Performance Analysis of Fridrich–Goljan Self-Embedding Authentication Method

Gabriel Domínguez-Conde, Pedro Comesaña, and Fernando Pérez-González

Abstract—This paper analyzes the performance of the image authentication method based on robust hashing proposed by J. Fridrich and M. Goljan. In this method, both the embedder and the detector generate the watermark from a perceptual digest of the image. Therefore, an accurate performance analysis requires assessing the relation between noise and hash bit errors. Our approach first derives the probability of hash bit error due to watermark embedding and/or the attack, and then uses such probability to derive the probabilities of false positive and false negative.

Index Terms—Content-based authentication, performance analysis, robust hash.

I. INTRODUCTION

In the last years, a number of powerful and user-friendly multimedia editing tools for digital video, image, and audio have proliferated, allowing an unskilled person to easily modify contents and produce forgeries. To regain confidence in digital objects, several authentication tools have been proposed, some of them relying on digital watermarking techniques. In watermarking-based authentication techniques, a low-power signal (watermark) is embedded in the digital content to be protected (host signal); then, the authenticity of the received signal is determined by verifying the presence of the correct watermark. Concerning the generation of such a watermark, sometimes it is just pseudorandomly produced using a secret key shared by the embedder and the detector, whereas in other cases, a perceptual digest of the digital content is also used. This low-dimensional summary of a digital content is usually referred to as a robust hash, perceptual hash, or soft hash.

In contrast to a cryptographic hash, a robust hash would be ideally sensitive only to perceptual changes, meaning that two perceptually identical digital objects should yield the same result. However, this requirement is difficult to be fully met in practice, as the design of a tractable mathematical model of human perception that accurately quantifies perceptual similarity is still an unsolved task. On the other hand, given that only authorized users should be able to generate a valid robust hash, the chosen hash functions usually depend on a secret key.

So far, several authentication techniques based on watermarking have been presented, most of them aimed at images. Schneider et al. [1] proposed, based on the ideas of Friedman [2], one of the earliest methods which uses a robust hash to aid the verification of the authenticity of digital contents. Bhattacharjee and Kutter [3] proposed a robust hashing authentication scheme by generating a set of feature-points with a set of Mexican-Hat wavelets. In [4], Kundur and Hatzinakos described a fragile method for tamper localization by using a quantization technique to embed the watermark in a transform domain. Venkatesan et al. [5] developed a robust hashing method which divides an image into random nonoverlapping blocks. The statistics of each block are used to generate the robust hash. Lin and Chang [6] introduced a robust authentication watermarking scheme, designed for JPEG compression, based on the relation between the discrete cosine transform (DCT) coefficients. In [7], Cannons and Moulin proposed a nonblind robust authentication method based on robust hashing and watermarking, and used a statistical description of the host image to analyze its performance. More recently, Swaminathan et al. have proposed a robust hashing method for image authentication, based on invariant characteristics of the images, e.g., the resilience to rotation, scaling, and translation of Fourier–Mellin transform [8]. Finally, in [9] Monga and Milčák have presented a robust hashing technique where a content-based binary vector is extracted by applying a nonnegative matrix factorization to the image.

From the set of authentication techniques which embed a watermark that depends on an image robust hash (also known as watermarking self-embedding authentication), the algorithm proposed by Fridrich and Goljan [10]–[12] is one of the most prominent, and has been extensively adopted as a reference in many other works and comparisons (e.g., [7], [8], [13]). In addition, this method uses a watermark synthesis function which successfully fills the gap between robust hashing and watermarking-based authentication; in fact, the watermark generation has the particular feature that it can be tuned to account for the error sensitivity of the overall authentication technique.

However, a performance analysis of this widely referenced self-embedding authentication algorithm is still lacking. This means that for comparison purposes, Monte Carlo techniques need be used, with the obvious drawback of requiring long simulation runs whenever small probabilities are to be estimated. In addition, this lack makes it very difficult to determine the influence of the various parameters of the Fridrich–Goljan method on performance. This paper aims at filling this gap by presenting a novel performance analysis and drawing some conclusions that can be extended to general hash-based authentication schemes.

In our analysis, we first compute the hash bit-error probability due to the watermark embedding and noise, and then we show its impact on the receiver operating characteristic (ROC) of the overall scheme. Notice that this methodology could be used to analyze other authentication schemes based on robust hashing, as the same principles would apply. To the best of our knowledge, this is the first time that the performance of a watermarking authentication scheme based on robust hashing is analyzed in this way.

In Section II, we give a brief introduction to the robust authentication method proposed by Fridrich and Goljan, and the embedding and detection processes, whereas performance is addressed in Section III. In Section IV, simulations are carried out to validate our approach; in addition, the modifications on the introduced performance analysis necessary for dealing with image compression, intensity change, and linear filtering are outlined. Finally, Section V presents the main conclusions and discusses some future lines.

A. Notation

We will denote scalar random variables with uppercase letters (e.g., $X$) and their outcomes with lowercase letters (e.g., $x$). The same notation criterion applies to random vectors and their outcomes, denoted in this case by bold letters (e.g., $\mathbf{X}$, $\mathbf{x}$), with transposes denoted by the
superindex $T$. The $i$th component of a vector $X$ is denoted as $X_i$. Images in the pixel domain will be partitioned in $N_b$ blocks and arranged as vectors.

II. DESCRIPTION OF FRIDRICH AND GOLJAN METHOD

In this section, a description of the method by Fridrich and Goljan [11], that will be analyzed in Section III, is given; furthermore, a correlation-based detector for that method is proposed.

