Obfuscation in Digital Fingerprinting*

Hans Georg Schaathun

NTNU, Norwegian University of Science and Technoloogy
Faculty of Engineering and Natural Sciences
Pb 1517
N-6025 Ålesund
Norway, ⟨georg@schaathun.net⟩

Minoru Kuribayashi

Okayama University
Graduate School of Natural Science and Technology
Okayama-shi
700-8530 Japan
⟨kminoru@okayama-u.ac.jp⟩

30th November 2017

Hans Georg Schaathun is cand.mag. 1996 (Mathematics, Economics, and Informatics), cand.scient. 1999 (Industrial and Applied Mathematics and Informatics), and dr.scient. 2002 (Informatics – Coding Theory), all from the University of Bergen, Norway. He was lecturer and post in coding and cryptography at the University of Bergen 2002 and Post.Doc. 2003-2006. As a lecturer and senior lecturer in computer science at the University of Surrey, England 2006-2010, his research focused on multimedia security including applications of coding theory and steganalysis using machine learning. He joined Ålesund University College (now NTNU) and became professor in 2011. Current research areas include software engineering and AI.

Minoru Kuribayashi received B.E., M.E., and D.E degrees from Kobe University, Kobe, Japan, in 1999, 2001, and 2004. From 2002 to 2007, he was a Research Associate in the Department of Electrical and Electronic Engineering, Kobe University. In 2007,

[*] A paper in honour of Gérard Cohen on the occasion of his 64th birthday. The first author is very grateful to Prof. Cohen, for his introduction to digital fingerprinting many years ago.

he was appointed as an Assistant Professor at the Division of Electrical and Electronic Engineering, Kobe University. Since 2015, he has been an Associate Professor in the Graduate School of Natural Science and Technology, Okayama University. His research interests include digital watermarking, information security, cryptography, and coding theory. He received the Young Professionals Award from IEEE Kansai Section in 2014. He is a senior member of IEEE and IEICE.

Abstract: Digital fingerprinting has been much studied in the literature for more than twenty years, motivated by applications in copyright protection. A popular and practical approach is to use spread spectrum watermarking to embed fingerprints in multimedia objects. Solutions are normally only validated against known attacks in simulation.

In this paper we review known attacks on spread spectrum fingerprinting, and give a mathematical argument for the efficacy of the so-called MMX attack. We also provide a new, mathematical description the obfuscation technique which we proposed at ICIP 2015, identifying some key properties which are necessary to resist the MMX and other attacks.

Keywords: Digital fingerprinting, watermarking, collusion-secure code, signal processing, coding theory, spread spectrum, copyright protection

1 Introduction

More than thirty years have passed since Wagner [Wag83] suggested digital fingerprinting as a means of copyright protection. Every legitimate copy of the copyrighted work is marked with the identity of the licensed user. If unauthorised copies ever appear, they can be traced back to the guilty source. Obviously, the mark, which we call a *fingerprint*, must be embedded in such a way that the users cannot remove it. The major challenge is collusion attacks, where multiple users compare their copies and make attacks based on the differences between copies.

Following the pioneering work of Boneh and Shaw [BS95], there has been substantial research into collusion-secure fingerprinting. Most works on fingerprinting assume (explicitly or implicitly) a layered model, consisting of a fingerprinting code (coding layer) and an embedding (watermarking) layer.

The fingerprinting layer can be described as a code, mapping user identities into codewords, or fingerprints, over some alphabet, and a decoding algorithm which makes the inverse mapping, also allowing for corrupt fingerprints. Studies on this layer typically assumes an abstract model, with a *marking assumption* which describes what attacks a collusion of users can perform. It is simply assumed that it is possible to embed the codeword symbol by symbol in a multimedia object so that the marking assumption is satisfied. A number of different marking assumptions have been proposed.

The embedding layer typically borrows a solution from digital watermarking, with spread spectrum watermarking (SS) being particularly popular. Additional care must be taken to make the system collusion-secure. It is common to combine spread-spectrum fingerprinting with well-known fingerprinting codes, such as the Tardos code [Tar05], in spite of the fact that SS watermarking does not satisfy the Marking Assumption.

Practical fingerprinting systems in the literature are usually tested experimentally to assess robustness against known attacks. Theoretical analysis is not possible without some attack model, such as the marking assumption. It is interesting to note that the practical solutions use much higher code rates than we would require based on the theoretical analysis of the fingerprinting layer on its own. Equally interesting, there are few, if any, practical fingerprinting system which resist all currently known attacks.

In this paper we review some of the known fingerprinting systems and attacks, focusing on spread spectrum constructions. We analyse the MMX and Uniform attacks which has previously proved effective in simulations, and we demonstrate their efficacy mathematically. Finally, we provide a new mathematical framework to study the obfuscation technique which has thwarted the MMX attack in simulations [KS15]. A security proof for obfuscation remains an open problem, but we do identify some interesting properties which may be relevant for further research.

2 The fingerprinting problem

Wagner [Wag83] introduced a taxonomy for digital watermarking as early as 1983. A *distributor* is the authorised supplier of fingerprinted objects, giving authorised access to *users*. The *opponent* is an entity who makes unauthorised use of objects, through one or more users. The *distributor's goal* is to identify the user(s) whom the opponent has compromised. The opponent's goal, conversely, is to prevent the identification, even when the distributor is able to inspect objects which have been used in unauthorised ways.

2.1 Collusion-secure codes

Boneh and Shaw [BS95, BS98] introduced a model for digital fingerprinting in the presence of collusion attacks, i.e. when the opponent has access to the copies of several users. Creating a *hybrid copy* based on multiple objects, the opponent can hope to prevent identification.

Each user is identified by a codeword, called a *fingerprint*, from some code $\mathcal{C}$, and each symbol from the codeword is embedded in the fingerprinted copy. Extracting a fingerprint from a hybrid copy, the distributor can get a *hybrid fingerprint*. This is analogous to a noisy codeword in conventional communication. The Marking Assumptions defines the opponent's space of opportunity, i.e. the set of hybrid fingerprints which can possibly be created in an attack. It says that for each coordinate position i, the hybrid copy can only contain a symbol which is seen by at least one of the users.

