A TUTORIAL ON DIGITAL WATERMARKING

Fernando Pérez-González and Juan R. Hernández

Dept. Tecnologías de las Comunicaciones,
ETSI Telecom., Universidad de Vigo, 36200 Vigo, Spain
email: fperez@tsc.uvigo.es, jhernan@tsc.uvigo.es

ABSTRACT

This paper gives a tutorial on the techniques and reference models used in digital watermarking. Distorsions, attacks and applications are described in some detail. Finally, the need for benchmarking is discussed.

1. INTRODUCTION

In his wonderful book *The Codebreakers* [1], D. Kahn recounts one of the stories in the *Histories* of Herodotus in which Histiaeus tatooed a message in the shaven head of a slave and waited for the new hair to grow before sending him to Aristagoras at Miletus with instructions to shave –again– the slave's head. Obviously, bandwidth was not a concern at those times, but methods haven't changed so much when compared with the state of the art in digital watermarking.

Since the publication of a seminal work by Tanaka et al. in 1990 [2], we have witnessed an extraordinary growth of techniques for copyright protection of different types of data, especially multimedia information. This interest is not surprising in view of the simplicity of digital copying and dissemination: digital copies can be made identical to the original and later reused or even manipulated. Cryptography is an effective solution to the distribution problem, but in most instances has to be tied to specialized –and costly– hardware to create tamper-proof devices that avoid direct access to data in digital format (even so, there exist software/hardware tools that allow to resample the analog output of the device with decent results). Moreover, most cryptographic protocols are concerned with secured communications instead of ulterior copyright infringements. For instance, access control in set-top-boxes used for digital television demodulation and decoding succeed in avoiding unathorized access to programs that are being broadcast in scrambled form [3] but fail in precluding further storage and illegal dissemination actions.

There is then an increasing need for software (or in the worst case, hardware) that allows for protection of ownership rights, and it is in this context where watermarking techniques come to our help. Perceptible marks of ownership or authenticity have been around for centuries in the form of stamps, seals, signatures or classical watermarks, nevertheless, given current data manipulation technologies, imperceptible digital watermarks are mandatory in most applications. A digital watermark is a distinguishing piece of information that is adhered to the data that it is intended to protect, this meaning that it should be very difficult to extract or remove the watermark from the watermarked object. Since watermarking can be applied to various types of data, the imperceptibility constraint will take different forms, depending on the properties of the recipient (i.e., the human senses in most practical cases).

In addition to imperceptibility there are some desirable characteristics that a watermark should possess, which are somewhat related to the so-called *robustness issue*. First, the watermark should be *resilient to standard manipulations* of unintentional as well as intentional nature. Second, it should be *statistically unremovable*, that is, a statistical analysis should not produce any advantage from the attacking point of view. Finally, the watermark should *withstand multiple watermarking* to facilitate traitor tracing, as discussed in Section

Work partially funded by CICYT under project TIC-96-0500-C10-10

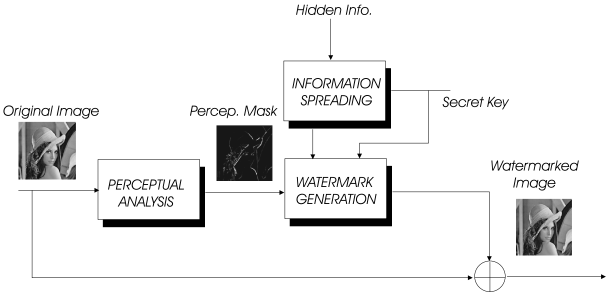

Figure 1: *Watermark insertion unit*

4. However, the type of manipulations and the attacker expected computational power heavily depend on the application.

Watermarking, like cryptography, also uses secret keys to map information to owners, although the way this mapping is actually performed considerably differs from what is done in cryptography, mainly because the watermarked object should keep its inteligibility. In most watermarking applications embedment of additional information is necessary. This information includes identifiers of the owner, recipient and/or distributor, transaction dates, serial numbers, etc. which play a crucial role in adding value to watermarking products. This will become clear when we briefly discuss a typical scenario in Section 3.

