Tatouage sûr et robuste appliqué au traçage de documents multimédia

Fuchun Xie

N° d'ordre : 4193 ANNÉE 2010

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Traitement du Signal et Télécommunication

Ecole doctorale MATISSE

présentée par

Fuchun XIE

préparée à l'INRIA Rennes - Bretagne Atlantique
Composante Universitaire : SPM

Intitulé de la thèse : Tatouage sûr et robuste appliqué au traçage de documents multimédia

Thèse soutenue à l'INRIA-Rennes le 23 Septembre 2010

devant le jury composé de :

Patrick BAS
CR CNRS École Centrale de Lille - LAGIS / Rapporteur

Andreas WESTFELD
Professeur HTW Dresden - Fakultät Informatik Mathematik / Rapporteur

Atilla BASKURT
Professeur INSA Lyon - LIRIS / Président

William PUECH
Professeur Université Montpellier II - LIRMM / Examinateur

Caroline FONTAINE
CR CNRS Lab-STICC-CID et Télécom Bretagne - ITI / Directrice de thèse

Teddy FURON
CR INRIA INRIA Rennes Bretagne Atlantique - TEMICS / Co-directeur de thèse

Robust and Secure Watermarking for Multimedia Traitor Tracing

Fuchun Xie

INRIA-Rennes Bretagne Atlantique Research Center

University of Rennes 1

A thesis for the degree of

Doctor of Philosophy

2010

Dedicate this thesis to my parents Qianshui XIE and Xuee YUAN,
and my wife Meiling LUO.

Acknowledgements

The making of this thesis would not have been possible without the support of a lot of helpful people. It is a pleasure for me to take the opportunity to thank them here.

First of all, I want to thank my advisor, Dr. Caroline FONTAINE, researcher of CNRS, for her generous support and priceless advice during my whole Ph.D study. With her support, I obtained the opportunity to do this thesis.

I would like to express my sincere gratitude to my co-supervisor, Dr. Teddy FURON, researcher of INRIA, for his constant guidance and encouragement during my research endeavors in the TEMICS team of INRIA-Rennes. He has played a significant role in both my professional and personal development. His knowledge, vision, patience, and endless pursuit of academic and professional excellence have influenced me with lifetime benefits. With his help, I developed the ability to conduct meaningful research. I deeply appreciate him for helping me reach this milestone in my life.

I would like to thank all the members of jury: Dr. Patrick BAS from École Centrale de Lille, Prof. Atilla BASKURT from INSA-Lyon, Prof. William PUECH from Université Montpellier II, and Prof. Andreas WESTFELD from Dresden Fakultät, for taking a lot of time to read this thesis, and providing lots of valuable advices to improve its quality.

I am grateful to Dr. Christine GUILLEMOT, the head of the TEMICS team of INRIA-Rennes, for accepting me to work in her research group, from which I obtained precious knowledge and invaluable help.

I would like to take this chance to thank Dr. Patrick BAS, researcher of CNRS working at École central de Lille, for his constructive discussions and suggestions; and thank Prof. Andreas WESTFELD of Dresden Fakultät in Germany, for his useful discussions.

I own my gratitude to my colleagues in the TEMICS research team, in particular Ana CHARPENTIER, Zhenzhong CHEN, Thomas COLLEU, Olivier CRAVE, Mathieu DESOUBEAUX, Angélique DRÉMEAU, Cédric HERZET, Denis KUBASOV, Simon MALINOWSKI, Fethi SMACH, Jonathan TAQUET, Velotiaray TOTO-ZARASOA, Mehmet TÜRKEN, and Joaquin ZEPEDA for creating a great ambiance in which to do research. Special thanks to Laurent GUILLO for always helping me to manage my computer, to our assistant Huguette BECHU for her help in daily life, and to my officemates, Zhenzhong CHEN, Mathieu DESOUBEAUX, and Fethi SMACH for the time spent together.

I am happy to thank my friends in IRISA: Xiao BAI, Zhan GAO, Xingwu LIU, Guang TAN, Huaibin TANG, Zhaoguang WANG and Fen ZHOU for their friendship and help.

I am grateful to Meiling LUO, my wife, for her love and encouragement during the past years. Finally, I give my heartfelt gratitude to my parents, my role model and the two most important persons in my life. Without their love, unconditional support and countless sacrifices, I could never accomplish so much and reach this milestone in my life. I dedicate this thesis to them.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

In the last decades, the rapid development of digital information technology has significantly facilitated our daily works and lives. The popularity of smart phone, digital camera, digital camcorder, and personal computer, has inspired us to create, enjoy and share multimedia content. Furthermore, the technological advancements in telecommunications and the popularization of broadband networks have made multimedia distribution and sharing over the Internet very easy and popular.

However, this simultaneously brings some serious problems such as unlimited duplication, arbitrarily modification, unauthorized upload, illegally redistribution etc. Because pirates can easily modify multimedia content by using some simple attacks, while maintaining a high quality, and then redistribute it without authorization, this inevitably infringes on the intellectual property of the multimedia holders. One example is that certain illicit users upload some famous films without authorization via the peer-to-peer streaming video network softwares, such as PPLive [1] and PPStream [2]. We can take another example of commercial market here, according to an investigation report by the Motion Picture Association of America (MPAA): worldwide losses are about $18.2 billion due to piracy in 2008, of which $322 million in France, $565 million in China, and $1.3 billion in United States etc [3]. Therefore, effective solutions to protect copyright hold-

ers are necessary, and although lots of works have already been done in the last decades, some challenging open problems are remaining.

Currently, Digital Rights Management (DRM) systems [4] are used to prevent multimedia content from unauthorized modification and redistribution, and thereby safeguard the intellectual property of right holders. Generally speaking, they include two basic parts for protecting multimedia content: 1) The first one is the access path control: the multimedia content is encrypted so that only the authorized users can access it; 2) The second one is multimedia forensics, a watermark is hidden in the protected multimedia by the holder, and then multimedia forensics is used to detect whether the multimedia content is illegal tampered and redistributed by users who have the right to access it. DRM system can be used for movies, television, audio CDs, Internet music, DVDs, computer games, computer software, E-books etc. Now it is widely applied by some famous companies such as Microsoft, Apple Inc., Adobe Systems Inc., Sony, America Online, BBC etc.

Digital watermarking is a branch of multimedia forensics, it can provide the posterior protection when the multimedia content is decrypted by the authorized users. It imperceptibly alters the original work (host signal) by hiding the identification information (watermark). Originally it has been developed for copyright protection and authentication. In these cases, the embedded watermark corresponds to the copyright owner's identity. The advantage of digital watermarking is that the embedded watermark does not spoil the usage of multimedia content, but can still be detected if needed.

Multimedia fingerprinting for traitor tracing is an application of digital watermarking, where the copyright holder hides a unique watermark in each user's copy. The purpose is to trace back the identities of the pirates (colluders) when an illicit copy is found, and then force them to be responsible for their actions. Here we can take the VOD (Video On Demand) application as an example (see Figure 1.1): for each video, the server divides it into lots of blocks, and then watermarks each block into q versions (here we suppose $q = 4$); finally, the server distributes a unique sequence to each user by swapping these different fingerprinted blocks. However, in order to avoid being accused, the pirates will try to employ a variety of attacks to remove the watermark. For example, in the

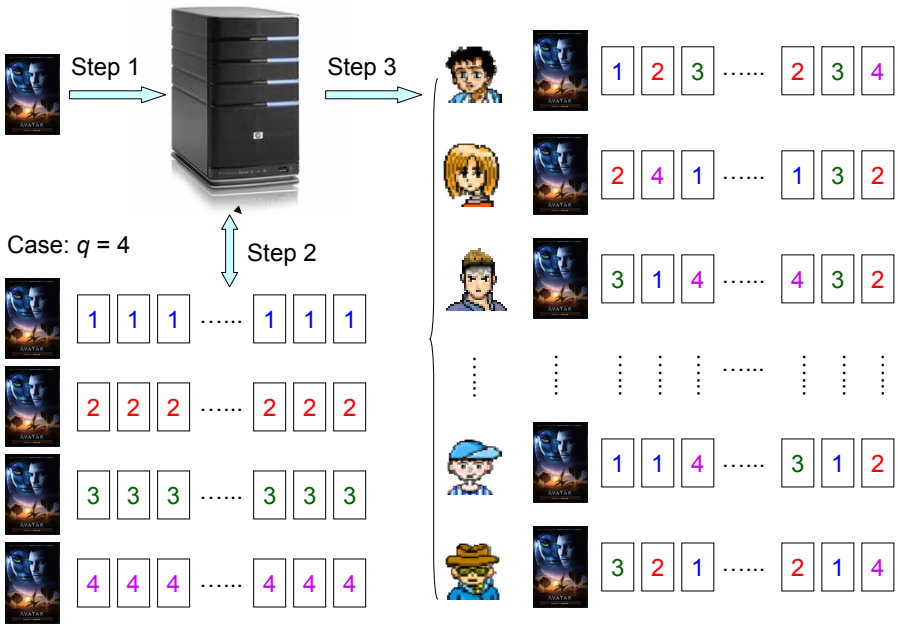

Figure 1.1: Multimedia fingerprinting example: VOD.

video fingerprinting scenario, if the sequences are watermarked with few secret keys according to the fingerprinting code, it is possible for a lonely pirate to extract enough information to estimate the secret keys, and thereby remove the watermark while preserving an excellent perceptual quality [5]. Another powerful attack is the collusion attacks, where several colluders mix several different copies of the same content into one hybrid version, in order to attenuate or even remove all the original fingerprints. If the multimedia fingerprinting system is designed improperly, it might fail to detect any fingerprints, although the attack is affected with only a few different copies. Consequently, these additional attacks bring new challenges to the multimedia fingerprinting system design, and especially relative to digital watermarking. In general, a good multimedia fingerprinting system must meet three conditions: firstly, the hidden fingerprint should be perceptually invisible; secondly, the system must be robust to some common signal processing operations like compression, denoising, scaling, filtering, rotation, etc; thirdly, it should be secure against intentionally attacks such as collusion and the security attack for the scenario where a number of fingerprinted contents

are available. Unfortunately, since these aspects are conflicting with each other. Because the imperceptibility means a upper limit for the capacity of the hidden information, but robustness and security performance require a lower limit for this capacity. Furthermore, robustness performance is relative to some common operations, while security aims at certain specific attacks, sometime we have to make a compromise between them. Therefore, these reasons makes a multimedia fingerprinting system design becoming a very challenging work.

1.2 Past Approaches and Thesis Objective

Since the first paper written by Wagner on the fingerprinting appeared in 1983 [6], lots of works on this topic have been produced. Their focus is mainly concerning about the potential attacks, the fingerprinting code design, or the combination between the fingerprinting code and some classical watermarking technique. Of course, the studies might be carried out from the theoretical or experimental point of views. In this section, we will summarize some significant progresses in this field.

Firstly, we will talk about the progress on the study of attacks against multimedia fingerprinting. Because there are practical tools to verify whether a designed multimedia fingerprinting system is successful or not, and a thorough understanding of these attacks will help us to create an effective multimedia fingerprinting system. The attacks for the multimedia fingerprinting system include all the attacks against the watermarking system, besides, they include some specific fusion attacks because of the difference among the fingerprinted copies. An early systematic study on these fusion attacks could be found in [7] [8]. Later, Schaathun introduced some novel fusion attacks [9], his attacks aim the spread-spectrum watermarking based multimedia fingerprinting. We will give a more detail description of these attacks in Section 2.2.

According to the literature, there are several different approaches to design a multimedia fingerprinting scheme. First of all, there is the approach from the cryptography and coding community. Early works can be traced back to [6] and [10]. Later, Boneh and Shaw introduced the now well-known concept of Marking Assumption, and constructed a two-level binary code [11]. This binary

code was later improved by Yacobi for the application of the multimedia signals [12]. Some recent works [13] [14] extended Boneh and Shaw's framework and considered the construction of codes with traceability, such as the identifiable parent property (IPP) code and the traceability (TA) code. Usually, we can split the codes into two parts: weak traceability versus strong traceability. Weak traceability codes should satisfy the basic property that a collusion may frame an innocent user or an innocent group of users, but with a very small probability. But for the strong traceability codes, it should satisfy another property: the success of identifying at least one colluder, and never frame an innocent user. Actually, strong traceability is not reachable in practice [15] [16]. Hence, all the below mentioned approaches are in the weak traceability category.

It is worth mentioning that another significant approach of the multimedia fingerprinting from the signal processing point of view, which was made by the scientists of the University of Maryland in USA. Wang et al. used the mutually independent (or orthogonal) basis signals as fingerprints to identify pirates [17]. Due to the computational burden of this independent fingerprinting signal, Trappe et al. employed the code modulation combining with the independent fingerprinting to construct the redundant fingerprint signals [18]. Furthermore, in order to make their fingerprinting system design more effective, some accessorial strategies were introduced [19] [20] [21]. Finally, some applications of the multimedia fingerprinting were explored [22] [23].

Another major breakthrough is the Tardos probabilistic fingerprinting code introduced from the statistics viewpoint. In 2003, Tardos proposed a fingerprinting code whose code length has a theoretically minimum order [24], which is known to be optimal with respect to the codelength and the collusion resistant capacity. Afterwards, Skoric et al. extended the original Tardos fingerprinting code generation from binary to q-ary [25]. Furthermore, there are other improvements which were introduced to reduce the codelength of original Tardos fingerprinting code [26] [27] [28] [29], or to optimize memory usage [30], or to improve the accusation function [31] [32]. We will give a more detail description for these past approaches in Chapter 2.

The objective of this thesis is completely different from the above approaches. Our goal is to design a secure and robust multimedia fingerprinting system. This

system should satisfy the following conditions: 1) robust for the unintentional attacks such as some common signal processing operations; 2) secure against intentional security attacks, whose aim is to explore the knowledge about the secret keys of the system by observing lots of fingerprinted content; 3) resistant to the fusion attacks, whereby a group of different copies are collected together to remove the fingerprints. This thesis firstly addresses the robustness issues of one watermarking scheme, which will be used for the multimedia fingerprinting system; based on this achievement, we try to improve the security performance of the watermarking scheme; finally we combine it with a Tardos fingerprinting code to construct a complete multimedia fingerprinting scheme.

1.3 Thesis Organization and Contribution

This dissertation is organized as follows. Chapter 2 starts with a general framework of multimedia fingerprinting and reviews potential attacks. Then, we briefly review the design of multimedia fingerprinting from the cryptography and coding viewpoint, and then discuss the approach from the signal processing viewpoint, which was introduced by the group of the university of Maryland, finally, we talk about the recent developments on the Tardos probabilistic fingerprinting code from the statistics point of view.

Based on the discussion on the state of the art in Chapter 2 and our analysis, Chapter 3 is dedicated to design an adequate watermarking scheme, which will serve for our robust and secure multimedia fingerprinting system. Since there is no watermarking scheme which is very robust as well as secure so far, we begin our study with a very robust watermarking scheme named “Broken Arrows” [33], which is state of the art. It has been specifically designed for the second contest of Break Our Watermarking System (BOWS-2) [34]; therefore it has been widely tested and showed a good robustness capacity. However, A. Westfeld found the worst attack in the first episode of the contest, which inevitably threaten the technique. In Chapter 3, we propose some solutions to plug this flaw. Our results show that the proposed solutions further improve the robustness of the watermarking technique. The obtained result in this Chapter has been published in the paper [32].

With the achievement obtained in Chapter 3, Chapter 4 focuses on improving the security performance of this scheme. Actually, security is a very different concept from robustness in watermarking; it relates to the intentional attacks whose purpose is to discover the secret keys used during embedding. For the studied watermarking scheme, two key estimation attacks have been proposed in [5]. In Chapter 4, we propose some solutions to block these security flaws, the simulation results show that the threats of these two security attacks are perfectly eliminated. However, our study shows that: although the improved watermarking technique is much more secure than before, it is not perfect, a slight security vulnerability still exist. In consequence, we design a novel robust and secure watermarking scheme via a maximization function under constraints, the watermarking technique seeks at maximizing a given robustness criterion under the perceptual constraint and the security constraint. As far as we know, this is the first time that security criteria is taken into the watermarking algorithm. The experimental results reveal that this new scheme provides an excellent security level, but its robustness is much weaker than the above mentioned watermarking techniques, this is the cost to pay. The results have been included in two conference papers [35] and [36].

In Chapter 5, a complete multimedia fingerprinting scheme is proposed, which includes a watermarking technique combined with the symmetric Tardos fingerprinting code. We test all the three watermarking techniques introduced in the last two chapters. Furthermore, we propose two different accusation functions for the fingerprinting detection scheme. The simulations show that, if the pirates employ the fusion attacks by collusion, it would rather help the accusation process than puzzle it. Up to now, the watermarking layer attacks are prevented by the proposed watermarking schemes, the fingerprinting layer attacks are averted by the Tardos fingerprinting code, and the fusion collusion attacks are avoided thanks to the efficient accusation functions. Therefore, we proposed a concrete and operational multimedia fingerprinting system. Part of the results has been published in the paper [37] and the paper [38].

Finally, Chapter 6 summarizes the main conclusions of this dissertation, and discusses the main challenges and important topics in the future works.

Chapter 2

Framework and State of The Art

In this chapter, we present the general framework and the background of the multimedia fingerprinting. Especially, we study the various attacks for multimedia fingerprinting, which have appeared in the literature. Finally, we talk about three significant approaches to design the multimedia fingerprinting system: the first approach is the cryptography and coding approach based on the *Marking Assumption*, the second approach is the signal processing approach of the group of Maryland university, the third approach is the statistical approach around the Tardos probabilistic fingerprinting code. The content of this section is mainly to provide the background knowledge for the remaining chapters.

2.1 General Multimedia Fingerprinting Framework

A completely multimedia fingerprinting framework is very similar to a communication chain, which has a transmitter, a channel and a receiver. Both the transmitter and the receiver correspond to the system: the fingerprint embedding for the transmitter, and the fingerprint extraction and pirates accusation for the receiver; while the collusion attacks are modeled by a transmission channel.

In general, this whole framework can also be divided into three parts: fingerprinting layer, watermarking layer and attacks channel. In detail, they can be divided into six major steps: fingerprint encoder, fingerprint embedding(watermarking),

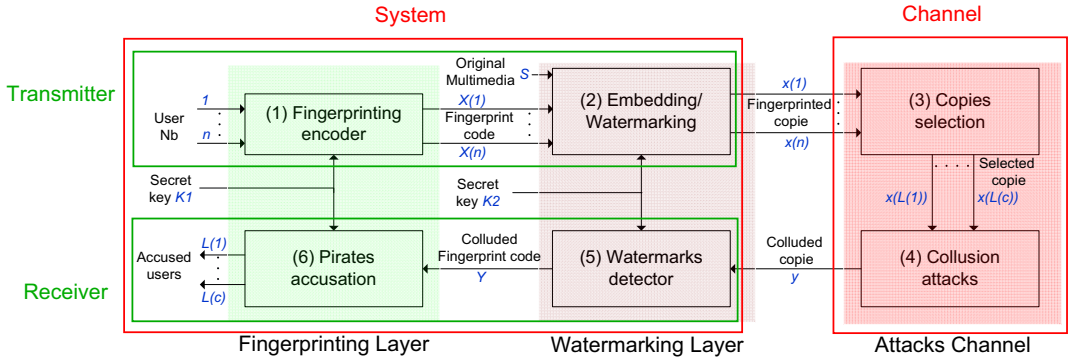

Figure 2.1: The multimedia fingerprinting design schema.

copies selection, collusion attacks, watermarks detector and pirates accusation, as depicted in Figure 2.1. We describe the role of each layer in the following subsections.

2.1.1 Fingerprinting Layer

The fingerprinting layer is a layer directly related to the users. In this layer, the multimedia content holder associates each user's identity with a unique fingerprinting codeword, which will be used in the embedding layer. Of course, the encoding technology is critical, since fingerprint must be unique. Furthermore, under the *Marking Assumption* [11] (in which we assumed colluders can change symbols in the blocks which have different symbols, and remain the same for the blocks that have identical symbols), a collusion of these fingerprints cannot frame an innocent user, or an innocent group of users. Furthermore, in the fingerprint detection process, this layer should have some efficient methods to correctly identify the pirated code. Because the existence of a variety of attacks brings a lot of diversity, the accusation process is thus becoming very complicated. So this two aspects propose a higher level of requirements for the fingerprinting layer. Some practical fingerprinting techniques will be mentioned in the other sections of this chapter, such as the Identifiable Parent Property code and the Traceability code 2.3, the Error Correction Code (ECC) based fingerprinting code in subsection 2.4 and Tardos probabilistic fingerprinting code 2.5.

2.1.2 Watermarking Layer

After each user's identity has been associated with a unique codeword, the watermarking layer takes charge of embedding it into the host signal to generate each users copy. Firstly the host signal is split into blocks, and then each symbol of the codeword is hidden into one block to produce the watermarked block. This is not an easy work, because the symbols should be embedded imperceptibly, robustly, and securely. Imperceptibility can be obtained by using a human visual model, as well as imposing a constraint to the visual distortion brought by the embedding, so that the fingerprinted version is perceptually similar to the original signal. Robustness requires the embedded fingerprints to resist some common signal processing operations, such as denoising and compression. Security means the embedded fingerprints can withstand some malicious attacks, such as a secret keys estimation attack. However, these three aspects constraint each other, making the design of a practical watermarking system hard, as it is difficult to strike a balance between them.

On the detection side, if the holder finds a pirated content, he will apply the watermarks decoder to retrieve the symbols, and then pass the results to the fingerprinting layer to identify the possible attackers. According to the availability of the host signal at the detection side, the watermarks detection process can be blind and non-blind. In non-blind detection, the host content is available to the watermark decoder, and it will be removed to avoid the interference during the detection process. Its advantage is that the decoding results have a far smaller variance and are thus more accurate. However, some people think that it is too complex to find back the host signal, and then perfectly synchronize the pirated version with the host to finally erase its contribution. It is true that people working with blind detector avoid these troubles. So in this thesis we mainly focus on blind decoding.

2.1.3 Attacks Channel

When the fingerprinted contents are distributed to the authorized users, some of these users (so-called *colluders*) may redistribute the content to someone else in order to pursue some illegal business interests. To avoid being accused, they

will use their several different copies and then perform an attack to erase the fingerprints or frame any other innocent user. For example, they can exchange the blocks between the different copies to produce a hybrid version; or they can mix the different blocks of the different copies together to produce a new block; even if they have just one copy, they can also modify the blocks in using some common signal processing such as de-noising, compression or noise addition. Hence, during the multimedia fingerprinting system design, all these possible attacks must be taken into account. At least, this can ensure that the designed system will not be destroyed by the existing attacks. For this purpose, we will give a full study for a number of attacks in the next section.

2.2 Various Attacks

In this section, we study the possible strategies which can be applied in the attack layer. Generally speaking, according to their targets, the various attacks can be divided into three classes: pure watermarking attacks, pure fingerprinting code attacks and fusion attacks. We will detail them in the following subsections.

2.2.1 Pure Watermarking Attacks

Pure watermarking attacks is a class of attacks which focus on the watermarking technique of the multimedia fingerprinting system. The purpose of these attacks is to break down the multimedia fingerprinting system by attacking the watermarking technique. If the watermarking system is broken down, the symbols of the fingerprint will not be correctly detected, and thus the multimedia fingerprinting system is unable to trace the colluders. Of course, these attacks can be effected even if just one copy is available.

Pure watermarking attacks can be classified into two families. The first family consist of robustness attacks, which includes some common signal processing actions, such as digital-to-analog and analog-to-digital conversion, sampling, scaling, quantification, compression, format transform etc. If we take the Broken Arrows watermarking technique [33] as an example, there is an efficient robustness attack, namely Westfeld's de-noising attack [39], which was introduced by

A. Westfled during the first episode of the BOWS-2 contest [34]. Its core idea is based on the estimation of the amplitude of a wavelet coefficient as function of the coefficients in its neighborhood via a linear regression.

Another family consist of security attacks, whose purpose is to take advantage of several observations to design a dedicated attack whose goal is to remove the watermarks or fingerprints through precise processes. Such attacks can for example proceed through secret key retrieval, or use oracles to get accurate informations about the watermarking region [40]. In the multimedia fingerprinting application, a huge number of contents are marked with the same key. For instance, the watermarking technique embeds the fingerprinting code in a video block by block, and for a given user all these blocks are watermarked with a few number of secret keys, the number of secret keys is equal to the symbol size of the fingerprinting codeword. So this may greatly help the pirate to estimate the secret key and therefore remove the fingerprints. We have two watermarking security attack examples here [5], which are also mainly against Broken Arrow watermarking technique. The first one is Westfeld clustering attack, which was introduced by A. Westfeld in the third episode of BOWS-2 contest, it extracts the estimated watermarks by using Westfeld's de-noising attack, and classifies them into several bins, and finally gets the attacked image by subtracting the average value of all the estimated watermarks of the bin which this image is in. The second one is Bas subspace estimation attack, which is later proposed by P. Bas [5]. He uses the OPAST algorithm and the Independent Component Analysis (ICA) technique to estimate the secret subspace, and then pushes the watermarked content outside the detection region in using this estimated secret subspace.

Of course, the type of these attacks can be avoided, if the watermarking technique of the multimedia fingerprinting system is robust enough, and secure enough. However, a sufficiently robust and secure watermarking system design is still a great challenge so far, due to the trade-off between robustness and security.

2.2.2 Pure Fingerprinting Code Attacks

Pure fingerprinting code attacks are some attacks which focus on the fingerprinting code layer of the multimedia fingerprinting system, and they do not "touch" the watermarking layer. In other words, we assume that the splitting of the host signal into blocks is an open knowledge, because we suppose that there is no way to keep this process secret. Therefore, the colluders know the blocks, and they will replace some blocks by the blocks of other copies. However, they regard each watermarked block as an inseparable whole and do not modify them. One general model of these attacks for the i-th block can be considered as:

$$y_i^{(p)} = \sum_{\ell=1}^{c} \omega_{i\ell} \cdot y_{ij_\ell} \tag{2.1}$$

with $y_i^{(p)}$ is the pirated copy, c is the number of the colluders, $\{j_1, \ldots, j_c\}$ are the indices of the c colluders, y_{ij_ℓ} are the available blocks of colluders, and $\omega_{i\ell}$ being c weighing vectors such that one $\omega_{i\ell} = 1$, $\forall \ell \in \{1, ..., c\}$, and the others equal zero, so the weights are exclusive. Comparing the available different versions of the i-th block, the pirates can know how many different symbols are embedded in their blocks, and their frequency, and then choose a method to select one block. According to the different selection methods, we enumerate some of such attacks here: Uniform exchange attack, minority vote attack, majority vote attack.

1. Uniform exchange attack: the block put in the pirated copy is one block which is independently and randomly selected from the different available blocks.

2. Minority vote attack: the block put in the pirated copy is the less frequent block in the available copies.

3. Majority vote attack: the block put in the pirated copy is the most frequent block in the available copies.

Of course, we only enumerate several typical attacks here, much more attacks might be introduced. However, please note that the constant symbol strategy, where the pirates always select the block with a given symbol inside, is not relevant a priori in our multimedia fingerprinting. Fingerprinting codes invented by

cryptographers foresee this case because the content is modelized as a long string of symbols directly observable by the colluders. In multimedia scenario, the colluders do not know the watermarking secret key to decode symbols embedded in their copies.

Any way, the class of these attacks are managed by the fingerprinting code, because it exactly matches the scenario envisaged by the cryptographers: the so-called *Marking Assumption* [11], which will be detailed in Section 2.3.1.