A. Hash Computation

The original host signal in the pixel domain is block-wise partitioned and arranged as $N_b$ vectors $X_i, 1 \leq i \leq N_b$, each of size $M$.1 For the sake of notational simplicity, we will avoid the block superindex; however, it must be clear that the authentication method operates at the block level. From each $X_i$, the variance of which will be denoted by $\sigma^2_{X_i}$, and depending on a set of $N_b$ length-$M$ pseudorandom sequences $s^i$ (generated irrespectively of $X_i$), $1 \leq j \leq N_b$, an $N_b$ bits hash vector $h$ is computed as

$$h_j = \begin{cases} 0, & \text{if } \frac{1}{M}|X^T_i \cdot s^i| < T_s, \\ 1, & \text{otherwise} \end{cases}$$

where $T_s$ is a quantization threshold (constant along the complete image) computed to comply with the constraint that the total number of 0’s over all the $N_b$ hash vectors $h$ of the image must be equal to the total number of 1’s. In our analysis, this threshold will be approximated by the median of the absolute value of the coefficients obtained by projecting the host image blocks onto the pseudorandom sequences, i.e., $(1/M)|X^T \cdot S|$. Noticing that the method operates at the block level, for our purposes, we will rely on a block-wise characterization of the host image. Therefore, for the derivations contained in this paper, the probability density function (pdf) of the absolute value of the projected host is obtained by averaging over the projected host blocks. Using this approximation, the number of zeros and ones cannot be guaranteed to be equal for every hash originated from a given image; however, their relative frequency will be asymptotically identical as $N_b \cdot N_h$ is increased.

Concerning the generation of the projection sequences $s^i$, each of them is produced from a $\sqrt{M} \cdot \sqrt{M}$ pseudorandom matrix (obtained depending on the system secret key), whose components take values uniformly in $[0,1]$. Each of these matrices is then low-pass filtered, mean adjusted, and rearranged as the length-$M$ vector $s^i$.

B. Watermark Computation

Each hash vector $h$ is permuted using $N_p$ permutations $\pi^k: \{0,1\}^{N_h} \rightarrow \{0,1\}^{N_h}$, with $k = 1, \ldots, N_p$. Next, the results are joined to define the length-$N_b$ vectors $t^i \triangleq (\pi^1(h), \pi^2(h), \ldots, \pi^{N_p}(h)), 1 \leq i \leq \ldots, N_b$. These $t^i$, jointly with the secret key of the system, and the index of the current image block, are used as seed of a pseudorandom number generator (PRNG) that generates a length-$M$ sequence with components uniformly distributed on $[-1, +1]$, and that we will denote by $v^i$.

Finally, the watermark $w$ corresponding to a block of the original host signal $X_i$, is constructed as $w = \sqrt{3/M} \sum_{j=1}^{N_h} v^i$. Assuming a good behavior of the PRNG, it is reasonable to think of the $v^i$ as being (almost) independent of $X_i$ and $s^i$ (both through $t^i$), so $w$ can be considered to be independent of $X_i$ and $s^i$ as well. Furthermore, given that different seeds are used for generating the vectors $v^i$, it follows that $w$ will be (almost) independent from each other; hence, for values of $N_b$ large enough, one can use the central limit theorem (CLT), and approximate the distribution of $W$ by $N(0, \text{I}_{3M \times 3M})$, with $\text{I}_{3M \times 3M}$ the identity matrix of size $M$.

This watermark $w$ is embedded in the host signal using additive spread spectrum [14] in the $\sqrt{M} \times \sqrt{M}$-block pixel domain, so a block of the watermarked image is obtained as $y = x + \gamma w$, where $\gamma$ is an embedding strength parameter.

C. Detection

On the detector side, the steps described above are followed to obtain an estimate $\hat{w}$ of the watermark $w$ from a block of the received signal $z$. Be aware that even in the absence of attacks, the considered signal $z$ will be different from the host image (due to the presence of the watermark), so the quantization threshold at the detector will be computed from $(1/M)|z^T \cdot S^t|$, and thus it will not necessarily coincide with that obtained at the embedder. In order to make this difference explicit, we will denote the threshold calculated at the detector as $T_d$. On the other hand, the same considerations made regarding the computation of $T_s$ in Section II-A can be raised here.

The main objective of the detector is to decide on the presence of the watermark estimate in the corresponding block of the received signal; if the estimate of the watermark is decided to be present, that block will be declared authentic (i.e., nonmodified); otherwise, it will be said to be manipulated.

Since in [11] detection is not addressed, we introduce next the detection process assumed in our analysis. First, the decision on the presence or absence of the estimate of the watermark can be formulated as a binary hypothesis test, namely,

$$H_0: \quad z = \eta(x + \gamma \hat{w}) + n$$

$$H_1: \quad z = x + \gamma \hat{w}$$

(1)

where $H_0$ represents the hypothesis of the received signal being the sum of a watermarked signal scaled by a given factor $\eta \in [0,1]$ and some complementary signal $n$ independent of $x$, $s$, and $w$, with zero mean and variance $\sigma^2_{X}$, whereas $H_1$ denotes the hypothesis of the received signal being the output of the embedder. This hypothesis test comprises several interesting detection scenarios. For example, by setting $\eta = 0$, the proposed hypothesis test can model the case of deciding whether a block was watermarked with a valid key or it was not watermarked (or watermarked with an invalid key). On the other hand, with $\eta \neq 1$, $\eta_0$ corresponds to the case where the image under test is the result of applying some unacceptable noise/processing (modeled by the addition of $n$) to an otherwise valid watermarked image. Moreover, other values of $\eta$ in $(0,1)$, may model other scenarios, e.g., scaling attacks, image fusion, etc.