Definition 1 (Boneh-Shaw Marking Assumption)**.** *Let $\mathcal{P} \subset \mathcal{C}$ be a collusion of t users, and suppose they create a hybrid fingerprint $\mathbf{r} = (r_1, \ldots, r_n)$. Then*

$$\forall i = 1, \ldots, n, \exists (c_1, \ldots, c_n) \in \mathcal{P} \text{ s.t. } c_i = r_i. \tag{1}$$

A code $\mathcal{C}$ is said to be collusion-secure if the distributor, observing the hybrid fingerprint $\mathbf{r}$, can identify at least one user fingerprint $\mathbf{c} \in \mathcal{P}$. Boneh and Shaw also provided a code construction which is collusion-secure with bounded error probability. The Tardos code [Tar05] is a more recent construction with better code rate. His construction has been studied, analysed, improved and generalised by several authors [ŠVCT06, AT09, NFH$^+$09, ŠKC08, CXFF09, SKSC11, OSD13, ISO13]. Combination of the Tardos code with specific modulation schemes for embedding in multimedia has also been studied [XFF08].

It is also possible to construct combinatorially secure codes, which always allow the distributor to correctly identify at least one user with zero error probability [SSW01, CS04].

A number of alternative marking assumptions exist. Guth and Pfitzmann [GP00] introduced a marking assumption allowing for random errors, meaning that the condition (1) could be broken in each position i with a bounded probability ϵ. This marking assumption is arguable much more realistic, because the watermarking system used to embed the fingerprint does not have to be perfect. The Boneh-Shaw code is secure also under the Guth-Pfitzmann Marking Assumption with only a modest sacrifice of code rate [Sch08b].

The Marking Assumptions establish abstract models, and the problem of making practical systems which satisfy the assumption has received significantly less attention than the construction of collusion-secure codes within the model.

2.2 Spread Spectrum Fingerprinting

Digital watermarking based on spread spectrum was proposed very early [CKLS97], and many authors have further applied this to digital fingerprinting. As with collusion-secure codes, each user is represented by a fingerprint $\mathbf{w}_i \in \mathcal{C}$, but now, the fingerprint is a real-valued signal, rather than a word over a discrete alphabet. The set $\mathcal{C} \subset \mathbb{R}^n$ of fingerprints is a a family of orthogonal or near-orthogonal sequences. The media file (object to be fingerprinted) can be viewed as a signal $\mathbf{x} \in \mathbb{R}^n$, and the fingerprint can be added thereto, to give the fingerprinted object for user i:

$$\mathbf{y}_i = \mathbf{x} + \alpha \mathbf{w}_i,$$

for some embedding strength α.

On the receiver side, the decoder is given a noisy version $\mathbf{y}'$ of the fingerprinted copy, and aims to recover the original watermark $\mathbf{w}_i$ or the corresponding user i. A non-blind decoder knows the original work $\mathbf{x}$, and can subtract it to obtain a noisy fingerprint

$$\mathbf{w}' = \frac{\mathbf{y}' - \mathbf{x}}{\alpha}.$$

Thus, the original work has no impact on a non-blind decoder. A blind decoder does not know the original host signal $\mathbf{x}$ and has to treat it as additional noise to the fingerprint. This paper will only consider non-blind decoding.

The main constraint in SS fingerprinting is that the distortion must be negligible. More precisely, the fingerprinted copy $\mathbf{y}_i$ must be perceptually indistinguishable from the original work $\mathbf{x}$. A common measure for the distortion is squared Euclidean distance,

$$||\mathbf{y}_i - \mathbf{x}||^2 = \alpha^2 ||\mathbf{w}_i||^2,$$

which is also known as the power of the fingerprint $\alpha \mathbf{w}_i$. For particular classes of media, it may be possible to make perceptual models which more accurately measure the relevant distortion. In this paper, we will stick to power as the distortion measure, because of its generality.

The fingerprinting strength α scales the fingerprint to make a trade-off between decodability and distortion. When the decoder is non-blind, we can ignore the host signal $\mathbf{x}$ and assume $\alpha = 1$ without loss of generality.

A simple and well-known spread spectrum decoder uses the correlation

$$s_i(\mathbf{w}') = \mathbf{w}' \cdot \mathbf{w}_i$$

as the *detection score* of user i. The decoder can output the user i maximising this score. Alternatively, a list decoder can output a set of users with scores above a given threshold. We observe that

$$s_i(\mathbf{w}_i) = ||\mathbf{w}_i||^2, \tag{2}$$

$$s_i(\mathbf{w}_{i'}) \approx 0, \quad \text{when } i \neq i' \tag{3}$$

because the sequences are near-orthogonal.

Each sample, or co-ordinate position, in the watermark is commonly referred to as a *mark* in the fingerprinting literature. Note that the detection score is a sum of one term for each mark:

$$s_i(\mathbf{w}_k) = \sum_j w_{i,j} w_{k,j}. \tag{4}$$

This is an important observation because it allows us to do most of the analysis based on a single mark. We will refer to the map $w_{i,j} \mapsto w_{i,j} w_{k,j}$ as the *mark detection score.*

In this paper we will focus on random binary sequences, to allow for simple proofs based on randomness. Each fingerprint $\mathbf{w}_i$ is selected independently and uniformly at random over the alphabet $\{\pm 1\}$. It is easily verified that the expected value $E(s_i(\mathbf{w}_j)) = 0$ when $i \neq j$ and the fingerprint $\mathbf{w}_j$ is drawn at random. Thus (3) holds when the fingerprint length n is sufficiently large.