2.2.3 Fusion Attacks

Another family of attacks is a number of most powerful attacks, called fusion attacks. These attacks span the watermarking layer and the fingerprinting layer. In these attacks, the colluders fuse their different copies into one pirated version to delude the accusation. These attacks are carried out under *two assumptions*: all the fingerprinted blocks are independents to each other; all the samples in one block are independent to each other. In this way, the process for the i-th block can be considered as a general model:

$$y_i^{(p)}(u,v) = \sum_{\ell=1}^{c} \gamma_{i\ell} \cdot y_{ij_\ell}(u,v) \tag{2.2}$$

where $y_i^{(p)}(u,v)$ is the sample of the pirated copy (i.e. the pixel value in spatial space, or the wavelet coefficient in the wavelet transform domain), $y_{ij_\ell}(u,v)$ ($\forall \ell \in \{1,...,c\}$) are the elements of the copies for collusion, $\gamma_{i\ell}$ being c weighing vectors such that $\sum_{\ell=1}^{c} \gamma_{i\ell} = 1$, and $\{j_1, \ldots, j_c\}$ are the indices of c colluders. We remind here this model is sample-wise, and not bloc-wise. It describes a lot of attacks, including (the definitions being given for all $(\ell, i) \in \{1,...,c\} \times \{1,...,m\}$, to simplify the expression):

1. Average: $\gamma_{i\ell} = 1/c$.
2. Minimum: $\gamma_{i\ell} = 1$ if $\ell = \arg(\min(y_{ij_\ell}(u,v)))$, else 0.
3. Maximum: $\gamma_{i\ell} = 1$ if $\ell = \arg(\max(y_{ij_\ell}(u,v)))$, else 0.
4. MinMax: $\gamma_{i\ell} = 1/2$ if $\ell \in \{\arg(\min(y_{ij_\ell}(u,v))), \arg(\max(y_{ij_\ell}(u,v))\}$, else 0.

5. Median: $\gamma_{i\ell} = 1$ if $\ell = \arg(\text{median}(y_{ij_\ell}(u,v)))$, else 0.

6. Modified Negative: $\gamma_{i\ell} = 1$ if $\ell \in \{\arg(\min(y_{ij_\ell}(u,v))), \arg(\max(y_{ij_\ell}(u,v))\}$, and $\gamma_{i\ell} = -1$ if $\ell = \arg(\text{median}(y_{ij_\ell}(u,v)))$, else 0.

7. Interleaving: $\gamma_{i\ell} = 1$ if ℓ is one index selected independently and uniformly at random from the set $\{1, ..., c\}$, else 0.

8. Uniform: $y_i^{(p)}(u,v) \in [\min(y_{ij_\ell}(u,v)), \max(y_{ij_\ell}(u,v))]$, which is a random value of this interval.

9. Randomized Negative: $\gamma_{i\ell} = 1$, if $\ell = \arg(\min(y_{ij_\ell}(u,v)))$, with probability p; or if $\ell = \arg(\max(y_{ij_\ell}(u,v)))$, with probability $1-p$; else 0.

10. Majority Extreme: $\gamma_{i\ell} = 1$ if $\ell = \arg(\min(y_{ij_\ell}(u,v)))$, when $\text{average}(y_{ij_\ell}(u,v)) < \text{median}(y_{ij_\ell}(u,v))$; or $\ell = \arg(\max(y_{ij_\ell}(u,v)))$, when $\text{average}(y_{ij_\ell}(u,v)) > \text{median}(y_{ij_\ell}(u,v))$; else 0.

11. Minority Extreme: $\gamma_{i\ell} = 1$, if $\ell = \arg(\min(y_{ij_\ell}(u,v)))$, when $\text{average}(y_{ij_\ell}(u,v)) > \text{median}(y_{ij_\ell}(u,v))$; or $\ell = \arg(\max(y_{ij_\ell}(u,v)))$, when $\text{average}(y_{ij_\ell}(u,v)) < \text{median}(y_{ij_\ell}(u,v))$; else 0.

12. Moderated Minority Extreme: $\gamma_{i\ell} = 1$, if $\ell = \arg(\min(y_{ij_\ell}(u,v)))$, when $Z > \theta$; or $y_i^{(p)}(u,v) = \text{average}(y_{ij_\ell}(u,v))$, when $\theta > Z > -\theta$; or $\ell = \arg(\max(y_{ij_\ell}(u,v)))$, when $Z < -\theta$; else 0. Here $Z = \text{average}(y_{ij_\ell}(u,v)) - \text{median}(y_{ij_\ell}(u,v))$ and θ is a threshold value.

We enumerate some typical fusion attacks here, however, some more complex collusion attacks can be considered as a combination of these basic collusion attacks.

In the literature, these fusion attacks have frequently been studied by the fingerprinting community. Zhao et al. gave a fairly comprehensive investigation about some typical nonlinear collusion attacks in [7] [8], they include Average attack, Minimum attack, Maximum attack, MinMax attack, Median attack, Uniform attack and Modified Negative attack. During the study, they used the independent Gaussian fingerprinting and the human visual model based spread

spectrum embedding. They gave a theoretical analysis for these attacks, and compared the effectiveness of these collusion attacks in [8], based on two criterias: the probability of falsely accusing at least one innocent user ε_1 and the probability of capturing at least one colluder $1-\varepsilon_2$, with ε_2 is the probability of failing to accuse any colluder. They showed that the Average attack gives the lowest distortion; the Median or MinMax attacks have comparable performances as the Average attack; the Minimum, Maximum and Randomized Negative attacks bring much larger distortion than others. Since the extracted colluded fingerprints under the Minimum, Maximum and Randomized Negative attacks have non-zero mean. However, a post processing could be applied to remove it before the detection process. From the viewpoint of Zhao et al, they considered that the Randomized Negative attack is the most effective attack without the post processing.

However, H.G. Schaathun denied this point as the Randomized Negative attack brings great distortion in practice, he thinks that the Minority choice attack is more powerful if correlation decoding is used [9]. Moreover, he designed an adaptive collusion attack so-called Minority Extreme attack to break the He and Wu joint Watermarking/Fingerinting scheme [41]. Furthermore, he introduced several novel collusion attacks against the orthogonal or random Gaussian fingerprinting that are based on the spread spectrum watermarking technique [41] [9]. These include Majority Extreme attack, Minority Extreme attack, Uniform attack and Hybrid attack. He proved the effectiveness of these attacks in the experimental evaluation, but did not give a theoretical analysis. Up to now, these attacks are still the major potential threats to the fingerprinting scheme.

2.3 Cryptography and Coding Approach

We have discussed the various attacks for the multimedia fingerprinting system in the last section. In this part, we review some former works of fingerprinting from the cryptography and coding viewpoint. We know that fingerprinting was mostly studied by cryptographers. The concept of fingerprinting was introduced in [6], and it has gained interests since Boneh and Shaw's work [11]. In this field, the fingerprinting problem can be decomposed into the following issues: setup of a mathematical model of the attacks, definition of features of the code, construction

of such a code. In a way, the assumption is that watermarkers will embed fingerprinting codes developed by cryptographers. Thanks to the mathematical model of the collusion attack, the conception of codes is decoupled from the physical layer. We now summarize the available literatures in cryptography and coding.

2.3.1 Marking Assumption

The marking assumption is a terminology coming from [11]. A watermark that in a position of the content can be in one of q different states, without causing perceptual artifact. The dishonest users of a collusion can spot the watermarks where the hidden symbols differ, and modify in some way these hidden symbols; but the undetected marks are unchanged. This is the marking assumption as stated in [11]. This stems in a first variation presented in [14]. In the narrow sense problem, colluders replace a detected mark by randomly selecting one of their marks at that position, whereas in the wide-sense problem, colluders replace a detected mark by randomly selecting a symbol in the alphabet. The second variation is the presence of erasures. Boneh and Shaw introduce the case where the colluders can erase the detected marks [11]. This has been generalised by Guth and Pfitzmann in a weak marking assumption [42], where a bounded number of erasures might occur everywhere in the sequence. It was used, for instance, in [43] and [14].

2.3.2 Types of Fingerprinting Codes

A perusal of the fingerprinting literature in cryptography shows a large diversity of types of codes. They can be classified into the following types: 1) the Frameproof code, which means that no collusion can frame another user not in the collusion by producing its codeword. 2) the Secure Frameproof code, which means that no collusion $\mathcal{C}$ can frame a disjoint collusion $\mathcal{C}'$ by producing its descendant. This introduces two branches of fingerprinting codes: either, we strengthen the feature of the code in order that such an event is impossible (strong traceability), either the probability of this event, called error probability, is small and decreasing as the code length increases [11] [14] (weak traceability). 3) the Identifiable Parent Property code, which means that no collusion can produce a codeword

that cannot be traced back to at least one member of the collusion. The goal here is to avoid the case where the potential collusions are indeed disjoint sets. 4) the Traceability code, which means that the pirated codeword is closer to the ones of the colluders than to others. Tracing traitors reduces to searching for the codewords that agree in most symbol positions with the pirated one. However, the tracing algorithm cannot expect to find all parents, since some of them may contribute with too few positions and cannot be traced.

2.3.3 Strong Traceability

Strong traceability is a very hard requirement and typical code lengths are extremely huge. Here we list some useful tools to build and decode traceability codes: 1) first of all, error correcting codes, which is often used to construct the strong traceability codes. For example, references [44] and [16] have independently suggested the use of Reed-Solomon codes. This is also the key idea in [45]. 2) the second tool is the q-ary alphabet. Boneh and Shaw show that strong traceability is not possible for $c \geq 2$ with binary alphabet. This implies the use of q-ary alphabet with $q > 2$ when we expect more than two colluders. Furthermore, strong traceability is a hard constraint implying long codes and/or large alphabets. It is feasible in practice with Reed-Solomon codes but at the cost of a large alphabet. 3) the third tool is the list decoding. The problem with fingerprinting decoding is the identification of several colluders. The input is a corrupted codeword, the output a list of codewords. This functionality is quite unusual in coding theory. For a short the traceability codes, the exhaustive decoding is practicable. But for a long codes such as Identifiable Parent Property, the exhaustive decoding is obviously not viable. The former work dealing with list decoding from Guruswami and Sudan [46] enforce this functionality, it succeeds in finding codewords within a given distance. The list decoding brings two advantages to the traceability codes based on the error correcting codes [45; 47]: a) Frist, list decoding takes into account soft information about the collusion attack model and the embedding technique, while the classical Reed-Solomon decoding algorithm do not take into account a priori information about the channel transmission; b) The second advantage is that it is a relatively efficient decoding,

it is slightly better than the exhaustive search with respect to the decoding complexity. 4) The last tool is the iterative decoding. Because list decoding has a drawback, it works great when colluders share evenly the risk, but, if one colluder contribute less than the others, list decoding doesn't succeed to find him. Thus, Fernandez and Soriano proposed the iterative decoding to deal with the situation where colluders' contributions are uneven [48]] [45]. They showed that when the dominating contributors of the collusion are caught, it is still possible to find the remaining colluders with a list decoding. In their algorithm, a first list decoding succeeds to find $j < c$ colluders. The received codeword is modified, creating a copy of the pirated codeword where symbols matching with caught colluders codewords are erased. A list decoding is performed on this new codeword. This process is iterated until c colluders are caught.

2.3.4 Weak Traceability

Strong traceability implies two big drawbacks: long codes over large alphabets. This stems in a lack of feasibility for some applications. This is the reason why weak traceability is usually preferred. The tracing algorithm usually allows the capture of only one colluder with an error probability ϵ exponentially decreasing as the code length increases. The code is said to be c-secure with ϵ-error. This probability of error ϵ introduces the notion of randomness. What is random here? The marking assumption states that colluders are able to locate the detectable marks. This does not imply any knowledge about the scheme. But, the colluders may or may not be able to read the embedded symbol or to write in place a given symbol. They might simply be able to change the symbol to another one which they do not control. In the same way, they may or may not know their codewords or the codewords not in the collusion. These issues are related to their knowledge about the fingerprinting scheme. To prevent this, the scheme usually uses some secret keys to make traitors in a blind state. In this thesis, we focus on weak traceable fingerprinting codes because they are the most experienced in literature.

We also lists some useful tools for constructing the weak traceability codes. 1) the first one is the permutation, this tool is firstly introduced in [11], which is

used combined with the replication of code. This permutation serves as a secret key, it scrambles the symbol positions before embedding them in content. Hence, colluders cannot notice repeated symbols (unless they notice only the detectable marks). The inverse permutation is applied just after symbol decoding to get back to the original code. 2) the second tool is the concatenation, we know that the code is not practical as its length grows roughly with the number of users. The most well known tool used in weak traceability in order to tackle big number of users is the concatenation of codes, which includes the inner code and the outer code. The decoding of such a concatenated code begins by decomposing the received word in blocks. Then, the decoding for the inner code is applied. This gives, for each block, a codeword or a set of codewords. This sequence of results is the input for the decoding for the outer code. At the encoding side, the outer code is called first, then the inner code. In the tracing algorithm, the inner code is decoded first, then the outer code; whence the terminology 'outer/inner'. The basic idea is that the inner code tackles the robustness against collusion of size c, but for a low number of users. The concatenation with the outer code allows to increase the number of users while keeping the property of the inner code. Yet, this kind of scheme is quite hard to fine tune. If we use a good inner code, with a low probability of error, the blocks are already quite long. There is clearly a tradeoff to strike between the features of the inner and outer codes.

2.4 Signal Processing Approach

In this section, we will talk about how to design a multimedia fingerprinting system from the signal processing point of view in the literature.

2.4.1 Independent Fingerprint Signals

Wang et al. use the orthogonal signals to represent the multimedia fingerprinting code since their orthogonality [49] [17]. Each user is identified by an orthogonal pseudo-random basis signal, so their correlations between each other are null or at least can be neglected, this character helps to decrease the probability of

false positive and leads to simplify detection schemes which employs the correlation. These orthogonal fingerprint signals are embedded into the multimedia content via some typical robust watermarking technique likes spread spectrum watermarking [50]. Then, the fingerprinted multimedia contents are distributed to many different users. The pirate collects several different copies together to perform the attack. Of course, the attack process can be any attack as mentioned in Section 2.2, however, it is often considered as an Average attack in their works [51] [7] [17], since they think that the Average attack is easy to realize and the colluded signal often have a lowest distortion compared to the original signal. Moreover, Zhao et al give some analyses on several nonlinear collusion attacks against the independent fingerprints [51], their experimental works show that the different attacks lead to almost the same perceptual distortion except the Randomized Negative attack.

At the detection side, the distributor finds the pirated copy, he subtracts the original content, here they assume that the detection process is nonblind, in order to get a better accuracy of accusation process. Then, he calculates n test statistics as the correlations of the obtained signal with the n orthogonal watermark signals. The used test statistic could be one of the three detection statistics introduced by Zhao et al [7]: Z-statistic, T_N-statistic and q-statistic. The number of correlation peaks is increasing with the colluder number c, but their amplitudes are decreased by a factor $1/c$. The collusion succeeds if the correlation peaks is below a decision threshold Z. This threshold is calculated according to the desired false positive probability ε_1, which is the probability of falsely accusing at least an innocent user. The power of the detection process is the probability of identifying a real colluder is ε_r. The false negative probability is the probability of failing to accuse any colluders, it could be written as: $\varepsilon_2 = (1 - \varepsilon_r)^c$. So the global probability to identify at least one colluder equals to $1 - \varepsilon_2 = 1 - (1 - \varepsilon_r)^c$. Note that the probability ε_1 given above is for one user. The total probability of false alarm is equal to $1 - (1 - \varepsilon_1)^{n-c} \sim (n - c)\varepsilon_1$. Consequently, the system is c-secure with ϵ-error, $\epsilon < \varepsilon_2 + (n - c)\varepsilon_1$.

One great advantage of the independent fingerprint scheme is its simple structure of the encoding and embedding process, this makes it attractive in identification application which has a small number of users. However, this scheme has

two drawbacks: a relatively high total probability of false alarm and a complexity proportional to the number of users as n correlations are calculated. In [18], the authors propose a recursive algorithm to reduce the number of correlations. However, the number of correlations is almost $2(\lceil \log_2 n \rceil - 1)$. When there are several colluders, the number of correlations increases and the difference with the exhaustive search is very small, and it becomes less interesting.

2.4.2 Redundant Fingerprint Signals

A more effective solution is to use the redundant fingerprint signals. The primitive works can be traced back to the Boneh and Shaw fingerprinting code [11], which is a two-level binary c-secure with ϵ-error. In fact, the independent fingerprint signals can be formulated as the On-Off Keying modulation of an orthogonal signals basis by binary messages. One can observe that it is not the best modulation. The BPSK modulation is more powerful, it can manage n users with only $\log_2 n$ basis signals. This modulation is efficient in the sense that it addresses more users than the number of basis signals. Indeed, shorter codes allow greater power per basis signal and also need less correlation calculus at the decoding side.

Having selected a modulation, the next step is to model the impact of the attacks on the fingerprinting code decoding. The main difference compared with the independent fingerprinting scheme is that they have a watermark decoder (that decodes bits) instead of a watermark detector (that looks for the presence (On) or the absence (Off) of watermark signals). For a great number of colluders, it is illusive to base a traitor tracing program on the estimation of the pirated fingerprint signal, due to the interferences with the host and noises. This is the reason why [18] only detect the symbol with a quite high threshold, correlations smaller than this threshold are decoded as '0'. This looks like the way Boneh and Shaw replace erasures by symbols '0'. The decoded bit is then a '1' if and only if all the embedded bits were set to '1'.

Thereby, Trappe et al. assume that the impact of the collusion approximates the bitwise AND operation in the message domain [18]. They then look for a AND-ACC (AND Anti Collusion Code) binary code. A trivial code is

$\{110, 101, 011\}$. The AND operation on two codewords leaves two '0' indicating the colluders. Furthermore, they create more powerful AND-ACC based on Balanced Incomplete Block Design, a mathematical tool which is out of scope of our study. They also improve the algorithm with a soft decoding. Experimental measurements are given in their article [18], they indicate the probability of false alarm and the power of the test against the number of colluders and the total number of users. Although AND-ACC are based on an average attack model, the authors asserted that more complicated attacks have the same impact if the colluders evenly share the risk. However, AND-ACC have never been tested against more malicious attacks using weighting vector as presented in [52].

In contrast to AND-ACC, He and Wu use Reed-Solomon codes to construct the ECC (Error Correction Code) based fingerprint code. Their reasons are that Reed-Solomon codes have the minimum distance, and some ECC have more efficient decoding algorithms. Furthermore, the ECC based fingerprint code requires much less orthogonal sequences than the independent fingerprint code. This implies that the ECC based fingerprint system has a simpler design and implementation at the embedding stage. A full study about the performances of ECC based fingerprint code is available in [53].

2.4.3 Accessory Strategies

In the complete fingerprinting system proposed by He and Wu [20], several accessory strategies are proposed to improve the collusion resistance and computation performance. In the fingerprinting code layer, He and Wu applied the prior group information to the redundant fingerprinting scheme [54]. This strategy is derived from the fact that an adversary is more likely to collude with some users than with other users due to geographic area or social background. This technique is firstly introduced by Wang et al [19], where they proposed group-oriented fingerprinting to enhance the collusion resistance of independent fingerprinting. In this way, the fingerprinting code for each user consists of user subcode and group subcode, which are hidden overlappingly into host signal via a watermarking technique. The detection is done in two levels, which identifies guilty groups through correlation and then narrows down to specific colluders inside the extracted guilty

groups. According to their simulation results [54], they stated that the improved fingerprinting system provides substantial improvement over the previous ECC based fingerprinting.

In the embedding layer, an other strategy called sub-block permutation is used for their ECC based multimedia fingerprinting [20]. This technique is firstly utilized for the construction of collusion-secure fingerprinting codes by Boneh and Shaw in [11]. The core idea is that the watermark embedding hides the fingerprint codes into the content block by block, each block can be divided into several sub-blocks, and these sub-blocks are permuted according to a secret key to obtain the final fingerprint signal for representing the user.At the detection side, the extracted fingerprint sequence is first inversely permuted before being passed to the detector. This technique can prevent the colluders from the attacks for the whole fingerprinted blocks, since they have no way to identify which sub-blocks belonging to one block. Moreover, this technology renders the collusion process similar in effect to an averaging, and averaging collusion is far less effective from the colluders' point of view. However, this solution has no effect on the fusion attacks.

At the detection side, He and Wu also propose a method that automatically adjusts the group detection threshold according to the detection statistics for group information to adapt to different collusion patterns [21]. Their strategy is similar as in [19]. The core idea is that they choose the group detection threshold by tuning system parameters to obtain the optimal colluder detection probability and the lowest false alarm probability. Their experimental results show that this adaptive detection outperforms the non-adaptive detection under a variety of collusion scenarios, and it can provide up to 10% improvement on the overall probability of detection.

These above accessory strategies play some important roles in He-Wu joint Watermarking/Fingerinting scheme [20], and this fingerprinting system was later used in video fingerprinting application for a large group of users [23]. However, it is broken by an adaptive collusion attack so-called minority extreme attack, which was introduced by Schaatun in 2007 [41]. In order to fill this hole, Lin et al. propose a new solution [55], where they introduce a row-permuted binary

orthogonal inner code along with an adaptive detector. However, it brings a lot of additional computation.

2.5 Statistical Approach

In 2003, Tardos proposed a fingerprinting code whose code length has the theoretically minimum order [24]. Afterwards, many improvement works have appeared from different directions. The related works mainly focused on the generalization of Tardos code from binary to q-ary, the reduction of the code length, the improvements of the accusation function. In this sections, we detail Tardos seminal fingerprinting codes and summarize the improvements.

2.5.1 Original Tardos Fingerprinting Code

Peikert et al. gave the lowest bound of $O(c^2 log(1/c\epsilon))$ on the length of the general fingerprinting code (if $c << n$) [56], with collusion size c and accusation error ϵ. Tardos was the first to publish a binary probabilistic fingerprinting code whose length touches this bound [24]. We now review Tardos seminal works here.

Tardos original fingerprinting code is generated as follows: first of all, we note n the number of users, m the code length of the binary codeword in the Tardos fingerprinting system. The distributor chooses m independent and identically distributed random variables $\{p_i\}_{i=1}^{m}$ from the interval $[t, 1-t]$, in other words $p_i \in [t, 1-t]$, where t is a small cutoff parameter and Tardos set $t = 1/(300c)$. The variables p_i are distributed according to the pdf:

$$f(p) = \frac{1}{2\arcsin(1-2t)} \frac{1}{\sqrt{p(1-p)}} \tag{2.3}$$

Secondly, he fills the columns of the $n \times m$ codewords matrix X by independently drawing random bits $X_{ji} \in \{0, 1\}$ according to the law $\mathbb{P}(X_{ji} = 1) = p_i$. X is the code-matrix, whose j-th row corresponds to the fingerprint distributed to the j-th user.

We define C to denote a collusion set which has c colluders, and the $c \times m$ matrix X_C to denote the codewords distributed to the colluders. The pirates mix their personalized copies to create a pirated copy y. The distributor later finds

this pirated copy and extracts the fingerprint Y. In order to identify at least one colluder in the coalition, he computes an accusation sum for each user. The accusation sum S_j for the j-th user can be calculated as:

$$S_j = \sum\nolimits_{i=1}^{m} U_T\left(Y_i, X_{ji}, p_i\right) \tag{2.4}$$

where

$$U_T(Y_i, X_{ji}, p_i) = \begin{cases} g_1(p_i) & \text{if} \quad Y_i = 1 \quad \text{and} \quad X_{ji} = 1 \\ g_0(p_i) & \text{if} \quad Y_i = 1 \quad \text{and} \quad X_{ji} = 0 \\ 0 & \text{if} \quad Y_i = 0 \quad \text{and} \quad X_{ji} = 1 \\ 0 & \text{if} \quad Y_i = 0 \quad \text{and} \quad X_{ji} = 0 \end{cases} \tag{2.5}$$

The accusation weight functions $g_1(p)$ and $g_0(p)$ are defined as:

$$g_1(p_i) = -g_0(1 - p_i) = \sqrt{\frac{1 - p_i}{p_i}} \tag{2.6}$$

The distributor decides that user j is guilty, if S_j is greater than a certain accusation threshold Z. Here Z mainly depends on the desired false positive probability ε_1 (the probability that an innocent user gets accused). Tardos chooses the fingerprinting code lengths m and the accusation threshold Z in the following ways:

$$m = 100c^2 \left\lceil \ln \varepsilon_1^{-1} \right\rceil ; \quad Z = 20c \left\lceil \ln \varepsilon_1^{-1} \right\rceil \tag{2.7}$$

Furthermore, he proved that, if the collusion size $\leq c$, and with $\varepsilon_2 = \varepsilon_1^{c/4}$, the achieved false positive and false negative error rates by his scheme are respectively lower than ε_1 and ε_2 [24] .

2.5.2 Symmetric Tardos Fingerprinting Code

The original Tardos fingerprinting scheme has two drawbacks. Firstly, the computation of Tardos accusation sum in equation (2.4) is asymmetric, because only those codewords $Y_i = 1$ in i-th block are taken into account, and the others($Y_i = 0$) are discarded. This way of exploiting the information hidden in the pirated copy is unfair and inefficient, since the $Y_i = 0$ blocks carry as much information as the $Y_i = 1$ blocks. Secondly, this construction cannot be extended to nonbinary alphabets due to its asymmetry.

Skoric et al. improve Tardos original scheme by two modifications [25]. Firstly, they generalize the fingerprinting code generation from binary to q-ary. Let $\mathcal{Q}$ be

an alphabet of q symbols and $\mathcal{Q} = \{0, 1, ..., q-1\}$, they use $X_{ji} \in \mathcal{Q}$ instead of bits $\{0, 1\}$. Instead of one random scales $\{p_i\}_{i=1}^m$, they employ the Dirichlet distribution function to generate m independent random vector $\mathbf{p}_i = (p_i^0, ..., p_i^{q-1})$ for $1 \leq i \leq m$, where the components satisfy $p_i^\alpha \in [t/(q-1), 1-t]$ and $\sum_{\alpha=0}^{q-1} p_i^\alpha = 1$. We note $\bar{\mathbf{p}} = \{\mathbf{p}_i\}_{i=1}^m$, the vectors $\mathbf{p}_i$ have the pdf $F(\mathbf{p}_i)$ which is invariant under any permutation over $\mathcal{Q}$. Thus this construction is symmetric in all symbols $\alpha \in \mathcal{Q}$. In the i-th column of X, random symbols are generated according to $\mathbf{p}_i$ such that $\mathbb{P}(X_{ji} = \alpha) = p_i^\alpha$. The j-th row of the matrix $\mathbf{X}$ will be used for the j-th user.

The second modification is the computation of the accusation sum. Unlike Tardos original scheme, their accusation sum computation take all fingerprint symbols that occur in the pirated copy into account. This accusation sum for a certain user at a certain symbol block is positive if he has the same symbol as in the pirated copy; otherwise it is negative. The expression of the accusation sum U_j can be written in the following way:

$$S_j = \sum\nolimits_{i=1}^{m} U(Y_i, X_{ji}, p_i^{Y_i}) \tag{2.8}$$

with

$$U(Y_i, X_{ji}, p_i^{Y_i}) = \delta(Y_i, X_{ji}) g_1(p_i^{Y_i}) + (1 - \delta(Y_i, X_{ji})) g_0(p_i^{Y_i}) \tag{2.9}$$

where $\delta(Y, X)$ denotes Kronecker delta, and they use the same accusation weight functions as in Equation (2.6).

In detail, the accusation sum of Equation (2.8) is computed as follows: if user j has the same symbol as in the i-th block as the pirated copy, his accusation score is added by a positive amount $g_1(p_i^{Y_i})$, where this score decreases with the symbol's probability. If user j has a different symbol than the pirated copy, then his score is added by a negative amount $g_1(p_i^{Y_i})$, which has the largest effect when the symbol y_i is likely to occur. Note that their improved accusation sum function (Equation (2.8)) is fully symmetric with the symbols, where the Kronecker deltas reduces the symbol space into two classes: $X_{ji} = Y_i$ and $X_{ji} \neq Y_i$. Therefore, the value of the accusation sum does not depend on the actual value X_{ji}, since it just depends on the similarity with the symbols appearing in the pirated copy.