Additionally, it is worth pointing out that the watermark detection problem (a.k.a., one-bit and zero-bit watermarking), although out of the scope of this paper, can be also studied in the framework defined by the hypothesis test in (1), just by setting $\eta = 0$ and interpreting $n$ as a nonwatermarked content.

When detection mistake costs are not set, or $a$ priori probabilities for the two hypotheses are not available (as it is the case in most practical scenarios), the Neyman–Pearson criterion is customarily used, as it minimizes the probability of false negative for a given probability of false positive [15]. This criterion implies the use of the likelihood ratio test, taking the form

$$L(z) = \frac{f(z|H_1)}{f(z|H_0)} \geq \lambda$$

where $\lambda$ is the detection threshold. When both the host signal and the noise are independent and Gaussian distributed, the correlation between the received block and the corresponding watermark, i.e., $\rho \triangleq (1/M)|z^T \cdot \hat{w}|$, is a sufficient statistic for this problem. In most practical
detection methods, and due to its simplicity, this statistic is still used, although the mentioned condition on the Gaussianity and independence of the signals is not verified. Given a false positive probability \( P_{fa} \) (the probability of deciding that the received signal was not modified, when indeed it was), \( \lambda \) is selected so that the following equation holds:

\[
P_{fa} = \int_{R(\lambda)} f_{\mathcal{M}|\mathcal{H}_0}(\tau) d\tau
\]

and the false negative (the probability of deciding that the received signal was modified, when it was not) is then calculated as

\[
P_{fn} = \int_{R(\lambda)} f_{\mathcal{M}|\mathcal{H}_1}(\tau) d\tau,
\]

where in the above expressions (2) and (3), \( R(\lambda) \) denotes the complement of \( R(\lambda) \), and \( f_{\mathcal{M}|\mathcal{H}_0} \) and \( f_{\mathcal{M}|\mathcal{H}_1} \) denote the pdfs of \( \mathcal{M} \) when, respectively, \( \mathcal{H}_0 \) and \( \mathcal{H}_1 \) are true (namely, \( \rho_{\mathcal{M}|\mathcal{H}_0} \) and \( \rho_{\mathcal{M}|\mathcal{H}_1} \)).

For the computation of \( f_{\mathcal{M}|\mathcal{H}_0} \) and \( f_{\mathcal{M}|\mathcal{H}_1} \) we will take into account that the ratio \( \rho \) is the result of adding \( N_{\mathcal{M}} \) random variables with finite variance, so for large values of both \( N_{\mathcal{M}} \) and \( M \), and whenever \( \sigma_{\mathcal{M}}^2 \gg \sigma_{\mathcal{H}}^2 \) and \( \sigma_{\mathcal{H}}^2 \gg \gamma^2 \), we can model both \( f_{\mathcal{M}|\mathcal{H}_0} \) and \( f_{\mathcal{M}|\mathcal{H}_1} \) as Gaussian distributions (see Appendix B for further details); therefore, the detection thresholds on \( \rho \) will be given by the solutions to the following equation (see [15, Ch. 2.6]):

\[
\frac{(\rho - E\{\rho_{\mathcal{M}|\mathcal{H}_0}\})^2}{2\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_0}\}} - \frac{(\rho - E\{\rho_{\mathcal{M}|\mathcal{H}_1}\})^2}{2\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_1}\}} + \log \left( \frac{\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_0}\}}{\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_1}\}} \right) = \log \lambda
\]

where \( E\{\cdot\} \) denotes expectation, and \( \text{Var}\{\cdot\} \) variance. Given that the last equation has in general two solutions in \( \rho \), the detection region is defined by two thresholds, \( T_0 \) and \( T_1 \) (with \( T_0 < T_1 \)), depending on the chosen hypothesis on the ratio \( \frac{\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_0}\}}{\text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_1}\}} \); if that ratio is larger than 1, \( \mathcal{H}_1 \) holds for \( \rho \in [T_0, T_1) \), and \( \mathcal{H}_0 \) elsewhere (conversely, if the ratio is smaller than 1, the hypothesis choice is the reverse). In the special case where the variances of both Gaussians are the same, (4) becomes a linear equation, and a single solution (one detection threshold) exists. Although in our performance analysis both \( T_0 \) and \( T_1 \) are used, the analytical expressions will be still valid when \( \text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_0}\} = \text{Var}\{\rho_{\mathcal{M}|\mathcal{H}_1}\} \) by setting \( T_1 \approx \infty \).

In order to quantify the performance of the method proposed by Fridrich and Goljan, we will derive its ROC. This is equivalent to evaluating both the probability of false positive \( P_{fa} \) and the corresponding probability of false negative \( P_{fn} \) for a range of values of \( \lambda \); this derivation is the objective of Section III.