Other authors have studied random Gaussian fingerprints [ZWWL05] or orthogonal families [Kur12]. The same principles apply in all cases. Alternative decoding scores can also be found [ZWWL05]. Various techniques can be added to improve performance against certain attacks, such as interference removal [Kur12] and the preprocessor proposed in [ZWWL05].

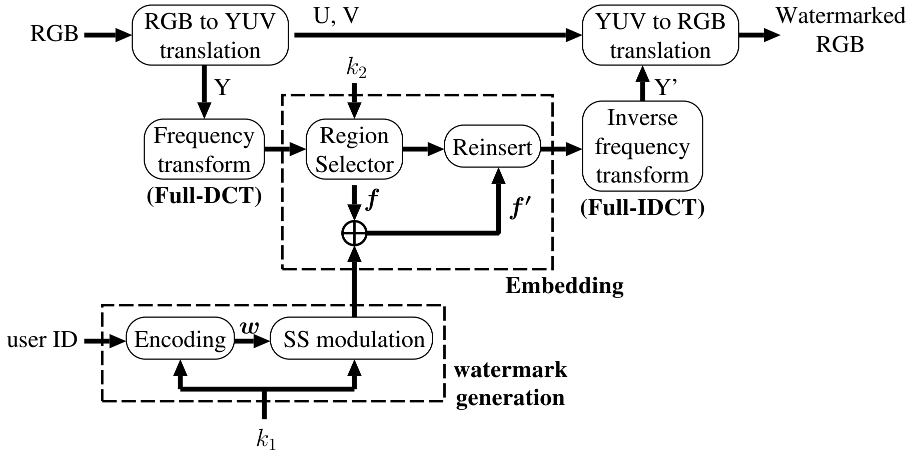

Figure 1: Block diagram of embedding procedure in [Kur14].

2.3 A complete fingerprinting system

The fingerprinted objects are usually assumed to be images or media objects. Here we give an example of how spread spectrum fingerprinting can be applied to images. We consider the recent scheme from [Kur14], as shown in Figure 1. This design is typical for many systems in the literature.

The system consists of several key components. Following [HW06] and [Sch08b], the modules are called *layers*. The coding layer (*watermark generation*) defines the code $\mathcal{C}$ of fingerprints. The watermarking (*embedding*) layer embeds fingerprints into host signals to create fingerprinted signals. Each of these components are parameterised by secret keys k_1 and k_2.

Finally, we need a layer to extract a suitable signal from the host objects (media files). In Figure 1, this consists of taking only the luminance (grey-scale) component and making a frequency transform (DCT). The DCT transform means that the region selector can avoid the highest frequencies where the fingerprint would be vulnerable to noise attacks and image resizing, and also the lowest frequencies where the perceptible impact would be unacceptable. Observe that there is no secret key in the host layer. Therefore, in accordance with Kerckhoffs' (second) principle, we assume that the attacker can extract the same signal as the decoder, and apply the attack on the watermarking layer, and then reinsert the attacked fingerprinted signal in the host.

For each layer, especially the host and coding layers, there are a large number of proposed constructions in the literature.

The core element of the Watermarking Layer is the *embedding operation*. In this paper, we will only consider additive spread spectrum (SS) fingerprinting with non-blind decoder, as described in the previous section. Additionally, the Watermarking Layer commonly includes a *region selector*, which pseudo-randomly selects coefficients to be used by the embedding operation. A key k_2 is used to seed the pseudo-random generator for this selection. The motivation for the region selector has been to make it harder for the opponent to know which coefficients to attack.

For the coding layer, we will only consider a random binary code over the alphabet $\{\pm 1\}$ with correlation decoding. Other spread spectrum constructions are likely to have similar properties.

3 Attacks on Spread Spectrum Fingerprinting

Digital fingerprinting is concerned with two types of attacks, collusive and non-collusive. Non-collusive attacks include attacks known from other applications of spread-spectrum, such as additive noise (Gaussian or otherwise). Spread spectrum is very robust against such attacks.

3.1 Collusion attacks

It is natural to expect the attacks to be most effective when they are applied to the same domain as the embedding. Assuming that the opponent knows the system design, and only the secret keys are secret (cf. Section 2.3), he can extract the signal in the host layer, and apply the attacks to the fingerprinted signals $\mathbf{y}$.

In a collusive attack, the opponent has access to a set of fingerprinted copies. Typically, the assumption is that the opponent is a collusion of users, each holding one fingerprinted copy, but it does not matter to the analysis how the opponent came by the fingerprinted copies. The colluding users are commonly referred to as colluders or pirates, and we use the term *pirate fingerprints* about the fingerprints embedded in the copies held by the opponent. The set of pirate fingerprints will be called the *collusion.*

The output of a collusion attack is not just a noisy version of one fingerprinted copy, but a hybrid of multiple copies with different fingerprints. Let P be a matrix with all the pirate copies as rows. We will only consider attacks which operate independently on each mark. Thus we can write the attack as a function $a : \mathbf{c} \mapsto z$, where $\mathbf{c}$ is a *column* of P. The attack a is applied to every column to produce the hybrid fingerprint.

It is well known that SS fingerprinting is robust against a number of collusive attacks based on basic signal processing operations, such as minimum, maximum, and average (see Table 1). It is also robust against non-collusive noise attacks, such as AWGN, and against a combination of averaging and AWGN.

In contrast, SS fingerprinting is very vulnerable to two less well-known attacks, namely the MMX attack [Sch07] and the Uniform attack [Sch08a]. These attacks are defined as follows.

