watermarking? (part i)," *IEEE Signal Processing Magazine*, 20, 5, 55–59, September 2003.

[4] M. Barni, "What is the future for watermarking? (part ii)," *IEEE Signal Processing Magazine*, 20, 6, 53–59, November 2003.

[5] M. Barni and F. Bartolini, *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*, Marcel Dekker, 2004.

[6] S. Katzenbeisser and F. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, 2000.

[7] N. Nikolaidis and I. Pitas, "Benchmarking of watermarking algorithms," in *Intelligent Watermarking Techniques*, J.-S. Pan, H.-C. Huang, and L. Jain, Eds., 315–347. World Scientific Publishing, 2004.

[8] G. Voyatzis and I. Pitas, "The use of watermarks in the protection of digital multimedia products," *Proceedings of the IEEE*, 87, 7, 1197–1207, July 1999.

[9] N. Nikolaidis and I. Pitas, "Digital image watermarking: an overview," in *Int. Conf. on Multimedia Computing and Systems (ICMCS'99)*, Florence, Italy, I, 1–6, 7–11 June 1999.

[10] Special issue on Identification & Protection of Multimedia Information," *Proceedings of the IEEE*, 87, 7, July 1999.

[11] Special issue on signal processing for data hiding in digital media and secure content delivery," *IEEE Transactions on Signal Processing*, 51, 4, April 2003.

[12] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn, "Information hiding—a survey," *Proceedings of the IEEE*, 87, 7, 1062–1078, July 1999.

[13] I. Cox, M. Miller, and J. Bloom, "Watermarking applications and their properties," in *International Conference on Information Technology: Coding and Computing 2000*, Las Vegas, 6–10, March 2000.

[14] A. Nikolaidis, S. Tsekeridou, A. Tefas, and V. Solachidis, "A survey on watermarking application scenarios and related attacks," in *2001 IEEE International Conference on Image Processing (ICIP 2001)*, Thessaloniki, Greece, 7–10 October 2001.

[15] M. Barni and F. Bartolini, "Data hiding for fighting piracy," *IEEE Signal Processing Magazine*, 21, 2, 28–39, March 2004.

[16] R. Venkatesan, S.M. Koon, M.H. Jakubowski, and P. Moulin, "Robust image hashing," in *IEEE International Conference on Image Processing*, Vancouver, Canada, October 2000.

[17] V. Monga and B. Evans, "Robust perceptual image hashing using feature points," in *IEEE International Conference on Image Processing, ICIP 04*, Singapore, October 2004.

[18] J.A. Bloom, I.J Cox, T. Kalker, M.I. Miller, and C.B.S. Traw, "Copy protection for dvd video," *Proceedings of the IEEE*, 87, 7, 1267–1276, July 1999.

[19] F. Bartolini, A. Tefas, M. Barni, and I. Pitas, "Image authentication techniques for surveillance applications," *Proceedings of the IEEE*, 89, 10, 1403–1418, October 2001.

[20] A.M. Alattar, "Smart images using digimarcs watermarking technology," in *IS&T/SPIEs 12th International Symposium on Electronic Imaging*, San Jose, CA, 3971, 25 January 2000.

[21] T. Furon, I. Venturini, and P. Duhamel, "A unified approach of asymmetric watermarking schemes," in *SPIE Electronic Imaging, Security and Watermarking of Multimedia Contents*, San Jose, CA, 4314, 269–279, January 2001.

[22] T. Furon and P. Duhamel, "An asymmetric watermarking method," *IEEE Transactions on Signal Processing*, 51, 4, 981–995, April 2003.

[23] I.J. Cox, M.I. Miller, and A.L. McKellips, "Watermarking as communications with side information," *Proceedings of the IEEE*, 87, 7, 1127–1141, July 1999.

[24] B. Chen and G. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, 47, 4, 1423–1443, May 2001.

[25] I.J. Cox, J. Kilian, F.T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Transactions on Image Processing*, 6, 12, 1673–1687, December 1997.

[26] J.K. Ruanaidh and T. Pun, "Rotation, scale and translation invariant spread spectrum digital image watermarking," *Elsevier Signal Processing, Sp. Issue on Copyright Protection and Access control*, 66, 3, 303–317, 1998.

[27] J. Cannons and P. Moulin, "Design and statistical analysis of a hash-aided image watermarking system," *IEEE Transactions on Image Processing*, 13, 10, 1393–1408, October 2004.