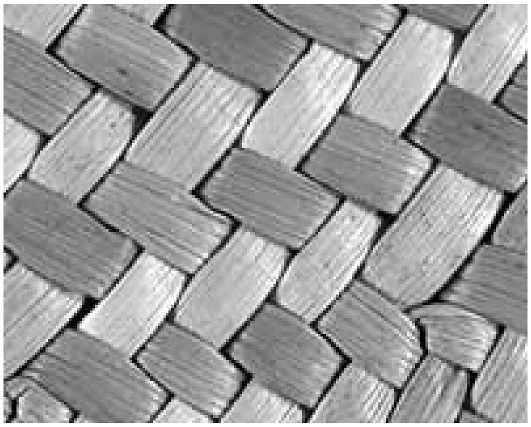

Figure 2: *Original ‘Fabric’ image*

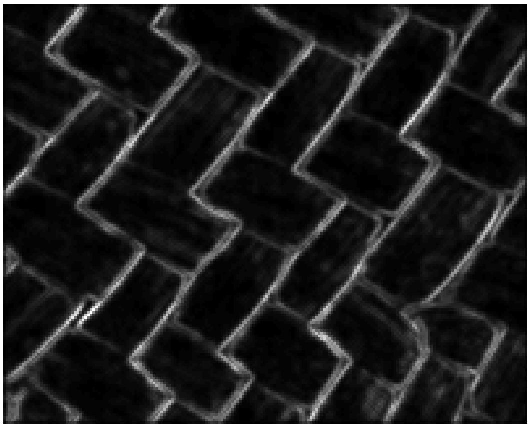

Figure 3: *Perceptual Mask*

2. STRUCTURE OF A TYPICAL WATERMARKING SYSTEM

Every watermarking system consists at least of two different parts: watermark embedding unit and watermark detection and extraction unit. Figure 1 shows an example of embedding unit for still images. The unmarked image is passed through a perceptual analysis block that determines how much a certain pixel can be altered so that the resulting watermarked image is indistinguishable from the original. This takes into account the human eye sensitivity to changes in flat areas and its relatively high tolerance to small changes in edges. After this so-called *perceptual-mask* has been computed, the information to be hidden is shaped by this mask and spread all over the original image. This spreading technique is similar to the interleaving used in other applications involving coding, such as compact disc storage, to prevent damage of the information caused by scratches or dust. In our case, the main reason for this spreading is to ensure that the hidden information survives cropping of the image. Moreover, the way this spreading is performed depends on the secret key, so it is difficult to recover the hidden information if one is not in possession of this key. In fact, a similar technique is used in spread spectrum systems (more precisely, in Code-Division Multiple Access) to extract the desired information from noise or other users. Additional key-dependent uncertainty can be introduced in pixel amplitudes (recall that the perceptual mask imposes only an upper limit). Finally, watermark is added to the original image.

Figure 3 represents the perceptual mask that results after analyzing the image presented in Figure 2. Higher intensity (i.e., whiter) levels imply that higher perturbations can be made at those pixels without perceptible distortion. Thus, the higher capacity areas for hiding information correspond to edges. These masks are computed by

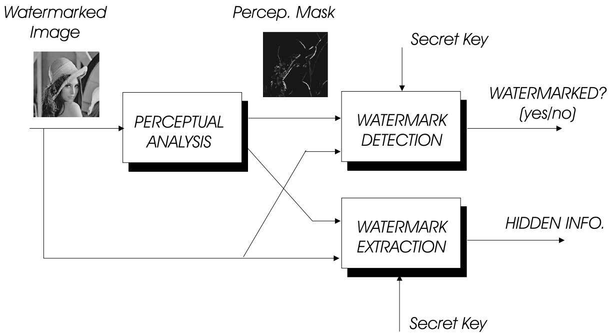

Figure 4: *Watermark detection and extraction unit*

using some known results on how the human eye works in the spatial domain. Different results are obtained when working on other domains, such as the DCT (Discrete Cosine Transform) or Wavelet transform. In fact, when working on the DCT coefficients domain one may take advantage of the relative independence between the maximum allowable perturbations at every coefficient. This is udseful when dealing with the mask for watermarking purposes.

