WATERMARK DETECTION: BENCHMARKING PERSPECTIVES

N.Nikolaidis, V.Solachidis, A.Tefas, I.Pitas

Department of Informatics
Aristotle University of Thessaloniki
Thessaloniki 54006, Greece
Tel,Fax: +30310-996304
e-mail: {nikolaid,vasilis,tefas,pitas@zeus.csd.auth.gr}

ABSTRACT

Benchmarking of watermarking algorithms is a complicated task that requires examination of a set of mutually dependent performance factors (algorithm complexity, decoding/detection performance, and perceptual quality). This paper will focus on detection/decoding performance evaluation and try to summarize its basic principles. A methodology for deriving the corresponding performance metrics will also be provided.

1. INTRODUCTION

Watermarking research evolved with a tremendous speed in the last few years [1, 2]. Up to now, performance evaluation and method comparison has been carried out in a non-standardized way, with no concrete supporting evidence. With the watermarking technology entering into a more mature era, backed up by concrete mathematical foundations, it is about time that a benchmarking methodology will be devised. This development would benefit both the watermarking technology suppliers, by allowing them to fine-tune their algorithms, and the technology users, by providing a systematic way of comparing existing solutions. Overall, the establishment of concrete benchmarking foundations would give the watermarking community the credibility that is largely lacking. A number of efforts towards this direction are underway and have resulted in the development of benchmarking platforms as well as to the introduction of the basic benchmarking principles [2, 3, 4, 5, 6, 7].

The mutually dependent parameters that should be taken into account when judging the performance of a watermarking algorithm are the following:

- **Detection/decoding performance:** In the context of this paper we make the distinction between watermark detection and message decoding. The term watermark detection is used to denote the ability of the watermarking algorithm to declare the presence or absence of a watermark on an image. As soon as the algorithm declares the image to be watermarked the embedded message (if any) should be decoded. Thus, 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 image.
- **Algorithmic Complexity** The complexity of the watermark embedding and the watermark detection-message decoding steps should be evaluated. The easiest but not the most appropriate way to measure complexity is by recording execution time in a fixed hardware/software suite. Other means of measuring the complexity, e.g. by theoretical evaluation of the number of required operations, are far more appropriate but difficult to implement, especially within the context of an automated benchmark.
- **Visual quality:** The perceptual quality of watermarked images should be measured in a quantitative way that correlates well with the way human observers perceive image quality, if an automated benchmarking system is to be constructed. Obviously quantitative measures that correlate better with the perceptual image quality than the widely used SNR and PSNR metrics should be devised. However no globally agreeable visual quality metric currently exists.

Obviously, the necessity to deal with a multidimensional performance space makes watermarking performance characterization, method comparison and result presentation a complicated task. The requirement for performance evaluation over various attacks and attack parameters (e.g. for various compression factors) adds one more complexity factor to the problem. To make things worse, detection performance cannot be measured by a single index but requires a pair of indices (probability of false alarm and probability of false rejection, see section 2). Among the performance aspects described above, visual quality is the only one that can be directly controlled by modifying the watermark embedding strength. Thus, fixing visual quality to values typical for the application under study and measuring the system performance (and proceed to comparisons) with respect to the remaining parameters can be a way to partially deal with the multidimensionality problem.

This paper will focus on the fundamentals of judging the detection/decoding performance, summarizing in a detailed and systematic way the corresponding principles and methodology. Discussion will be limited to the so-called robust watermarks, leaving aside benchmarking considerations for the so-called fragile or semi-fragile algorithms. Despite the fact that the paper concentrates on still images, the proposed procedures and metrics can be applied for the benchmarking of watermarking techniques for other digital media (audio, video, 3-D models). The detection/decoding performance metrics and methodology that will be presented below can be used for benchmarking both blind and non-blind methods.

2. WATERMARK DETECTION PERFORMANCE

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

- H0 : the image under test hosts the watermark under investigation.
- H1 : the image under test does not host the watermark under investigation.

Hypothesis H1 can be further divided into two sub-hypotheses:

- H1a : the image under test is not watermarked.
- H1b : the image under test hosts a different watermark.