2.5.3 Reduction of Tardos Code Length

The length of the fingerprinting code plays an important role in the multimedia application, so there are numerous works aiming at finding a tighter lower bound on the Tardos code length [57] [25] [26] [27] [30] [28] [29]. In the seminal work [24], Tardos proposed a code length: $m = 100c^2 \log \frac{1}{\varepsilon_1}$ for n users that are ε_1-secure against c pirates. This length is shorten than Boneh and Shaw's result in $O(c^4 \log \frac{1}{n\varepsilon_1} \log \frac{1}{\varepsilon_1})$. Afterwards, Skoric et al. showed that for the sufficiently large value c, the code length can be reduced to approximately $m = 4\pi^2 c^2 \log \frac{1}{\varepsilon_1}$ in typical content distribution applications, if a high false negative error probability ε_2 can be tolerated [57]. Furthermore, they found that, for the colluders size c sufficiently large, the accusation sums of the innocent user and of the colluder have probability distributions that are very close to Gaussian; and they indicated that if these distributions are perfectly Gaussian, a code length of $m \approx 2\pi^2 c^2 \log \frac{1}{\varepsilon_1}$ is sufficient for achieving the desired error probabilities ε_1 and ε_2 if they are independents. Later, they proposed a symmetric version of Tardos fingerprinting code used for arbitrary alphabets [25]. When c is large, the code length can be shortened to $m \approx \pi^2 c^2 \log \frac{1}{\varepsilon_1}$ for binary alphabets, by improving the accusation function. Furthermore, invoking the Central Limit Theorem in the case of sufficiently large c, they showed that even a code length of $m \approx \frac{1}{2}\pi^2 c^2 \log \frac{1}{\varepsilon_1}$ is adequate. Moreover, they indicated that the code length can be further reduced by using a q-ary alphabet to replace a binary alphabet, their numerical results show that a reduction of 35% is achievable for $q = 3$ and 80% for $q = 10$.

Blayer and Tassa took another direction to reduce of the code length by improving the parameter choice for Tardos codes [29]. They replaced the constants with parameters in Tardos' original scheme and derived a set of inequalities that those parameters must satisfy, then, they looked for a solution of those inequalities in which the codes length m is minimal. In this way, the code length can be reduced by a factor of approximately 4. Moreover, they pointed out that this value can be further reduced by decoupling ε_1 and ε_2. In their simulation, the code length can be reduced to 6.426% of the Tardos original scheme, in the case of $n = 100$, $c = 20$ and $\varepsilon_1 = \varepsilon_2 = 0.01$.

Hagiwara et al. also shortened the code length of Tardos code for a small number of colluders, and gave a primitive construction for the discrete Tardos fingerprinting code in [58]. Following this work, Nuida et al. have done some improvement works [26] [27] [30] [28] which concern the practical implementation issues of Tardos code and the reduction of code length. Especially in [28], they showed that, when the collusion size c goes to infinity, the code length are significantly reduced to 5.35%, thanks to the modification of the tracing algorithm object: tracing only the pirate with the highest score leads to a small error probability.

Recently, Amiri and Tardos introduced a high rate fingerprinting [59] by combining the two approaches [24] [60]. They gave a tight estimation about the rate of the fingerprinting code as the code lenght m goes to infinity, and they declared that the rate of their code achieves the fingerprinting capacity. However, the computational complexity in the accusation algorithm of their fingerprinting code makes it very difficult in practical application. But their work pointed out a direction for the future research.

Contrasted to these theoretical improvements, Furon et al. found an experimental approach to estimate the minimal code length of binary symmetric Tardos code [61]. Firstly they assessed the worst case attack that the colluders can lead, then they used a rare event analysis algorithm to compute the probabilities of error ε_1 and ε_2. Finally, for a given collusion size, they were able to estimate the minimal length of the code satisfying some error probabilities constraints. Their experimental result indicated that the estimated code length are far smaller than the previously known theoretical lower bounds.

2.5.4 Improvements of Accusation Function

Another major improvement about Tardos code concerns the accusation function. The first improvement is the symmetric accusation function introduced by Skoric et al. as we mentioned in subsection 2.5.2. Later in [31], Furon et al. indicated that Tardos' original accusation functions and Skoric's symmetric accusation functions are very conservatives: they used the same generic accusation functions for any kind of collusion strategy. They are optimum with respect

to the original assumptions, but some more efficient functions can be derived if the knowledge of the collusion strategy is available. In the subsequent work [32], Charpentier et al. proposed a practical way to improve the accusation function in binary case through an iterative estimation of the colluders's strategy. With the iterative structure and their improved accusation functions, Tardos fingerprinting code works much better than the original ones when matching the collusion strategy. The efficiency is verified in their experimental work when the collusion size c is big; but if c is too small, the Expectation Maximization algorithm fails and the estimated collusion strategy is not accurate at all, that's because the accuracy of estimation strongly depend on the ratio c/n.

2.6 Chapter Summary

This chapter mainly provides the background and the state of the art in multimedia fingerprinting field. We give a general framework of multimedia fingerprinting system in Section 2.1, which is based on three layers: fingerprinting layer, watermarking layer and attacks layer. Then, we fully analysis the various attacks for multimedia fingerprinting in Section 2.2, they include the attacks in watermarking layer, the attacks in fingerprinting code layer, and the fusion attacks which lies across these two layers. These attacks are very important in the multimedia fingerprinting system design, because they are the criterion for testing whether a multimedia fingerprinting system is successful or failing.

We also review three multimedia fingerprinting systems: the first approach is the cryptography and coding approach initiated by Boneh and Shaw, which based on the Marking Assumption (Section 2.3). The second approach is introduced by the researchers who worked in Maryland university, they tried to design a practical multimedia fingerprinting system from the signal processing point of view (Section 2.4); The third approach is from the statistical viewpoint, which is mainly around the Tardos probabilistic fingerprinting code (Section 2.5). The first approach focuses on the fingerprinting code design. The second approach further improved the fingerprinting code design and investigated the method of combination between the fingerprinting code and the watermarking layer. The last approach mainly concerns the fingerprinting code and shortens the code

length. These approaches have a common drawback, that is, they do not pay sufficient attention to the watermarking layer. Actually, the watermarking layer is very important for the full system design of multimedia fingerprinting, because a good watermarking is a necessary condition to establish a powerful multimedia fingerprinting. If the fingerprints can be removed by application of normal watermarking attacks, the multimedia fingerprinting system cannot be considered as a successful design. So in the following two chapters, we will explore some watermarking schemes, which have good robustness and security, and finally they will be used in our secure and robust multimedia fingerprinting system design.

Chapter 3

Robust Watermarking for Multimedia Fingerprinting

Any robust and secure multimedia fingerprinting scheme must rely on a robust and secure watermarking technique to embed users' fingerprinting codes. However, as far as we know, a robust and secure watermarking technique design is a challenging work, since robustness and security are considered as two different concepts in the watermarking area [62] [63] [64]. Watermarking robustness concerns common signal processing actions without a precise strategy. In contrast, watermarking security deals with attacks based on the knowledge of the watermarking technique and the observations of several contents marked with the same key. Therefore, during these attacks, the attacker carries out precise actions to estimate the secret key/subspace and perform a surgical strike which removes the watermark with as less distortion as possible. These are two indispensable aspects of watermarking, they influence and restrict each other.

In this chapter, we study the watermarking technique for multimedia fingerprinting, especially we focus on its robustness performance. We firstly analyze the robustness and security performances of some classical existing watermarking techniques. Based on these analysis, we decide to take a very robust watermarking technique, "Broken Arrows", as a starting point, and further enhance its robustness capability in this chapter.

3.1 Watermarking Choice for Multimedia Fingerprinting

Lots of watermarking schemes have been presented in the literature. The watermark can be embedded in pixel space or transformed space, such as Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Curvelet Transform, or Contourlet Transform etc. In order to choose a suitable watermarking, we briefly compare two main families of watermarking methods: spread spectrum watermarking and quantization based methods.

3.1.1 Spread Spectrum Watermarking VS Quantization Index Modulation

Spread spectrum watermarking introduced by Cox et al. is probably one of the important breakthroughs in watermarking field [50], their result greatly increased the robustness of the watermarking scheme to resist the common signal processing operations and certain intentional attacks. In this method, the host signal is firstly transformed into a frequency subspace; then the pseudo-random generator generates a pseudo-random sequence depending on the secret key, it is always considered as secret carrier; next the watermarking embedder modulates this secret carrier according to the message to be hidden to obtain the watermark signal, and finally adds it to the original host signal. In this way, the watermark is concealed secretly, and therefore preventing the unauthorized users to access or remove it. This watermarking method can be considered as Additive Spread Spectrum, afterwards, some improvements have been proposed, such as Scale Spread Spectrum [65], and Improved Spread Spectrum [66]. Even so far, spread spectrum method is still widely used; many actual embedding schemes are based on this method.

Another important family of watermarking schemes is the Quantization Index Modulation (QIM) proposed by Chen and Wornell [67]. The main inspiration came from considering watermarking as communications with side information by

Cox et al. [68], and the rediscovery of Costa's result about the interference cancellation channel [69]. In the literature, the implementation of QIM includes Distortion Compensation-Dither Modulation (DC-DM), Spread Transform-Dither Modulation (ST-DM) [67], Trellis Code Quantization [70] [71], and Lattice-Quantizer Index Modulation (Lattice QIM) [72]. These quantization-based methods provably outperformed the former spread spectrum methods in the sense of the traditional watermarking evaluation criteria: the distortion and robustness.

However, the security performance of quantization-based methods is not so good. Pérez Freire et al. have deeply studied and compared the security levels of these two classes of watermarking methods [73] [74] [75]. They took the lattice based QIM as an example to represent the quantization-based methods, their results showed that the security level of lattice based QIM scheme is lower than for spread spectrum methods. Moreover, they indicated the reasons of this gap: this is due to the host-rejection properties of lattice based QIM scheme, and to the host interference of the spread spectrum watermarking (see [76, Chapter 7]). We can also find some previous works for assessing the security level of quantization-based schemes, for instance the dirty paper trellis, in [77]. These analyses revealed the relatively weak security levels of advanced robust techniques based on quantization.

In this thesis, our objective is to design a robust and secure watermarking for the multimedia fingerprinting application. Based on the above analysis, we will focus on spread spectrum watermarking in this thesis.

3.1.2 Why Choose "Broken Arrows"?

How to design a robust and secure watermarking technique? Especially in using spread spectrum methods? This is a recent hot topic. Cayre and Bas have realized a trial, they gave a detailed analysis for the security of spread spectrum methods; afterwards, they proposed two new watermarking modulations in the Watermarked-Only-Attack framework (Only the watermarked contents are available for the attacker) from theoretical point of view, called natural watermarking and circular watermarking [78]. Natural watermarking is shown to provide perfect security, however, the price is the significant degradation of the robustness

relative to the previous spread spectrum method. Circular watermarking method was proposed to improve robustness at the cost of achieving lower security levels. Mathon et al. confirmed and optimized these results for the experimental point of view [79] [80]. These schemes considered security as a top priority, then evaluated the robustness. The gap is quite big compared to past advanced robust techniques. It is just this reason that motivates us to seek a breakthrough from other directions.

In this thesis, our approach is the reverse, we start with a very robust zero-bit watermarking technique, “Broken Arrows”, and then try to increase its security levels. “Broken Arrows” [33] has been designed for the second contest of Break Our Watermarking System (BOWS-2) [34]. Its performances in terms of robustness and imperceptibility are state-of-the-art. One another great advantage is that it has been intensively put to a lot of usual attacks during BOWS-2 contest and resisted well, this superiority is incomparable with other watermarking techniques. First of all, we will review the original “Broken Arrows” watermarking scheme in the next section.

3.2 “Broken Arrows” Watermarking Scheme

We briefly review the original “Broken Arrows” watermarking technique in this section. Its embedding and detection processes involve three spaces conversion among the following four nested spaces: the pixel space, the wavelet subspace, the secret subspace and the MCB (Miller, Cox and Bloom) plane.

3.2.1 Watermark Generation

We summarize the main processes of watermark generation in “Broken Arrows” in this subsection. An illustration for this scheme is shown in Figure 3.1. Firstly, we take the original image $\mathbf{i}_X$ in the pixel space, which is the $H_\mathbf{i} \times W_\mathbf{i}$ matrix of 8-bit luminance values; then we perform the 2D wavelet transform (Daubechies 9/7) on three levels of decomposition of $\mathbf{i}_X$, and select the coefficients from all the bands in the wavelet subspace except the low-frequency LL band. These

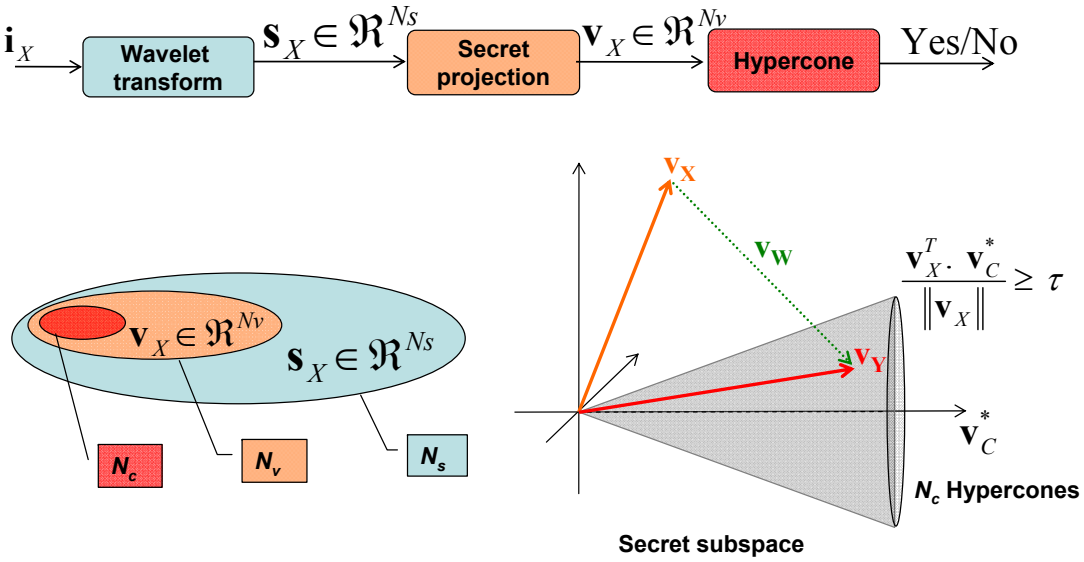

Figure 3.1: The illustration for "Broken Arrows" watermarking scheme.

$N_s = H_\mathbf{i} \times W_\mathbf{i}(1-1/64)$ wavelet coefficients are then stored as the host in wavelet space $\mathbf{s}_X$.

Secondly, the host coefficients in the wavelet space is transformed to the secret space, we use N_v secret binary antipodal carriers signals of size N_s: $\mathbf{S}_{C,j} \in \left\{-1/\sqrt{N_s}, 1/\sqrt{N_s}\right\}^{N_s}$, $\forall j \in \{1, ..., N_v\}$, produced by a pseudo-random generator seeded by the secret key K. The host signal is projected onto these carrier signals: $v_X(j) = \mathbf{S}_{C,j}^T \mathbf{s}_X$, these N_v correlations being stored as $\mathbf{v}_X = (v_X(1), ..., v_X(N_v))^T$. This means that $\mathbf{v}_X$ represents the host signal in the secret subspace. We can write this projection with the $N_s \times N_v$ matrix $\mathbf{S}_C$ whose columns are the carrier signals:

$$\mathbf{v}_X = \mathbf{S}_C^T . \mathbf{s}_X \tag{3.1}$$

Note that the norm is conserved because the secret carriers are assumed to constitute a basis of the secret subspace: $\|\mathbf{v}_X\|^2 = \mathbf{s}_X^T \mathbf{S}_C \mathbf{S}_C^T \mathbf{s}_X \approx \|\mathbf{s}_X\|^2$.

Finally, we transfer the host secret vector $\mathbf{v}_X$ to the MCB plane. In the secret subspace, we define a set $\mathcal{V}$ of secret directions with N_c secret unitary vectors: $\mathcal{V} = \{\mathbf{v}_{C,k}\}_{k=1}^{N_c}$. With the host signal being represented by $\mathbf{s}_X$ in this space, we

look for the "nearest" secret direction from the host vector:

$$\mathbf{v}_C^* = \text{sign}(\mathbf{v}_C^T \mathbf{v}_X)\mathbf{v}_C \tag{3.2}$$

with

$$\mathbf{v}_C = \arg \max_{k \in \{1,\ldots,N_c\}} \left| \mathbf{v}_X^T \mathbf{v}_{C,k} \right| \tag{3.3}$$

With the secret vector $\mathbf{v}_C^* \in \mathbb{R}^{N_v}$ in the secret subspace, the basis of the MCB plane is given by ($\mathbf{v_1}$,$\mathbf{v_2}$) as in [33, Eq.(3)]:

$$\mathbf{v}_1 = \mathbf{v}_C^*, \quad \mathbf{v}_2 = \frac{\mathbf{v}_X - \left(\mathbf{v}_X^T \mathbf{v}_1\right) \mathbf{v}_1}{\left\| \mathbf{v}_X - \left(\mathbf{v}_X^T \mathbf{v}_1\right) \mathbf{v}_1 \right\|}. \tag{3.4}$$

Hence, the MCB plane contains $\mathbf{v}_C^*$ and $\mathbf{v_X}$. The coordinates representing the host are $\mathbf{c}_X = (c_X(1), c_X(2))^T$ with $c_X(1) = \mathbf{v}_X^T \mathbf{v}_1$ and $c_X(2) = \mathbf{v}_X^T \mathbf{v}_2$. According to a certain criterion for maximizing the robustness (see [33, Sec 3.1.2]), the watermarked coordinates $\mathbf{c}_Y = (c_Y(1), c_Y(2))^T$ are presented as:

$$\mathbf{c}_Y = \begin{cases} \left(c_X(1) + \sqrt{\rho^2 - c_X(2)^2}, 0\right)^T & \text{for} \quad c_X(2) \leq \rho \cos(\theta) \\ \mathbf{c}_X + \rho(\sin(\theta), -\cos(\theta))^T & \text{for} \quad c_X(2) > \rho \cos(\theta) \end{cases} \tag{3.5}$$

Here the parameter ρ is related to the embedding distortion constraint, which we will detail it in the next subsection, and θ is an angle defining the cone of the detection region, which is a function of the parameter N_v and the probability of false alarm p_{fa} for watermark detection. We can compute it according to the paper [81]:

$$T = \frac{\sqrt{2}}{\sqrt{N_v - 2}} \text{erf}^{-1}(1 - 2 * p_{fa}), \tag{3.6}$$

$$\theta = \text{acos}(\frac{e^{2T} - 1}{1 + e^{2T}}). \tag{3.7}$$

Therefore, the generated watermark signal in the MCB plane can be represented by $\mathbf{c}_W = \mathbf{c}_Y - \mathbf{c}_X$.

3.2.2 Watermark Embedding

In order to reconstruct the watermarked signal, the watermark signal should be projected back to the wavelet subspace. Firstly, $\mathbf{c}_W$ in the MCB plane is projected

back to the secret space as:

$$\mathbf{v}_W = (\mathbf{v}_1, \mathbf{v}_2).\mathbf{c}_W \tag{3.8}$$

and the watermarked signal in the secret space is:

$$\mathbf{v}_Y = \mathbf{v}_X + \mathbf{v}_W \tag{3.9}$$

here $\mathbf{v}_W$ is the watermark vector of secret space, we note its norm is ρ. Then, $\mathbf{v}_W$ is projected back in the wavelet subspace to get the watermark signal in the wavelet domain $\mathbf{s}_W$, which can be written as

$$\mathbf{s}_W = \mathbf{S}_C.\mathbf{v}_W \tag{3.10}$$

The norm of $\mathbf{s}_W$ satisfies: $\|\mathbf{s}_W\|^2 = \mathbf{v}_W^T \mathbf{S}_C^T \mathbf{S}_C \mathbf{v}_W$. In BA, $\mathbf{S}_C$ was considered as orthonormal since it is a very long random matrix, so we have $\mathbf{S}_C^T \mathbf{S}_C \approx \mathbf{I}_{N_v}$ and :

$$\|\mathbf{s}_W\| \approx \|\mathbf{v}_W\| = \rho. \tag{3.11}$$

The watermark signal is added to $\mathbf{s}_X$ with some perceptual mask $\mathbf{M}$:

$$\mathbf{s}_Y = \mathbf{s}_X + \mathbf{M}.\mathbf{s}_W. \tag{3.12}$$

In "Broken Arrows", the mask is indeed proportional to the absolute value of the host wavelet coefficients: $\mathbf{M} = |\mathbf{s}_X|$. The mask has two important impacts we will take into account in the following subsections.

3.2.2.1 Effect on The Embedding Distortion

We model the masking weights by random variables statistically independent of $\mathbf{s}_W$, and with second order moment empirically measured as $\overline{M^2} = N_s^{-1} \sum_{i=1}^{N_s} M(i)^2$. This assumption allows us to write that:

$$\|\mathbf{s}_Y - \mathbf{s}_X\|^2 \approx \overline{M^2}.\|\mathbf{s}_W\|^2 = \overline{M^2}\rho^2. \tag{3.13}$$

This squared norm is also equal to the Mean Squared Error over the image, times the number of pixels, because the wavelet transform conserves the Euclidean norm. Hence the following relationship between ρ and the required PSNR:

$$\rho = \frac{255.\sqrt{W_\mathbf{i} H_\mathbf{i}}}{\sqrt{\overline{M^2}}} 10^{-\mathrm{PSNR}/20}, \tag{3.14}$$

where $(W_\mathbf{i}, H_\mathbf{i})$ is the width and height of the original image.

3.2.2.2 Effect on The Projection Vector

A difficulty stems from the fact that the mask disturbs the vector retro-projection. When we mix the generated watermark signal $\mathbf{s}_W$ in the wavelet space, and then retro-project the watermarked signal $\mathbf{s}_Y$ back onto the secret space, it is not located where we expect, that is, not in $\mathbf{v}_X + \mathbf{v}_W$. Actually, the retro-projection denoted by $\mathbf{v}_Y$ works as follows:

$$v_Y(k) = v_X(k) + \sum_{j=1}^{N_v} v_W(j) \sum_{i=1}^{N_s} M(i) s_{C,j}(i) s_{C,k}(i), \tag{3.15}$$

We need to assume that:
i) the involved variables can be treated as independent random vector;
ii) the second sum over N_s coefficients can be seen as the empirical average equaling the expectation;
iii) $\mathbf{S}_C^T \mathbf{S}_C \approx \mathbf{I}_{N_v}$, to derive this simplification:

$$v_Y(k) \approx v_X(k) + v_W(k)\overline{M}. \tag{3.16}$$

with $\overline{M} = N_s^{-1} \sum_{i=1}^{N_s} M(i)$. Therefore, from a vector $\mathbf{v}_W$ of norm ρ created at the embedding side, we end up with a vector $\overline{M}\mathbf{v}_W$, at the detection side. We must take this 'amplification' into account when looking for the best watermark vector. For this, we modify ρ by a factor $\overline{M}$, we get its final expression:

$$\rho_{fin} = \frac{255.\overline{M}\sqrt{W_{\mathbf{i}} H_{\mathbf{i}}}}{\sqrt{\overline{M^2}}} 10^{-\mathrm{PSNR}/20} \tag{3.17}$$

$$s_Y(k) = s_X(k) + M(k).s_W(k)/\overline{M}. \tag{3.18}$$

In this way, the embedding can yield perceptually acceptable watermarked pictures for PSNR above 40 dB. In our experiments, we set the targeted PSNR = 43 dB as in the BOWS-2 contest.

3.2.3 Technique Flaw

With the PSNR greater than 40 dB, it appears that the amplitude of the samples of $\mathbf{s}_W$ are almost all lower than 1. Therefore, this embedding technique conserves the sign of the wavelet coefficients. In our opinion, this is the real flaw of the

technique: the watermarking amplitude is proportional to the host signal, and the sign of the wavelet coefficient of the watermarked signal are unchanged. An attack not modifying the sign of the coefficients automatically preserves this important part of the original content. Therefore, if the amplitude of the attacked coefficients is sufficiently different while preserving the quality of the content, the watermark can no longer be detected. This is indeed the case with the attack mounted by A. Westfeld [39] in the first episode of BOWS-2 contest. He designed a specific attack for wavelet-based schemes, which can be regarded as a de-noising process. It is mainly based on the estimation of the amplitude of any wavelet coefficient as a function of the coefficients in its neighborhood via a linear regression. Our purpose in this chapter is to provide an improved version of "Broken Arrows", which is robust against this specific attack, as well as the normal signal processing operations. Our countermeasures are to design a less predictable mask, with a stronger watermark amplitude. We will detail them in the next section.

3.3 Robustness Enhancement of "Broken Arrows"

In order to strengthen the robustness performance of the "Broken Arrows" embedding technique, we propose two improvement directions: (i) Balancing the Wavelet Coefficients of three subbands in the same transformation level (BWC), and (ii) Averaging the Wavelet Coefficient with four neighbouring coefficients in the same subband (AWC). We will present them in the following subsections.

3.3.1 BWC Proportional Embedding

Our first study consists in correlating coefficients of the three subbands in the same wavelet transformation level. In each level of wavelet transformation, we balance the wavelet coefficients of the three subbands. The Level 0 case is given as an example in Figure 3.2. We denote $\mathbf{M_{BWC}}(LH0)$ (resp. $\mathbf{M_{BWC}}(HH0)$ and $\mathbf{M_{BWC}}(HL0)$) to represent the sub-masks of the perceptual mask to modulate the watermark signal in subband $LH0$ (resp. $HH0$ and $HL0$). For Level 0 of wavelet

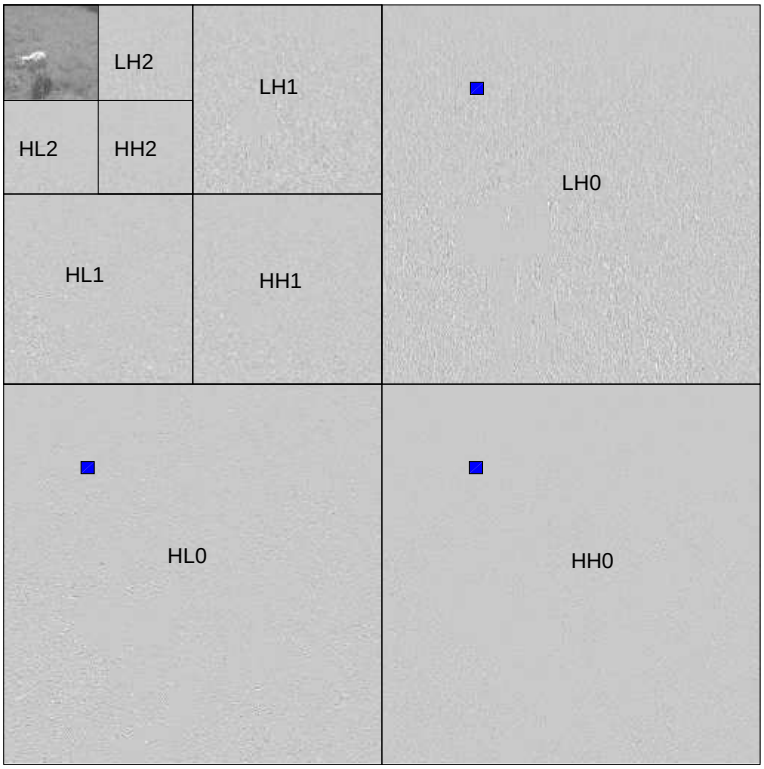

Figure 3.2: Balancing the Wavelet Coefficients of three subbands in each transformation level (BWC).

transformation, their three sub-masks for $LH0$, $HH0$, $HL0$ three subbands are all set to:

$$\begin{aligned}&\mathbf{M_{BWC}}(LH0(m,n)) = \mathbf{M_{BWC}}(HH0(m,n)) = \mathbf{M_{BWC}}(HL0(m,n))\\&= \frac{|\mathbf{s}_X(LH0(m,n)) + \mathbf{s}_X(HH0(m,n)) + \mathbf{s}_X(HL0(m,n))|}{3}\end{aligned} \tag{3.19}$$

where $\mathbf{s}_X(LH0(m,n))$ (resp. $\mathbf{s}_X(HH0(m,n))$ and $\mathbf{s}_X(HL0(m,n))$) denotes the original wavelet coefficient in position (m,n) of the host signal in subband $LH0$ (resp. $HH0$ and $HL0$). In the same way, we can obtain the sub-masks for the Level 1 and Level 2 of the wavelet decomposition. The mask $\mathbf{M_{BWC}}$ is the only difference between the original BA and BWC proportional embedding methods.

Intuitively, this embedding enhances the dependency between the subbands of the wavelet coefficients of the watermark signal. For a given position, the mask has a value might bigger than the smallest amplitude of the three considered coefficients. Depending on the value of the watermark signal at this position, the embedding might consequently change the sign the wavelet coefficient. In other words, the presence of the watermark is not only hidden in the amplitudes of the coefficients but also in some of their signs. We will verify our statement in the subsection 3.3.3. The robustness performance of the improved embedding technique will be confirmed by the experimental results in Section 3.4 for resisting Westfeld denoising attack as well as the common signal processing.