Definition 2 (Moderated Minority Extreme (MMX))**.** *Let* $\Delta = a^{\mathrm{avg}}(\mathbf{c}) - a^{\mathrm{mid}}(\mathbf{c})$*. The MMX attack for a given threshold* θ *is defined as*

$$a_\theta^{\mathrm{MMX}}(\mathbf{c}) = \begin{cases} a^{\mathrm{min}}(\mathbf{c}) & \textit{if } \Delta \geq \theta \\ a^{\mathrm{avg}}(\mathbf{c}) & \textit{if } |\Delta| < \theta \\ a^{\mathrm{max}}(\mathbf{c}) & \textit{if } \Delta \leq -\theta \end{cases} \tag{5}$$

Average:	$a^{\mathrm{avg}}(\mathbf{c}) = \dfrac{1}{t}\displaystyle\sum_{j=1}^{t} y_j$
Minimum:	$a^{\mathrm{min}}(\mathbf{c}) = \min_j c_j$
Maximum:	$a^{\mathrm{max}}(\mathbf{c}) = \max_j c_j$
Median:	$a^{\mathrm{med}}(\mathbf{c}) = \mathrm{median}_j\ c_j$
Midpoint(MinMax):	$a^{\mathrm{mid}}(\mathbf{c}) = \dfrac{a^{\mathrm{min}}(\mathbf{c}) + a^{\mathrm{max}}(\mathbf{c})}{2}$

Table 1: Common signal processing attacks.

Definition 3 (Uniform attack)**.** *The uniform attack a^{U}_{α} with scaling factor α, $(0 \le \alpha \le 1)$ is a probabilistic attack. Given a column $\mathbf{c}$ it outputs a hybrid mark z drawn uniformly at random from the range $a^{\mathrm{mid}}(\mathbf{c}) \pm \alpha d_i$ where $d_i = (a^{\mathrm{max}}(\mathbf{c}) - a^{\mathrm{min}}(\mathbf{c}))/2$.*

3.2 Classification of Attacks

The attacks, as discussed above, are applied to the fingerprinted copies, and one might assume that the effect depends on the host signal. This is not the case, at least not when the embedding is additive watermarking. We introduce the concept of homomorphic attacks. By abuse of notation, we take $\mathbf{c} + P$ for any n-dimensional row vector $\mathbf{c}$ and $n \times m$ matrix P to mean the matrix obtained by adding $\mathbf{c}$ to each row of P.

Definition 4 (Homomorphic Attack)**.** *An attack a is said to be homomorphic if it satisfies*

$$a(\mathbf{x} + P) = \mathbf{x} + a(P) \tag{6}$$

for any collusion P and any signal $\mathbf{x}$.

Homomorphic attacks can be studied independently of the host signal. If we let $\mathbf{x}$ be the host signal, and P a matrix with the pirate fingerprints as rows, then (6) means that the attack applied to the fingerprinted copies gives the same signal as the attack applied to the pirate fingerprints and then added to the host. In other words, we can ignore the host signal in the analysis of the attack.

Remark 1. *It is easy to confirm that all the attacks considered above (MMX, uniform, MAX, MIN, and average) are homomorphic.*

3.3 Attack analysis

We consider a single mark. Both the attack a and the decoding score s is well-defined on each mark. Let x_i be the mark seen by the ith pirate. The score associated with pirate i is

$$S_i = s_i(a(x_1, x_2, \ldots, x_t)) = x_i \cdot a(x_1, x_2, \ldots, x_t).$$

which is a stochastic variable with a probability distribution induced by the probability distribution of the random fingerprints, and also by the attack in the event of a probabilistic attack. All the attacks we have considered are symmetric in the pirates, so S_i has the same distribution for all i. In the sequel, we omit the subscript i.

Below, we shall find $E(S)$ under different attacks. The code is a random binary code over ±1. We will write $X^{\min} = a^{\min}(X_1, \ldots, X_t)$ for the sake of brevity, and similarly for $X^{\max}$, X^{mid}, X^{MMX}_θ, etc.

Proposition 1. *Let X_i for $i = 1, 2, \ldots, t$ be uniformly and independently distributed over the alphabet $\{\pm 1\}$. Let*

$$S = X_1 \cdot a^{\min}(X_1, X_2, \ldots, X_t).$$

Then we have

$$E(S) = 0.5^{t-1} \tag{7}$$
$$\mathsf{var}(S) = 1 - 0.5^{2t-2} \tag{8}$$

Proof. Both $X^{\min}$ and X_1 are ±1, and hence $S = \pm 1$. With probability 0.5, we get $X_1 = -1$, and in this case, we always get $X^{\min} = -1$ and $S = 1$. With probability 0.5, $X_1 = +1$, and in this case, we get $X^{\min} = +1$ if $X_i = +1$ for all i, something which happens with probability 0.5^{t-1}. Otherwise $X^{\min} = -1$. Thus we get the following probabilities:

$$\mathsf{P}(X^{\min} = +1, X_1 = -1) = 0, \tag{9}$$
$$\mathsf{P}(X^{\min} = +1, X_1 = +1) = 0.5^t, \tag{10}$$
$$\mathsf{P}(X^{\min} = -1, X_1 = -1) = 0.5 \tag{11}$$
$$\mathsf{P}(X^{\min} = -1, X_1 = +1) = 0.5(1 - 0.5^{t-1}). \tag{12}$$

It follows that

$$\mathsf{P}(S = -1) = 0.5 - 0.5^t, \tag{13}$$
$$\mathsf{P}(S = +1) = 0.5 + 0.5^t. \tag{14}$$

We get

$$E(S) = -(0.5 - 0.5^t) + (0.5 + 0.5^t) = 0.5^{t-1}. \tag{15}$$

The variance is easily calculated using (13)–(15). □

Proposition 2. *Let X_i for $i = 1, 2, \ldots, t$ be uniformly and independently distributed over the alphabet $\{\pm 1\}$. Let*

$$S = X_1 \cdot a^{\max}(X_1, X_2, \ldots, X_t).$$

Then we have

$$E(S) = 0.5^{t-1} \tag{16}$$
$$\mathsf{var}(S) = 1 - 0.5^{2t-2} \tag{17}$$

The proof is similar to that of Proposition 1 by symmetry.