Figure 4 shows the typical configuration of a watermark detection and extraction unit. Watermark detection involves deciding whether a certain image has been watermarked with a given key. Note then that a watermark detector produces a binary output. Important considerations here are the probability of correct detection P_D (i.e., the probability of correctly deciding that a watermark is present) and the probability of false alarm P_F (i.e., the probability of incorrectly deciding that an image has been watermarked with a certain key). These two measures allow us to compare different watermarking schemes: One method will be superior if achieves a higher P_D for a fixed P_F. Note also that for a watermarking algorithm to be useful it must work with extremely low probabilities of false alarm. Watermark detection is usually done by correlating the watermarked image with a locally generated version of the watermark at the receiver side. This correlation yields a high value when the watermark has been obtained with the proper key. As we have shown in [4], it is possible to improve the performance of the detector by eliminating original image-induced noise with signal processing. It is worthy of remark that some authors [5] propose using the original image in the detection process. Although this simplifies further treatment of the watermark in the reeiver end, it is quite unrealistic for most applications, particularly those related to E-commerce.

Once the presence of the watermark has been correctly detected, it is possible to extract the hidden information. The procedure is also generally done by means of a cross-correlation but in this case, an independent decision has to be taken for every information bit with a sign slicer. In fact, we have also shown that this correlation structure has not been well-founded and significant improvements are achievable when image statistics are available. For instance, the widely-used DCT coefficients used in the JPEG and MPEG-2 standards are well approximated by *generalized gaussian probability density functions* that yield a considerably different extraction scheme. Obviously, when extracting the information the most adequate parameter for comparison purposes is the *probability of bit error* P_b, identical to that used in digital communications. This is not surprising because watermarking creates a hidden (sometimes called *steganographic*) channel on which information is conveyed.

3. A REFERENCE MODEL FOR COPYRIGHT MANAGEMENT

In this Section, we briefly describe a subset of the Common Reference Set, developed under the European project IMPRIMATUR which defines a conceptual framework for the development of electronic copyright management systems. The usefulness of this model lies in its ability to map the different tasks and agents involved in copyright management into clearly defined entities. This is extremely important considering that the main application of watermarking is related to electronic commerce with multimedia data, especially in Internet. This simplified model is represented in Figure 5.

The term *creator* covers every possible content that begins the value chain, including composers, photographers, video creators, etc. The *creator provider* makes contents available to the public in a form that can be later distributed, for instance, through WWW servers by *distributors*. The former would include publishers, multimedia companies, agencies, etc. The *rights holder* (not depicted) manages entitlements and responsibil-