3.3.2 AWC Proportional Embedding

Another avenue to improve the robustness of the embedding scheme is to take into account the dependency between the neighbouring coefficients. The main idea is inspired from Westfeld denoising attack. We replace any coefficient of the wavelet transform by an average of five coefficients: itself and four local neighbors. In the Figure 3.3, we take the wavelet coefficients in the subband LH0 as an example. We obtain the mask:

$$\mathbf{M_{AWC}}(m,n) = \frac{1}{5}\left|\sum_{r=m-1}^{m+1}\sum_{t=n-1}^{n+1}\mathbf{s}_X(r,t)\right| \tag{3.20}$$

where $\mathbf{s}_X(m,n)$ denotes the wavelet coefficient of the position (m,n) in $\mathbf{s}_X$ for any band except the low frequency LL band. $\mathbf{s}_X(m-1,n)$, $\mathbf{s}_X(m,n-1)$, $\mathbf{s}_X(m+1,n)$,

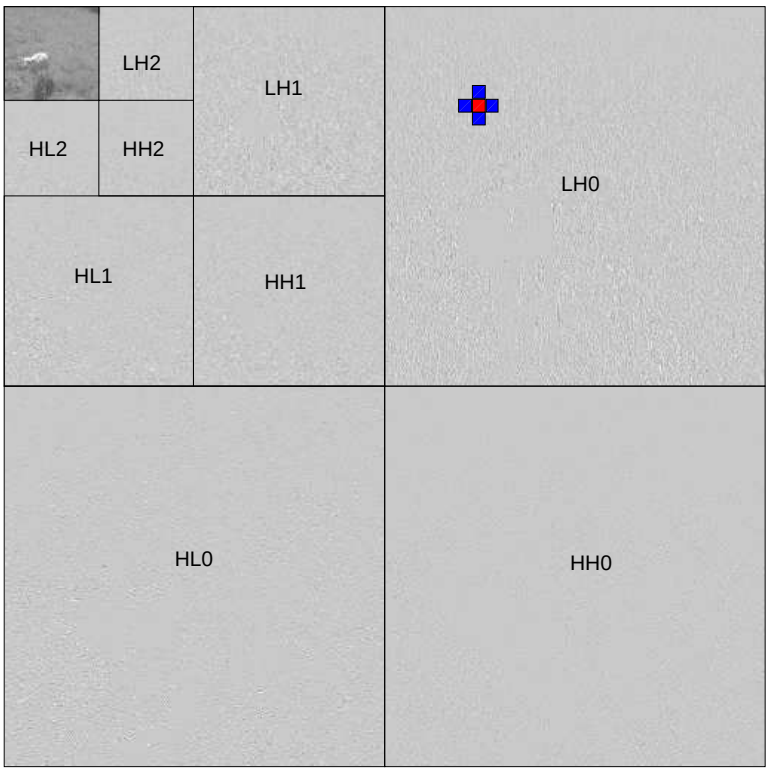

Figure 3.3: Averaging the Wavelet Coefficient with four neighboring coefficients in the same subband (AWC).

Figure 3.4: The test image "sheep" in the database of BOWS-2 contest.

and $\mathbf{s}_X(m, n+1)$ are its four neighbors. Putting all the $\mathbf{M_{AWC}}(m, n)$ together, we get another mask $\mathbf{M_{AWC}}$, which will serve in the AWC proportional embedding. The masks $\mathbf{M_{AWC}}$, $\mathbf{M_{BWC}}$, $\mathbf{M_{BA}}$ are the only differences between the AWC, BWC and BA proportional embedding methods.

In this way, the watermark signal might modify the signs of the host coefficients. As BWC, it is also efficient solution to cope with Westfeld denoising attack. We now confirm our statements concerning the BWC and AWC embedding techniques through the experiments.

3.3.3 Experimental Validation

In order to verify our proposed solutions in the last two subsections, we test the image "sheep" as shown in Figure 3.4, which is taken from the database of BOWS-2 contest. We keep the same test conditions as in [33], except change the visual mask. In simulation, three different embedding strategies are compared: the original BA proportional embedding, the BWC proportional embedding and the AWC proportional embedding. The expected PSNR is set to 43 dB; the real PSNR of the watermarked images are 42.88 dB for BA, 42.88 dB for BWC, and 42.81 dB for AWC. The histogram of the visual mask $\mathbf{M_{BA}}$ of the BA proportional embedding (resp. $\mathbf{M_{BWC}}$ of the BWC proportional embedding and $\mathbf{M_{AWC}}$ of the AWC proportional embedding) is shown as Figure 3.5 (resp. Figure 3.6 and Figure 3.7). According to these histograms, we can see that the values of the visual masks $\mathbf{M_{BWC}}$ and $\mathbf{M_{AWC}}$ are more centered on 0. Thereby, the factors, which amplify the watermark signal $\mathbf{v}_W$ during the embedding, are smaller for the two improved embeddings BWC and AWC. Actually, we can confirm this result by the experiment, since we have $\overline{M}_{\mathbf{BA}} = 16.02$, $\overline{M}_{\mathbf{BWC}} = 9.71$, and $\overline{M}_{\mathbf{AWC}} = 7.08$.

However, on the other side, for each kind of embedding technique, we want to have almost the same PSNR for the final watermarked image, or the Mean Squared Error over the image. In other word, we should have almost the same value of ρ_{fin} (see Equation 3.17) for the different embedding techniques. In order to achieve this objective, we should increase the norm value ρ of the watermark signal $\mathbf{v}_W$ (see Equation 3.14) for the two improved embedding techniques. This is indeed we have done in the experiment. In the test, we obtained the $\overline{M^2}_{\mathbf{BA}} = 23.95$, $\overline{M^2}_{\mathbf{BWC}} = 13.87$, and $\overline{M^2}_{\mathbf{AWC}} = 11.08$. According to the Equation 3.14, the norm of the watermark signal to be embedded for these three watermarking techniques are: $\rho_{\mathbf{BA}} = 38.59$, $\rho_{\mathbf{BWC}} = 66.63$ and $\rho_{\mathbf{AWC}} = 83.39$.

We know that with the PSNR greater than 40 dB, the amplitude of the samples of $\mathbf{s}_W$ are almost all lower than 1, so the BA original proportional embedding almost conserves the signs of the wavelet coefficients. Actually, according to the test for the image "sheep", only 0.76% wavelet coefficients change their signs after

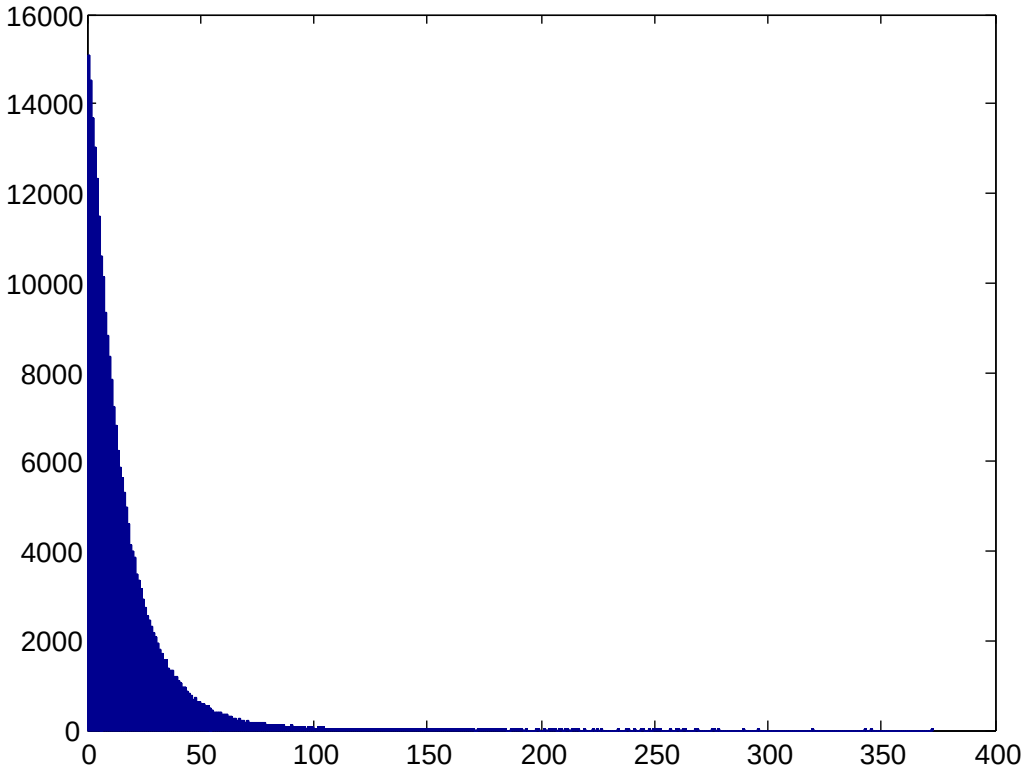

Figure 3.5: The histogram of the BA visual mask $\mathbf{M_{BA}}$ of image "sheep".

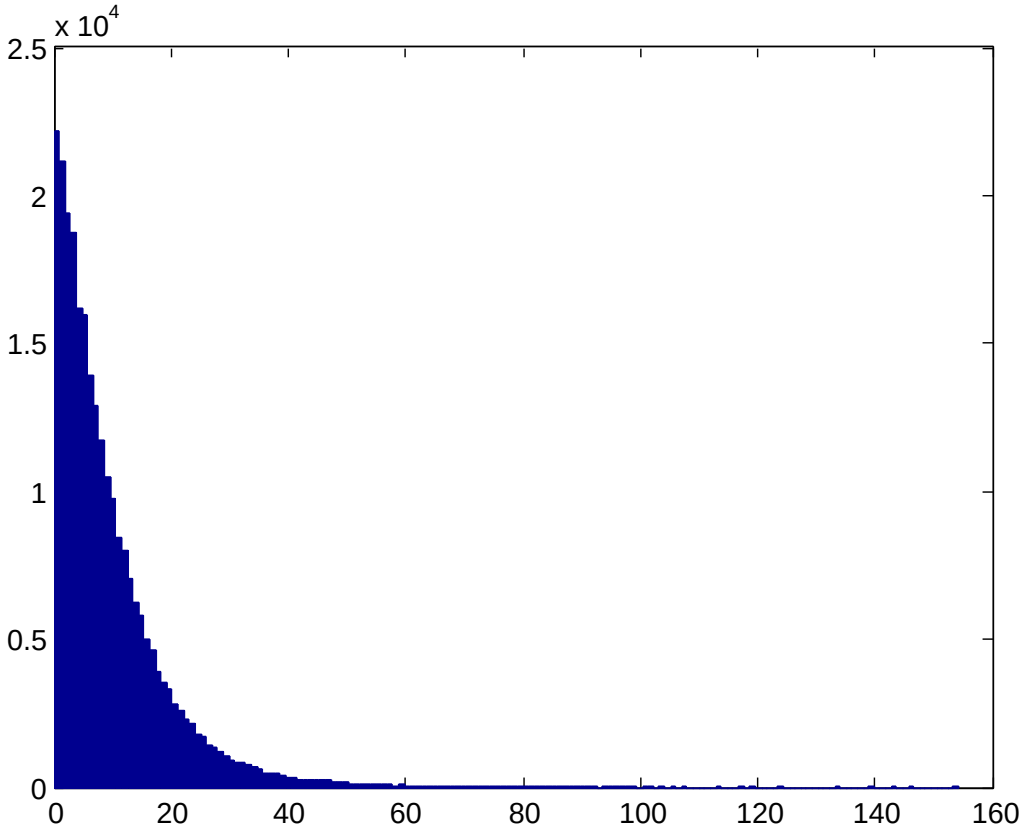

Figure 3.6: The histogram of BWC visual mask $\mathbf{M_{BWC}}$ of image "sheep".

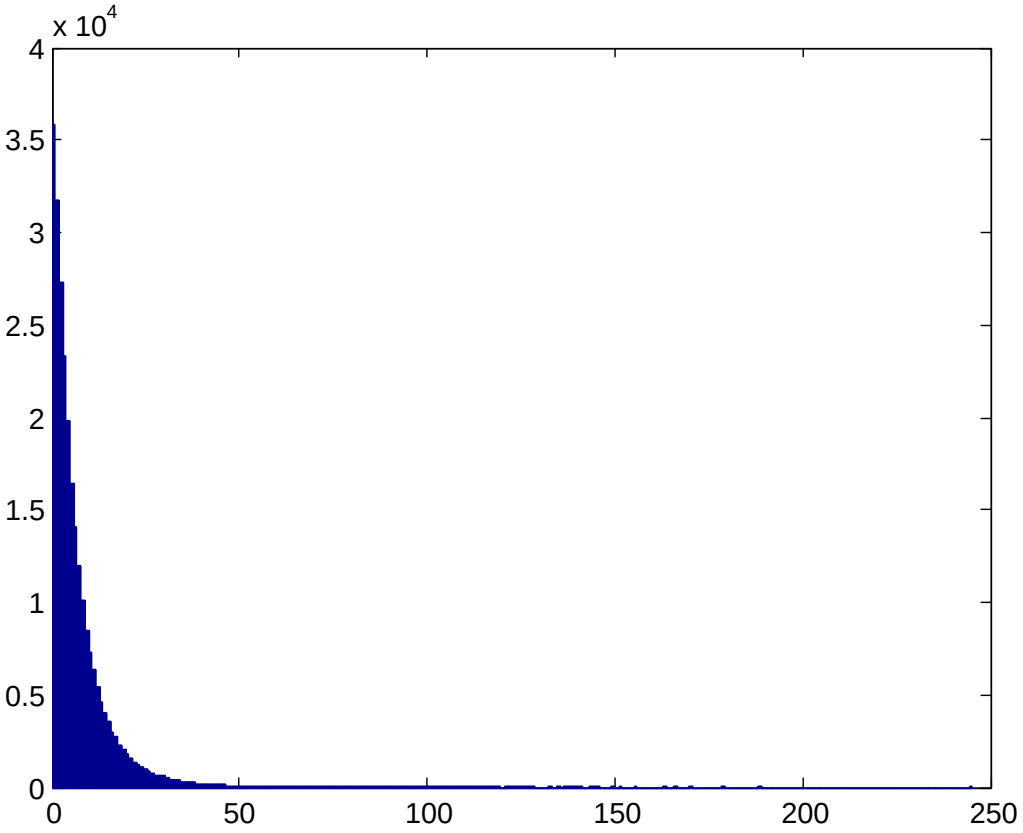

Figure 3.7: The histogram of AWC visual mask $\mathbf{M_{AWC}}$ of image "sheep".

the BA embedding process. Therefore, we can say that the watermark signal is almost hidden in the amplitude. However, the norm of the watermark signal of the two improved embedding techniques are so big, the watermark signal inevitably change certain signs of the wavelet coefficients. In our simulation, the 2.36% wavelet coefficients have changes the signs after the BWC proportional embedding, and the 2.16% wavelet coefficients have changes the signs after the AWC proportional embedding. Consequently, the watermark signal is not only hidden in the amplitude of the wavelet coefficients, but also concealed in their sign. Furthermore, the Figure 3.8 shows the mask distributions for three embedding techniques(BA, AWC, and BWC), for the image "sheep". The results confirm that: the maskes of our improved embedding techniques are less predictable than before.

In the next section, we will evaluate the robust performance for both BWC and AWC.

3.4 Robustness Evaluations

We first compare the robustness of the improved BWC and AWC proportional embeddings with the original BA proportional embedding, using the same benchmark as in the original paper [33]. Then we discuss the robustness performance for resisting the Westfeld denoising attack.

3.4.1 Facing Common Attacks

We use the same $2,000$ luminance images of size 512×512 as in [33]. These pictures represent natural and urban landscapes, people, or objects, taken with many different cameras from 2 to 5 millions of pixels. Three different watermark embedding strategies are compared: the original BA proportional embedding and the variants BWC and AWC. During embedding, the input PSNR is set to 43 dB; the real PSNR of the watermarked images is in between 42.5 dB and 43 dB. As for the original BA, the visual distortions are invisible for almost all images when using BWC or AWC proportional embeddings. Indeed, these latter schemes yield slightly better quality (this is a subjective assessment).

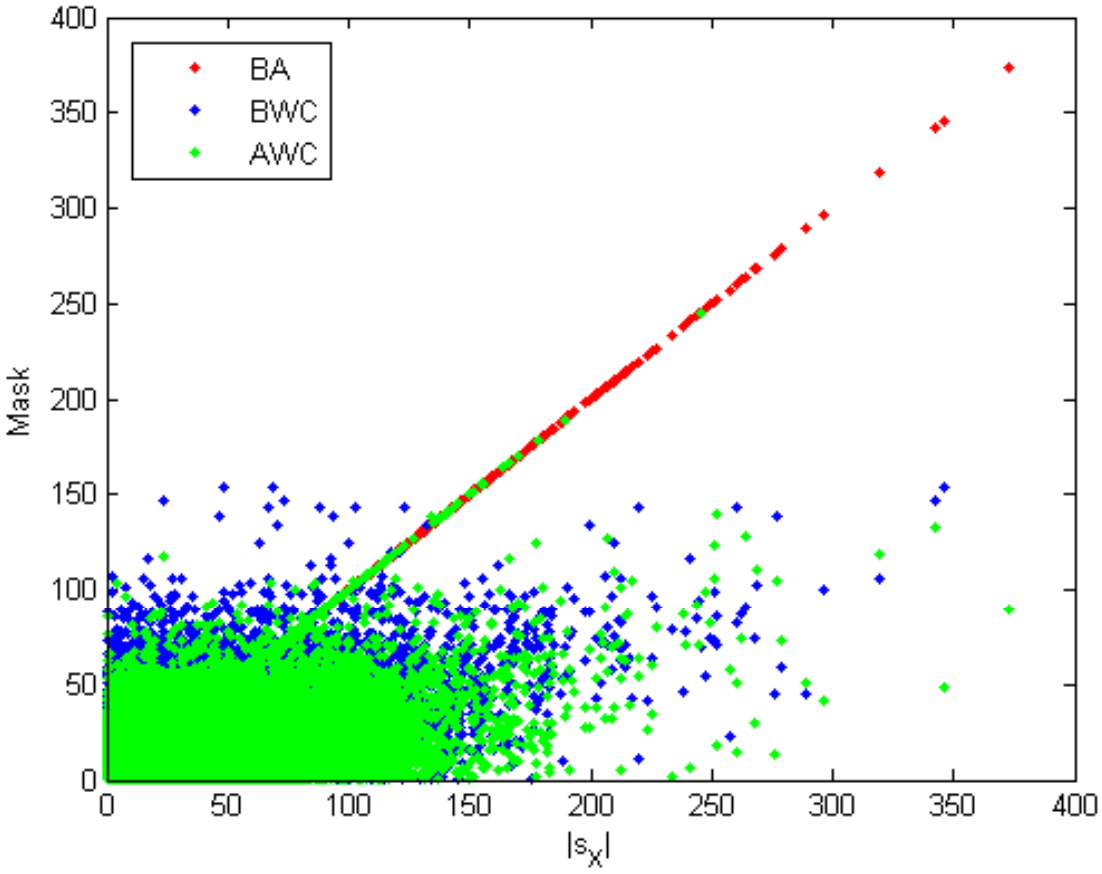

Figure 3.8: The mask distributions for three embedding techniques for the image "sheep".

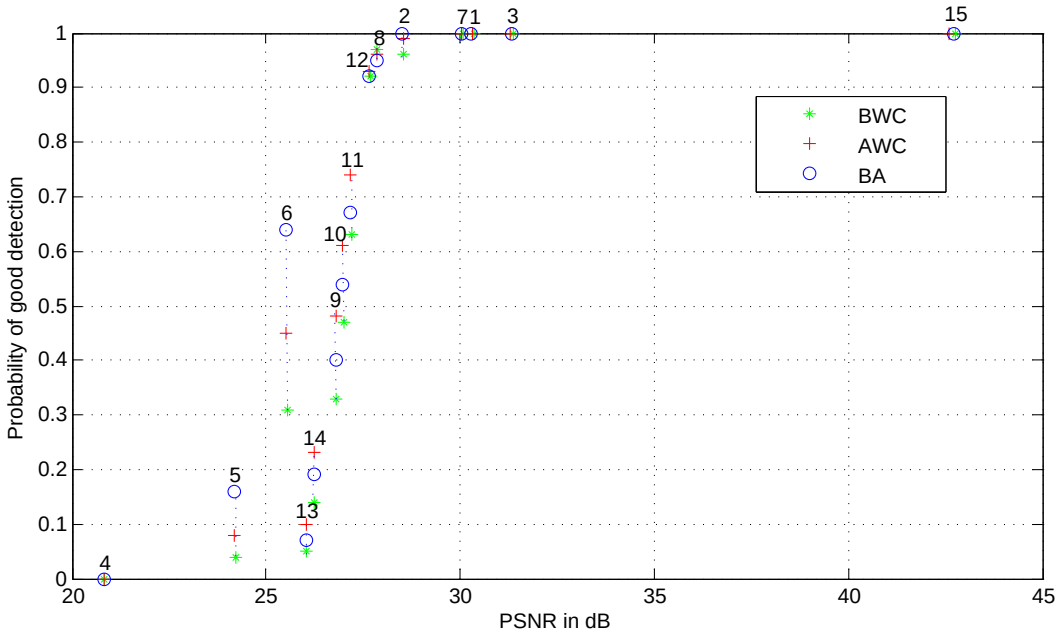

Figure 3.9: Probability of good detection versus average PSNR of the attacked images for the three watermark embedding techniques: BWC, AWC and BA.

We apply the same benchmark on watermarked images as in BA's original paper, that is, a number of attacks mainly composed of combinations of JPEG and JPEG 2000 compressions at different quality factors, low-pass filtering, wavelet subband erasure, and a simple de-noising algorithm. Figure 3.9 reveals the impact of 15 most significant attacks on the three embedding techniques. The selected attacks include: 1) denoise threshold 20; 2) denoise threshold 30; 3) JPEG Q = 20; 4) JPEG2000 r = 0.001; 5) JPEG2000 r = 0.003; 6) JPEG2000 r = 0.005; 7) scale 1/2; 8) scale 1/3; 9) scale 1/3 + JPEG Q = 50; 10) scale 1/3 + JPEG Q = 60; 11) scale 1/3 + JPEG Q = 70; 12) scale 1/3 + JPEG Q = 90; 13) scale 1/4 + JPEG Q = 70; 14) scale 1/4 + JPEG Q = 80; 15) no attack. The probability of detecting the watermark (i.e. number of good detections divided by $2,000$) is plotted with respect to the average PSNR of the attacked images. Because these classical attacks produce almost the same average PSNR, the three points for a given attack are almost vertically aligned. The impact on the probability of detection is interesting: each watermark embedding technique has its advantage for resisting different attacks. AWC proportional embedding is more robust than others technique for resisting Attacks 9-14. BA proportional embedding is better for resisting Attacks 2, 5, and 6. For Attacks 1, 3, 4, 7, and 15, the three embedding technique have a comparable performance. Although the BWC proportional embedding has a tiny predominance for Attack 8, its overall performance is worse than the two other techniques.

3.4.2 Facing Westfeld Denoising Attack

In this section, we evaluate the robustness of the three watermarking embedding techniques (BA, BWC and AWC) against Westfeld denoising attack. To get a result comparable with the above experiment, we keep the same testing conditions and use the same $2,000$ images. This is the main difference with Westfeld's test presented in [39], as he used all the $10,000$ images available on BOWS-2's website [34].

The PSNR of the attacked images ranges from 19.9 to 46.2 dB. This result is almost the same as Westfeld's (from 19.7 to 45.0 dB). Figure 3.10 shows the decreasing percentage of successfully broken images for increasing PSNR. For BA

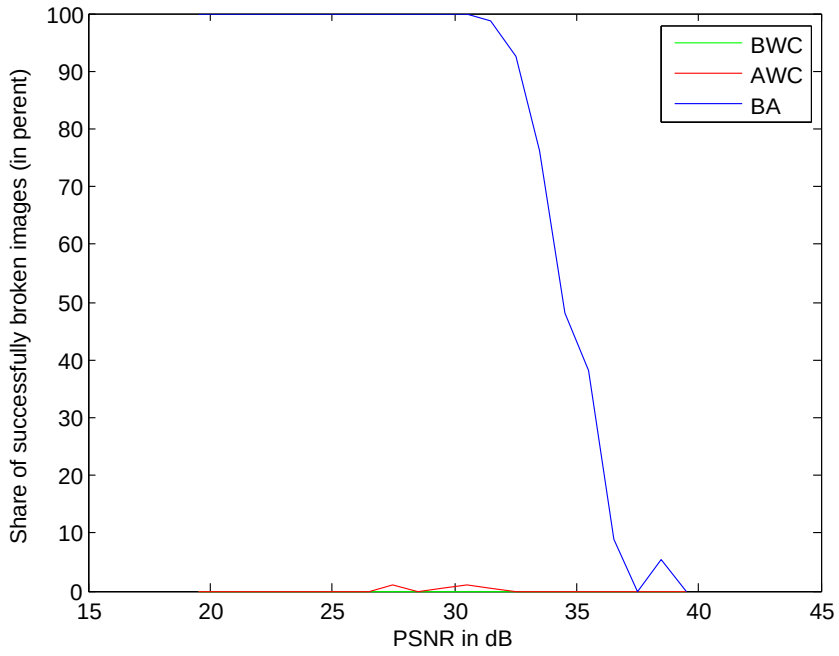

Figure 3.10: Operating curve of the estimated images from the BOWS2 database.

proportional embedding, Westfeld denoising attack is really powerful, successful for 100% of the images when the PSNR is less than 30 dB, and even if its efficiency decreases when the PSNR is growing, it is still successful for 40% of the images when the PSNR is around 35 dB. Note that this does not exactly fit Westfeld's experimental result, because he used a bigger set of images. Nevertheless, our set is large enough to illustrate the power of his attack on the original BA.

Figure 3.10 also shows the results of both variants BWC and AWC. For BWC, Westfeld denoising attack does not work at all: the percentage of successfully broken images is 0 for any PSNR. For AWC, Westfeld denoising attack works for very few images for a PSNR ranging from 26 to 32 dB. Any of our two improved embedding techniques is sufficient to cope with Westfeld denoising attack. However, some robustness is lost against some common attacks, especially for the BWC proportional embedding. Therefore, in order to prevent Westfeld denoising attack as well as the others, we have to make a trade-off, and the AWC proportional embedding seems to be the best choice.

3.5 Chapter Summary

One task of this chapter is to seek a promising watermarking technique for constructing robust and secure multimedia fingerprinting scheme. For this purpose, we briefly compared the robustness and security performance of two main categories of watermarking techniques: spread spectrum watermarking and quantization based methods. According to some previous works, we found that quantization based methods provided a better robustness against AWGN, but weak security levels. Therefore, we explored the spread spectrum based watermarking schemes.

Some schemes have exploited spread spectrum methods by considering security as a priority, and then evaluated its robustness, but the gap is still large to the advanced robust techniques. Our approach is different, we start with a very robust zero-bit watermarking technique, "Broken Arrows", and try to increase its security levels. Because it had been tested by many attacks during BOWS-2 contest and gave a good performance.

To further improve the robustness performance of "Broken Arrows" watermarking technique, we propose two improved embedding techniques AWC and BWC. These proposed techniques perfectly prevent Westfeld denoising attack, which is the worst robustness attack in the first episode of the BOWS-2 contest. In the next chapter, we will continue to enhance the security aspect of the watermarking technique.

Chapter 4

Secure Watermarking for Multimedia Fingerprinting

This chapter considers the security aspect of watermarking for multimedia fingerprinting. We recall that in the multimedia fingerprinting scenario, the watermarking technique embeds the fingerprinting code into multimedia content block by block. For a given user, all these blocks are watermarked with a few number of secret keys according to the fingerprinting code. It is easy for the pirate to extract enough information for estimating the secret keys, and thereby remove the watermark while preserving an excellent perceptual quality. So the security level of watermarking technique plays an important role in the security of multimedia fingerprinting.