Proposition 3. *Let X_i for $i = 1, 2, \dots, t$ be uniformly and independently distributed over the alphabet $\{\pm 1\}$. If*

$$S = X_1 \cdot a^{\text{avg}}(X_1, X_2, \dots, X_t),$$

then

$$E(S) = \frac{1}{t} \tag{18}$$

$$\mathsf{var}(S) = \frac{t-1}{2t^2}. \tag{19}$$

Proof. Spelling out the attack function, we get

$$S = \frac{1}{t}\sum_{i=1}^{t} X_i X_1 = \frac{X_1^2}{t} + \frac{1}{t}\sum_{i=2}^{t} X_i X_1$$

The first term is always $1/t$. The expected value can be written as

$$E(S) = \frac{1}{t} + \sum_{x=\pm 1} \mathsf{P}(X_1 = x) \cdot x \cdot \frac{1}{t}\sum_{i=2}^{t} E(X_i)$$

and since $E(X_i) = 0$, we get $E(S) = 1/t$.

The variance is given as

$$\mathsf{var}(S) = \sum_{x_1=\pm 1}\sum_{x_2=\pm 1}\cdots\sum_{x_t=\pm 1}\left(\frac{1}{2}\right)^t \left(\frac{x_1^2}{t} + \frac{1}{t}\sum_{i=2}^{t} x_i x_1 - \frac{1}{t}\right)^2. \tag{20}$$

Because $x_1^2 \equiv 1$, we get

$$\mathsf{var}(S) = \sum_{x_1=\pm 1}\sum_{x_2=\pm 1}\cdots\sum_{x_t=\pm 1}\left(\frac{1}{2}\right)^t \frac{1}{t^2} x_1^2 \left(\sum_{i=2}^{t} x_i\right)^2 = \frac{1}{2t^2}\cdot V, \tag{21}$$

where V is the variance of a sum of $t-1$ stochastic variables, independently and uniformly distributed on ± 1. Hence $V = t - 1$, and the variance follows as stated. □

Proposition 4. *Let X_i for $i = 1, 2, \dots, t$ be uniformly and independently distributed over the alphabet $\{\pm 1\}$, and*

$$S = X_1 \cdot a_\theta^{\text{MMX}}(X_1, X_2, \dots, X_t).$$

Then

$$E(S) = \frac{1}{2^{t-1}}\left[1 - \sum_{i=1}^{(1-\theta)t/2} \binom{t}{i}\frac{t-2i}{t} + \frac{1}{2t}\cdot \sum_{i=(1-\theta)t/2}^{(1+\theta)t/2} \binom{t}{i}\right] \tag{22}$$

Note that if $\theta = 0$, then the last term is zero, and $E(S) \ll 0$. If $\theta = 1$, then the second term is zero, and $E(S) \gg 0$. Furthermore, $E(S)$ is monotonically increasing in θ, so $E(S)$ can be tuned by changing θ.

Proof. Let C denote the number indices i such that $X_i = +1$. Obviously C is binomially distributed with t trials and trial probability 0.5. Write $\Delta = X^{\mathrm{avg}} - X^{\mathrm{mid}}$. We get

$$\Delta = \begin{cases} 0, & \text{when } C = 0 \text{ or } C = t, \\ \frac{2C-t}{t}, & \text{otherwise.} \end{cases}$$

To analyse S, we distinguish between three cases.

1. $C = 0$ or $C = t$.
2. $C = i$ or $C = t - i$, and also $0 < i < \dfrac{(1-\theta)t}{2}$.
3. other values of C, i.e. close to $t/2$.

Case 1 corresponds to columns where all pirates have the same symbol. In this case $S = 1$, and

$$E(S|\text{Case } 1) = 1.$$

In Case 2, the MMX attack outputs the minority choice. If X_1 is part of the minority, then $S = 1$, otherwise $S = -1$. The probability that X_1 is part of the minority is i/t. Hence, we get

$$E(S|\text{Case } 2) = -\frac{t-i}{t} + \frac{i}{t} = -\frac{t-2i}{t}.$$

In Case 3, the MMX attack returns the average, and $S = X_1 X^{\mathrm{avg}}$. There is a useful symmetry in Case 3. If $C = i$ is a Case 3 event, then so is $C = t - i$. It follows, that $(X_1, X_2, \ldots, X_t) = \mathbf{x}$ is a Case 3 event, then so is $(X_1, X_2, \ldots, X_t) = -\mathbf{x}$. Furthermore, $\mathbf{x}$ and $-\mathbf{x}$ give the same score,

$$X_1 X^{\mathrm{avg}} = (-X_1)(-X)^{\mathrm{avg}}.$$

Thus, we can use an argument similar to the proof of Lemma 3. The score is given as

$$S = \frac{1}{t} \sum_{i=1}^{t} X_i X_1 = \frac{1}{t} + \frac{X_1}{t} \sum_{i=2}^{t} X_i,$$

and the expected value is

$$E(S|\text{Case } 3) = \frac{1}{t} + \sum_{x=\pm 1} \mathsf{P}(X_1 = x) \cdot x \cdot \frac{1}{t} \sum_{i=2}^{t} E(X_i) = \frac{1}{t}.$$

The expected value of the score is the weighted sum of the conditional expected values

$$E(S) = \sum_{i=1,2,3} \mathsf{P}(\text{Case } i) E(S|\text{Case } i), \tag{23}$$

and the proposition follows. □

Proposition 5. *Let X_i for $i = 1, 2, \ldots, t$ be uniformly and independently distributed over the alphabet $\{\pm 1\}$, Let*

$$S = X_1 \cdot a^{\mathrm{U}}_{\alpha}(X_1, X_2, \ldots, X_t).$$

Then

$$E(S) = \frac{1}{2^{t-1}} \tag{24}$$

Proof. Note that with probability 0.5^{t-1}, all the X_i are equal. In this case $X^{\mathrm{U}} = \pm 1$ and $X_i \cdot X^{\mathrm{U}} = 1$ for any i. In all remaining cases, the pirates can see both $+1$ and -1 and X^{U} is uniformly distributed on the interval $\pm\alpha$. Thus the expected value $E(S|\neg X_1 = X_2 = \ldots = X_t) = 0$. The lemma follows. □

3.4 Discussion

The study of the expected value of the decoding score in the presence of attacks explains why the MMX attack is so effective. The expected score can be made negative. Averaging in contrast, is not effective. The expected score $E(S) = 1/t$ declines only linearly in the collusion size t. The minimum and maximum attacks, as well as the uniform attack. are effective if the collusion is large, since the expected score $E(S) = 2^{1-t}$ declines exponentially. There are other decoding techniques which make spread spectrum more robust against minimum and maximum, but these techniques are not effective against the uniform attack [Sch08a].