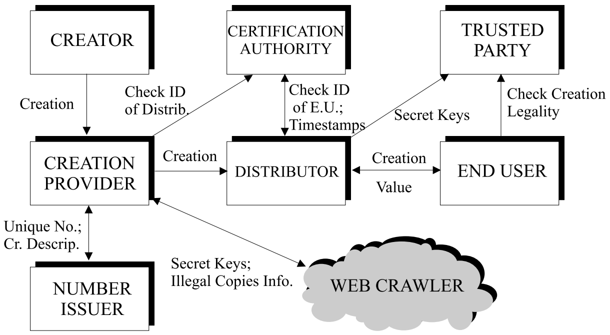

Figure 5: *Reference Model for Copyright Management*

ities for the creator and facilitates licensing and royalties collection. Each creation is identified by a *creation identification number* that allows the rights holder to sell exploitation licences. An example is the ISBN that identifies books. This number should become part of the hidden information described in Section 1, and it is issued by an authorized organization that has been aproved by a community of creator providers. The *purchaser* represents the end user of this model and includes not only individuals but organizations. It is at this level where the cost of the technology for copyright management is critical, which means that specific hardware (even software) should be avoided. At this side, the user should be able to check whether an object has been legally acquired. This means that it should perform the watermark detection and extraction tasks described in Section 2. Unfortunately, since no public key watermarking schemes have been developed yet, it is not possible to carry out such tasks without tamperproof hardware that use the secret key. An alternative to this is resorting to a *trusted third party* that checks the validity of the watermark and sends the pertinent part of the hidden information encrypted to the end user (this latter operation may use public key protocols). Note that a *certification authority* may be required to verify the identity of the various agents involved in the process and to issue timestamps whose utility will be clarified in Section 4. The certification authority and the trusted third party may be joined in a singke entity, depending on the application. Finally, note that if rights are to be purchased, there should exist exchange of information between the end user and the rights holder.

There is also a vital part of the system, not explicitely included in the IMPRIMATUR model, which is an agent that searches the network for illegal copies. It would be too naive to think that all end users would check the rights of the objects that acquire. This is currently happening in Internet with MP3-compressed music files, which are downloaded by most users without considering copyright infringements. Thus, it is important that the creation provider has a way of locating illegal distribution sites. Obviously, the mere existence of this web spider makes the development of attacks at protocol level (see [6]) rewarding.

4. DISTORSIONS AND ATTACKS

In practice, a watermarked object may be altered either on purpose or accidentally, so the watermarking system should still be able to detect and extract the watermark. Obviously, the distorsions are limited to those that do not produce excessive degradations, since otherwise the transformed object would be unusable. These distorsions also introduce a degradation on the performance of the system as measured by the probabilities defined in the previous section (i.e., P_D and P_b would decrease for a fixed P_F). For intentional attacks, the goal of the attacker is to maximize the reduction in these probabilities while minimizing the impact that his/her transformation produces on the object; this has to be done without knowing the value of the secret key used in the watermarking insertion process, which is where all the security of the algorithm lies.

Next, we introduce some of the best known attacks. Some of them may be intentional or unintentional, depending on the application:

Additive Noise. This may stem in certain applications from the use of D/A and A/D converters or from transmission errors. However, an attacker may introduce perceptually shaped noise (thus, imperceptible) with the maximum unnoticeable power. This will typically force to increase the threshold at which the correlation detector works.

Filtering. Low-pass filtering, for instance, does not introduce considerable degradation in watermarked images or audio, but can dramatically affect the performance, since spread-spectrum-like watermarks have a non negligible high-frequency spectral contents.

Cropping. This is a very common attack since in many cases the attacker is interested in a small portion of the watermarked object, such as parts of a certain picture or frames of a video sequence. With this in mind, in order to survive, the watermark needs to be spread over the dimensions where this attack takes place.

Compression. This is generally an unintentional attack which appears very often in multimedia applications. Practically all the audio, video and images that are currently being distributed via Internet have been compressed. If the watermark is required to resist different levels of compression, it is usually advisable to perform the watermark insertion task in the same domain where the compression takes place. For instance, DCT-domain image watermarking is more robust to JPEG compression than spatial-domain watermarking.

Rotation and Scaling. This has been the true battlehorse of digital watermarking, especially because of its success with still images. Correlation-based detection and extraction fail when rotation or scaling are performed on the watermarked image because the embedded watermark and the locally generated version do not share the same spatial pattern anymore. Obviously, it would be possible to do exhaustive search on different rotation angles and scaling factors until a correlation peak is found, but this is prohibitively complex. Note that estimating the two parameters becomes simple when the original image is present, but we have argumented against this possibility in previous sections. In [7] the authors have shown that although the problem resembles synchronization for digital communications, the techniques applied there fail loudly. Some authors have recently proposed the use of rotation and scaling-invariant transforms (such as the Fourier-Mellin [8]) but this dramatically reduces the capacity for message hiding. In any case, publicly available programs like Strirmark break the uniform axes transformation by creating an imperceptible non-linear resampling of the image [6] that renders invariant transforms unusable. In audio watermarking it is also quite simple to perform a non-linear transformation of the time axis that considerably difficults watermark detection.