In the previous chapter, we have selected "Broken Arrows" as the watermarking technique for multimedia fingerprinting application, and proposed an enhancement called AWC proportional embedding, which further strengthened the robustness to against Westfeld denoising attack [39], which was the worst attack disclosed during the first episode of BOWS-2 challenge. However, security flaws have been disclosed and discussed [5] during the second and third episodes of BOWS-2, during which the pirates could observe plenty of watermarked pictures with the same secret key. These flaws clearly prevent the use of "Broken Arrows" in multimedia fingerprinting applications. But the second episode of BOWS-2 is concerning about the oracle attacks, which is not relevant in multimedia fingerprinting. Thereby, one goal of this chapter is to propose counterattacks to

strengthen the security performance of "Broken Arrows".

4.1 Security Attacks for "Broken Arrows"

First of all, we focus in this section on two security attacks specifically dedicated to "Broken Arrows"watermarking scheme [5].

Algorithm 1: Cluster by exclusion

$l_1 := 1$ (we used $l_1 := 3661$, image Sheep, without restricting generality);
for $k = 2 \dots N_c - 1$ **do**
    $l_{k+1} := \text{argmin}_{j=1}^{|\mathcal{D}|} \max_{i=1}^{k}(|\text{cor}(\hat{s}_{W(l_i)}, \hat{s}_{W(j)})|)$;
    for $m = 1 \dots k$ **do**
        $l_m := \text{argmax}_{j=1}^{|\mathcal{D}|}|\text{cor}(\hat{s}_{W(l_m)}, \hat{s}_{W(j)})|$ for $j \neq l_m$;
    end
end

4.1.1 Westfeld Clustering Attack

A security attack called Westfeld clustering attack was introduced by A. Westfeld [5], which is the worst security attack in the third episode of BOWS-2 challenge. Its main steps can be summarized as follows:

1. He applies Westfeld denoising attack [39] to the (10,000 in BOWS-2) watermarked images.

2. As these attacked images look like estimated original images, he removes them from the watermarked images to estimate the watermark signals.

3. These estimated watermarks are sorted into several bins ($N_c = 30$ in BOWS-2), by using a clustering method as show in Algorithm 1.

4. For a given bin, he averages all the estimated watermarks of the bin to estimate the secret carrier of this bin, and subtracts it from images watermarked with this carrier. Finally he obtains the attacked images.

4.1.2 Bas Subspace Estimation Attack

Another watermarking security attack is the subspace estimation attack proposed by P. Bas [5]. We sum up its main process here:

1. Through a huge number of observations, the fast and efficient OPAST algorithm estimates the projection matrix $\mathbf{W}$, whose size is $N_s \times N_p$, here N_s is the number of the wavelet coefficients to be watermarked and N_p represents the number of principal components. In order to assess the performance of his subspace estimation algorithm, he used the Square Chordal Distance(SCD) between the secret subspace Span($\mathbf{S}_C$) and the estimated subspace Span($\hat{\mathbf{W}}$) during this step. SCD was firstly proposed by Pérez-Freire et al. [75] in the watermarking security domain, the computation of the SCD is defined by the principal angles (the minimal angles between two orthogonal bases [82]). Here he uses it to compute the distance between the secret subspace and the estimated subspace. The smaller the SCD, the better the subspace estimation.

2. With this projection matrix $\hat{\mathbf{W}}$, he uses Independent Component Analysis (ICA) technique to estimate each axis direction and thereby the whole secret matrix $\hat{\mathbf{C}}$.

3. Finally, he pushes the watermarked content outside the detection region by making use of the estimated secret matrix $\hat{\mathbf{C}}$. In this way, the watermark is removed with a high PSNR.

Our experimentation confirmed its good performance. OPAST (Orthogonal Projection Approximation Subspace Tracking) is the key ingredient of this attack. The usual Principal Component Analysis (PCA) algorithm based on eigenvalue decomposition cannot operate a so big data set. The designers of "Broken Arrows" thought that therefore PCA was no longer a threat. However, the discovery of the inline and iterative OPAST proved that they were wrong.

4.2 Security Improvements of "Broken Arrows"

In order to prevent these two attacks pulling down the watermarking scheme and thereby the multimedia fingerprinting system, we have to find some ways to enhance the watermarking security. In this part, several efficient solutions will be introduced.

4.2.1 Countermeasure to Westfeld Clustering Attack

As we mentioned in the last chapter, AWC proportional embedding is introduced as one of the best efficient solutions to prevent the Westfeld denoising attack while maintaining a good robustness against a lot of usual attacks. But its impact on the security performance has never been examined before.

In other words, Westfeld's clustering attack was the worst security attack during the third episode of BOWS-2 [34], and Westfeld denoising attack is a core step in this clustering attack (See Section 4.1.1). Therefore, since AWC proportional embedding prevents Westfeld denoising attack to estimate the watermarks correctly, Westfeld clustering attack may no longer do a good classification of the watermarks.

In order to confirm this idea, we implemented Westfeld's clustering attack in Matlab and found two slight improvements of his Algorithm (see Algorithm 1) [5]. In his algorithm, all the bin leaders are updated at each iteration (step 4 to step 6). This operation does a lot of repetitive work, and wastes a lot of computing power. We give the reason here: for a given bin leader, we compute its correlations with all the observations which are not the current bin leaders, during each iteration; and then choose the one who has the biggest correlation value to replace the previous bin leader. In this way, the algorithm converges towards a stable region very quickly. After certain number of iterations, the cluster inevitably converges to two observations, they are both the one which has the biggest correlation for each other. So in this case, the iteration is not necessary to continue. According to our experiment, for a given cluster, it usually needs 3 or 4 iterations to find a stable bin leader, while the following iterations output the last 2 leaders alternatively. The observation of this phenomenon is used as a stopping condition. This greatly reduces the computing time by 25%.

Another improvement is that in our experiment, we also consider the condition to avoid the repetition of the bin leader. In Westfeld's clustering algorithm (Algorithm 1 [5]), the observation which has the smallest correlation with all the existing bin leaders, is selected as a leader of a new bin. But, with this initialization of the new bin, it is possible to select as the bin leader a vector which was already a leader of another bin. This tends to split a 'correct' bin into several small clusters. So in our simulation, we pay attention in truly finding new leaders. In this way, the probability of splitting bins is reduced, and this improves the accuracy of the classification.

We keep the same test condition as A. Westfeld in order to obtain comparable results. Firstly, we take the m images of the BOWS-2 database ($m = 10,000$), then during the watermark embedding, we save the cone index information for every image. This allows to build a ground truth classification. Denote $\mathcal{B}_{ref}(i)$ the subset of all images which have been watermarked with the i-th secret cone. As there are N_c secret cones, the m watermarked images are classified into a partition $\mathcal{B}_{ref}$ of N_c subsets: $\mathcal{B}_{ref} = \bigcup_{i=1}^{N_c} \mathcal{B}_{ref}(i)$. This partition is the ground truth and it will be used to evaluate the accuracy of the clustering attack.

Secondly, we apply Westfeld denoising attack on all the watermarked images to get an estimation of the original images. This yields an estimation of the watermark signal: $\hat{\mathbf{s}}_W = \mathbf{s}_Y - \hat{\mathbf{s}}_X$. Thirdly, we run our improved Westfeld's clustering attack with a targeted bin number N_t in the range $\{1, ..., N_c\}$. This yields a partition $\mathcal{B}_{est}$ of N_t subsets: $\mathcal{B}_{est} = \bigcup_{i=1}^{N_t} \mathcal{B}_{est}(i)$.

The question is now how to evaluate the accuracy of this clustering attack. Note that N_t might not be equal to N_c. For that purpose, the confusion matrix $\mathbf{P}_{conf}$ is first computed:

$$P_{conf}(k,l) = \frac{|\mathcal{B}_{est}(k) \cap \mathcal{B}_{ref}(l)|}{M}, \quad \forall (k,l) \in \{1,..,N_t\} \times \{1,..,N_c\}. \tag{4.1}$$

This confusion matrix can be considered as the probability transition of a noisy Discrete Memoryless Channel. The subset indices of ground truth partition are the symbols of the source to be broadcast through this channel. Their probabilities are given by $P_{ref}(l) = |\mathcal{B}_{ref}(l)|/M$, $l \in \{1,..,N_c\}$. The indices of the partition induced by the clustering attack are the received symbols. Denote

$P_{est}(k) = |\mathcal{B}_{est}(k)|/M$, $k \in \{1, .., N_t\}$. Then, the accuracy of the attack is measured as the quantity of information its clustering carries about the ground truth partition, i.e. the mutual information between the index of the clustering (the 'received symbols') and the index of the ground truth partition (the 'emitted symbols'):

$$MI(\mathcal{B}_{est}, \mathcal{B}_{ref}) = \sum_{k=1}^{N_t} \sum_{l=1}^{N_c} P_{conf}(k, l) \log \frac{P_{conf}(k, l)}{P_{ref}(l).P_{est}(k)} \tag{4.2}$$

Now, the problem is that this measure is not very well calibrated. For instance, if the attack doesn't work at all, producing a clustering which is purely random and independent of the ground truth, the expected value of the mutual information will depend on N_t and N_c. This prevents us from comparing clustering accuracy for different values of N_t. The adjusted mutual information (AMI) has been recently proposed by Vinh et al.[83] as a calibrated measure:

$$AMI(\mathcal{B}_{est}, \mathcal{B}_{ref}) = \frac{MI(\mathcal{B}_{est}, \mathcal{B}_{ref}) - \mathbb{E}\{MI(\mathcal{R}(N_t), \mathcal{R}(N_c))\}}{\max_{\mathcal{C}}\{MI(\mathcal{C}, \mathcal{B}_{ref})\} - \mathbb{E}\{MI(\mathcal{R}(N_t), \mathcal{R}(N_c))\}} \tag{4.3}$$

where $\mathbb{E}\{MI(\mathcal{R}(N_t), \mathcal{R}(N_c))\}$ is the expected mutual information between two random clusterings $\mathcal{R}(N_t)$ of size N_t and $\mathcal{R}(N_c)$ of size N_c, under a given statistical model; and $\max_{\mathcal{C}}\{MI(\mathcal{C}, \mathcal{B}_{ref})\}$ is maximum value of the mutual information for this particular ground truth partition (indeed the maximum of the entropies of the 'emitted' and 'received' symbols).

In our simulation we measure the accuracy of the clustering attack by the adjusted mutual information $AMI(\mathcal{B}_{est}, \mathcal{B}_{ref})$ and then we plot it with respect to the ratio N_t/N_c in Figure 4.1. For the original BA embedding technique, with $N_t/N_c = 1$, the adjusted mutual information AMI is 0.7, this means that the estimated clustering $\mathcal{B}_{est}$ is very similar to the ground truth partition, and the Westfeld clustering attack succeeds in estimating the secret cones with a decent accuracy. However, this classifier does not work with our improved embedding method AWC: for almost all ratio N_t/N_c from 0 to 1, the adjusted mutual information is smaller than 0.05. The reason is that: the estimation of the watermark signal plays a crucial role in Westfeld clustering attack, and our improved embedding AWC prevents the estimation and thus the attack. So AWC proportional embedding is an efficient solution to block Westfeld's clustering attack.

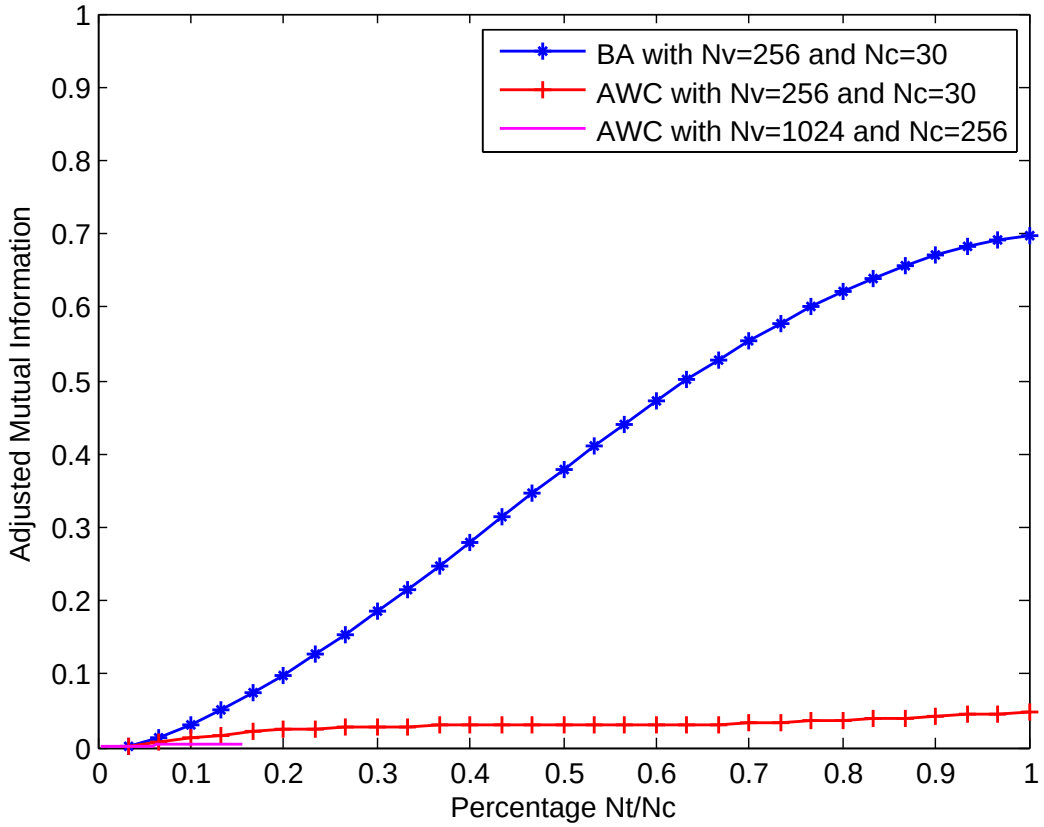

Figure 4.1: Probability of the good classification for Westfeld classifier against BA and AWC, with different N_v and N_c

4.2.2 Countermeasure to Bas Subspace Estimation Attack

Bas subspace estimation attack does not use Westfeld denoising attack. Therefore, AWC is likely not to provide any hint. This subspace estimation is based on the fact that the embedding deeply changes the power of the signal in the secret space. This leaves a clue for disclosing this space.

4.2.2.1 Security Measurement

The main idea of BA is to project the signal composed of the wavelet coefficients onto a secret subspace of dimension $N_v = 256$. From 2000 images of the BA databases [33], the results of the power distributions of the original projected vector $\mathbf{v}_X$ and the watermarked projected vector $\mathbf{v}_Y$ are shown in Figure 4.2. The power P_X in the secret space is uniformly distributed: there is no particular reason why this vector could have more power in any given direction of the secret space. We can measure the power for the original host signal $\mathbf{v}_X$ in the secret space by:

$$P_X = \frac{1}{N_v}\mathbb{E}(c_X(1)^2 + c_X(2)^2) \tag{4.4}$$

Yet, the power P_Y of $\mathbf{v}_Y$ is very different: The embedding process has changed the power distributions. In order to maximize the robustness, we push the watermarked vector inside the detection region as deep as possible. This operation inevitably increases the power along the secret cone direction, and decreases the power of the other directions.

We model the power distribution as follows: The embedder selects one secret cone among N_c ones with a uniform probability $p_s = 1/N_c$. Once a given secret cone is selected, the power is $P_s = \mathbb{E}(c_Y(1)^2)$; otherwise, the power is $P_n = \mathbb{E}(\frac{c_Y(2)^2}{N_v-1})$, because the $N_v - 1$ elements share the energy of $c_Y(2)^2$. See notations in the paper of Furon and Bas [33]. Thus, for the first N_c components of $\mathbf{v}_Y$, the power (expectation of the energy per component) $P_Y(k)$ is:

$$\begin{aligned} P_Y(k) &= p_s.P_s + (1-p_s).P_n \\ &= \tfrac{1}{N_c}\mathbb{E}(c_Y(1)^2) + \tfrac{N_c-1}{N_c(N_v-1)}\mathbb{E}(c_Y(2)^2) \end{aligned} \tag{4.5}$$

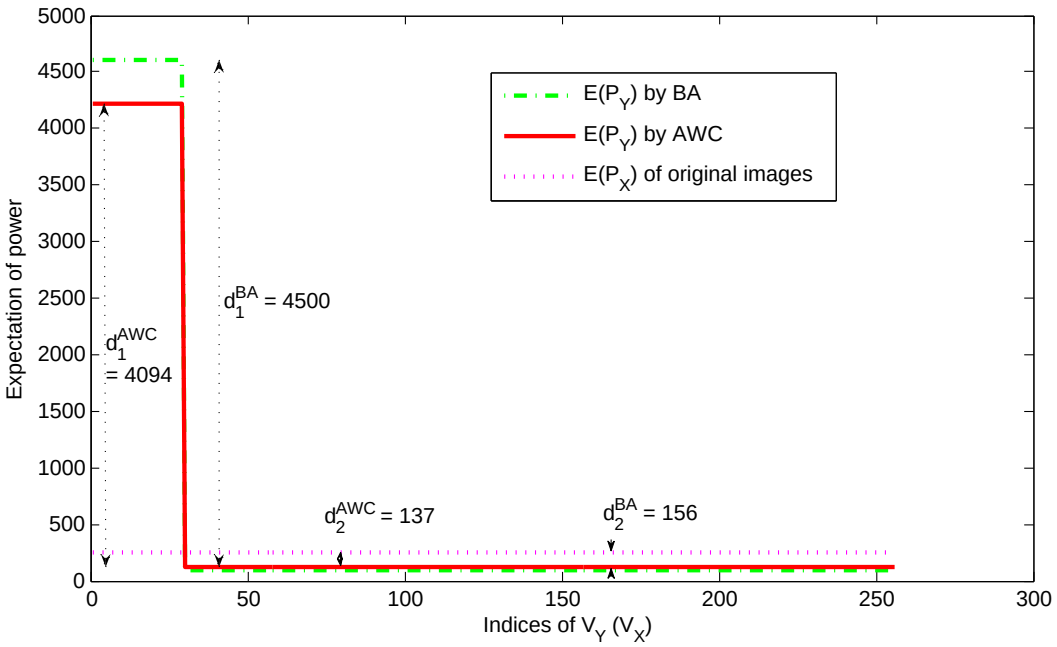

Figure 4.2: Power distribution of the correlation vectors $\mathbf{v}_Y(\mathbf{v}_X)$ with BA and AWC proportional embeddings (with N_v=256 and N_c=30).

The $(N_v - N_c)$ remaining directions of the secret space are not secret cone support. The expection of the power $P_Y(k)$ for $N_c < k \leq N_v$ is:

$$P_Y(k) = \frac{1}{N_v - 1}\mathbb{E}(c_Y(2)^2) \tag{4.6}$$

The difference of power between a direction selected as a secret cone support and a direction not selected as secret cone support in the watermarked correlation $\mathbf{v}_Y$ can be written as:

$$\begin{aligned} d_1 &= |P_Y(1) - P_Y(N_v)| && (4.7)\\ &= \frac{1}{N_c}(\mathbb{E}(c_Y(1)^2) - \frac{1}{N_v - 1}\mathbb{E}(c_Y(2)^2)) && (4.8) \end{aligned}$$

The difference of power between a direction of the secret space (not a secret cone support) and a direction not in the secret space is denoted d_2:

$$\begin{aligned} d_2 &= |P_X - P_Y(N_v))| && (4.9)\\ &= \frac{1}{N_v}\mathbb{E}((c_X(1)^2 + c_X(2)^2)) - \frac{1}{N_v - 1}\mathbb{E}(c_Y(2)^2) && (4.10) \end{aligned}$$

Bigger values of (d_1, d_2) ease the pirate job in disclosing the secret subspace, and in this subspace, the directions used as secret cone directions. An embedding technique lowering this two values provides a better level against Bas' subspace estimation attack, but it is impossible to achieve the ideal case: $d_1 = 0$ and $d_2 = 0$. We compare the distances for these two embedding techniques: original BA with proportional embedding and BA with AWC embedding. They have almost the same values for d_2; AWC has a distance d_1 just a little bit smaller than the one of the original BA (Figure 4.2). Whereas AWC embedding is a good counterattack against Westfeld denoising attack and clustering attack, it does not help against Bas' subspace estimation attack.

4.2.2.2 Regulated Parameters

Inserting (3.5) in (4.7) and (4.9), we have:

$$\begin{aligned} d_1 &= \pi\frac{\mathbb{E}((c_X(1) + \sqrt{\rho^2 - c_X(2)^2})^2)}{N_c}\\ &+ (1-\pi)\left(\frac{\mathbb{E}((c_X(1) + \rho\sin(\theta))^2)}{N_c} - \frac{\mathbb{E}((c_X(2) - \rho\cos(\theta))^2)}{N_v - 1}\right) && (4.11) \end{aligned}$$

and

$$\begin{aligned} d_2 &= \pi \frac{\mathbb{E}(c_X(1)^2 + c_X(2)^2)}{N_v} \\ &+ (1-\pi)\left(\frac{\mathbb{E}(c_X(1)^2 + c_X(2)^2)}{N_v} - \frac{\mathbb{E}((c_X(2) - \rho\cos(\theta))^2)}{N_v - 1}\right) \end{aligned} \quad (4.12)$$

where π is the probability that $c_X(2) \leq \rho\cos(\theta)$. In these two equations, ρ is a parameter related to the embedding distortion, we can not modify it arbitrarily since we need a high quality watermarked content and an acceptable PSNR. Parameter θ is the angle of the detection cone region; it cannot be modified because it fixes the false detection probability.

Therefore, we can only tune parameters N_v and N_c. The analysis is quite involved since the statistics of $\mathbf{c}_X$, π and θ also depends on these parameters. Firstly, we fix N_c and analyze the impact of N_v. Changing N_v has almost no effect on d_1, whereas d_2 is clearly decreasing with N_v. Thereby, after carefully considering that the complexity of the embedding and detection algorithms are in $O(N_v)$, we choose $N_v = 1024$.

Now we study the last parameter N_c: d_1 is decreasing function with regard to N_c. In this regard, we should choose N_c as big as possible. But, a bigger N_c lowers the value of θ giving birth to a small detection region, and this significantly degrades the robustness of the system. We make a trade-off with $N_c = 256$.

We evaluate the power distribution of the correlation vectors for two embedding techniques with the regulated parameters, Figure 4.3 shows the result. We can see that both the distance d_1 and d_2 are much smaller than before ($\approx 1/8$).

4.2.2.3 Security Evaluations Against Attacks

With the new parameters $N_v = 1024$ and $N_c = 256$, we check the security level of the embedding techniques against these attacks with the same database as in the third episode of BOWS-2 (10,000 images). First of all, we test Westfeld's clustering attack. Figure 4.1 shows the performance of Westfeld's classifier against AWC with N_v=1024 and N_c=256, up to $N_t \leq 40$: the adjusted mutual information AMI is around 0.006, i.e. much smaller than for the case of AWC embedding with $N_v = 256$ and $N_c = 30$. So AWC with the regulated parameters has even a better security level than before.

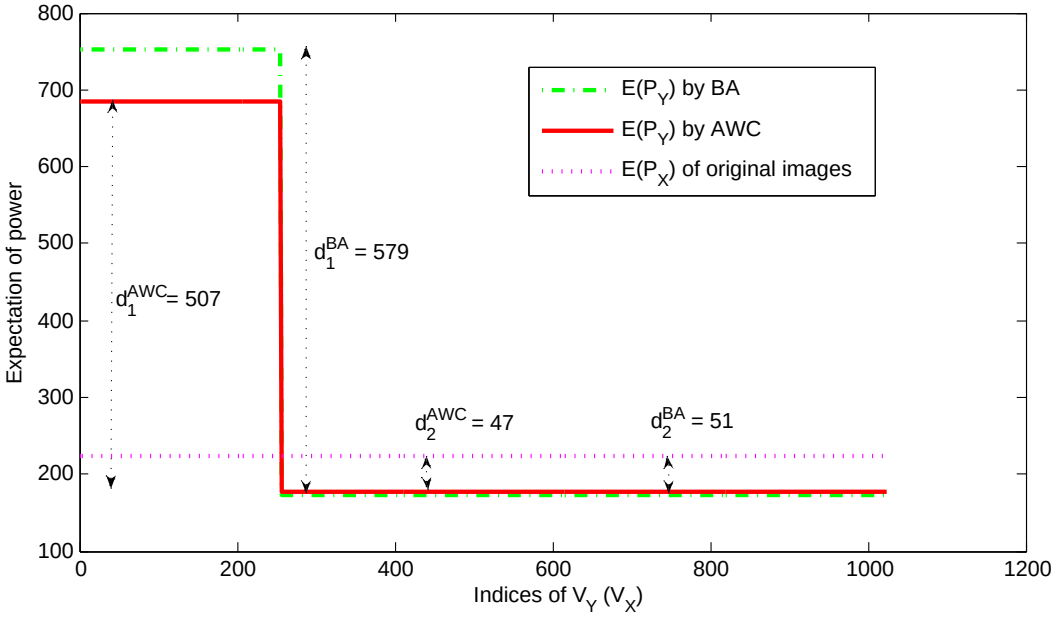

Figure 4.3: Power distribution of the correlation vectors $\mathbf{v}_Y(\mathbf{v}_X)$ with BA and AWC proportional embeddings (with N_v=1024 and N_c=256).

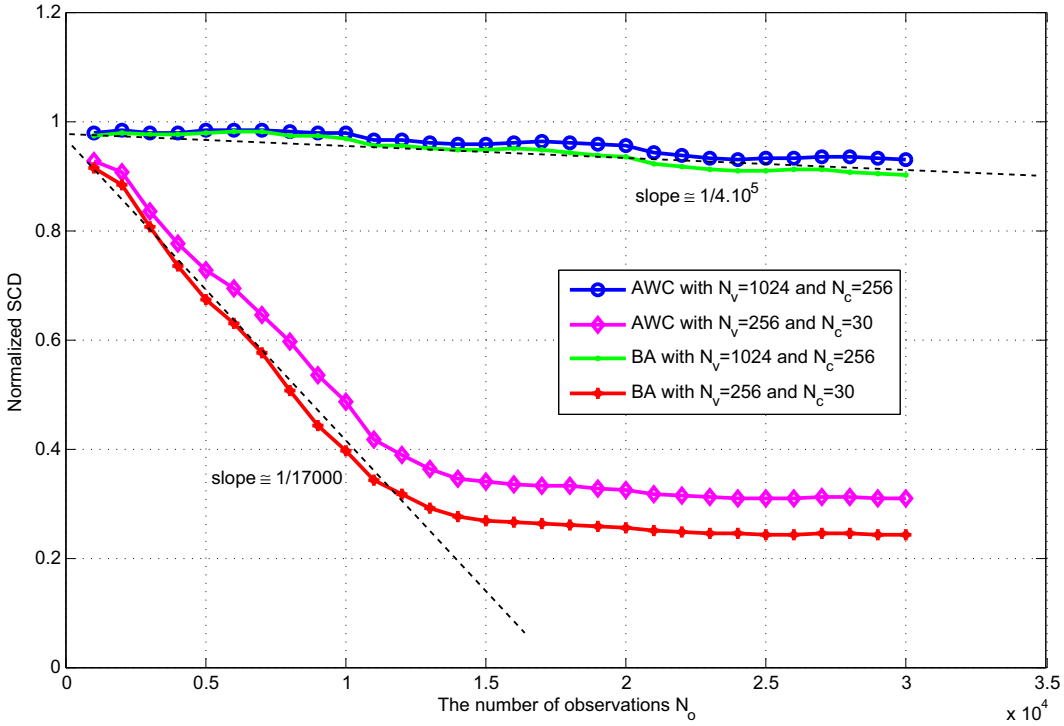

Figure 4.4: Normalized SCD for the embedding techniques BA and AWC with different parameters N_v and N_c

We also evaluate its security performance against Bas' subspace estimation attack. In order to compare the performances of this attack, we define a normalized square chordal distance: $SCD_{norm} = SCD/N_c$. Note that here SCD is the Square Chordal Distance between the secret space and the estimated subspace, and N_c is the number of the secret cone directions in the embedding process. $SCD_{norm} = 0$ means that the estimated space is equal to the secret space; $SCD_{norm} = 1$ means the subspaces are orthogonal and, therefore, the attack has failed. Figure 4.4 shows the results of the OPAST algorithm applied against the original BA and AWC embedding with different parameters. N_o is the number of observations. We can see that, for BA embedding technique with $N_v = 256$ and $N_c = 30$, SCD_{norm} is decreasing with the number of observations, and the estimation keeps on improving very quickly. This confirms Patrick Bas' results [5]. However, for BA and AWC embedding techniques with the regulated parameters $N_v = 1024$ and $N_c = 256$, SCD_{norm} decreases very slowly, even after 3.10^4 observations, SCD_{norm} is always very close to 1 (more than 0.9), this shows that the OPAST algorithm cannot effectively estimate the secret subspace any longer.