4 Modulation and Obfuscation

To our knowledge, there is only one technique which claims to thwart both the MMX and uniform attacks, namely the obfuscation described in [Kur16, KS15]. Previous works have applied obfuscation as one of several advanced features in a complex system, and evaluated the complete system only. Here, we will introduce obfuscation as a generic technique and analyse its merits theoretically.

4.1 Obfuscation and modulation

Obfuscation, as introduced in [KS15], is a new layer between the host and watermarking layers. This is shown in Figure 2. The obfuscation key k_3 is used to generate a pseudo-random sequence $\mathbf{s}$ over $\{\pm 1\}$. The signal extracted by the host layer is first multiplied (element-wise) by $\mathbf{s}$ and subsequently subject to a DCT transform. The DCT transform is linear, and can be written as multiplication by the matrix T in the figure. Element-wise multiplication by $\mathbf{s}$ is equivalent to multiplication by the diagonal matrix S, with $\mathbf{s}$ on the main diagonal. Clearly $S^{-1} = S$.

In the original work, the host signal $\mathbf{x}$ was a matrix, and the DCT transform was two-dimensional. This does not affect the analysis. The 2D DCT transform can also be

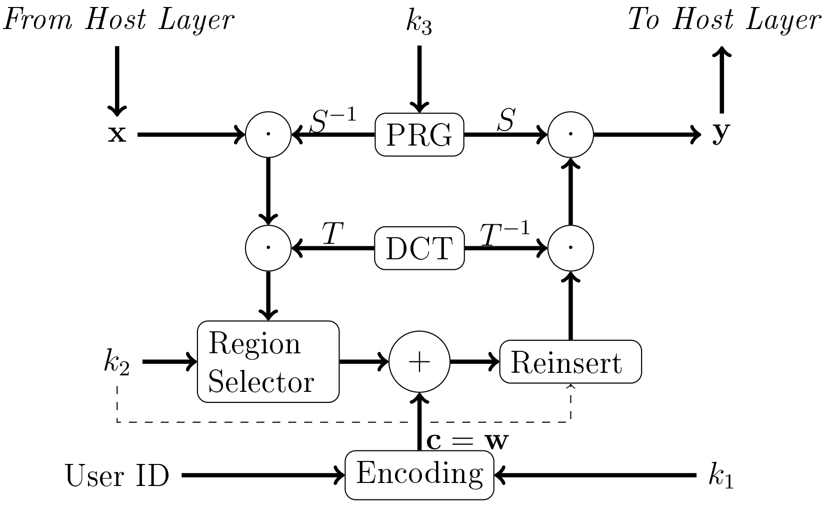

Figure 2: Additive spread spectrum fingerprinting with obfuscation.

written as $\mathbf{x}T$ for a vector $\mathbf{x}$ and some matrix T, although not the same matrix T as we would use for a 1D transform. In this paper, we will view the host signal $\mathbf{x}$ is a vector, and T can really be thought of as any invertible, publicly known matrix.

Disregarding the region selector just for a moment, we can write the obfuscation and watermarking layers combined as

$$\mathbf{y} = (\mathbf{x} \cdot S \cdot T + \mathbf{w}) \cdot T^{-1} \cdot S^{-1}, \tag{25}$$

where $\mathbf{x}$ is the input from the host layer, $\mathbf{w}$ is the input from the coding layer, and $\mathbf{y}$ is the output to the host layer. Because the transforms are linear, the equation can be rewritten as

$$\begin{aligned} \mathbf{y} &= \mathbf{x} \cdot S \cdot T \cdot T^{-1} \cdot S^{-1} + \mathbf{w} \cdot T^{-1} \cdot S^{-1} \\ &= \mathbf{x} + \mathbf{w} \cdot T^{-1} \cdot S^{-1}. \end{aligned} \tag{26}$$

In other words, the obfuscation layer can equivalently be implemented between the fingerprinting and watermarking layers. In this case, we would prefer to let $\mathbf{c}$ denote the codeword from the fingerprinting layer, and let $\mathbf{w} = \mathbf{c}T^{-1}S^{-1}$ denote the watermark to be embedded. Given a received watermark $\mathbf{w}'$, the receiver would calculate $\mathbf{c}' = \mathbf{w}'ST$ in the watermarking layer, and pass the received word $\mathbf{c}'$ for decoding in the coding layer.

The region selector, which we disregarded in the discussion, can also be written as matrix multiplication, using a permutation matrix.

Definition 5. *An $m \times n$ permutation matrix is an $m \times n$ matrix over $\{0, 1\}$ with $m \leq n$, where each column has at most one 1-entry, and each row has exactly one 1-entry.*

Note that a permutation matrix R is semi-orthogonal, in the sense that $RR^T = I$. In Figure 2, the region selector multiplies the host by R^T for some permutation matrix R. The reinsertion is also trivial, but not as neat. Multiplying by R leaves a number of zero entries, where the host samples which were not used for watermarking must be reinserted.

It is easier and more transparent to implement the region selector in the coding layer. In Equation (26) we can add a permutation matrix R to pad the watermark $\mathbf{w}$ with zeros corresponding to each unused sample in the host signal. At the receiver, R^T removes the unused samples and returns the (hybrid) fingerprint. Thus

$$\mathbf{y} = \mathbf{x} + \mathbf{w} \cdot R \cdot T^{-1} \cdot S^{-1}. \tag{27}$$

Applying the permutation R before the DCT transform T^{-1} means that we can use a relatively short codeword $\mathbf{w}$, and have the energy spread across a longer host signal.