**III. PERFORMANCE ANALYSIS CONSIDERING THE WATERMARK EMBEDDING**

In this section we will analyze the effect of the watermark embedding and the attack on the estimate of the hash on the detector side, and how a nonperfect estimate of the watermark will deteriorate the overall performance. Our first step will be the characterization of the random variable \( \mathcal{D}_T \approx (1/M)\mathbf{X}^T \cdot \mathbf{S}^T \). Reasoning that projecting onto \( \mathbf{s}^T \) resembles computing an almost orthogonal transform somewhat similar to the DCT, whose coefficients have been previously characterized in the literature by a generalized Gaussian distribution (GGD) [16], we propose to model \( \mathcal{D}_T \) by a GGD, i.e.,

\[
f_{\mathcal{D}_T}(x) \approx \mathcal{A}_\mathcal{N} e^{-x^2/2\gamma^2}\mathcal{N}(x),
\]

where in the above expression \( \mathcal{A}_\mathcal{N} \), \( \beta_N \) and the shaping parameter \( \gamma_N \) are fitted for each block of the image to the experimental data using maximum likelihood estimation. This crucial hypothesis has been validated using the luminance component of a set of 100 images. This set was built by randomly selecting 100 images from the Uncompressed Colour Image Database of Austin University [17], resizing the chosen images to 256 x 256 pixels. For each image, the Kullback–Leibler divergence (KLD) between the histogram obtained by projecting its blocks of size 64 x 64 over 10,000 random vectors and the corresponding GGD with parameters optimized for those projections has been computed. The mean value of the KLD over the 100 images is as small as 2.5 x 10^{-4}. For the sake of comparison, we have also obtained the KLD between the histogram of the projected coefficients for each image and a Gaussian distribution with zero mean and variance empirically estimated from those coefficients; in this case, the mean value of KLD is 3.1 x 10^{-2}, i.e., an order of magnitude difference between the two KLD’s which supports the use of a GGD.

In order to derive the probability of the errors produced by watermark embedding, the probability of flipping one bit of the hash obtained at the detector with respect to the robust hash computed at the embedder, has to be calculated. As stated above, for large \( N_M \), the components of the watermark can be well-approximated by a \( \mathcal{N}(0,1) \) distribution; furthermore, \( \mathbf{W} \) and \( \mathbf{S} \) can be considered as independent, as stated above. Therefore, for the characterization of the pdf of \( \mathcal{D}_T = \mathbf{W}^T \cdot \mathbf{S}^T \), we will take into account that for an arbitrary value of the vector \( \mathbf{s}^T \), \( (1/M)\mathbf{W}^T \cdot \mathbf{s}^T \) follows a \( \mathcal{N}(0, \sigma_s^2/\gamma^2\mathcal{N}) \), so in order to derive the pdf of \( (1/M)\mathbf{W}^T \cdot \mathbf{S}^T \) one should average the obtained Gaussian distribution over the possible values of \( \sigma_s^2 \) and \( \gamma^2 \). Thus, given that \( \mathbf{S}^T \) is obtained by processing an independent identically distributed (i.i.d.) random vector with a low-pass filter and subtracting the mean, due to the law of large numbers (see [18, Ch. VIII]) for large values of \( M \), the distribution of \( \sigma_s^2/\gamma^2\mathcal{N} \) will converge to \( \sigma_\mathcal{S}^2 \cdot \|\mathbf{g}\|^2 \), where \( \mathbf{g} \) denotes the coefficients of the mentioned filter and \( \sigma_\mathcal{S}^2 \) is the variance of the original i.i.d. signal. Let \( \sigma_\mathcal{S}^2 \approx \sigma_\mathcal{S}^2 \cdot \|\mathbf{g}\|^2 \), then we can approximately project the watermark (with \( \gamma = \gamma_N \)) onto \( \mathbf{W} \) by \( N(0,\sigma_\mathcal{S}^2/\gamma^2\mathcal{N}) \). This characterization is valid for both cases where \( \mathbf{N} \) is an image (since, as we have discussed at the beginning of this section, its projection will be well-modeled by a GGD) or Gaussian additive noise (for which \( \epsilon_N \approx \mathcal{N} \)).

Thus, as shown in Appendix A, the probability of a hash bit error under hypothesis \( \mathcal{H}_0 \) can be expressed as (5), shown at the top of the next page, where \( \mathcal{Q}(x) \approx (1/\sqrt{2\pi}) \int e^{-x^2/2\gamma^2}\mathcal{N}(x) \). It is worth pointing out that (5) is valid whenever \( \eta > 0 \); in the particular case where \( \eta = 0 \), due to the assumption of independence between \( \mathbf{X} \) and \( \mathbf{N} \), it is clear that \( P_e \approx 1/2 \). Additionally, (5) can be easily adapted to hypothesis \( \mathcal{H}_1 \) by setting \( \eta = 1 \) and \( \|\mathbf{S}^T\| = 0 \) (i.e., \( \beta_N = \infty \)).

Expression (5) admits further simplifications under certain circumstances. First, recalling that the threshold \( \mathcal{T} \), is set so that it is exceeded half of the time, the denominators of the two summands in (5) will be approximately 1 when the statistics of the different projected blocks are similar. Moreover, when \( \mathcal{N} \) corresponds to Gaussian noise, the outer integral in (5) can be explicitly evaluated. To this end, we will focus on the first summand in (5), since a similar derivation applies to the second

---

2Border effects were not considered in this derivation due to their marginal consequences.
\[ P_c \approx \frac{2}{\eta} \int_{-\infty}^{\infty} A_N e^{-\frac{3\pi}{\eta^2}} \int_{0}^{\infty} \left( 1 - Q \left( \frac{\sqrt{M(T_d - t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M(T_d + t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) d\tau \]

\[ P_c \approx \frac{2}{\eta} \int_{-\infty}^{\infty} A_N e^{-\frac{3\pi}{\eta^2}} \int_{0}^{\infty} \left( 1 - Q \left( \frac{\sqrt{M(T_d - t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M(T_d + t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) d\tau \]

\[ P_{tr} \approx \sum_{n=0}^{N_h} \Pr(N_x = n) \left( Q \left( \frac{\sqrt{M}(T_0 - (N_h - n) \frac{2}{N_h})}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M}(T_1 - (N_h - n) \frac{2}{N_h})}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) \]

\[ P_{in} \approx \sum_{n=0}^{N_h} \Pr(N_x = n) \left( Q \left( \frac{\sqrt{M}(T_0 - (N_h - n) \frac{2}{N_h})}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M}(T_1 - (N_h - n) \frac{2}{N_h})}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) \]