4.2.2.4 Robustness Evaluations

To examine the robustness impact brought by the proposed solution in the watermarking layer, we apply the same benchmark as in BA's original paper. Since we have already reviewed this benchmark in Subsection 3.4.1, we do not repeat it here any more. Figure 4.5 reveals the impact of 15 most significant attacks on the two embedding techniques. The probability of detecting the watermark (i.e. number of good detections divided by $2,000$) is plotted with respect to the average PSNR of the attacked images. Because these classical attacks produce almost the same average PSNR, the three points for a given attack are almost vertically aligned.

The probability of detection is slightly decreased when N_v (resp. N_c) increases from 256 (resp. 30) to 1024 (resp. 256) for BA and AWC embeddings. The proposed counterattacks trade a great improvement of security levels against a little bit of robustness. Comparing to the original BA embedding, the AWC

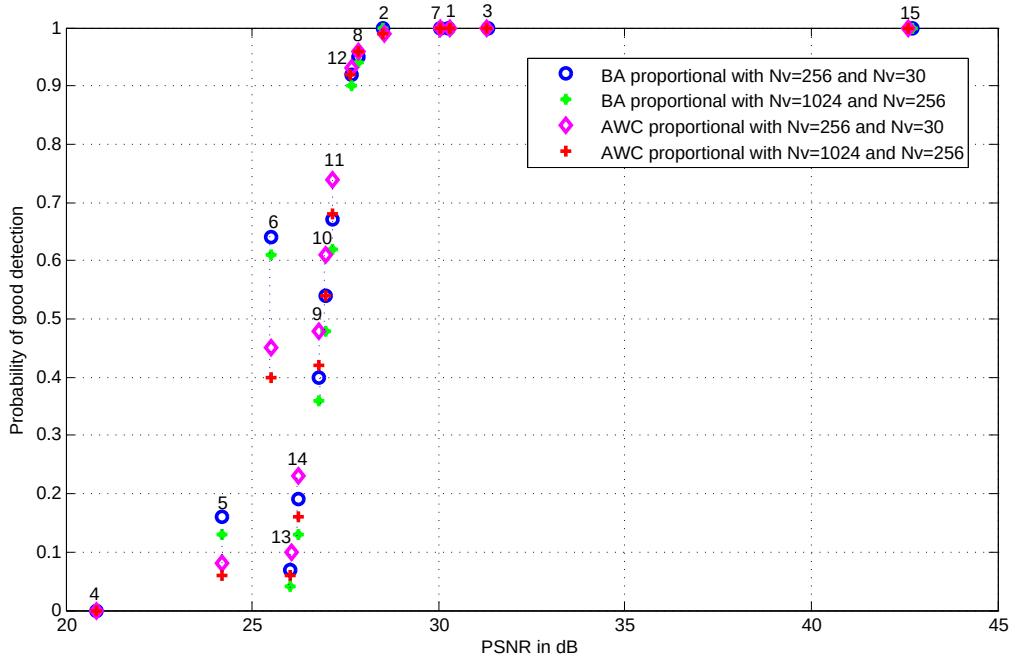

Figure 4.5: Probability of good detection versus average PSNR of the attacked images for the three watermark embedding techniques: 'BA' proportional embedding with N_v=256 and Nc=30 'o', 'BA' proportional embedding with N_v=1024 and Nc=256 '*', 'AWC' proportional embedding with N_v=256 and Nc=30 '◊', 'AWC' proportional embedding with N_v=1024 and Nc=256 '+'.

embedding is more robust against attacks 9-14, but less robust against attacks 5 and 6; and comparable for attacks 1-4, 7, and 15.

4.2.3 Extension to On-Off Keying

So far, we have discussed about BA as a zero-bit watermarking technique and independently of any particular scenario. In the multimedia fingerprinting application, BA should be used in conjunction with an anti-collusion code such as q-ary Tardos code. Therefore, the secret subspace is decomposed of q complementary spaces: each one of them gathers the secret cone directions associated to one symbol. Therefore, we force $N_v = q.N_c$. Since the symbols to be embedded are uniformly distributed, all the directions of the secret subspace have the same probability to serve as a secret cone support. This cancels the use of distance d_1 (or in other words, this automatically sets $d_1 = 0$). Another advantage of this solution is that it also reduces d_2.

In order to verify our arguments, we use 2000 images (as in BA [33]) to evaluate distance d_2 in the multimedia fingerprinting scenario. The PSNR of the watermarked images is controlled around 43 dB. The alphabet size for the fingerprinting codewords is $q = 4$. The reason is that: when we watermark an image with q different symbols, and then perform an averaging attack with a JPEG compression ($Q = 20$) for them, the probability for detecting a watermark from the the watermarked image is small than 50%. So there is no advantage in having q higher than 4 with this technique. We will give a detailed explaination in Section 5.3.1. We also fix the parameters N_v=1024 and N_c=256 for both BA and AWC proportional embeddings, and thereby $N_v = q * N_c$.

In this experiment, we assume that the fingerprinting symbols are uniformly distributed over all the 2000 images. Figure 4.6 shows that, with our proposed solution, d_1 is artificially reduced to 0 in this application. This is a significant progress, since in a pure zero-bit scenario d_1 has a huge value for both embedding methods ($d_1^{AWC} \approx 4100$ and $d_1^{BA} \approx 4500$, see Figure 4.2). On the other hand, d_2 is also slightly decreased: $d_2^{AWC} = 110$ and $d_2^{BA} = 124$ (before $d_2^{AWC} = 137$ and $d_2^{BA} = 156$, see Figure 4.2).

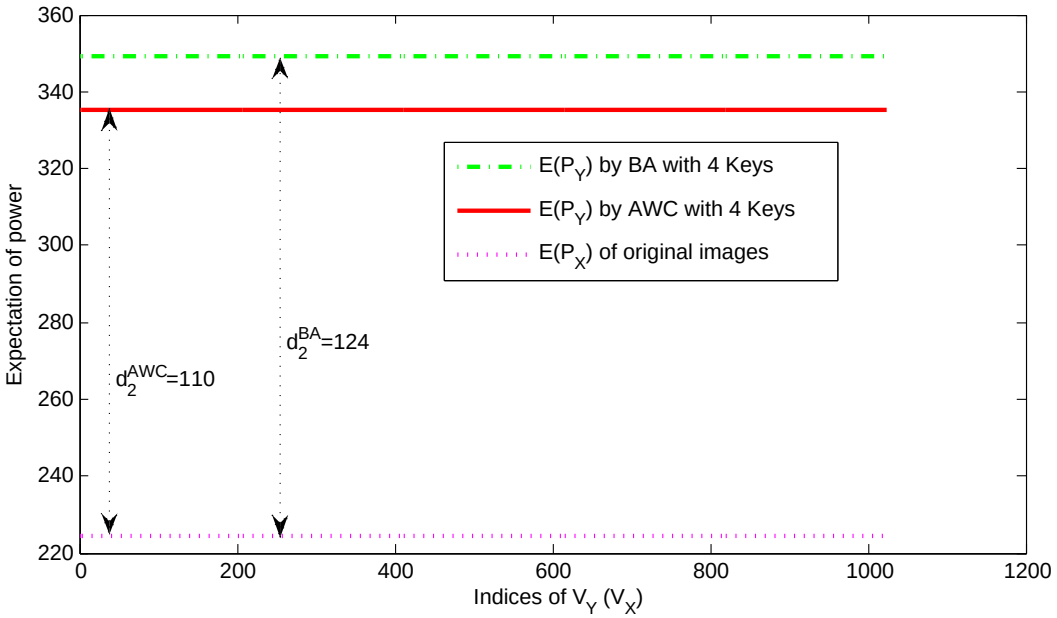

Figure 4.6: Power distribution of the correlation vectors $\mathbf{v}_Y(\mathbf{v}_X)$ with BA and AWC proportional embeddings extended to multi-bits (with N_v=1024 and N_c=256).

4.2.4 Result Discussion

We proposed counterattacks to known attacks against the BA watermarking technique. Thanks to a conjunction of the AWC embedding, the regulated parameters N_v and N_c plus the conditions of use in the traitor tracing scenario, the distribution of the signal power is much more uniform than before, and this is sufficient to ruin the above mentioned attacks. The cost of a better security levels is a small loss in robustness compared to the original BA technique, and slower embedding and detection algorithms (by a factor of 4).

However, the assessment of a higher security level is not completed: we addressed only some known attacks, worse threats certainly still exist. It might be possible for the pirate to use more powerful implementations of PCA than OPAST, and to collect much more watermarked images to disclose the secret space. Moreover, the main counterattack simply suggests to use a 'bigger' secret (a 'longer secret key' would say a cryptographer), which is not a new idea. In the next section, our work will try to design a watermarking technique which has a better security levels.

4.3 Novel Robust and Secure Watermarking

In this part, we present yet another attempt towards robust and secure watermarking. We start from the above mentioned watermarking technique, and propose changes in order to strengthen the scheme against attacks based on second order statistics such as PCA, by striking a perfectly even distribution of the power. These changes include the introduction of a security criterion, an embedding process implemented as a maximization of a robustness metric under the perceptual and the security constraints, and a watermarking detection seen as a contrario decision test. To the best of our knowledge, this is the first time that watermarking security is enforced right into the embedding algorithm. We will give a detailed description of this robust and secure watermarking technique in the following subsections.

4.3.1 A Contrario Decision

As far as we know, there is no known optimum zero-bit watermarking technique for multimedia contents. This is mostly due to the lack of stationarity and to the wide variety of distribution from a content to another. From the theoretical viewpoint, [84] proposes a unifying theory of zero-bit watermarking, but its main drawback is its lack of universality: the embedder and detector must know the statistical distribution of the host content. More recently, Comesaña et al. have found the optimum scheme under the restrictive assumption that the host distribution belongs to the white and Gaussian family [85], whatever its variance. To apply this theoretical result into a real application, BA [33] projects many wavelet coefficients $\mathbf{s}_X \in \mathbb{R}^{N_s}$ of the image (with $N_s > 200,000$) into a secret (i.e. pseudo-randomly generated) subspace $\mathcal{S} = \text{Span}(\mathbf{S}_C)$ of very low dimension ($N_v = 256$). This makes the projection vector $\mathbf{v}_X = \mathbf{S}_C^T \mathbf{s}_X \in \mathbb{R}^{N_v}$ almost white and Gaussian distributed. In the end, we can write that $\mathbf{v}_X \sim \mathcal{N}(\mathbf{0}, \sigma_X^2 \mathbf{I}_{N_v})$, where $\sigma_X^2 = ||\mathbf{s}_X||^2$, i.e. the variance varies from a content to another. Several tweaks are also deployed to tackle a perceptual model and to improve the robustness against common image processing (see [33]).

However, as already mentioned in the last two sections, BA is robust but not secure. The embedding pushes the vector $\mathbf{v}_X$ deep into the acceptance region which is an hypercone, focusing most of the embedding energy along its directions. This brings in an uneven distribution of the watermarked signals power in the space, and a PCA algorithm can disclose the directions of the hypercone.

We propose here a very different paradigm, inspired by works in computer vision proposing a partial gestalt system based on the Helmholtz principle [86]. This principle groups a set of observed objects into one class if it is very unlikely that random could have generated such configuration. This is also known in statistics as *a contrario detection*. The detection decides one of the two hypotheses H_0 or H_1 based on some observations $\mathbf{v}$. A statistical model is assumed under H_0 as a distribution p_{H_0}, but the alternative H_1 is far too broad to support such a model, and/or what happens under H_1 is not well known. Therefore, the detector evaluates the probability to observe $\mathbf{v}$ under p_{H_0}, and if this event is unlikely, it decides for H_1.

Here we exactly consider the same idea: for original natural images, we assume that the projection vector $\mathbf{v}_R$ of the received image is white Gaussian distributed: $p_{H_0} \propto \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{N_v})$. Watermark detection is triggered on when this vector is not typical from that distribution. Therefore, embedding amounts to render $\mathbf{v}_X$ as much as possible (i.e. while fulfilling the constraints of embedding distortion and security) not typical with regard to p_{H_0}. This method is quite different from usual approaches where embedding transforms host vector $\mathbf{v}_X \sim p_{H_0}$ into a watermarked vector $\mathbf{v}_Y \sim p_{H_1}$ and where the detector measures the likelihood ratio $p_{H_1}(\mathbf{v}_R)/p_{H_0}(\mathbf{v}_R)$ (or its derivative for a LMP test) to decide whether $\mathbf{v}_R$ is watermarked or not.

4.3.2 The Embedding Core Process

We detail in this subsection how the concepts introduced in the above sections are implemented. We postpone the details concerning image processing to the next subsection. We assume here that we have extracted a Gaussian distributed vector $\mathbf{v}_X$ from the host content, and that a targeted PSNR controls the watermark embedding.

4.3.2.1 Three Functions

As mentioned above, we have to manage three criteria during the embedding: the perceptual quality of the watermarked image, the global security of the scheme, and the Gaussianity of the signals.

Quality Function: we assume the targeted PSNR imposes a constraint on the Euclidean distance of the watermark signal $\mathbf{v}_W = \mathbf{v}_Y - \mathbf{v}_X$, such that $\|\mathbf{v}_W\| < \rho$. We will see later that parameter ρ is not only a function of the targeted PSNR but also on some statistics of the host image. For this reason, it might vary from a content to another, even if the required PSNR is fixed. We define the following function:

$$c_Q(\mathbf{v}_W) = \frac{\|\mathbf{v}_W\| - \rho^2}{\rho^2}. \tag{4.13}$$

Security Function: as mentioned in the Section 4.2, a counter-attack to the OPAST threat (and any algorithm implementing a PCA) is to make sure that, on average, the power of the watermarked signal in the secret space $\mathcal{S}$ is equal to the

one in the complementary space $\mathcal{S}^{\perp}$: $P_Y(\mathcal{S}) = P_Y(\mathcal{S}^{\perp})$. We assume that for natural images, the power of the host signal is evenly distributed: $P_X(\mathcal{S}) = P_X(\mathcal{S}^{\perp})$. Since the embedding only modifies the signal in $\mathcal{S}$, we need to enforce a conservation of the power in this subspace: $P_Y(\mathcal{S}) = P_X(\mathcal{S})$.

This argument holds on average: $P_X(\mathcal{S})$ is the average power present in $\mathcal{S}$ over a very large collection of natural images. However, we have noticed that the energy of the projection onto $\mathcal{S}$, namely $\|\mathbf{v}_X\|$, greatly varies from one image to another, and it might be hazardous to bet on any average value. Therefore, we enforce a stricter rule: $\|\mathbf{v}_Y\| = \|\mathbf{v}_X\|$. If the energy is conserved for any host image, then it must be true on average. We define the following function:

$$c_S(\mathbf{v}_W, \mathbf{v}_X) = \frac{\|\mathbf{v}_X + \mathbf{v}_W\| - \|\mathbf{v}_X\|}{\|\mathbf{v}_X\|}. \tag{4.14}$$

Gaussianity Function: there are plenty of tests to decide whether a collection of i.i.d. data has been drawn from a given probability distribution. For continuous distributions and especially the Gaussian distribution, the Anderson-Darling test is one of the most famous. In brief, the test computes the statistic $f(\mathbf{v})$ from the samples $\mathbf{v} = \{v(i)\}_{i=1}^{N_v}$ as follows:

1. Mean μ and the standard deviation σ are estimated from the samples.
2. Samples are normalized: $v^n(i) = (v(i) - \mu)/\sigma$.
3. Samples are then sorted in increasing order to get $(v^s(1), \ldots, v^s(N_v))$.
4. Compute $f(\mathbf{v}) = (-N_v - T)(1 + \frac{4}{N} - \frac{25}{N^2})$.
5. If $f(\mathbf{v}) > \alpha$, then the samples are not Gaussian distributed.

with

$$\begin{aligned} T = N_v^{-1} \sum_{i=1}^{N_v} & \quad [(2i-1)\ln \Phi(v^s(i)) \\ + & \quad (2(N-i)+1)\ln\left(1 - \Phi(v^s(i))\right)]. \end{aligned} \tag{4.15}$$

The value of α depends on the level of the test: the bigger is α, the smaller is the level. We give some critical values α for the normal distribution: 0.632 for a

0.1 level, 0.751 for a 0.05 level, 0.870 for a 0.025 level, and 1.029 for a 0.01 level (see [87, Table 1, part (a), page 239]). For instance, if $f(\mathbf{v}) > 1.029$, the data are deemed non Gaussian, and the probability of being wrong is 0.01. We take $f(\mathbf{v})$ as the detection score for the vector $\mathbf{v}$ extracted from the received image. The image is declared as watermarked if $f(\mathbf{v}) > \alpha$ and the probability of false alarm is indeed the level of the test corresponding to this threshold.

4.3.2.2 Constrained Optimization

We consider the watermark embedding as a maximization under constraints. The embedding looks for the watermark vector $\mathbf{v}_W^\star$ which maximizes the objective function $f(\mathbf{v}_X + \mathbf{v}_W)$ under the constraints that $c_Q(\mathbf{v}_W) \leq 0$ and $c_S(\mathbf{v}_W, \mathbf{v}_X) = 0$:

$$\mathbf{v}_W^\star = \arg \max_{\mathbf{v}_W \in \mathbb{R}^{N_v} : c_Q(\mathbf{v}_W) \leq 0, c_S(\mathbf{v}_W, \mathbf{v}_X) = 0} f(\mathbf{v}_X + \mathbf{v}_W) \tag{4.16}$$

However, we can not give a concrete description for the distribution of the watermarked vector $\mathbf{v}_Y = \mathbf{v}_X + \mathbf{v}_W$, because this mainly depends on the original image, but we know the fact that the watermarked vector does not follow the Gaussian distribution.

Necessary Conditions: so far, the constraints can be trivially fulfilled by setting $\mathbf{v}_W = \mathbf{0}$. Therefore, we are sure to maximize an objective function over a non empty set. However, since watermarking is always a matter of trade-off between distortion and robustness, we would like to consume all the allowed distortion to maximize our chance of being robust. In other words, we would like to replace the inequality by the equality $c_Q(\mathbf{v}_W) = 0$. There is a necessary condition so that both equality constraints can be satisfied. The constraint on quality describes a hypersphere of radius ρ centered on $\mathbf{v}_X$, whereas the constraint on security defines a hypersphere of radius $\|\mathbf{v}_X\|$ centered on $\mathbf{0}$. Both constraints can be fulfilled if the intersection of those two regions is not empty. This holds if the necessary condition is true:

$$\rho/2 \leq \|\mathbf{v}_X\|. \tag{4.17}$$

The equality holds in this equation when the hyperspheres are tangent in a point $\mathbf{v}_W = -2\mathbf{v}_X$ so that $\mathbf{v}_Y = -\mathbf{v}_X$. Since $\mathbf{v}_X$ is assumed to be Gaussian distributed,

so is $\mathbf{v}_Y$, and consequently an embedding restricted to consume all the distortion budget fails in this case.

To avoid this situation, we need to properly design the technique so that over a vast majority of images, Inequality (4.17) holds. However, it is clear that some pictures will not be watermarked, such as a uniform image. This is indeed quite sound. Watermarking content when the host power is too weak raises a security flaw as the host is not properly hiding the watermarking signal. Since $v_X(i) \sim \mathcal{N}(0, \overline{\Sigma^2})$, with $\overline{\Sigma^2}$ the average power of the wavelet coefficients (see [33, Sec. 3.3]), then $\mathbb{E}[\|\mathbf{v}_X\|^2] = N_v\overline{\Sigma^2}$. This shows that the dimension reduction operated by the projection from $\mathbb{R}^{N_s}$ to $\mathbb{R}^{N_v}$ must not be too strong. This is the reason why we increase N_v from 256 (as set in the original BA technique) to 1024. There is clearly a trade-off with the complexity of the embedder and detector.

For some rare images, this precaution is not enough and (4.17) does not hold. We then reduce the embedding distortion to 90% of the maximum $2\|\mathbf{v}_X\|$ and we stay with an equality quality constraint. Therefore, we hope that the maximization is done over a large enough set, and with a large enough embedding distortion budget in order to find a big and robust extremum of f.

Numerical Algorithm: we use the Matlab implementation of the 'Interior-Point Algorithm' to solve this maximization under constraints. This program can tackle large scale problems and is robust, as it can recover from 'NaN' or 'Inf' results. It also takes benefit from 'user-supplied' derivatives, and Hessian of the objective and the constraints functions. An important point is that these functions are not convex; therefore, there are *a priori* local maxima. When starting from different initial points, the algorithm might end at different local maxima. Here is a way to find a suitable initial point:

1. Define the constants
$$\alpha = 1 - \frac{\rho^2}{2\|\mathbf{v}_X\|^2},\ \beta = \|\mathbf{v}_X\|\sqrt{(1-\alpha^2)}$$
2. Randomly draw a vector $\mathbf{n} \in \mathbb{R}^{N_v}$.
3. Compute $\mathbf{n}' = \mathbf{n} - \frac{\mathbf{n}^T\mathbf{v}_X}{\|\mathbf{v}_X\|^2}\mathbf{v}_X$
4. Set $\mathbf{v}_W^{(0)} = (\alpha - 1)\mathbf{v}_X + \beta\frac{\mathbf{n}'}{\|\mathbf{n}'\|}$.

It is easy to see that $c_Q(\mathbf{v}_W^{(0)}) = c_S(\mathbf{v}_W^{(0)}, \mathbf{v}_X) = 0$. Since the creation of such an initial vector is not a computational burden, we generate plenty of them, we compute their scores with the function f, and we give Matlab the one with the biggest score as an initial vector.

4.3.3 Plugging "Broken Arrows"

This section focuses on the embedding process and how the above algorithm is plugged into BA still image watermarking technique. Actually, the spaces conversions of the watermarking embedding process are almost identical as we have described in Section 3.2.2 and 3.2.1, however, we do not need the last spaces conversion step, which concerns the secret subspace to the MCB plane. Another big difference is the generated watermark signal and the AWC mask. Thereby, here we give a brief review for the main steps.

1. The discrete wavelet transform of the original image is computed. All the wavelet coefficients except the LL subband are stored in a vector $\mathbf{s}_X$. Its projection onto the secret subspace is $\mathbf{v}_X = \mathbf{S}_C^T \mathbf{s}_X$. This matrix is composed of N_v binary carriers of $\{+1, -1\}^{N_s}$, normalized by a factor of $1/\sqrt{N_s}$.

2. From the input $\mathbf{v}_X$, the optimization algorithm mentioned in the last section generates a watermark vector $\mathbf{v}_W$ of norm ρ.

3. Finally, the watermark signal $\mathbf{v}_W$ is mapped back into the wavelet domain via the AWC mask, to reconstruct the final watermarked image. In Section 3.2.2, we have given a detail description on the embedding process with a perceptual mask, and explained the impacts brought by the mask, here we do not repeat it any more.

Except these important impacts of the perceptual mask that we have taken into account in the embedding process, we will introduce two other improvements to further improve the embedding accuracy. In fact, a big difference with BA is that the score is calculated from a vector of big dimension N_v ($f : \mathbb{R}^{N_v} \to \mathbb{R}$), whereas in BA this vector was projected again on a 2D space before the score was computed ($\mathbb{R}^2 \to \mathbb{R}^+$). Figure 8 in [33] shows that there is some inaccuracy in the

embedding: in this 2D space, the embedding targets a given location, and at the detection side, the watermarked image is projected on a different position nearby. This inaccuracy is due to the approximations we made so far, and we noticed that its impact is even bigger with our proposed scheme, certainly because the score is now computed on a much higher dimension space. We propose here to reduce this inaccuracy.

4.3.3.1 Orthonormal Matrix

As discussed in the last Subsection 3.2.2.2, Relation (3.16) is obtained thanks to the three listed assumptions, and it is quite easy to get rid off the last one. As already mentioned, the generation of the secret matrix $\mathbf{S}_C$ is very fast, but this matrix is not exactly orthonormal. $\mathbf{S}_C^T\mathbf{S}_C$ is positive definite, and is thus diagonalizable as $\mathbf{V}^T\Lambda\mathbf{V}$ with $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}_{N_v}$ and Λ a diagonal matrix. Let us denote by $\mathbf{C}$ the square root of the inverse matrix: $\mathbf{C} = \mathbf{V}^T\Lambda^{-1/2}\mathbf{V}$. For a given secret key, we compute in advance this $N_v \times N_v$ matrix. Finally, we modify the projection steps as follows:

$$\mathbf{v} = \mathbf{C}\mathbf{S}_C^T\mathbf{s}, \tag{4.18}$$

$$\mathbf{s} = \mathbf{S}_C\mathbf{C}\mathbf{v}. \tag{4.19}$$

This renders our scheme more accurate for two reasons:

- Now we have exactly $\|\mathbf{s}_W\|^2 = \mathbf{v}_W^T\mathbf{C}^T\mathbf{S}_C^T\mathbf{S}_C\mathbf{C}\mathbf{v}_W = \mathbf{v}_W^T\Lambda^{-1/2}\Lambda\Lambda^{-1/2}\mathbf{v}_W = \|\mathbf{v}_W\|^2$ instead of the Approximation (3.11).
- If Assumptions i) and ii) of the last Subsection 3.2.2.2 hold, then at the detection side we would get $\mathbf{v}_Y = \mathbf{v}_X + \overline{M}\mathbf{v}_W$.

The storage of the pre-computed matrix $\mathbf{C}$ however needs 4 MB for $N_v = 1024$.

4.3.3.2 Iterative Embedding

The remaining assumptions i) and ii) of the last Subsection 3.2.2.2 do not exactly hold in practice, and even with (3.17) and (3.18), we get $\mathbf{v}_Y^{(1)} = \mathbf{v}_Y + \epsilon$ with $\|\epsilon\| \ll \|\mathbf{v}_Y\|$. We propose the following iterative embedding sketched in Figure 4.7 to combat this effect.

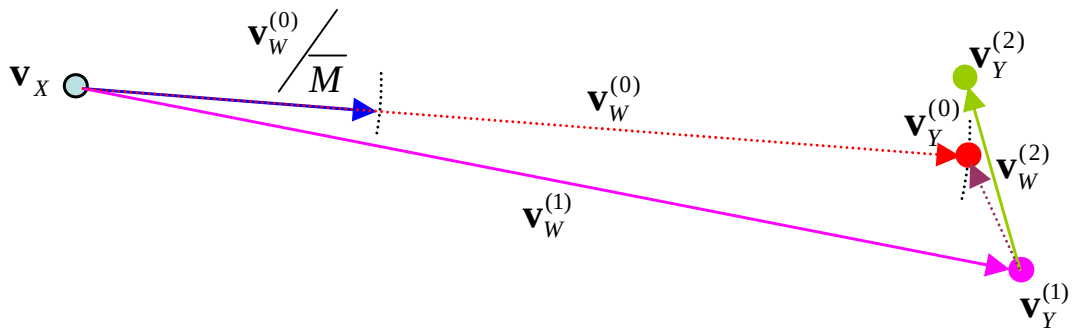

Figure 4.7: The iterative projection scheme

1. Initialization: Once the wavelet transform completed and the extracted coefficients stored in $\mathbf{s}_X$, compute $\mathbf{v}_X$ by (4.18) and ρ by (3.17). Denote $\mathbf{v}_W^{(0)}$ the result of the maximization under constraints (4.16) and $\mathbf{v}_Y^{(0)} = \mathbf{v}_X + \mathbf{v}_W^{(0)}$. Compute $\mathbf{s}_W^{(0)}$ (4.19) and embed it in the host signal (3.12) to get $\mathbf{s}_Y^{(1)}$.