Thus, detection performance can be characterized by the false alarm (or false positive) errors and their corresponding probability P_{fa} i.e., the probability to detect a watermark in an image that is not watermarked or is watermarked by a different key than the one under investigation, and the false rejection (or false negative) errors, described by the false rejection probability P_{fr} i.e., the probability of not detecting a watermark in an image that is indeed watermarked. Depending on the application, these two types of errors might have different significance. However one should never neglect the importance of false alarms when designing a watermarking algorithm. To understand this fact one can imagine a detection function constructed so as to always report ”watermark detected”. Such a detection function would have $P_{fr} = 0$. However its false alarm probability would be 1 and, obviously, the system would be useless. P_{fa} can be evaluated using detection trials with erroneous watermarks (hypothesis H1b) or detection trials on non-watermarked images (hypothesis H1a). The former might sometimes be preferable since it corresponds to the worst case scenario. 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 of producing distinct watermarks, and thus lead to estimates of the ”effective” keyspace. One can distinguish between three types of false alarms and false rejections [2]: 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 following we will deal with ways of measuring P_{fa}, P_{fr} for the multiple keys - single image case. Combination of results from different images in order to come up with metrics for the multiple keys - multiple images case will be studied in section 4.

Watermark detectors can be of two different types. Hard decision detectors generate a binary output (watermark detected, watermark not detected) which usually results by comparing internally the test statistic of the corresponding hypothesis test against a decision threshold. Soft decision detectors provide as output the test statistic itself i.e., a real number that is related to detection reliability. In this case, thresholding in order to reach a binary decision is done in a separate, subsequent step. In a real application, detectors will most probably be of the hard decision type although the soft decision output (detection test statistic) can accompany the binary output in order to give an indication of the decision reliability. However, during the development stage of an algorithm one should look at the detection as a soft decision procedure because, as will be described in the sequel, this perspective allows judging the performance of the algorithm under all possible operating conditions and facilitates final threshold selection.

In order to estimate P_{fr}, P_{fa} one should conduct experiments involving a set of images $\mathbf{I} = \{I_i/i = 1...N_I\}$, a set of keys $\mathbf{W} = \{K_i/i = 1...N_W\}$ and a set of messages $\mathbf{M} = \{M_i/i = 1...N_M\}$ (to be used later on for the message decoding evaluation). Each image I_i is watermarked with a watermark W_j and a message M_k is encoded. The procedure is repeated for all elements of the sets $\mathbf{I}, \mathbf{W}, \mathbf{M}$ and a set $\mathbf{I}^w$ of watermarked images is generated. The cardinality of $\mathbf{I}^w$ equals $N_W \times N_I \times N_M$. Subsequently, the images in $\mathbf{I}^w$ are distorted using the attack under study and the set $\mathbf{I}^a$ of attacked images, comprising of of $N_W \times N_I \times N_M$ elements, is generated. Finally, watermark detection is performed to all images of $\mathbf{I}^a$. Trials with the watermark W_i that has been indeed embedded in the image I_i^a and with an erroneous watermark W_j, $(i \neq j)$ are conducted. Alternatively, one can conduct experiments involving detection of watermark W_i in the original, un-watermarked version of the image under study. Thus, for each image two pairs of detector outputs $\mathbf{D}^c$, $\mathbf{D}^e$, for the correct and the erroneous watermark (or the no-watermark) case respectively are extracted. Message decoding is also being conducted along with watermark detection but this procedure will be described in section 3. In the following subsections we will describe the detection performance metrics that one can derive using the ”raw” results $\mathbf{D}^c$, $\mathbf{D}^e$, for a single image in $\mathbf{I}$.

2.1. Hard decision detector

In this case, one can use the number N_{fa} of the erroneously detected watermarks and the number N_{fr} of the missed watermarks from sets $\mathbf{D}^e$ and $\mathbf{D}^c$ to evaluate a P_{fa}, P_{fr} pair:

$$P_{fa} = \frac{N_{fa}}{|\mathbf{D}^e|}, \quad P_{fr} = \frac{N_{fr}}{|\mathbf{D}^c|}$$

where $|\mathbf{D}|$ denotes the number of elements in $\mathbf{D}$. Since a single performance index can facilitate method comparison, one can evaluate the weighted sum $P_{er} = p_1 P_{fa} + p_2 P_{fr}$. The constants p_1, p_2 should be selected so as to to reflect the relative importance of P_{fa}, P_{fr} in a certain application scenario.