The numerator of the chosen term as

Recalling that \( N^T S^t \sim \mathcal{N}(0, (\sigma^2 \sigma^2_N / M)) \), we can write the numerator of the chosen term as

\[ \int_{-\infty}^{\infty} A_N e^{-\frac{3\pi}{\eta^2}} \int_{0}^{\infty} \left( 1 - Q \left( \frac{\sqrt{M(T_d - t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M(T_d + t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) d\tau \]

\[ \int_{-\infty}^{\infty} A_N e^{-\frac{3\pi}{\eta^2}} \int_{0}^{\infty} \left( 1 - Q \left( \frac{\sqrt{M(T_d - t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M(T_d + t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) d\tau \]

Given that the inner integral in (9) is the convolution of \( \mathcal{N}(0, (\sigma^2 \sigma^2_N / M)) \) and \( \mathcal{N}(0, (\gamma^2 \sigma^2 / M)) \), we can rewrite (9) as

\[ \int_{-\infty}^{\infty} A_N e^{-\frac{3\pi}{\eta^2}} \int_{0}^{\infty} \left( 1 - Q \left( \frac{\sqrt{M(T_d - t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) - Q \left( \frac{\sqrt{M(T_d + t)}}{\sigma^2 + \gamma^2 \sigma^2_N} \right) \right) d\tau \]

Summarizing, when \( N \) is Gaussian-distributed, and the projected host blocks have similar statistics, \( P_c \) becomes (6), shown at the top of the page.

Once we have obtained the probability of error of each hash bit, we want to relate this quantity to the errors made in the estimate of the watermark. As it was described in Section II-B, the estimated watermark vector is generated from \( N \) permutations of the reconstructed hash vector \( \hat{h}_1 \); one bit of each of these permutations is picked to form the vector \( t^i \), \( 1 \leq i \leq N_h \). Thus, \( N \) errors in the estimate of the hash vector, with \( N \leq N_h \), will be spread to at most \( N \cdot N \cdot N \) different vectors \( t^i \). This implies that the correlation between \( w \) and \( \hat{w} \) for a given block will depend, through the generation of \( \mathcal{N} \), on the number of wrong vectors \( t^i \), that we will denote by \( N_h \). Hence, it is necessary to know the probability that the number of wrong vectors \( t^i \) is \( N_h \), when there are \( n \) bit errors in the estimate of the hash. In Appendix B, it is shown that the values of the probability of false positive and false negative are given by (7) and (8), respectively, shown at the top of the page. In both expressions, \( \Pr(N_x = n) \) and \( \Pr(N_x = n) \) (which are derived in Appendix B) denote the probability that the number of wrong vectors \( t^i \) is \( n \); under hypotheses \( H_0 \) and \( H_1 \), respectively, and when \( \var{H_0} \neq \var{H_1} \), \( \xi \approx \text{sign} (\var{H_1} - \var{H_0}) \); for the case where \( \var{H_0} = \var{H_1} \), the obtained expressions are still valid by making \( T_1 = \infty \) and \( \xi = 1 \). In addition, \( \sigma^2_h \) represents the standard deviation of the projection of the original watermark \( w \) onto \( \hat{w} \) computed at the detector when the number of wrong vectors \( t^i \) is \( N_h \).

IV. EXPERIMENTAL RESULTS

Next we present the results of several experiments conducted on the set of images described above. First, we experimentally verify the validity of our model. In order to do so, in our first experiment, we study the scenario where the detector must decide whether a given image bears the right watermark. In this setup, the null hypothesis \( H_0 \) is particularized to \( \eta = 0 \) and \( n \) is the block of a nonwatermarked image. The aforementioned set of 100 images was used, with block size of 64 x 64 pixels, \( N_h = 16 \) and \( N_w = 5 \). The results are plotted in Fig. 1, where the empirical and analytical ROC curves almost perfectly match. Furthermore, the curves for different values of \( \gamma \) (\( \gamma \in \{2, 4, 8, 10\} \)) show that a better performance, in terms of the ROC, is achieved with larger values of \( \gamma \), although one should also consider that a larger \( \gamma \) also implies a larger distortion. Hence, in this case, a trade-off between distortion and performance should be achieved. However, the reasoning “larger distortion implies better performance” is not always verified.
For example, if one considers the case $\eta = 1$, and a fixed distribution of $\mathbf{N}$, an increase of $\gamma$ reduces the detection probability for a given probability of false positive, as the effect of $\mathbf{N}$, which in this scenario is the part of the received signal that helps us to decide what hypothesis holds, is masked by the watermark.

In the second considered scenario, we analyze the performance of the Fridrich–Goljan authentication method when trying to detect luminance scaling attacks. Fig. 2 compares the analytical and empirical ROC curves for $\eta \in \{0.3, 0.5, 0.7\}$ and $\gamma = 10$ with the same parameters of the previous experiment (i.e., blocks of $64 \times 64$ pixels, $N_b = 16$ and $N_q = 5$) for Lena without additive noise, i.e., $n = 0$. The close resemblance between the analytical and empirical results shows again the goodness of the proposed model and the subsequent analysis.