2. Iteration K: From $\mathbf{s}_Y^{(K)}$, project back onto the secret space (4.18). This gives $\mathbf{v}_Y^{(K)}$. Compute $\mathbf{v}_W^{(K)} = \mathbf{v}_W^{(K-1)} + (\mathbf{v}_Y^{(K)} - \mathbf{v}_Y^{(0)})$. Compute $\mathbf{s}_W^{(K)}$ (4.19) and embed it the host signal (3.12) to get $\mathbf{s}_Y^{(K+1)}$.

3. Stop: We stop iterating when we are close to the desired point $\mathbf{v}_Y^{(0)}$, i.e. $\|\mathbf{v}_Y^{(K)} - \mathbf{v}_Y^{(0)}\| < \eta$. The watermarked coefficients are copied back in the wavelet domain and the inverse wavelet transform gives the watermarked image $\mathbf{i}_Y$.

This iterative process does take time but, in practice, just one iteration is enough, as it already greatly reduces the inaccuracy of the embedding.

4.3.4 Experimental Results

The perceptual distortion and the security performance of the proposed watermarking system are ensured in the watermark signal generation as we have shown above. So, in this section, we just assess the robustness of the proposed watermarking scheme. First of all, we give the experimental setup.

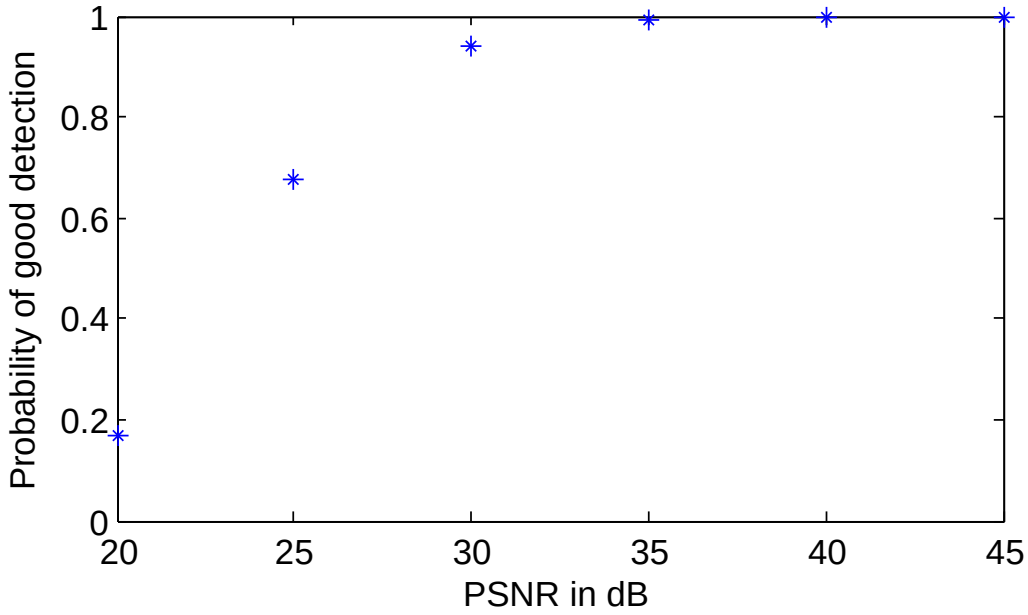

Figure 4.8: The good detection probability with the Gaussian noise attack.

4.3.4.1 Setup

We test 2000 luminance images of size 512×512. These pictures represent natural and urban landscapes, people, or objects, taken with many different cameras from 2 to 5 millions of pixels. A three-level wavelet decomposition is performed for each image, using a Daubechies 9/7 biorthogonal wavelet. Then, the selected wavelet coefficients are projected to the secret matrix, which is generated by the Mersenne Twister pseudorandom number generator seeded by a secret key. The dimension of the secret space is $N_v = 1024$. The embedding distortion is set by a targeted PSNR of 43dB (except in Subsection 4.3.4.2). The number of the tested starting vectors is set to 100. The critical value α is set to 1.029 for a 1% level. The options of the Matlab Interior Point algorithm are set as follows: TolFun=10^{-6}, TolCon=10^{-6}, MaxIter=120, gradConstr = 'on', gradObj = 'on', DerivativeCheck = 'off', FunValCheck = 'on', Hessian = 'user-supplied'.

4.3.4.2 Noise Attacks

To evaluate the robustness against noise attack, we add the noise directly to the watermarked vector $\mathbf{v}_Y$:

$$\mathbf{v}'_Y = \mathbf{v}_Y + \sigma_N \mathbf{n} \tag{4.20}$$

where $\mathbf{n}$ is drawn from a normal distribution, and σ_N is the power of the attack in relation with a $\mathrm{PSNR_a}$ between the attacked image and the watermarked image:

$$\sigma_N = 255\sqrt{\frac{W_\mathbf{i} H_\mathbf{i}}{N_s}}.10^{-\frac{\mathrm{PSNR_a}}{20}}. \tag{4.21}$$

This artificial attack allows to benchmark the key ideas of our proposed watermarking technique :
i) include the security criterion right into the embedding process;
ii) *a contrario decision* test, while decoupling it from this image processing implementation.

To get a better simulation, for each image, we test it with 30 different noise patterns, and compute the average acceptance rate. Figure 4.8 plots the good detection probability against $\mathrm{PSNR_a}$ in dB.

4.3.4.3 Common Attacks

The benchmark of the real still image watermarking technique is the same as in [33]: the attacks are mainly composed of combinations of JPEG and JPEG 2000 compressions at different quality factors, low-pass filtering, wavelet subband erasure, and a simple denoising algorithm. The selected 15 attacks are exactly identical as in the Subsection 3.4.1. However, we use a different probability of false alarm, which is set to 1.10^{-2}. Figure 4.9 shows the average PSNR of the attacked images and the average probability of good detection for these 15 attacks on the proposed watermarking technique. The result of the benchmark is quite similar to the ones shown in [33, Figure 11] or Figure 4.5. In order to give a convenient comparison, we put all the results together. In Figure 4.9, we can see that, in general, the robustness of the proposed watermarking is weaker than the three previous ones. For a given attack, the probability of good detection of the attacked images for the proposed technique is always more or less smaller than

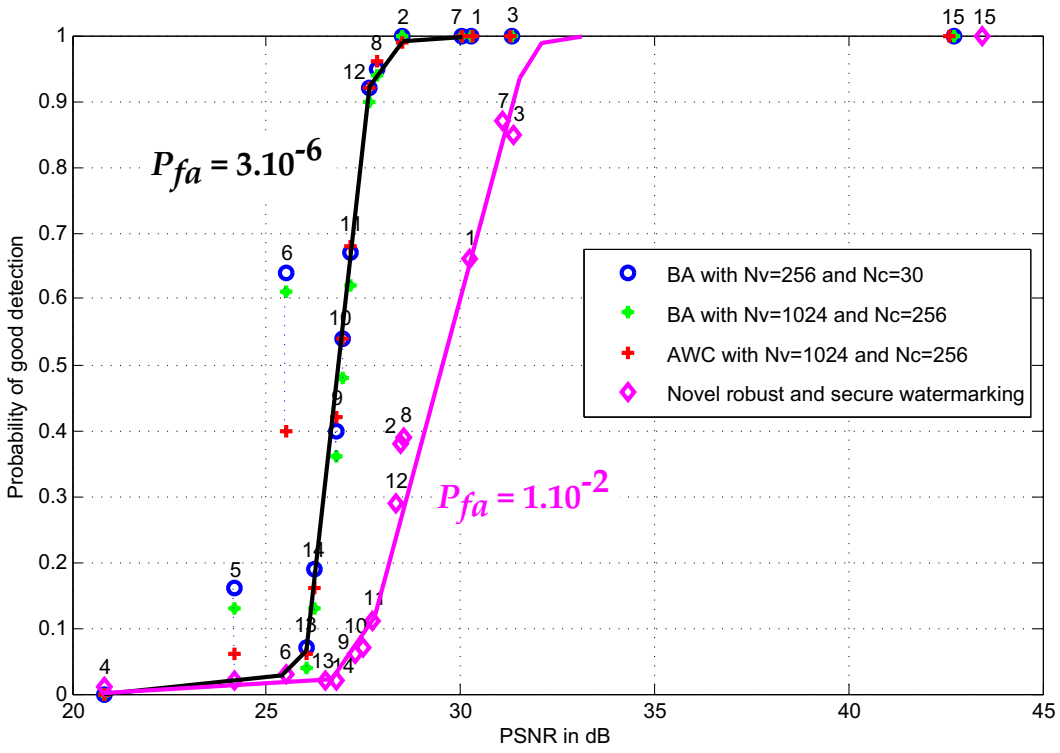

Figure 4.9: Probability of good detection versus average PSNR of the attacked images for the proposed robust and secure watermarking technique and three previous ones.

Attacks	$L = 0$	$L = 1$	$L = 2$
Average	0.0195	0.0255	0.9550
Interleaving	0.0295	0.0275	0.9430
Maximum	0.0215	0.0270	0.9515
Minimum	0.0195	0.0300	0.9505
Uniform	0.0210	0.0285	0.9505

Table 4.1: The probabilities Prob(L) of detecting L watermarks from the attacked images $\mathbf{i}'_Y$.

the ones of three previous techniques. If we want to get the same probability of good detection, the PSNR of the attacked images for the proposed technique must have $2-4$ dB more than the ones of three previous techniques. On the other hand, the Figure 4.9 also shows us that: for the three previous techniques, the probability of good detection decreases very quickly between 26 dB to 28 dB, but for the proposed robust and secure watermarking, the decline interval is larger (between 26 dB to 32 dB), and the decline speed is slower than these previous ones. However, this technique is much weaker than the previous ones because the probability of false alarm here is set to 1.10^{-2} whereas the previous levels were at 3.10^{-6}. This is the price to pay for a good security level.

4.3.4.4 Collusion Attacks

We speak of *collusion attacks* when several copies of the same piece of content, watermarked with different secret keys are mixed to forge an illegal copy. This is the typical scenario of multimedia fingerprinting or traitor tracing: a watermarking technique as the embedding layer coupling with a fingerprinting codes, different message symbols being related to different keys. If we focus on binary anti-collusion codes, as Tardos codes, one bit is embedded in each block of the piece of content. So, each block i exists only in two versions: $\mathbf{i}_{Y1}$ and $\mathbf{i}_{Y2}$. We have tested the following fusion processes at the mixing step of the attack:

1. Average: $i'_Y(i,j) = (i_{Y1}(i,j) + i_{Y2}(i,j))/2$;

2. Interleaving: $i'_Y(i,j) \in \{i_{Y1}(i,j), i_{Y2}(i,j)\}$ with probability $\text{Prob}(i'_Y(i,j) = i_{Y1}(i,j)) = 1/2$;

3. Maximum: $i'_Y(i,j) = \max\{i_{Y1}(i,j), i_{Y2}(i,j)\}$;

4. Minimum: $i'_Y(i,j) = \min\{i_{Y1}(i,j), i_{Y2}(i,j)\}$;

5. Uniform: $i'_Y(i,j)$ is a random vector $\sim \mathcal{U}([i_{Y1}(i,j), i_{Y2}(i,j)])$.

Over a set of 2,000 images, Table 4.1 shows the estimated probability $\text{Prob}(L)$ of detecting $L \in \{0,1,2\}$ watermarks from the attacked images $\mathbf{i}'_Y$. The collusion succeeds in erasing both watermarks only with a maximum rate not larger than 3%. All attacks yield double detection with a big probability (more than 94%), which greatly improves the performances of the tracing algorithm [88]. Note that the high probability of false alarm is not a problem in this application because we are only dealing with the attacked watermarked images and because the anti-collusion fingerprinting code can deal with a small amount of detection errors.

4.4 Chapter Summary

This chapter focuses on the security aspect of the watermarking technique for multimedia fingerprinting application. Based on the obtained result of robustness improvement in the last chapter, we proposed some counterattacks for the used watermarking technique against known security attacks, such as Westfeld clustering attack and Bas subspace estimation attack. We further exploited the security performance of the AWC embedding, regulated the system parameters N_v and N_c, and extended the conditions of use to the traitor tracing scenario. Thanks to the improvements, the proposed watermarking scheme has a better security levels, these security attacks are no longer threats. But the cost is a small loss in robustness compared to the original BA technique, and the computational complexity during the embedding and detection processes is increased by a factor of 4.

However, the assessment of a higher security level is not completed: we addressed only some known attacks, worse threats certainly still exist. Moreover, the main counterattack simply suggests to use a 'bigger' secret, which is not a

new idea. For this reason, we proposed some modifications, and introduced a novel robust and secure watermarking scheme. We presented *a contrario* decision test, a constrained maximization embedding method, which aims at maximizing robustness under the perceptual distortion and security constraints. Despite the introduction of these new conceptions, we face once again the same trade-off between security and robustness: To keep the same performances in terms of good detection against common image processing, we had to increase the probability of false alarm.

Despite its poor trade-off probability of false alarm vs. robustness, we believe that this scheme has some serious potential in the images (or video) watermarking applications, especially in multimedia fingerprinting (a.k.a. traitor tracing), since the contents are all watermarked in this scenario the probability of false alarm is no longer a problem. The question is more about the symbols likely to be hidden in the pirated copy. So far, as far as we know, the watermark detector outputs binary decision about the presence or absence of the watermarks (this includes potential multiple detections). The *a contrario* decision test can indeed provide a probability of the presence of a given symbol; i.e. a soft output bringing more information for the Tardos accusation step.

Chapter 5

Robust and Secure Multimedia Fingerprinting

A complete multimedia fingerprinting system indispensably includes two parts: a fingerprinting codes and a watermarking technique. So far, the designs of these two technologies have often been made separately. Fingerprinting codes have been mostly proposed by the cryptographic community with models of the collusion process defined on the sequence space since the pioneering work [11]. Watermarking techniques are mainly studied by people in the image or signal processing community. Hence, it is crucial to verify that a collusion of watermarked contents is compliant with the assumptions made by fingerprinting designers.

In this chapter, we propose a full multimedia fingerprinting scheme based on *on-off keying* modulation. This scheme includes the symmetric Tardos fingerprinting code combined with a watermarking technique. The watermarking technique could be one of the three watermarking techniques mentioned in the last two chapters. Furthermore, as fusion attacks (described in Secection 2.2.3) are not naturally averted by the watermarking layer nor the fingerprinting layer, we propose two different accusation functions to improve the fingerprinting detection process and tackle them. These proposed solutions force the fusion attacks of the pirates to help the accusation process rather than puzzle it. Some experimental evaluations will be given to confirm our statements.

5.1 A Full Multimedia Fingerprinting Scheme

We propose a full robust and secure multimedia fingerprinting system, which combines a symmetric Tardos fingerprinting code [25] with one watermarking techniques via on-off keying modulation. In detail, it can be divided into four parts: fingerprint construction, fingerprint embedding, fingerprint detection and pirates accusation. We give a more detail description for them in the following subsections.

5.1.1 Fingerprint Construction

A fingerprinting code is a set of n different symbol sequences $\{\mathbf{X}_j\}_{j=1}^{n}$ with size of m. The code has the property that observing a mixture of a bounded number of code sequences, the decoding can retrieve a subset of the original sequences used for this forgery. In our multimedia fingerprinting scheme, we employ the symmetric Tardos fingerprinting code introduced by Skoric et al. [25], the symbols belong to a q-ary discrete alphabet: $X_{ji} \in \mathcal{Q}, \forall j \in \{1, \ldots, n\}$ and $\forall i \in \{1, \ldots, m\}$, with $\mathcal{Q} = \{0, 1, ..., q-1\}$. In detail, we use the Dirichlet distribution function to generate a set of independent random vector $\mathbf{p}_i = (p_i^0, ..., p_i^{q-1})$, according to the Dirichlet distribution shape parameter $\kappa_i = (\kappa_i^0, ..., \kappa_i^{q-1})$, where the components satisfy $p_i^\alpha \in [t/(q-1), 1-t]$ and $\sum_{\alpha=0}^{q-1} p_i^\alpha = 1$, with $t = 1/(300c)$ and c is the collusion size. We note $\bar{\mathbf{p}} = \{\mathbf{p}_i\}_{i=1}^{m}$, the vectors $\mathbf{p}_i$ have the pdf $F(\mathbf{p}_i)$ which is invariant under any permutation over $\mathcal{Q}$, if $\kappa_i = cte$. Thus this construction is symmetric for all symbols $\alpha \in \mathcal{Q}$. In the i-th column of matrix $\mathbf{X}$, random symbols are generated according to $\mathbf{p}_i$ such that $\mathbb{P}(X_{ji} = \alpha) = p_i^\alpha$. The j-th row of matrix $\mathbf{X}$ will be used as the fingerprinting code for the j-th user.

5.1.2 Fingerprint Embedding

5.1.2.1 Block Based Embedding

Each fingerprint sequence identifying a user has to be hidden in his personal copy through a watermarking technique. In our system, the embedding process is block based: it divides the content into consecutive blocks and it hides a symbol per block. We assume here that the content (a video or an audio clip) is long enough

so that there is at least m blocks. The watermarking process starts by extracting a long sequence $\mathbf{s}^{(o)}$ of L coefficients (such as DCT, DFT, DWT coefficient...) from the original content. This sequence is split into m blocks of l samples $\{\mathbf{s}_X^{(i)}\}_{i=1}^m$ (we suppose that $L = ml$), such that $\mathbf{s}_X^{(i)} = (s^{(o)}(il+1), \ldots, s^{(o)}((i+1)l))$. The watermark embedding hides the symbol X_{ji} into the block $\mathbf{s}_X^{(i)}$ producing the i-th watermarked block for the j-th user:

$$\mathbf{s}_{Y,j}^{(i)} = \mathbf{s}_X^{(i)} + \mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)}), \tag{5.1}$$

where $\mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)})$ denotes the embedded watermark signal. Packing back all the watermarked blocks together, this yields the watermarked sequence $\mathbf{s}_{y,j}$, delivered to the j-th user.

This block based embedding has two advantages. First, the blocks of content are watermarked offline in q versions containing a different symbol, the online content server is just a switch that ships the right blocks according to the user sequence. Second, the pirated copy is processed only once by the computationally greedy watermark decoding for retrieving the pirated symbol sequence $\mathbf{Y}'$. Then, the accusation process of the fingerprinting code accuses some users (or nobody) based on this 'pirated' sequence $\mathbf{Y}'$.

5.1.2.2 On-Off Keying Modulation

Among all the attacks the colluders may perform, we distinguish pure watermarking attacks, pure fingerprinting attacks, and fusion attacks (see Section 2.2). The most challenging class in the design of a fingerprinting scheme is the fusion attacks, as general robustness is not a priori sufficient to resist such an attack. Our goal is to choose a suitable combination between the watermarking technique and the symmetric Tardos codes, which could face a fusion attack with several copies of the multimedia content.

In our multimedia fingerprinting system, we choose to use a block based embedding, and to embed one q-ary symbol in each block. We know that zero-bit watermarking is similar to on-off keying in digital communications. This modulation is used on very rare applications: fiber communication where it is not possible to modulate the light emission, except by switching it on and off. Some

theoretical works also show that on-off keying is the last solution to communicate when the channel transmission quality is really too bad (e.g., the delay spread of the fading is less than the symbol duration, so that channel estimation and equalization is not possible) [89]. The use of zero-bit watermarking is not new in multimedia fingerprinting. For instance, Safavi-Naini and Yang embed q-ary symbols in pictures using q different secret keys of a classical spread spectrum scheme [43]. We use some different zero-bit watermarking techniques which are side-informed.

Since a zero-bit watermarking technique is based on on-off keying, the combination between watermarking technique and fingerprinting code of our multimedia fingerprinting system can also be considered as an on-off keying modulation. By nature, a zero-bit technique does not carry any specific message, but just the presence of a watermark. To embed symbols of a q-ary alphabet, we defined q secret keys, and use key $\mathcal{K}(X_{ji})$ to embed symbol X_{ji}. Since two different keys produce two almost independent watermark signals, the fusion is now deemed as a scaling and the addition of an independent noise. Note that this modulation is not costly, for example, the embedding and detection of the AWC watermarking involve three nested spaces, and the modulation is just related to the last one, so, the computational complexity overhead is only related to this last space.

5.1.2.3 Selected Watermarking Techniques

Three watermarking techniques are proposed as a practical watermarking solution in our experimentation: the improved version of Broken Arrows after the robustness enhancement as mentioned in Section 3.3, another improved version of Broken Arrow after the robustness improvement as well as security improvement mentioned in Section 4.2, and the new robust and secure watermarking (RSW) proposed in Section 4.3. However, in order to adapt for on-off keying modulation, we have to adjust the parameter N_c, since we use $q = 4$ different secret keys: $N_c = N_v/q$. The reason why we set $q = 4$ will be given later. We enumerate these three used watermarking techniques as follows:

1. First watermarking technique: the AWC watermarking with $N_v = 256$ and $N_c = 64$;

2. Second watermarking technique: the AWC watermarking with $N_v = 1024$ and $N_c = 256$;

3. Third watermarking technique: the robust and secure watermarking.

These three watermarking schemes have a common property: one specific form of zero-bit watermarking. Actually, these watermarking techniques do not carry any specific message, but just the presence of a watermark (or a symbol).

We know that for the zero-rate watermarking scheme, the watermark signal consists of a random number with zero mean and unit variance. Compared to the zero-rate watermarking, the q possible watermark signals $\{\mathbf{w}(X, \mathbf{s}_X^{(i)})\}_{X \in \mathcal{Q}}$ generated by these zero-bit watermarking techniques are not strictly independent, because they all take advantage of the side-information $\mathbf{s}_X^{(i)}$. But they are less dependent, since they are generated from q independent secret keys. Hence, a fusion attacks is more similar to the scaling of the present watermark and the addition of an independent noise (the watermarks in the others copies in the fusion attack). Another advantage of these zero-bit watermarking is that it is very unlikely to frame the resulting watermark inside the detection region of symbol (i.e., a secret key) which does not belong to $\{X_{ji}\}_{j \in \mathcal{C}}$, with $\mathcal{C}$ is the set of colluders. The rationale is that the colluders cannot succeed to watermark a block without knowing this secret key from signals which are independent from this detection region. Another way to see this is that, assuming the fusion is linear, the forged block remains in an affine space passing by the point $\mathbf{s}_X^{(i)}$ and spanned by the watermark signals $\{\mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)})\}_{j \in \mathcal{C}}$, which is almost orthogonal to the detection region related to the other keys. Hence, this event should be as rare as a false alarm.

There is a difference between the first two watermarking techniques and the last one, with respect to the way to adapt the on-off keying modulation. For the first two watermarking methods, we firstly perform a wavelet transform for a given image; and then project the wavelet coefficients from the wavelet subspace to the secret subspace according to a secret key, this step obtains a secret vector with size N_v; finally, this vector is divided into q sets of $N_c = N_v/q$ components each, according to q different secret keys. Here each secret key $\mathcal{K}(X_{ji})$ is defined by an unique symbol X_{ji} of fingerprinting code. Therefore, everything remains

the same except that the set in use during embedding is given by the symbol X_{ji}. These sets of secret directions are independent, whence all is as if the embedding was done with q different secret keys.

However, for the last watermarking technique, the modulation method is different. For a given image, we first perform a wavelet transform, this step being the same as in the first two watermarking methods; but in the next step, we should use q different secret keys to project the wavelet coefficients to q different secret subspaces, each secret key $\mathcal{K}(X_{ji})$ is also defined by a unique symbol X_{ji} of the fingerprinting code. Of course, these q secret subspaces are independent, since they were projected with q different secret matrices generated by q different secret keys.

5.1.3 Fingerprint Detection

The content server distributes the fingerprinted copies to lots of users, some of them are colluders, they collect their personalized copies and preform one or more attacks to generate a pirated copy for their illegal redistribution purpose. In the experiments of Section 5.3, we will test several typical fusion attacks. Later, the copyright holder finds this pirated copy, and he wants to trace back the identities of these colluders, and then to investigate their relevant legal responsibility. First of all, he should keep the synchronization and divide the pirated content into consecutive blocks, as in the fingerprint embedding process. Then, he has to detect all the symbols hidden in each block. Please note that in our scheme, the original image is not required at the fingerprint detection side (blind detection). Finally, he runs the accusation process to identify the colluders, of course under an allowable small false alarm. In this section, we talk about how to detect the fingerprint symbols in each block for the above three watermarking techniques, the pirates accusation process will be discussed in the next section.

5.1.3.1 AWC Watermarking Detection

Among three watermarking techniques to be tested, two are AWC watermarking. So we firstly detail the AWC watermarking detection process. For an individual separated block of the pirated copy, the detection should output the indices of

the sets which have given a positive output (the signal is inside one of their hypercones). The following steps is necessary to determine whether a given symbol X_{ji} is present or not, according to the secret key $\mathcal{K}(X_{ji})$. Here we take an image as a example.

1. We perform the 2D wavelet transform (Daubechies 9/7, decomposition of 3 levels) to the test block $\mathbf{i}_Y^{\prime(i)}$, then we select the coefficients from all the bands except the low-frequency LL band, and store them into a signal $\mathbf{s}_Y^{\prime(i)}$.

2. We project the $\mathbf{s}_Y^{\prime(i)}$ from the wavelet space to the secret space to obtain the secret vector $\mathbf{v}_Y^{\prime(i)}$. This step is realized in using the secret matrix $\mathbf{S}_C$ which seeded by the same secret key K as in the embedding process.

3. With the help of the secret vector $\mathbf{v}_c^*$, which is generated according to the secret key $\mathcal{K}(X_{ji})$, the host signal vector $\mathbf{v}_Y^{\prime(i)}$ is projected to the MCB plane.

$$\mathbf{v}_1^{\prime(i)} = \mathbf{v}_C^*, \quad \mathbf{v}_2^{\prime(i)} = \frac{\mathbf{v}_Y^{\prime(i)} - \left(\mathbf{v}_Y^{\prime(i)^T}\mathbf{v}_1^{\prime(i)}\right)\mathbf{v}_1^{\prime(i)}}{\left\|\mathbf{v}_Y^{\prime(i)} - \left(\mathbf{v}_Y^{\prime(i)^T}\mathbf{v}_1^{\prime(i)}\right)\mathbf{v}_1^{\prime(i)}\right\|}. \tag{5.2}$$

   The coordinates representing the pirated block are $\mathbf{c}_Y^{\prime(i)} = (c_Y^{\prime(i)}(1), c_Y^{\prime(i)}(2))^T$, with $c_Y^{\prime(i)}(1) = \mathbf{v}_Y^{\prime(i)^T}\mathbf{v}_1^{\prime(i)}$ and $c_Y^{\prime(i)}(2) = \mathbf{v}_Y^{\prime(i)^T}\mathbf{v}_2^{\prime(i)}$.

4. Finally, we consider that the symbol X_{ji} is present in this block if:

$$\frac{\left|(1,0)\cdot\mathbf{c}_Y^{\prime(i)}\right|}{\left\|\mathbf{c}_Y^{\prime(i)}\right\|} = \frac{\left|c_Y^{\prime(i)}(1)\right|}{\left\|\mathbf{c}_Y^{\prime(i)}\right\|} > \cos(\theta). \tag{5.3}$$

   here θ is an angle defined by the parameters N_v, N_c and the probability of false alarm. Actually, the cone of angle θ represents the watermark detection region.

5.1.3.2 RSW Watermarking Detection

Another watermarking technique to be tested is the new robust and secure watermarking (RSW) that we proposed in Section 4.3. For a given pirated block, the detection outputs the indices of the sets in which the secret vector does not

follow the Gaussian distribution. In order to determine whether the symbol X_{ji} is present or not for the given pirated block $\mathbf{i}_Y^{\prime(i)}$, three necessary steps are as follows:

1. We perform the 2D wavelet transform (Daubechies 9/7, decomposition of 3 levels) to the test block $\mathbf{i}_Y^{\prime(i)}$, then we select the coefficients from all the bands except the low-frequency LL band, and store them into a signal $\mathbf{s}_Y^{\prime(i)}$.

2. We project the $\mathbf{s}_Y^{\prime(i)}$ from the wavelet space to the secret space to obtain the secret vector $\mathbf{v}_Y^{\prime(i)}$. This step is realized in using the secret matrix $\mathbf{S}_C$ which seeded by the same secret key $\mathcal{K}(X_{ji})$ as in the embedding process.

3. Finally, we calculate the Anderson-Darling Test result $f(\mathbf{v}_Y^{\prime(i)})$ of the vector $\mathbf{v}_Y^{\prime(i)}$, and then compare it with the critical value β for a given level. If $f(\mathbf{v}_Y^{\prime(i)}) \leq \beta$, we think that this tested vector follows the normal distribution, and thus we decide that the tested image is not watermarked. Otherwise, we consider a watermark is present.