2.2. Soft decision detectors

In case of soft decision detectors one can use sets $\mathbf{D}^c$ and $\mathbf{D}^e$ (now containing real-valued numbers instead of binary values) to derive the empirical probability distribution functions (histograms) of the detection test statistic for both hypotheses H0 and H1b (or H1a). By utilizing these empirical distributions the probabilities of false alarm and false rejection as a function of the detection threshold T can be extracted. Let T_1 and T_2 be the minimum and the maximum value within $\mathbf{D}^c$, $\mathbf{D}^e$:

$$T_1 = \min\{\mathbf{D}^c, \mathbf{D}^e\}, \quad T_2 = \max\{\mathbf{D}^c, \mathbf{D}^e\}$$

Then for a (sufficiently large) set of discrete threshold values T_k between T_1 and T_2, $P_{fa}(T_k)$ and $P_{fr}(T_k)$ can be calculated:

$$P_{fa}(T_k) = \frac{|\mathbf{D^e_{T_k}}|}{|\mathbf{D^e}|}, \text{where } \mathbf{D^e_{T_k}} = \{x_i > T_k \mid x_i \in \mathbf{D^e}\}$$

$$P_{fr}(T_k) = \frac{|\mathbf{D^c_{T_k}}|}{|\mathbf{D^c}|}, \text{where } \mathbf{D^c_{T_k}} = \{x_i < T_k \mid x_i \in \mathbf{D^c}\}$$

where $|\mathbf{D}|$ denotes the number of the elements (cardinality) of the set $\mathbf{D}$. Using $P_{fa}(T_k), P_{fr}(T_k)$ we can evaluate the *Receiver Operating Characteristic* (ROC), i.e., the plot of the probability of false alarm P_{fa} versus probability of false rejection P_{fr}. The

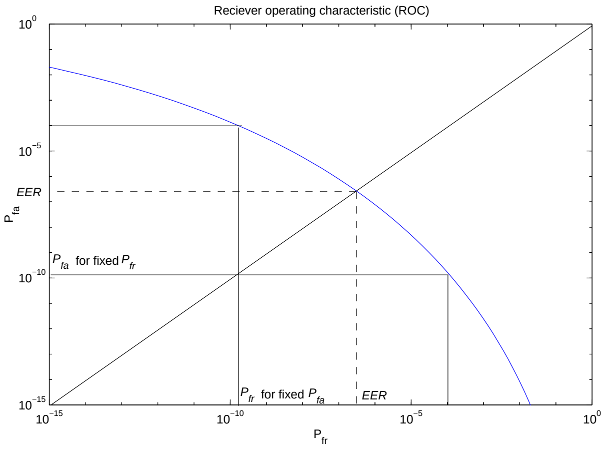

Figure 1: Detection performance measures.

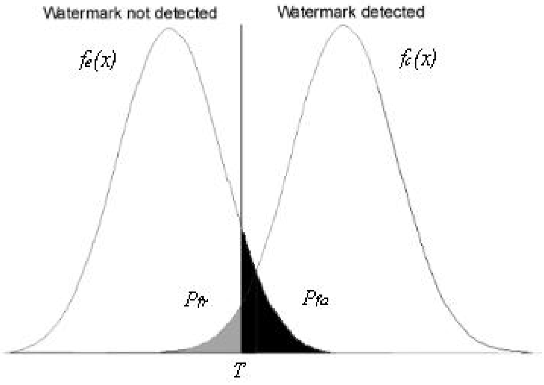

Figure 2: False alarm and false rejection probabilities.

ROC curve (figure 1) is the most complete way to describe an algorithm detection performance since it allows forming an idea about the algorithm performance in various operating conditions. Using the ROC curve, one can select the threshold value that gives the desired P_{fa}, P_{fr} pair. Having evaluated the ROC, one can also evaluate the following performance measures (figure 1):

- P_{fa} for a fixed, user-defined P_{fr}.
- P_{fr} for a fixed, user-defined P_{fa}.
- Equal error rate (EER), i.e, the point on the ROC where P_{fa}=P_{fr}.