It is worth discussing the role of the number $N_p$ of permutations of $\mathbf{h}$ in the generation of $\mathbf{w}$. This parameter is used to control the performance degradation due to image modification, as it was outlined at the end of Section III. On one hand, it is important to avoid that a convex combination of a block from an authentic signal and another from a nonauthentic signal be used to produce a forgery. In that case, we would be interested in having a large value of $N_p$, producing a sharp degradation of the correlation statistic, and therefore, implying a reduction in the false positive probability. Nevertheless, we are also interested in having a smooth degradation of the correlation statistic with the number of mistakes in the hash, as the authentication scheme should be robust to slight modifications, such as transcoding or the watermark embedding itself. In that sense, the smaller the value of $N_p$, the smoother the performance degradation, and therefore the more robust the authentication scheme. As a conclusion, we can establish that a trade-off exists between robustness and probability of false positive.

In order to illustrate the dependence of the probability of false negative on $N_p$, in Fig. 3 the behavior of $P_{fn}$ is plotted as a function of $N_p$ for different values of $N_b$ for Lena (with $M = 4096$, $\eta = 0$ and $\gamma = 15$), after fixing the probability of false positive to $P_{fp} = 0.005$. In the particular case studied in Fig. 3, the value of $N_p$ which minimizes $P_{fn}$ just slightly depends on $N_b$. Notice that if $\mathbf{z}$ is not watermarked or the detector and the embedder use different keys then $P_{fn} \approx 1/2$. Therefore, if we increase $N_p$, the error probability in the seed used to generate $\mathbf{v}$ will rise and consequently $P_{fn}$ will also be larger.

The dependence of $P_{fn}$ on the parameters of the GGD used for modeling $\mathbf{X}^T \cdot \mathbf{S} \cdot \mathbf{X}$ when $\mathbf{N}_b$ holds is illustrated in Fig. 4, where $P_{fn}$ is plotted as a function of the shaping factor $c_X$ for different standard deviations of the projected blocks $\sigma_X$, $M = 4096$ and $\gamma = 10$. The considered range of $\sigma_X$ corresponds to typical values of the standard deviation obtained for the projected coefficients of real images. On one hand, and according to intuition, it can be seen that for larger values of the document-to-watermark ratio (proportional to $\sigma^2_X / \gamma^2$), the probability that the watermark embedding flips bits of the original robust hash will be smaller. On the other hand, the results in Fig. 4 indicate that $P_{fn}$ decreases with $c_X$.

Finally, in order to identify the limitations of the analysis presented above, we will show how it can be adapted to deal with typical image processing attacks such as JPEG compression, intensity transformations, and linear filtering.

The JPEG standard compresses an image by quantizing its $8 \times 8$ block-DCT coefficients so that the step-size of the used uniform scalar quantizer depends on both the desired JPEG quality factor (QF) and the frequency of each particular coefficient (quantization step-sizes for low-frequency coefficients are usually smaller than high-frequency ones). Our analytical framework can be adapted to deal with large QF JPEG compression by using a fine quantization approximation, i.e., by modeling the effect of the quantization error in the $8 \times 8$ block-DCT
with a noise random variable independent of the to-be-quantized coefficient and uniformly distributed in $[-\Delta_i/2, \Delta_i/2]$, where $\Delta_i$ is the step-size of the $i$th coefficient scalar quantizer. Considering the $8 \times 8$ IDCT transform, one can obtain the distribution of $N$ in (1). Obviously, when the QF is decreased, the fine quantization approximation fails and our analysis can be no longer applied. This is illustrated in Fig. 5 where we compare the values of the complementary cumulative density function of $\rho_0$ (i.e., $1 - F_{\rho_0}(T)$) as function of the QF for the actual JPEG compression, the analytical results obtained by adapting our analysis considering fine-quantization, and the empirical results obtained when noise uniformly distributed in $[-\Delta_i/2, \Delta_i/2]$ is added to the corresponding $8 \times 8$ DCT coefficients. As expected, the plots corresponding to the two latter are very similar for the full range of QF, whereas both of them are close to the actual JPEG compression just for QF > 90, i.e., while the fine-quantization approximation holds.

Affine point-wise intensity transformations consist in scaling the luminance values of an image and centering the histogram of the resulting image by adding an offset. This processing can be modeled by changing $\eta$ and $\mathbf{n}$ in (1). Analytical and empirical curves for different intensity change affine correction functions can be found in Fig. 6, showing the good match between both curves; the only plot with a significant difference between both results corresponds to $\eta = 1/0.7$, where the luminance transformation becomes nonlinear due to clipping. Notice that nonlinear transformations are not encompassed by our model.

In addition, the proposed analysis can be also adapted to deal with linear space invariant (LSI) image filtering, by modeling both the image block projection distribution and the distribution of the correlation between the watermark and the filtered watermarked signal. To this end, it is reasonable to neglect border effects and approximate image filtering by a $64 \times 64$ block circular filtering, so the pdf of the mentioned correlation $\rho$ can be calculated by using the circular convolution theorem. Due to space limitations, we have chosen low-pass (Gaussian) and high-pass (Sobel) filters as representatives of the class of LSI filters. Concerning the Gaussian filtering, it is worth noting that the obtained $P_c$ will be very similar to the $P_c$ without filtering, as the Gaussian filters just slightly modify the low frequency components used to generate the robust hash. Knowing that host signal variance is concentrated in the low frequencies, and we are using a white watermark, for large document-to-watermark ratios the correlation between the Gaussian filtered watermarked signal and the reconstructed watermark will have a variance similar to the nonfiltered case, whereas its mean will be reduced. On the other hand, when the image is Sobel-filtered, $P_c$ can be approximated by 0.5, as most frequencies used to produce the robust hash are erased; as a consequence, the mean of the correlation between the Sobel filtered watermarked signal and the reconstructed watermark will be approximately null, whereas its variance will be dramatically reduced due to the low frequencies host power allocation. The behavior of both filters is shown in Fig. 6, in accordance to our previous discussion.