Statistical Averaging. An attacker may try to estimate the watermark and then ‘unwatermark’ the object by substracting the estimate. This is dangerous if the watermark does not depend substantially on the data. Note that with different watermarked objects it would be possible to improve the estimate by simple averaging. This is a good reason for using perceptual masks to create the watermark.

Multiple Watermarking. An attacker may watermark an already watermarked object and later make claims of ownership. The easiest solution is to timestamp the hidden information by a certification authority.

Attacks at Other Levels. There are a number of attacks that are directed to the way the watermark is manipulated. For instance, it is possible to circumvent copy control mechanisms discussed below by superscrambling data so that the watermark is lost [9] or to deceive web crawlers searching for certain watermarks by creating a presentation layer that alters they way data are ordered. The latter is sometimes called ‘mosaic attack’ [6].

5. APPLICATIONS

In this section we discuss some of the scenarios where watermarking is being already used as well as other potential applications. The list given here is by no means complete and intends to give a perspective of the broad range of bussiness possibilities that digital watermarking opens.

Video Watermarking. In this case, most considerations made in previous sections hold. However, now the temporal axis can be exploited to increase the redundancy of the watermark. As in the still images case, watermarks can be created either in the spatial or in the DCT domains. In the latter, the results can be directly extrapolated to MPEG-2 sequences, although different actions must be taken for I, P and B frames. Note that perhaps the set of attacks that can be performed intentionally is not smaller but definitely more expensive than for still images.

Audio Watermarking. Again, previous considerations are valid. In this case, time and frequency masking properties of the human ear are used to conceal the watermark and make it inaudible. The greatest difficulty lies in synchronizing the watermark and the watermarked audio file, but techniques that overcome this problem have been proposed.

Hardware/Software Watermarking. This is a good paradigm that allows us to understand how almost every kind of data can be copyright protected. If one is able to find two different ways of expressing the same information, then one bit of information can be concealed, something that can be easily generalized to any number of bits. This is why it is generally said that a perfect compression scheme does not leave room for watermarking. In the hardware context, Boolean equivalences can be exploited to yield instances that use different types of gates [10] and that can be addressed by the hidden information bits. Software can be also protected not only by finding equivalences between instructions, variable names, or memory addresses, but also by altering the order of non-critical instructions. All this can be accomplished at compiler level.

Text Watermarking. This problem, which in fact was one of the first that was studied within the information hiding area can be solved at two levels. At the printout level, information can be encoded in the way the textlines or words are separated (this facilitates the survival of the watermark even to photocopying). At the semantic level (necessary when raw text files are provided), equivalences between words or expressions can be used, although special care has to be taken not to destruct the possible intention of the author.

Executable Watermarks. Once the hidden channel has been created it is possible to include even executable contents, provided that the corresponding applet is running on the end user side.

Labeling. The hidden message could also contain labels that allow for example to annotate images or audio. Of course, the annotation may also been included in a separate file, but with watermarking it results more difficult to destroy or loose this label, since it becomes closely tied to the object that annotates. This is especially useful in medical applications since it prevents dangerous errors.

Fingerprinting. This is similar to the previous application and allows acquisition devices (such as video cameras, audio recorders, etc) to insert information about the specific device (e.g., an ID number) and date of creation. This can also be done with conventional digital signature techniques but with watermarking it becomes considerably more difficult to excise or alter the signature. Some digital cameras already include this feature.

Authentication. This is a variant of the previous application, in an area where cryptographic techniques have already made their way. However, there are two significant benefits that arise from using watermarking: first, as in the previous case, the signature becomes embedded in the message, second, it is possible to create 'soft authentication' algorithms that offer a multivalued 'perceptual closeness' measure that accounts for different unintentional transformations that the data may have suffered (an example is image compression with different levels), instead of the classical yes/no answer given by cryptography-based authentication. Unfortunately, the major drawback of watermarking-based authentication is the lack of public key algorithms that force either to put secret keys in risk or to resort to trusted parties.