These detection performance indices are very useful since they provide single-value metrics for characterizing the algorithm and thus allow easy comparison between algorithms. Furthermore, they allow checking the appropriateness of an algorithm for a certain application scenario, through the comparison of the metric against a performance threshold. If, for example, a certain application requires a specific P_{fa} value, one can fix this value and compare two algorithms with respect to the corresponding P_{fr} values.

Despite its simplicity, the ROC curve evaluation approach presented above has a major drawback; in order to obtain accurate estimates of $P_{fa}(T)$ and $P_{fr}(T)$ one has to conduct experiments involving an extremely large number of different keys. This is particularly true for the threshold values that correspond to the tails of the empirical distributions, where, for a well-behaved algorithm, the error probabilities might be extremely low and thus very difficult to measure. These are actually the operating points of most interest for a watermarking algorithm. A solution to this problem is to fit appropriate distribution models $f_c(x)$ and $f_e(x)$ on the experimental data $\mathbf{D}^c$ and $\mathbf{D}^e$ and proceed to ROC evaluation using these models. In this case $P_{fa}(T)$, $P_{fr}(T)$ can be calculated as follows:

$$P_{fa}(T) = \int_{T}^{\infty} f_e(x)dx \ , \qquad P_{fr}(T) = \int_{-\infty}^{T} f_c(x)dx$$

In other words, P_{fr} is given by the area of $f_c(x)$ left of threshold whereas P_{fa} is the area of $f_e(x)$ right of threshold, as it is illustrated in Figure 2.

The success of this approach depends on how accurately the theoretical pdf models the experimental data. For correlation-based detection schemes and due to the central limit theorem, the empirical data can be sufficiently well approximated by a Gaussian pdf. Other embedding / detection approaches might also allow for theoretical modeling of the detector output distribution. In the context of an automated benchmarking system, where the embedding/detection procedures are not known (black box case), the following approach can be used: apply goodness-of-fit tests (e.g. the Chi-Square test or the Kolmogorov-Smirnov test) on the data within sets $\mathbf{D}^c$ and $\mathbf{D}^e$ to check whether they come from a certain distribution among a pre-selected set of distribution models (using the same significance level for all tests). According to the tests outputs, select the model that best fits the data, or, if more than one models fit the data, the one with the highest value of the test statistic. The problem of evaluating the false alarm probability has been also treated in [8].

3. MESSAGE DECODING PERFORMANCE

If the watermarking method supports message encoding, its decoding performance can be judged by evaluating the Bit Error Rate (BER) i.e., the mean number of erroneously decoded bits. In certain applications, the message might consist of various parts, each conveying information of different type and importance. In such a case, BER should be evaluated separately for each part of the message. Since message decoding is assumed to be performed only in 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.

In order to evaluate decoding performance, a message M_i is embedded in every image in addition to the watermark, as already described in section 2. Then, watermark detection is performed to all images of $\mathbf{I}^a$. As a result of the detection procedure, two sets of decoder outputs $\mathbf{B}^c$, $\mathbf{B}^e$ are extracted from the detection trials with the correct and the erroneous watermark respectively. In case of hard decision detectors, a single BER value is evaluated by comparing the message M_i that has been embedded in the image with the decoded message $\widehat{M_i}$ for all messages in $\mathbf{B}^c$, $\mathbf{B}^e$ (and not only messages in $\mathbf{B}^c$ as one can initially assume) that are associated with successfully detected watermarks (either correct or erroneous).

In case of soft decision detectors, BER should be evaluated as a function of the detection threshold T (or equivalently as a function of P_{fa} or P_{fr}). This can be done by evaluating for each threshold T the mean number of erroneously decoded bits (BER), for all messages associated with watermarks (either correct or erroneous) that have resulted in a detector output greater than T, i.e., for all successfully detected watermarks. The BER for fixed P_{fa} or fixed P_{fr} can be used as performance measure in this case.