Another approach to prevent attacks on the individual co-ordinate positions is modulation [KS15], which is essentially obfuscation as in Equation (26) with a slightly different interpretation. If $S{\cdot}T$ is an orthogonal matrix, which is the case in many implementation of DCT, we get

$$T^{-1}S^{-1} = (ST)^{-1} = (ST)^{\mathrm{T}}. \tag{28}$$

We can view $(ST)^{-1}$ as the codebook of spread spectrum sequences. When the encoder calculates $\mathbf{w}(ST)^{-1}$, it encodes each bit of $\mathbf{w}$ as an SS sequence, and these sequences are added together and embedded as one watermark. When the receiver applies the inverse transform $ST = ((ST)^{-1})^{\mathrm{T}}$, it is equivalent to correlation decoding.

Obfuscation and modulation as defined in (26) or (28) can be viewed either as part of the coding layer or as additional layers between the coding and watermarking layers. We make the following definition to facilitate comparison of coding layers with and without obfuscation.

Definition 6. *We define a fingerprinting scheme to be a pair (C, d) of a fingerprinting code C and a decoding score $d : \mathbb{R}^n \times C \to \mathbb{R}$. For any fingerprinting scheme, and any semi-orthogonal matrix U, we define the obfuscated fingerprinting scheme (C_U, d_U) where*

$$C_U = \{\mathbf{c}U | \mathbf{c} \in C\}, \tag{29}$$

$$d_U(\mathbf{x}, \mathbf{c}U) = d(\mathbf{x}U^T, \mathbf{c}). \tag{30}$$

The matrix U is called the obfuscation matrix.

Our previous construction [KS15] uses two layers of obfuscation, or, as we phrased it then, obfuscation in addition to modulation. Effectively this gives the following embedding of a codeword $\mathbf{w}$:

$$\mathbf{y} = \mathbf{x} + \mathbf{w} \cdot R_1 \cdot T_1^{-1} \cdot S_1 \cdot R_2 \cdot T_1^{-1} \cdot S_2, \tag{31}$$

where the S_i are two distinct pseudo-random signature matrices, and R_2 is a pseudo-random permutation which also increases the word length by padding with zeroes. Each of these three pseudo-random matrices are determined by a distinct secret key. The R_1

matrix only pads the codeword with zeroes at the end. The first DCT transform T_1 is 1D while T_2 is 2D. Clearly, this is equivalent to one layer of obfuscation using

$$U = R_1 \cdot T_1^{-1} \cdot S_1 \cdot R_2 \cdot T_1^{-1} \cdot S_2$$

as the obfuscation matrix.

4.2 Is obfuscation secure?

We have tried to answer this question analytically as well as empirically. To some extent the available results are contradictory, and they leave several open questions.

The main idea of obfuscation is to change the basis in which the fingerprint is represented. The MMX attack is designed to maximise the 'attack noise' in each individual mark, which correspond directly to a co-ordinate position in the fingerprint (codeword). When the basis is changed as in (31) or (26), this correspondence is broken, and the attack noise in one mark, is spread across all the co-ordinate positions of the fingerprint.

Let us first consider the correlation decoder in an obfuscated scheme.

Proposition 6. *Let C be a fingerprinting code, and let d be the correlation decoder. For any semi-orthogonal matrix U, the obfuscated fingerprinting scheme (C_U, d_U) is equivalent to (C_U, d).*

Proof. The correlation decoder calculates $d(\mathbf{r}, \mathbf{c}) = \mathbf{r}\cdot\mathbf{c}$, and the corresponding decoder in the obfuscated scheme calculates $d_U(\mathbf{r}, \mathbf{c}U) = \mathbf{r}U^T \cdot \mathbf{c}$. Applying the correlation decoder to a codeword $\mathbf{c}U \in C_U$ we have

$$d(\mathbf{r}, \mathbf{c}U) = \mathbf{r} \cdot \mathbf{c}U = \mathbf{r}(\mathbf{c}U)^T = \mathbf{r}U^T\mathbf{c}^T = d_U(\mathbf{r}, \mathbf{c}U),$$

as required. □

The implication of this proposition is that obfuscation of any fingerprinting scheme with correlation decoding changes the codebook only, and not the decoding algorithm. Most fingerprinting codes are random, and obfuscation then changes the probability distribution. Effectively, each symbol in the obfuscated code is a sum of random symbols, and in many cases the central limit theorem will apply, making the the obfuscated code a random code with Gaussian distribution. The MMX attack has previously proved effective against Gaussian fingerprints with correlation decoding [Sch08a].

A study of obfuscation of other fingerprinting codes is beyond the scope of this paper. Suffice it to note that an obfuscated Nuida code has proved effective in simulations [KS15]. Thus the change of basis evidently has some merit. Both single and double obfuscation (Equations (27) and (31)) resisted the Uniform and MMX attacks applied to the signal $\mathbf{y}$. However, single obfuscation was circumvented by applying the attacks to a DCT transform of $\mathbf{y}$, i.e. by using the attack

$$a' = \mathrm{DCT}^{-1} \circ a \circ \mathrm{DCT},$$

where a is the MMX or uniform attack.

5 Discussion and Conclusion

We have given a theoretical explanation to the efficacy of the MMX attack against spread spectrum fingerprinting, and provided a theoretical framework to analyse the obfuscation technique which we have previously proposed [KS15].

Obfuscation of the Nuida code appears to prevent known attacks, if the obfuscation transform U has a structure which cannot be circumvented without knowledge of the secret key. It is still an open question if the current construction with two layers, each using a secret permutation, a DCT transform, and a secret signature matrix, will suffice.