Copy and Playback Control. The message carried by the watermark may also contain information regarding copy and display permissions. Then, a secure module can be added in copy or playback equipment to automatically extract this permission information and block further processing if required. In order to be effective, this protection approach requires agreements between content providers and consumer electronics manufacturers to introduce compliant watermark detectors in their video players and recorders. This approach is being taken in Digital Video Disc (DVD).

Signalling. The imperceptibility constraint is helpful when transmitting signalling information in the hidden channel. The advantage of using this channel is that no bandwidth increase is required. An interesting application in broadcasting consists in watermarking commercials with signalling information that permits an automatic counting device to assess the number of times that the commercial has been broadcast during a certain period. An alternative to this would require complex recognition software.

6. THE NEED FOR BENCHMARKING

Although watermarking technology is relatively young, papers and products related to the subject have mushroomed. However, very few efforts have been exerted to provide user/provider requirements, tools and procedures that would eventually end with standardization efforts. This scene is not different from what it was in cryptography some years ago, where algorithms flourished but civil cryptanalysis was underdeveloped. Unfortunately, it seems that this is not the best time to 'play' proposing yet a new watermarking algorithm, since this lack of references deters right holders and technology suppliers from using watermarking techniques, which in turn difficults E-Commerce applications that the general public is eager to absorbe.

Two research lines will eventually prove their ability to solve this problem: first, theoretical work in watermarking-analysis will bring objective performance measures and will improve existing methods in a more knowledgeable way than mere trial-and-error; second, development of benchmarking technology will help to assess the superiority of certain algorithms and to find a representative multimedia database that will end with the ad-hoc procedure that it is being used. Different standards, such as MPEG-4 and JPEG-2000 await that these efforts come true.

7. REFERENCES

[1] D. Kahn, *The Codebreakers; The Comprehensive History of Secret Communication from Ancient Times to the Internet.* Scribner, December 1996.

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

[3] B. Macq and J. Quisquater, "Cryptology for digital TV broadcasting," *Proceedings of the IEEE*, vol. 83, pp. 944–957, February 1995.

[4] J. R. Hernández, F. Pérez-González, J. M. Rodríguez, and G. Nieto, "Performance analysis of a 2d-multipulse amplitude modulation scheme for data hiding and watermarking of still images," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 510–524, May 1998.

[5] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "A secure, robust watermark for multimedia," in *Information Hiding* (G. Goos, J. Hartmanis, and J. Leeuwen, eds.), vol. 1174 of *Lecture Notes in Computer Science*, (University of Cambridge, UK), pp. 185–206, Springer-Verlag, May 1996.

[6] F. Petitcolas, R. Anderson, and M. Kuhn, "Attacks on copyright marking systems," in *Information Hiding* (D. Aucsmith, ed.), vol. 1525 of *Lecture Notes in Computer Science*, (Berlin), pp. 218–238, Springer-Verlag, 1998.

[7] J. R. Hernández, F. Pérez-González, and J. M. Rodríguez, "Coding and synchronization: A boost and a bottleneck for the development of image watermarking," in *Proc. of the COST #254 workshop on Intelligent Communications*, (L'Aquila, Italia), pp. 77–82, SSGRR, June 1998.

[8] A. Herrigel, J. O'Ruanaidh, H. Petersen, S. Pererira, and T. Pun, "Secure copyright protection techniques for digital images," in *Information Hiding* (D. Aucsmith, ed.), vol. 1525 of *Lecture Notes in Computer Science*, (Berlin), pp. 169–190, Springer-Verlag, 1998.

[9] I. J. Cox and J.-P. M. G. Linnartz, "Some general methods for tampering with watermarks," *IEEE J. Select. Areas Commun*, vol. 16, pp. 587–593, May 1998.

[10] J. Lach, W. Mangione-Smith, and M. Potknojak, "Fingerprinting digital circuits on programmable hardware," in *Information*

Hiding (D. Aucsmith, ed.), vol. 1525 of *Lecture Notes in Computer Science*, (Berlin), pp. 16–31, Springer-Verlag, 1998.