[28] A. Nikolaidis and I. Pitas, "Region-based image watermarking," *IEEE Transactions on Image Processing*, 10, 11, 1726–1740, November 2001.

[29] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Transactions on Image Processing*, 10, 5, 783–791, May 2001.

[30] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2d dft domain," *IEEE Transactions on Image Processing*, 10, 11, 1741–1753, November 2001.

[31] Juan R. Hernandez, Martin Amado, and Fernando Perez-Gonzalez, "Dct-domain watermarking techniques for still images: Detector performance analysis and a new structure," *IEEE Transactions on Image Processing*, 9, 1, 55–68, Januray 2000.

[32] M. Barni, F. Bartolini, A. DeRosa, and A. Piva, "A new decoder for the optimum recovery of nonadditive watermarks," *IEEE Transactions on Image Processing*, 10, 5, 755–766, May 2001.

[33] M. Barni, F. Bartolini, A. DeRosa, and A. Piva, "Optimum decoding and detection of multiplicative watermarks," *IEEE Transactions on Signal Processing*, 51, 4, 1118–1123, April 2003.

[34] Qiang Cheng and T.S. Huang, "Robust optimum detection of transform domain multiplicative watermarks," *IEEE Transactions on Signal Processing*, 51, 4, 906–924, April 2003.

[35] V. Solachidis and I. Pitas, "Optimal detector for multiplicative watermarks embedded in the dft domain of non-white signals," *Journal of Applied Signal Processing*, accepted for publication, 2004.

[36] A. Tefas and I. Pitas, "Robust spatial image watermarking using progressive detection," in *Proc. of 2001 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2001)*, Salt Lake City, Utah, 1973 –1976, 7–11 May 2001.

[37] D.P. Mukherjee, S. Maitra, and S.T. Acton, "Spatial domain digital watermarking of multimedia objects for buyer authentication," *IEEE Transactions on Multimedia*, 6, 1, 1–15, February 2004.

[38] M. Barni, F. Bartolini, V. Cappelini, and A. Piva, "A DCT-domain system for robust image watermarking," *Elsevier Signal Processing*, 66, 3, 357–372, 1998.

[39] W.C. Chu, "Dct-based image watermarking using subsampling," *IEEE Transactions on Multimedia*, 5, 1, 34–38, March 2003.

[40] C.-Y. Lin, M. Wu, J. Bloom, I. Cox, M. Miller, and Y. Lui, "Rotation, scale, and translation resilient watermarking for images," *IEEE Transactions on Image Processing*, 10, 5, 767–782, May 2001.

[41] Y. Wang, J. Doherty, and R. Van Dyck, "A wavelet-based watermarking algorithm for ownership verification of digital images," *IEEE Transactions on Image Processing*, 11, 2, 77–88, February 2002.

[42] ITU, "Methodology for the subjective assessment of the quality of television pictures," *ITU-R Recommendation BT.500-11*, June 2002.

[43] A. Netravali and B. Haskell, *Digital Pictures, Representation and Compression*, Plenum Press, 1988.

[44] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun, "Attack modelling: Towards a second generation benchmark," *Signal Processing*, 81, 6, 1177–1214, June 2001.

[45] A. Tefas and I. Pitas, "Multi-bit image watermarking robust to geometric distortions," in *in CD-ROM Proc. of IEEE Int. Conf. on Image Processing (ICIP'2000)*, Vancouver, Canada, 710–713, 10–13 September 2000.

[46] F.A.P. Petitcolas, "Watermarking schemes evaluation," *IEEE Signal Processing Magazine*, 17, 5, 58–64, September 2000.

[47] Martin Kutter, Sviatoshlav Voloshynovskiy, and Alexander Herrigel, "The watermark copy attack," in *Electronic Imaging 2000, Security and Watermarking of Multimedia Content II*, 3971, 2000.

[48] S. Craver, N. Memon, B.-L. Yeo, and M. Yeung, "Resolving rightful ownerships with invisible watermarking techniques: Limitations, attacks and implications," *IEEE Journal of Selected Areas in Communications*, 16, 4, May 1998.

[49] F. Petitcolas, M. Steinebach, F. Raynal, J. Dittmann, C. Fontaine, and N. Fates, "A public automated web-based evaluation service for watermarking schemes: Stirmark benchmark," in *SPIE Electronic Imaging 2001, Security and Watermarking of Multimedia Contents*, San Jose, CA, 4314, January 2001.