Obfuscation is equivalent to modulation by spread spectrum sequences, and this can potentially lead to a practical implementation of the Marking Assumption with Random Errors (as suggested by Guth and Pfitzmann). Given the existing theory on spread spectrum, and on collusion-secure codes, it may be possible to get practical fingerprinting systems with formal proofs and bounded error probabilities. It is worth investigating.

References

[AT09] Ehsan Amiri and Gábor Tardos. High rate fingerprinting codes and the fingerprinting capacity. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 336–345, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

[BS95] Dan Boneh and James Shaw. Collusion-secure fingerprinting for digital data. In *Advances in Cryptology - CRYPTO'95*, volume 963 of *Springer Lecture Notes in Computer Science*, pages 452–465, 1995.

[BS98] Dan Boneh and James Shaw. Collusion-secure fingerprinting for digital data. *IEEE Trans. Inform. Theory*, 44(5):1897–1905, 1998. Presented in part at CRYPTO'95.

[CKLS97] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamson. Secure spread spectrum watermarking for multimedia. *IEEE Trans. Image Processing*, 6(12):1673–1687, 1997.

[CS04] Gérard D. Cohen and Hans Georg Schaathun. Upper bounds on separating codes. *IEEE Trans. Inform. Theory*, 50(6):1291–1295, 2004.

[CXFF09] A Charpentier, F Xie, C Fontaine, and T Furon. Expectation maximization decoding of Tardos probabilistic fingerprinting code (proceedings paper). SPIE, 2009.

[GP00] Hans-Jürgen Guth and Birgit Pfitzmann. Error- and collusion-secure fingerprinting for digital data. In *Information Hiding '99, Proceedings*, volume 1768 of *Springer Lecture Notes in Computer Science*, pages 134–145. Springer-Verlag, 2000.

[HW06] S. He and M. Wu. Joint coding and embedding techniques for multimedia fingerprinting. 1:231–248, June 2006.

[ISO13] Sarah Ibrahimi, Boris Skoric, and Jan-Jaap Oosterwijk. Riding the saddle point: asymptotics of the capacity-achieving simple decoder for bias-based traitor tracing. *IACR Cryptology ePrint Archive*, 2013:809, 2013.

[KS15] Minoru Kuribayashi and Hans Georg Schaathun. Image fingerprinting system based on collusion secure code and watermarking method. In *ICIP, Quebec*, September 2015. Accepted for publication.

[Kur12] Minoru Kuribayashi. Interference removal operation for spread spectrum fingerprinting scheme. *Information Forensics and Security, IEEE Transactions on*, 7(2):403–417, 2012.

[Kur14] Minoru Kuribayashi. Simplified MAP detector for binary fingerprinting code embedded by spread spectrum watermarking scheme. *Information Forensics and Security, IEEE Transactions on*, 9(4):610–623, April 2014.

[Kur16] Minoru Kuribayashi. Simple countermeasure to non-linear collusion attacks targeted for spread-spectrum fingerprinting scheme. *IEICE TRANSACTIONS on Information and Systems*, 99(1):50–59, 2016.

[NFH$^+$09] Koji Nuida, Satoshi Fujitsu, Manabu Hagiwara, Takashi Kitagawa, Hajime Watanabe, Kazuto Ogawa, and Hideki Imai. An improvement of discrete tardos fingerprinting codes. *Designs, Codes and Cryptography*, 52(3):339–362, 2009.

[OSD13] Jan-Jaap Oosterwijk, Boris Skoric, and Jeroen Doumen. A capacity-achieving simple decoder for bias-based traitor tracing schemes. Cryptology ePrint Archive, Report 2013/389, 2013. `http://eprint.iacr.org/`.

[Sch07] Hans Georg Schaathun. Attack analysis for He&Wu's joint watermarking/fingerprinting scheme. In *The 6th International Workshop on Digital Watermarking*, volume 3304 of *Springer Lecture Notes in Computer Science*, 2007. Canton (Guangzhou) China.

[Sch08a] Hans Georg Schaathun. Novel attacks on spread-spectrum fingerprinting. *EURASIP J. Information Security*, 2008, 2008.

[Sch08b] Hans Georg Schaathun. On error-correcting fingerprinting codes for use with watermarking. *Multimedia Systems*, 13(5–6):331–344, February 2008.

[ŠKC08] Boris Škorić, Stefan Katzenbeisser, and Mehmet U. Celik. Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes. *Designs, Codes and Cryptography*, 46(2):137–166, 2008.

[SKSC11] Boris Skoric, Stefan Katzenbeisser, Hans Georg Schaathun, and Mehmet Utku Celik. Tardos fingerprinting codes in the combined digit model. *IEEE Transactions on Information Forensics and Security*, 6(3):906–919, September 2011.

[SSW01] Jessica N. Staddon, Douglas R. Stinson, and Ruizhong Wei. Combinatorial properties of frameproof and traceability codes. *IEEE Trans. Inform. Theory*, 47(3):1042–1049, 2001.

[ŠVCT06] B. Škorić, T.U. Vladimirova, M. Celik, and J.C. Talstra. Tardos fingerprinting is better than we thought. Technical report, 2006. arXiv:cs.CR/0607131 v1.

[Tar05] Gábor Tardos. Optimal probabilistic fingerprint codes. *Journal of the ACM*, 2005. `http://www.renyi.hu/~tardos/fingerprint.ps`. To appear. In part at STOC'03.

[Wag83] Neal R. Wagner. Fingerprinting. In *Proceedings of the 1983 Symposium on Security and Privacy*, pages 18–22, 1983.

[XFF08] Fuchun Xie, Teddy Furon, and Caroline Fontaine. On-off keying modulation and tardos fingerprinting. In *Proceedings of the 10th ACM workshop on Multimedia and security*, pages 101–106. ACM, 2008.

[ZWWL05] Hong Zhao, Min Wu, Z. June Wang, and K. J. Ray Liu. Forensic analysis of nonlinear collusion attacks for multimedia fingerprinting. *IEEE Trans. Image Proc.*, 14(5):646–661, 2005.