Finally, we would like to emphasize that, as stated above, the computation time needed to assess the performance of the analyzed self-embedding authentication algorithm using our analytical methodology is dramatically smaller than the time required by Monte Carlo techniques; as an example, a common PC with an Intel Core 2 Quad CPU at 2.4 GHz and 4 Gb of RAM with Matlab 7.4 spends more than six days carrying out the Monte Carlo simulations needed to obtain the curves of Fig. 1. In contrast, less than five minutes are needed to generate the corresponding analytical plots.
\[
Pr(N_{e,k} = m_k | N_{e,k-1} = m_{k-1}, N_e = n_e) = \begin{cases} 
1, & \text{if } k = 1 \text{ and } m_k = n_e \\
0, & \text{if } k = 1 \text{ and } m_k \neq n_e \\
\sum_{m_{k-1} = 0}^{N_{e,k-1}} Pr(N_{e,k} = m_k | N_{e,k-1} = m_{k-1}, N_e = n_e) , & \text{otherwise.}
\end{cases}
\]

V. CONCLUSION

A thorough performance analysis of the self-embedding authentication method proposed by Fridrich and Goljan was given in this paper. In this analysis, the hash bit error is first computed; then, the pdf of the projection of the received image blocks onto the reconstructed watermark is approximated depending on that robust hash bit-error probability. This allows us to obtain closed formulas for the false positive and false negative probabilities.

An important characteristic of the self-embedding authentication method is that the embedding process itself can modify the robust hash of the image and consequently corrupt the reconstructed watermark; however, in practical cases, a larger embedding distortion could increase the correlation between the received signal and the watermark estimated at the detector. Our analysis shows that, although the hash bit-error probabilities usually increase with the embedding distortion, the overall performance of the authentication system improves. We have also observed, based on (7) and (8), that given \( P_{\eta} \), \( P_{\eta} \) is a convex function of \( N_{e,k} \) with the optimal \( N_{e,k} \) almost invariant with \( N_{e} \). Furthermore, we have seen how \( P_{\eta} \) depends on the standard deviation and the shaping parameter of the projection (modeled by a GGD) of the image blocks onto the pseudorandom diagram.

Future research will focus on the analysis of self-embedding authentication techniques where the spread-spectrum embedding is replaced by informative embedding strategies, e.g., [19].

APPENDIX A

DERIVATION OF HASH-BIT-ERROR PROBABILITY

Defining \( G_j \triangleq (1/M)Z_j^T S^j = (1/M)(\eta X^T S^j + \gamma \eta W^T S^j + N^T S^j) \), the hash-bit error can be written as:

\[
P_{\eta} = \frac{1}{2} Pr(|G_j| < T_d | H_j = 1) + \frac{1}{2} Pr(|G_j| > T_d | H_j = 0)
= \frac{1}{2} Pr(\{G_j| < T_d 1/M|\eta X^T S^j| > \eta T_d\})
+ \frac{1}{2} Pr(\{G_j| > T_d 1/M|\eta X^T S^j| < \eta T_d\})
= \frac{Pr(\{G_j| < T_d 1/M|\eta X^T S^j| > \eta T_d\})}{Pr(\{G_j| > T_d 1/M|\eta X^T S^j| < \eta T_d\})}
+ \frac{Pr(\{G_j| > T_d 1/M|\eta X^T S^j| < \eta T_d\})}{Pr(\{G_j| < T_d 1/M|\eta X^T S^j| > \eta T_d\})}
= \int_{-\infty}^{\infty} Pr\left(\frac{1}{M}NT^T S^j = t\right)
\left[\int_{-\infty}^{\eta T_d} Pr\left(\frac{1}{M}F_j < t < T_d, \frac{1}{M}|\eta X^T S^j| = \tau\right)dt + \int_{\eta T_d}^{\infty} Pr\left(\frac{1}{M}F_j > t > T_d, \frac{1}{M}|\eta X^T S^j| = \tau\right)dt\right]
\times dt
\]

which yields (5) and where in (12) \( F_j \triangleq \eta X^T S^j + \gamma \eta W^T S^j \).

APPENDIX B

DERIVATION OF THE DETECTION ERROR PROBABILITIES

First, we will derive the probability that given a number of bit errors \( n_e \) in the estimate of the hash, the number of wrong vectors \( t^j \) is \( n_e \). In order to do so, we will obtain the probabilities \( Pr(N_{e,k} = m_k | N_{e,k-1} = m_{k-1}, N_e = n_e) \), where \( N_{e,k} \) denotes the number of wrong vectors \( t^j \) after \( k \) permutations of the reconstructed hash vector \( 1 \leq k \leq N_e \). These probabilities can be shown to be given by (10), shown at the top of the page, where \( 2 \leq k \leq N_e, n_e \leq N_e \). From there, it is possible to write \( Pr(N_{e,k} = m_k | N_e = n_e) \) as (11), shown at the top of the page. Finally, given that the probability of having \( n_e \) mistakes in the hash estimate is

\[
Pr(N_e = n_e) = \binom{N_e}{n_e} p^n \eta (1 - p)^{N_e - n_e}
\]

the probability of having \( n_e \) wrong vectors \( t^j \) after \( N_e \) permutations can be written as

\[
Pr(N_e = n_e) = \sum_{n_e=0}^{N_e} Pr(N_e, N_p = m | N_e = n_e) Pr(N_e = n_e). \quad (13)
\]