[50] S. Pereira, S. Voloshynovskiy, M. Madueno, S. Marchand-Maillet, and T. Pun, "Second generation benchmarking and application oriented evaluation," in *Information Hiding Workshop III*, Pittsburgh, PA, April 2001.

[51] V. Solachidis, A. Tefas, N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, and I. Pitas, "A benchmarking protocol for watermarking methods," in *Proc. of ICIP '01*, Thessaloniki, Greece, 1023–1026, 7–10 October 2001.

[52] M. Simon, J. Omura, R. Scholtz, and B Levitt, *Spread Spectrum Communications Handbook*, McGraw-Hill, 1994.

[53] I. Pitas and T.H. Kaskalis, "Applying signatures on digital images," in *Proc. of 1995 IEEE Workshop on Nonlinear Signal and Image Processing (NSIP'95)*, N. Marmaras, Greece, 460–463, 20–22 June 1995.

[54] N. Nikolaidis and I. Pitas, "Copyright protection of images using robust digital signatures," in *Proc. of ICASSP'96*, Atlanta, GA, 4, 2168–2171, 1996.

[55] A. Tefas, A. Nikolaidis, N. Nikolaidis, V. Solachidis, S. Tsekeridou, and I. Pitas, "Performance analysis of correlation-based watermarking schemes employing markov chaotic sequences," *IEEE Transactions on Signal Processing*, 51, 7, 1979–1994, July 2003.

[56] G. Voyatzis and I. Pitas, "Chaotic watermarks for embedding in the spatial digital image domain," in *Proc. of ICIP'98*, Chicago, USA, II, 432–436, 4–7 October 1998.

[57] A. Nikolaidis and I. Pitas, "Comparison of different chaotic maps with application to image watermarking," in *Proc. of ISCAS'00*, Geneva, Switzerland, V, 509–512, 28–31 May 2000.

[58] S. Tsekeridou, N. Nikolaidis, N. Sidiropoulos, and I. Pitas, "Copyright protection of still images using self-similar chaotic watermarks," in *Proc. of ICIP'00*, Vancouver, Canada, 411–414, 10–13 September 2000, to appear.

[59] N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, A. Tefas, V. Solachidis, and I. Pitas, "Applications of chaotic signal processing techniques to multimedia watermarking," in *Proceedings of the IEEE workshop on Nonlinear Dynamics in Electronic Systems*, Catania, Italy, 1–7, 18–20 May 2000.

[60] A. Tefas, A. Nikolaidis, N. Nikolaidis, V. Solachidis, S. Tsekeridou, and I. Pitas, "Statistical analysis of markov chaotic sequences for watermarking applications," in *2001 IEEE International Symposium on Circuits and Systems (ISCAS 2001)*, Sydney, Australia, 57–60, 6–9 May 2001.

[61] C. Lin, M. Wu, J.A. Bloom, I.J. Cox, M.L. Miller, and Y.M. Lui, "Rotation, scale, and translation resilient public watermarking for images," in *Proc. of SPIE:Electronic Imaging 2000*, 3971, 90–98, January 2000.

[62] I. Djurovic S. Stankovic and I. Pitas, "Watermarking in the space/spatial domain using two-dimensional radon wigner distribution," *IEEE Transactions on Image Processing*, 10, 4, 650–658, April 2001.

[63] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE Trans. on Image Processing*, 9, 6, 1123–1129, June 2000.

[64] S. Pereira, J. Ruanaidh, F. Deguillaume, G. Csurka, and T. Pun, "Template based recovery of fourier-based watermarks using log-polar and log-log maps," in *Proc. of ICMCS'99*, Florence, Italy, I, 870–874, 7–11 June 1999.

[65] S. Voloshynovskiy, F. Deguillaume, and T. Pun, "Multibit digital watermarking robust against local non-linear geometrical transformations," in *ICIP 2001, Int. Conf. Image Processing*, October 2001.

[66] A. Herrigel, S. Voloshynovskiy, and Y. Rytsar, "Template removal attack," in *Proc. of SPIE: Electronic Imaging 2001*, January 2001.

[67] V. Solachidis and I. Pitas, "Circularly symmetric watermark embedding in 2-d dft domain," in *Proc. of ICASPP'99*, Phoenix, Arizona, 3469–3472, 15–19 March 1999.