Another aspect of a watermarking algorithm decoding performance is its payload, which can be defined as the maximum number of bits that can be encoded in a fixed amount of 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. Payload evaluation can be performed by embedding messages of increasing length in a fixed amount of data or messages of fixed length to a decreasing amount of data until BER reaches the specified limit. As stated above, in case of soft decision detectors, BER is a function of the detection threshold T. As a consequence, the payload of the method should also be evaluated as a function of the threshold T (or equivalently as a function of P_{fa} or P_{fr}). A way for comparing two soft decision methods with respect to payload is to compare their payloads for fixed P_{fa}.

4. RESULT SUMMARIZATION

The methodology presented above leads to decoding/detection metrics for the single image - multiple keys (or messages) case, i.e., to metrics that refer to a single image from the set **I**. Using results obtained for all images in this set, one can proceed in deriving metrics for the multiple images -multiple keys (or messages) case. Such a derivation is meaningful only if all images in **I** are watermarked using embedding strength values that lead to watermarked images having the same perceptual quality.

For hard decision decoders, P_{fa}, P_{fr} values for multiple images can be obtained using a weighted averaging function:

$$P_{fr} = \sum_i w_i P_{fr_i}, \quad P_{fa} = \sum_i w_i P_{fa_i}, \ i = 1...N_I$$

For the above formula to be valid, the same number of keys should be used for obtaining P_{fa_i}, P_{fr_i} for all images. Weighted averaging is superior to simple averaging since weights that reflect the probability of occurrence of an image in a certain application scenario can be used.

For soft decision decoders, one can generate a multiple images-multiple watermarks ROC curve by first averaging $P_{fa_i}(T_k), P_{fr_i}(T_k)$ over all images for each threshold value T_k:

$$P_{fr}(T_k) = \sum_{i=1}^{N_I} w_i P_{fr_i}(T_k), \quad P_{fa}(T_k) = \sum_{i=1}^{N_I} w_i P_{fa_i}(T_k)$$

The above formula is valid only if the same set of discrete threshold values has been used for each image. Furthermore, the number of keys used for obtaining $P_{fa_i}(T_k), P_{fr_i}(T_k)$ should be equal for all images. Using $P_{fa}(T_k), P_{fr}(T_k)$ one can proceed in evaluating the ”cumulative” ROC curve.

A similar approach can be used in order to obtain a single ROC curve or a single P_{fa}, P_{fr} pair for a set of attacks, and thus judge the overall performance of the algorithm with respect to these attacks. In this case, weights that reflect the probability of occurrence of a certain attack on the application scenario under study should be used. Summarization of decoding performance metrics (BER) can be done in an analogous way. The above procedure can be seen as a progressive information compaction scheme that leads from [multiple watermarks-single image-single attack] results to [multiple watermarks-multiple images-single attack] results and further [to multiple watermarks-multiple images-multiple attack] results.

5. CONCLUSIONS

The basic principles of watermark detection/decoding performance evaluation along with a methodology for deriving the corresponding performance metrics have been presented in this paper. Despite the progress that has been achieved in the area of watermarking benchmarking, there are still a number of open theoretical and practical issues that have to be solved. Such an issue is how one can measure very small probability values, such as those related with false negatives, false positives and BER, without the need to conduct prohibitively large numbers of trials. Research towards these issues will hopefully lead to efficient benchmarking tools in the near future.

6. ACKNOWLEDGEMENTS

This work has been partially funded by the EU project IST-1999-10987 CERTIMARK.

7. REFERENCES

[1] “Identification & protection of multimedia information,” *Special issue on Proceedings of the IEEE*, vol. 87, no. 7, July 1999.

[2] I. Cox, M. Miller, and J. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, 2001.

[3] S. Katzenbeisser and F. A. P. Petitcolas (editors), *Information hiding techniques for steganography and digital watermarking*, Artech House, 2000.

[4] F. A. P. Petitcolas, “Watermarking schemes evaluation,” *IEEE Signal Processing Magazine*, vol. 17, no. 5, pp. 58–64, September 2000.

[5] 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, 7-10 October 2001.

[6] 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, USA, April 2001.

[7] EU project IST-1999-10987 CERTIMARK, *http://www.certimark.org*.

[8] M. L. Miller and J. A. Bloom, “Computing the probability of false watermark detection,” in *Proceedings of the Third International Workshop on Information Hiding*, 1999, pp. 146–158.