On the other hand, the correlation statistic under the hypothesis \( H_0 \) is given by

\[
\rho_{H_0} = \frac{1}{M}(\eta X^T \tilde{w} + \gamma \eta W^T \tilde{w} + n^T \tilde{w})
= \frac{1}{M}\left(\eta X^T \tilde{w} + \frac{3}{N_h} \eta \sum_{j=1}^{N_h} \tilde{v}_j^T \tilde{v}_j + n^T \tilde{w}\right). \quad (14)
\]

When \( x \) and \( \tilde{w} \) are uncorrelated, so are \( v^i \) and \( \tilde{v}^i \) when \( i \neq j \), while if \( N_e = n_e \), the vectors \( v^i \) and \( \tilde{v}^i \) coincide in precisely \( N_e - n_e \) values; then, using (14), it is possible to conclude that the mean of \( \rho_{H_0} \) is

\[
E\{\rho_{H_0} | N_e = n_e\} = (N_h - n_e) \frac{\gamma \eta}{N_h}. \quad (15)
\]

Similarly, the variance of \( \rho_{H_0} \) becomes

\[
\text{Var}\{\rho_{H_0} | N_e = n_e\} = \frac{1}{M}(\eta \gamma^2 \sum_{i=1}^{M} \sigma_i^2 + \eta^2 \sum_{i=1}^{M} \sigma_i^2 + \sigma_0^2 + \sigma_N^2) \quad (16)
\]

where

\[
\sigma_i^2 = \frac{1}{N_e \sigma_i^2} \left(\sum_{j=1}^{N_e} (N_e - n_e) \sigma_i^2 + n_e \sigma_i^2 + (N_e - 1)N_h \sigma_i^2\right)
\]

\[
+ (N_e - n_e)(N_h - n_e - 1)\sigma_i^2
\]

\[
\sigma_0^2 \triangleq \text{Var}\{V_i^2\}, \quad i = 1, \ldots, M. \quad \text{Since } V_i \sim \mathcal{N}(0, 1), \quad \text{then } \sigma_1^2 = 1/9 \text{ and } \sigma_2^2 = 4/45. \quad \text{In order to analyze the distribution of } \rho_{H_0}, \quad \text{one can straightforwardly adapt the discussion in Section III about the distribution of } W^T \cdot S \text{ to obtain the distribution of } X^T \cdot W \text{ and } W \cdot W \text{, and apply the CLT for approximating the pdf of}
\]

Authorized licensed use limited to: Roberto Lopez-Valcarraro. Downloaded on January 25, 2010 at 04:06 from IEEE Xplore. Restrictions apply.
\[ \mathbf{W}^T \cdot \mathbf{W} \] by a Gaussian for large values of \( N_k \) and \( M \). The resulting approximation is \( \hat{\rho}_{H_1|N_k=n \cdot} \sim \mathcal{N}(\mathbf{X}_k - \mathbf{n} \cdot (\gamma_{\hat{\rho}_{H_1}}/\mathcal{N}_k, (1/M)(\gamma_{\hat{\rho}_{H_1}}/(\gamma_{\hat{\rho}_{H_1}} + \gamma^2 \cdot n^2 \cdot \sigma_{\hat{\rho}_{H_1}}^2 + \sigma_{\hat{\rho}_{H_1}}^2)) \). Equivalently, it is clear that for \( \hat{H}_1 \) we can approximate \( \hat{\rho}_{H_1|N_k=n \cdot} \sim \mathcal{N}(\mathbf{X}_k - \mathbf{n} \cdot (\gamma_{\hat{\rho}_{H_1}}/\mathcal{N}_k, (1/M)(\gamma_{\hat{\rho}_{H_1}}^2 + \gamma^2 \cdot n^2 \cdot \sigma_{\hat{\rho}_{H_1}}^2 + \sigma_{\hat{\rho}_{H_1}}^2)) \). In this way, when \( \text{Var} \{ \rho_{H_0} \} / \text{Var} \{ \rho_{H_1} \} > 1 \), \( P_0 = \sum_{n=0}^{N_k} \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_0} \} \hat{\rho}_{H_0} \hat{\rho}_{H_0} \leq \hat{\rho}_{H_0} \leq T_1 \) and \( P_1 = \sum_{n=0}^{N_k} \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_1} \} \hat{\rho}_{H_1} \hat{\rho}_{H_1} \leq \hat{\rho}_{H_1} \leq T_1 \). Finally, when \( \text{Var} \{ \rho_{H_0} \} / \text{Var} \{ \rho_{H_1} \} < 1 \), \( P_0 = \sum_{n=0}^{N_k} \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_0} \} \hat{\rho}_{H_0} \hat{\rho}_{H_0} \leq \hat{\rho}_{H_0} \leq T_1 \) and \( P_1 = \sum_{n=0}^{N_k} \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_1} \} \hat{\rho}_{H_1} \hat{\rho}_{H_1} \leq \hat{\rho}_{H_1} \leq T_1 \). Replacing in the previous expressions the probabilities of \( \hat{\rho}_{H_1} \) by their integral form, one obtains (7) and (8), respectively. Aside from the obvious differences between the expressions (7) and (8), it is important to note that \( \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_0} \} \) and \( \text{Pr} \{ N_k = n \cdot \hat{\rho}_{H_1} \} \) are different, since the expression for \( P_1 \) depends on the considered hypothesis, as stated in Section III.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their comments, which helped to improve the present manuscript.

REFERENCES