[68] S. Tsekeridou and I. Pitas, "Embedding self-similar watermarks in the wavelet domain," in *Proc. of ICASSP'00*, Istanbul, Turkey, 1967–1970, 5–9 June 2000.

[69] M. Kutter, "Watermarking resisting to translation, rotation and scaling," in *Proc. of SPIE*, November 1998.

[70] H.C. Papadopoulos and C.E.W. Sundberg, "Simultaneus broadcasting of analogus fm and digital audio signals by means of precanceling techniques," in *Proceedings of the IEEE International Conference on Communications*, 728–732, 1998.

[71] M. Costa, "Writing on dirty paper," *IEEE transactions on Information Theory*, 29, 439–441, 1983.

[72] P. Moulin and J.A. O'Sullivan, "Information-theoretic analysis of information hiding," *IEEE Trans. on Information Theory*, 49, 3, 563–593, March 2003.

[73] B. Chen and G. Wornell, "Quantization index modulation methods for digital watermarking and information embedding in multimedia," *Journal of VLSI Signal Processing*, 27, 7–33, 2001.

[74] C.I. Podilchuk and W. Zeng, "Image-adaptive watermarking using visual models," *IEEE Journal on Selected Areas in Communications*, 16, 4, 525–539, May 1998.

[75] A.B. Watson, "Dct quantization matrices optimizedfor individual images," *Human vision, visual processing and digital display*, SPIE-1903, 202–216, 1993.

[76] N. Baumgartner S. Voloshynovskiy, A. Herrigel and T. Pun, "A stochastic approach to content adaptive digital image watermarking," in *Proc. of International Workshop on Information Hiding*, Dresden, Germany, 211–236, September 1999.

[77] J. Fridrich and A. Goljan, "Images with self-correcting capabilities," in *Proc. IEEE Conf. on Image Processing*, 25–28, October 1999.

[78] G.L. Friedman, "The trustworthy digital camera: Restoring credibility to the photographic image," *IEEE Transactions on Consumer Electronics*, 39, 4, 905–910, November 1993.

[79] S. Bhattacharjee and M. Kutter, "Compression tolerant image authentication," in *Proc. of ICIP'98*, Chicago, I, 425–429, 4–7 October 1998.

[80] B. Zhu, M.D. Swanson, and A.H. Tewfik, "Transparent robust authentication and distortion measurement technique for images," in *Proc. of DSP'96*, Loen, Norway, 45–48, September 1996.

[81] P.W. Wong, "A public key watermark for image verification and authentication," in *Proc. of ICIP'98*, Chicago, I, 425–429, 4–7 October 1998.

[82] C.-Y. Lin and S.-F. Chang, "Semi-fragile watermarking for authenticating jpeg visual content," in *SPIE EI'00*, 3971, 2000.

[83] M. Wu and B. Liu, "Watermarking for image authentication," in *Proc. of ICIP'98*, Chicago, I, 425–429, 4–7 October 1998.

[84] M.M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Proc. of ICIP'97*, Atlanta, GA, II, 680–683, October 1997.

[85] M. Holliman and N. Memon, "Counterfeiting attacks on oblivious blockwise independent invisible watermarking

schemes," *IEEE Trans. on Image Processing*, 9, 3, 432–441, March 2000.

[86] L. Xie and G.R. Arce, "Joint wavelet compression and authentication watermarking," in *Proc. of ICIP'98*, Chicago, II, 427–431, 4–7 October 1998.

[87] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proceedings of the IEEE*, 87, 7, 1167–1180, July 1999.

[88] F. Mintzer, G.W. Braudaway, and M.M. Yeung, "Effective and ineffective digital watermarks," in *Proc. of ICIP'97*, Atlanta, GA, III, 9–12, October 1997.

[89] P.W. Wong, "A watermark for image integrity and ownership verification," in *Proc. of IS&T PINC Conf*, Portland, OR, 1997.

[90] A. Tefas and I. Pitas, "Image authentication based on chaotic mixing," in *in CD-ROM Proc. of IEEE Int. Symposium on Circuits and Systems (ISCAS'2000)*, Geneva, Switzerland, 216–219, 28–31 May 2000.

[91] A. Tefas and I. Pitas, "Image authentication and tamper proofing using mathematical morphology," in *Proc. of European Signal Processing Conf. (EUSIPCO'2000)*, Tampere, Finland, 5–8 September 2000.