5.1.3.3 Detection Effect Brought by Fusion Attacks

These above detection processes work very well for the normal watermarked blocks, however, for the pirated blocks after fusion attacks mentioned in Secection 2.2.3, their detection efficiency is different because of the additional interference. We take an average attack with c different copies as an example. We note that the watermarked signal is the sum of the original and watermark signal: $\mathbf{s}_Y^{(i)} = \mathbf{s}_X^{(i)} + \mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)})$, and the pirated block after average attack can be written as: $\mathbf{s}_Y^{\prime(i)} = \mathbf{s}_X^{(i)} + c^{-1}\mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)}) + \omega$. This is very different from the pure watermarking layer attacks because of the scaling factor c^{-1} and the noise ω, sum of the other watermark signals, which is not strictly independent of the host or the watermark signals.

At the detection side, the most important thing is that, for a given symbol of colluders, the remaining scaled watermark $c^{-1}\mathbf{w}(X_{ji}, \mathbf{s}_X^{(i)})$ in the secret space of this symbol is sufficient to make the watermark detector output a positive answer. Of course, the detection results is directly upper bounded by the number of symbols q and the collusion size c. The more symbols participated in the

fusion, the smaller is the probability that these symbols are detected. That is why we should give a careful choice for the parameter q in the fingerprinting system design (see Section 5.3.1).

5.1.4 Skoric's Accusation Functions

Now we talk about how to make use of the watermark detection results to accuse the colluders. The number of detection outputs is indeed $2^q > q$, as, for each of the q secret keys, the detector will give a binary decision. Hence there are cases where several watermark signals are detected. At i-th block, a set of symbols $\mathcal{Y}'_i = \{Y'_i(d)\}_{d=1}^{D_i}$ is detected, here D_i represents the number of symbols detected at i-th block. What kind of fingerprinting code can take advantage of this feature? We found in literature the following two candidates.

Many strong c-traceable codes are based on algebraic error correcting codes such as Reed-Solomon codes. This feature allows two strategies: list decoding or iterative decoding. List decoding finds a group of nearest code sequences (from the pirated sequence) [47] beyond the decoding distance, and its algorithm like Guruswami-Sudan [46] takes into account some reliability measures about the decoded symbols, which could be based on the decoded symbols $\mathcal{Y}'_i$. Another strategy is to decode iteratively the pirated sequence to find several colluders. In [45], symbols of the pirate sequence are replaced by erasures when they match symbols of code sequences decoded in previous iterations. This new pirated sequence is again decoded at the next iteration. Here, we can replace erasure by another symbol decoded in the block.

However, the Skoric's original accusation function of the symmetric Tardos fingerprinting code is not suitable for the two above decodings methods, since it is confined to the cases of one decoded symbol for one pirated block, such as, by the exchange attack. We recall the Skoric's accusation functions as shown in Section 2.5.2, which computes the accusation sum for j-th user as follows:

$$S_j = \sum\nolimits_{i=1}^{m} U(Y'_i, X_{ji}, p_i^{Y'_i}) \tag{5.4}$$

with

$$U(Y'_i, X_{ji}, p_i^{Y'_i}) = \delta(Y'_i, X_{ji}) g_1(p_i^{Y'_i}) + (1 - \delta(Y'_i, X_{ji})) g_0(p_i^{Y'_i}) \tag{5.5}$$

where $\delta(Y', X)$ denotes Kronecker delta, and here the accusation weight functions are the same as Equation (2.6):

$$g_1(p_i) = -g_0(1 - p_i) = \sqrt{\frac{1 - p_i}{p_i}} \tag{5.6}$$

Finally, we consider the j-th user is one colluder if his accusation score S_j is greater than the accusation threshold Z. This is indeed the Skoric's accusation method.

Obviously, this accusation function does not make use of the information of all decoded symbols. Hence, in the next section, we propose two new accusation approaches to improve the efficiency of accusation process.

5.2 Our Proposed Accusation Approaches

In this section, our objective is to propose some novel accusation processes, in order to take into account the fact that a list of symbols $\mathcal{Y}'_i = \{Y'_i(d)\}_{d=1}^{D_i}$, are decoded from the i-th block. Note that the i-th watermark detection does not bring any information whether user j is guilty if $D_i = 0$ ($\mathcal{Y}'_i$ is an empty set) or $D_i = q$ ($\mathcal{Y}'_i = \mathcal{Q}$). Now we present two extended accusation processes in the following subsections.

5.2.1 First Accusation Method

In the first method, we propose the following score sum for the j-th user, with U defined in (5.5):

$$S_j = \sum_{i=1}^{m} \sum_{d=1}^{D_i} U(Y'_i(d), X_{ji}, p_i^{Y'_i(d)}). \tag{5.7}$$

In this way, it is equivalent as if the code length has increased from m to $m\bar{D}$, with $\bar{D} = m^{-1} \sum_{i=1}^{m} D_i$. As far as we know, the longer the fingerprinting code is, the more reliable is the accusation process. Therefore, this proposed method should be more reliable than the previous one. However, this rationale justifying the idea is not correct because the summands are not independent, but the experimental

section shows however that it works great. Here we compute the average of the score for an innocent user $\mu_I = m\mathbb{E}(\sum_{Y'(d)\in\mathcal{Y}'} U(Y'(d), X, p^{Y'_i(d)}))$:

$$\begin{aligned}
\mu_I m^{-1} &= \sum_{\mathcal{Y}',X} p^X p^{\mathcal{Y}'} \sum_{Y'(d)\in\mathcal{Y}'} \delta(Y'(d),X) g_1(p^{Y'_i(d)}) + (1-\delta(Y'(d),X)) g_0(p^{Y'_i(d)}) \\
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \sum_{X,Y'(d)\in\mathcal{Y}'} p^X \left(\delta(Y'(d),X) g_1(p^{Y'_i(d)}) + (1-\delta(Y'(d),X)) g_0(p^{Y'_i(d)})\right) \\
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \sum_{X,Y'(d)\in\mathcal{Y}'} p^X \left(\delta(Y'(d),X)\sqrt{\frac{1-p^{Y'_i(d)}}{p^{Y'_i(d)}}} - (1-\delta(Y'(d),X))\sqrt{\frac{p^{Y'_i(d)}}{1-p^{Y'_i(d)}}}\right) \\
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \sum_{Y'(d)\in\mathcal{Y}'} \sqrt{p^{Y'_i(d)}(1-p^{Y'_i(d)})} - \sqrt{p^{Y'_i(d)}(1-p^{Y'_i(d)})} \\
&= 0
\end{aligned}$$

Here $p^{\mathcal{Y}'} = \sum_{d=1}^{D} p^{Y'(d)}$. We remind here that it is impossible to compute the variance of the accusation score, because the number of the detected symbols for each attacked block is indeterminate.

5.2.2 Second Accusation Method

In the second method, we keep the same score sum as Equation (5.4), but change the summands as follows:

$$U(\mathcal{Y}'_i, X_{ji}, p_i^{\mathcal{Y}'_i}) = \delta(\mathcal{Y}'_i, X_{ji}) g_1(p_i^{\mathcal{Y}'_i}) + (1-\delta(\mathcal{Y}'_i, X_{ji})) g_0(p_i^{\mathcal{Y}'_i}) \tag{5.8}$$

with $\delta(\mathcal{Y}'_i, X_{ji}) = 1$ if $X_{ji} \in \mathcal{Y}'_i$, else 0, and $p_i^{\mathcal{Y}'_i} = \sum_{d=1}^{D_i} p_i^{Y'_i(d)}$. The rationale of this method is that it can decrease the variance of the colluders' scores: whatever their symbol $X_{ji} \in \mathcal{Y}'_i$, they receive the same penalization $g_1(p_i^{\mathcal{Y}'_i})$. The experimental results in the next section will show the excellent performances of these two extended accusation methods. Furthermore, we will prove their effectiveness to prevent the colluders from the fusion attacks through the experiment. For this second method, we also compute the average of the score for an innocent user $\mu_I = m\mathbb{E}(\mathcal{Y}', X, p^{\mathcal{Y}'})$:

$$\mu_I m^{-1} = \sum_{\mathcal{Y}',X} p^X p^{\mathcal{Y}'} \left(\delta(\mathcal{Y}',X) g_1(p^{\mathcal{Y}'}) + (1-\delta(\mathcal{Y}',X)) g_0(p^{\mathcal{Y}'})\right)$$

$$
\begin{aligned}
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \sum_{X} p^{X} (\delta(\mathcal{Y}', X) g_1(p^{\mathcal{Y}'}) + (1 - \delta(\mathcal{Y}', X)) g_0(p^{\mathcal{Y}'})) \\
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \sum_{X} p^{X} (\delta(\mathcal{Y}', X) \sqrt{\frac{1 - p^{\mathcal{Y}'}}{p^{\mathcal{Y}'}}} - (1 - \delta(\mathcal{Y}', X)) \sqrt{\frac{p^{\mathcal{Y}'}}{1 - p^{\mathcal{Y}'}}}) \\
&= \sum_{\mathcal{Y}'} p^{\mathcal{Y}'} \left(\sqrt{p^{\mathcal{Y}'}(1 - p^{\mathcal{Y}'})} - \sqrt{p^{\mathcal{Y}'}(1 - p^{\mathcal{Y}'})} \right) \\
&= 0
\end{aligned}
$$

Furthermore, we compute the variance of the score $\sigma_I^2 = m\mathbb{E}(U(\mathcal{Y}', X, p^{\mathcal{Y}'})^2)$, for an innocent user:

$$
\begin{aligned}
\sigma_I^2 m^{-1} &= \sum_{\mathcal{Y}' \notin \{\emptyset, \mathcal{X}\}, X} p^{X} p^{\mathcal{Y}'} \left(\delta(\mathcal{Y}', X) g_1(p^{\mathcal{Y}'})) + (1 - \delta(\mathcal{Y}', X)) g_0(p^{\mathcal{Y}'}) \right)^2 \\
&= \sum_{\mathcal{Y}' \notin \{\emptyset, \mathcal{X}\}, X} p^{\mathcal{Y}'} \left(\sum_{X \in \mathcal{Y}'} p^{X} g_1(p^{\mathcal{Y}'})^2 + \sum_{X \notin \mathcal{Y}'} p^{X} g_0(p^{\mathcal{Y}'})^2 \right) \\
&= \sum_{\mathcal{Y}' \notin \{\emptyset, \mathcal{X}\}} p^{\mathcal{Y}'} (1 - p^{\mathcal{Y}'} + p^{\mathcal{Y}'}) = 1 - p^{\mathcal{Y}'}(\emptyset) - p^{\mathcal{Y}'}(\mathcal{X})
\end{aligned}
$$

5.3 Experimental Evaluations

First of all, the experimental work is dedicated to evaluate the performances of three watermarking techniques; then, we evaluate the performance of the symmetric Tardos fingerprinting code; Finally, we assess the overall performance of the complete multimedia fingerprinting scheme, and evaluate the effectiveness of our two proposed accusation methods.

5.3.1 Evaluations of Three Watermarking Techniques

In order to evaluate the collusion resistance of these three watermarking techniques, we apply four typical collusion attacks mentioned in Section 2.2: 1) Averaging attack; 2) Interleaving attack; 3) Maximum attack; 4) Moderated minority extreme attack (MMX). We have tested 2000 images with size $512 * 512$ the PSNR of the watermarked images is around 43 dB. For each case, the images are watermarked by one among the three given watermarking techniques, and then

Table index	Averaging	Interleaving	Maximum	MMX
AWC ($N_v = 256$ $N_c = 64$)	Table B.1	Table B.2	Table B.3	Table B.4
AWC ($N_v = 1024$ $N_c = 256$)	Table B.5	Table B.6	Table B.7	Table B.8
RSW	Table B.9	Table B.10	Table B.11	Table B.12

Table 5.1: The index of the tables that indicate the conditional probabilities $P(D|\ell)$ of detecting D watermarks from the attacked image. The images are watermarked by one among watermarking techniques, and then attacked by one of four fusion attacks, finally followed by a JPEG compression with a quality factor $Q = 20$.

attacked by one of the four given fusion attacks with ℓ different fingerprinted images, followed by a JPEG compression with a quality factor $Q = 20$. Finally, we compute the probability $P(D|\ell)$ of detecting D different watermarks from the final attacked image. Table 5.1 shows the indices of the following 12 tables (Table B.1 - Table B.12 in Appendix B), which provide the results of the conditional probabilities $P(D|\ell)$.

For the first watermarking technique AWC ($N_v = 256$, $N_c = 64$), (Table B.1 - Table B.4), we can see that, when the collusion size ℓ is small (1 or 2), these four attacks have a similar performance, the detection probability is quite good (~ 0.98). However, when mixing more than 2 watermarked images, their performances become worse for all these attacks, because the strength of one watermark becomes smaller as more images are mixed. For $\ell = 4$, more than half of the time, we are not able detect any watermark from the images attacked by these attacks. Especially, for averaging attack, interleaving attack and maximum attack, the probabilities of detecting no watermark are more that 0.73 when $\ell = 4$. The maximum number of averaged watermarked images being $\min(q, c)$, there is no point in having q higher than 4. This is just the reason for setting $q = 4$. If we compare these four attacks, we can see that maximum attack is the most powerful when $\ell = 3$ and $\ell = 4$; while moderated minority extreme attack has the worst attack capability.

For the second watermarking technique AWC ($N_v = 1024$, $N_c = 256$), (Table B.5 - Table B.8), we can see that, when the collusion size ℓ is small (1 or 2),

the detection probability is very good (~ 0.99), even better than for the first watermarking technique. However, when mixing more than 2 watermarked images, the detection probabilities become worse like for the last watermarking technique. For $\ell = 4$, more than half of the time, we are not able to detect any watermark from the attacked images, except from the ones that were subjected to the moderated minority extreme attack. These results are similar to the one of the first watermarking technique, but much better. In brief, the fusion attack resistance of this second watermarking technique is better than the one of the first watermarking technique, but the cost to pay is a larger computational complexity, due to the increases of N_v and N_c.

The results for the proposed robust and secure watermarking technique are quite different from the ones of the two above watermarking techniques. They are provided in Tables B.9 to B.12. Firstly, when ℓ is small (1 or 2), the probability of detecting no watermark is much bigger than the ones of two former watermarking techniques; in detail, this probability increases by 0.14 when $\ell = 1$, and 0.25 when $\ell = 2$. Secondly, there is a small false watermark detection in this technique (with a probability ~ 0.01). For example, when the attacked image is done with two different fingerprinted images, it is possible to detect three watermarks from this attacked image. This is due to the compromise for choosing the threshold to get an acceptable detection probability and a detection false alarm. Thirdly, except these two weaknesses, this watermarking technique still has a big advantage. With the increase of the collusion size ℓ, the probability of detecting no watermark decreases very slowly, while this probability decreases very quickly for the last two watermarking techniques. This means that this watermarking technique has potential in multimedia fingerprinting design, especially when ℓ is big.

5.3.2 Evaluation of Fingerprinting Code

The introduced accusation methods presented in Sections 5.2.1 and 5.2.2 amount to the same accusation process as Skoric's in Section 5.1.4 when $\mathcal{Y}'$ is a singleton. This occurs when the colluders choose the pure fingerprinting code attack of Section 2.2.2. The attacks are just limited to the fingerprinting code layer (such as: the block exchange attacked, which is just an exchange of the fingerprinted

block to produce a pirated version), and do not touch the watermarking layer. According to Skoric et al., one of their best attacks within this class is the so-called 'extremal' strategy defined in [25, Eq.(58)]. These authors also noticed that there exist an optimal shape parameter κ to counter-attack this worst case scenario.

In our experiment, we set $m = 300$, $q = 4$, $c = 20$ and κ is varying from 0.1 to 0.5. Figure 5.1 shows the experimental measures of the expectations of the scores of an innocent $\mu_{I,0}$ and of a colluder $\mu_{C,0}$. Figure 5.2 shows the experimental measures of the variances of the scores of an innocent $\sigma^2_{I,0}$ and of a colluder $\sigma^2_{C,0}$. Noticeable features of Skoric's method are that $\mu_{I,0} = 0$ and $\sigma^2_{I,0} = m$, our experiments well verifies these features.

Furthermore, we assume that the scores are Gaussian distributed, therefore, we can compute the Kullback Leibler distance D_{KL} between the two pdfs of these two scores as follows:

$$D_{KL}(I;C) = \frac{1}{2}\left(\frac{(\mu_I - \mu_C)^2}{\sigma_C^2} + \frac{\sigma_I^2}{\sigma_C^2} - 1 + \log\frac{\sigma_C^2}{\sigma_I^2}\right). \tag{5.9}$$

I did not see your answer in the manuscript. "you should motivate more why you use the divergence as a comparison measure here. Why not the achievable rate?" I did not see your answer in the manuscript. This distance roughly shows the performances of the focused accusation process: the higher D_{KL} is, the more powerful is the test. The motivation for using this divergence as a comparison measure is that: it does not need lots of test. For example, if we use the achievable rate as the comparison measure, we should computer a lot of scores for the colluders and innocent users to get a precise result. Figure 5.3 shows the Kullback Leibler distance D_{KL} for κ from 0.1 to 0.5, we can find that $\kappa = 0.23$ is optimal for this experimental setup, which more or less confirms Skoric et al. optimal value of 0.27. The slight difference is not surprising because we use a completely different optimality criterion, while Skoric's optimality criterion is the shortest code length.

In this section, we just evaluate the performance of the fingerprinting code. In the next section, we keep the same test conditions and take the effect brought by the watermarking layer into account, and then give a comparison between Skoric's method and our two new proposed methods.

5.3.3 Evaluations of Our New Accusation Methods

The new accusation methods of Section 5.2 enter in the picture when the collusion chooses the fusion attacks of Secection 2.2.3. We repeat that classical cryptographic fingerprinting codes are not designed for this kind of collusion. Our proposal (the q-ary Tardos symmetric fingerprinting code, one of three proposed zero-bit watermarkings, and one of two improved accusation functions) raises interests if we can show that the fusion strategy is worse from the colluders' point of view. Therefore, the collusion will reject it and it will stick to the pure fingerprinting code attacks, for which fingerprinting codes have been designed.

We first investigate how frequently our methods yield different score than the regular Skoric's accusation process. One necessary condition is that c colluders have more than one hidden symbol at the i-th block. Table 5.2 shows that this occurs with a probability greater than 0.57 if $c \geq 3$. Another condition is that the number of decoded symbols after the fusion is neither 0 nor q, else the summands at that index are zeros. The performances of the watermarking techniques against the fusion attacks has a clear impact on this condition. Combining 12 tables (Tables B.1 to B.12) and Table 5.2, we easily have the probability of decoding D symbols: $P(D|c) = \sum_{d=1}^{\min(q,c)} P(D|d)P(d|c)$. Another 12 tables (Table C.1 - Table C.12) show that $P(D = q|c)$ is negligible, and that $P(D = 0|c)$ is slowly increasing with c thanks to the good robustness of three proposed watermarking techniques. Even for $c = 20$, our methods are active for more than 75% blocks for the first two proposed watermarking techniques, and 66% blocks for the third watermarking technique. Once again, the result confirms that the third watermarking technique has less robustness than two others.

Now we compare the performances of our two new accusation methods described in Section 5.2 with the one of Skoric's method as done in Section 5.3.2. The experimentation setup is the same. In order to simplify the test, we only test one case at first: the images are watermarked by the *AWC watermarking with Nv=256 and Nc=64*, then they are attacked by the *averaging attack* of ℓ images, and finally followed by the JPEG compression with a quality factor $Q = 20$. Therefore, the collusion is based on Table B.1 to simulate a fusion: whenever the

c	$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$
2	0.60	0.40	0	0
3	0.43	0.50	0.07	0
4	0.34	0.52	0.14	0.00
5	0.28	0.52	0.19	0.01
6	0.24	0.51	0.23	0.02
10	0.15	0.46	0.33	0.06
15	0.11	0.40	0.38	0.10
20	0.08	0.36	0.42	0.14

Table 5.2: The conditional probabilities $P(\ell|c)$ that the colluders have ℓ watermarked versions of a block for $2 \leq c \leq 20$ and $\kappa = 0.23$.

collusion has ℓ different symbols, we randomly pick up D of them. Statistics are established from 32,000 scores for the innocents and 8,000 scores for the colluders. Figure 5.1 shows that the expectation of an innocent's score is zero whereas the one of the colluder is roughly the same for both methods and especially much higher than previously. Figure 5.2 shows that the variance of the scores (innocent's and colluder's) are smaller than previously for both methods. The first method is very good at lowering σ_I^2 whereas the second method has the smallest σ_C^2. The overall performances measured by the Kullback Leibler distance in Figure 5.3 confirm that the collusion has no interest in adopting the fusion strategy such as averaging attack.

Furthermore, to evaluate the overall performances of the different watermarking techniques to against the different attacks, we compute the Kullback Leibler distances between the innocent's and colluder's scores pdfs, for a fixed Dirichlet distribution shape parameter $\kappa = 0.23$. We always use the block exchange attack for Skoric's accusation method. Figure 5.4 shows the results, we can see that thanks to our proposed accusation methods, the Kullback Leibler distances after all the fusion attack is higher than the one of the block exchange attack and Skoric's accusation method. Especially for the AWC watermarking with Nv = 256 and Nc = 64, the AWC watermarking with Nv = 1024 and Nc = 256, the distances are increased by a factor of 3 to 8. However, our proposed accusation

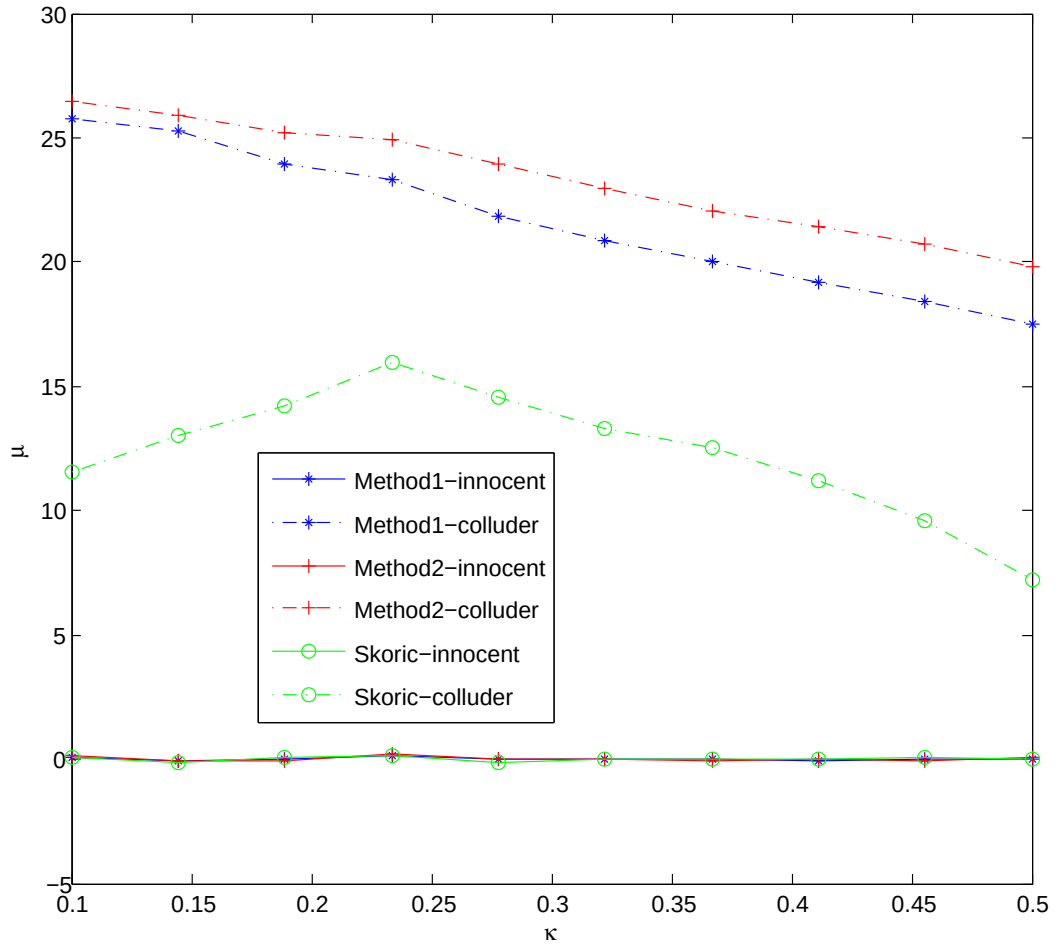

Figure 5.1: Expectation of an innocent's (solid) and a colluder's (dash) score against Dirichlet distribution shape parameter κ for the block exchange attack and Skoric's accusation method, the averaging fusion attack and our first accusation method, the averaging fusion attack and our second accusation method.

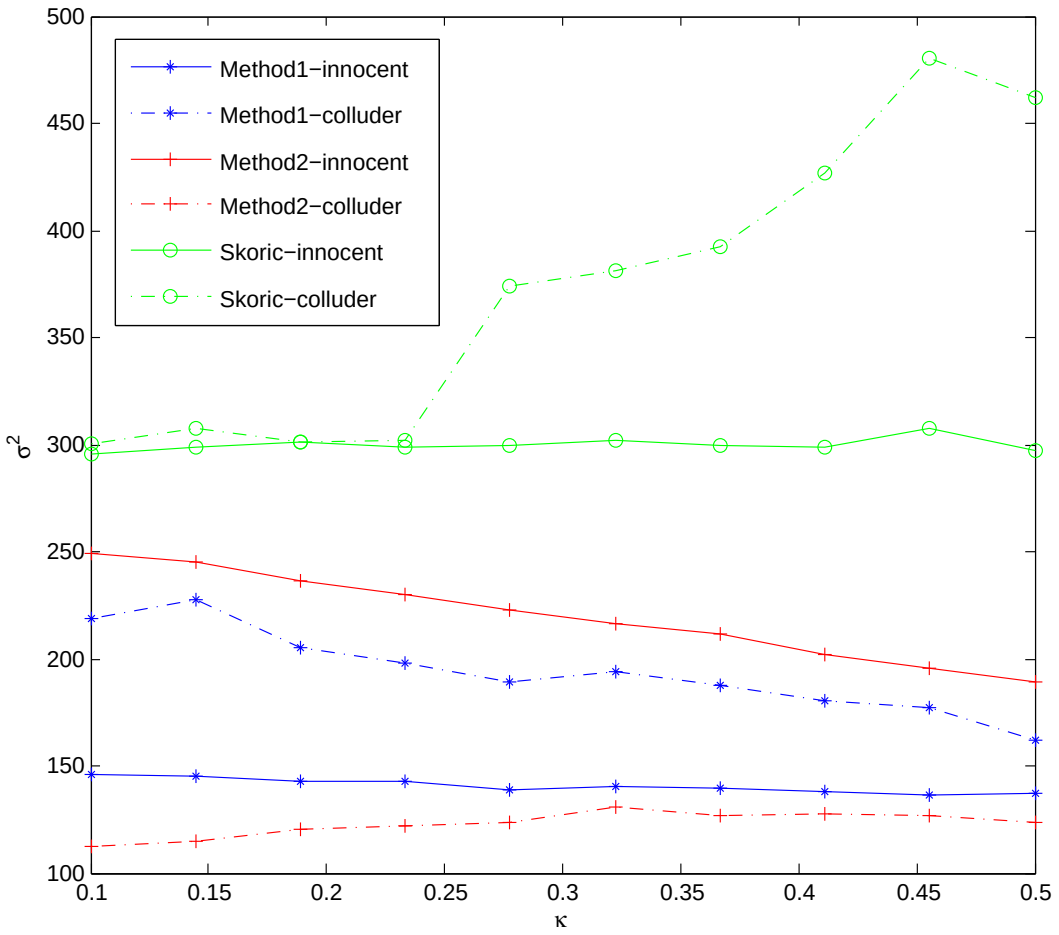

Figure 5.2: Variance of an innocent (solid) and a colluder's (dash) score against Dirichlet distribution shape parameter κ for the block exchange attack and Skoric's accusation method, the averaging fusion attack and our first accusation method, the averaging fusion attack and our second accusation method.

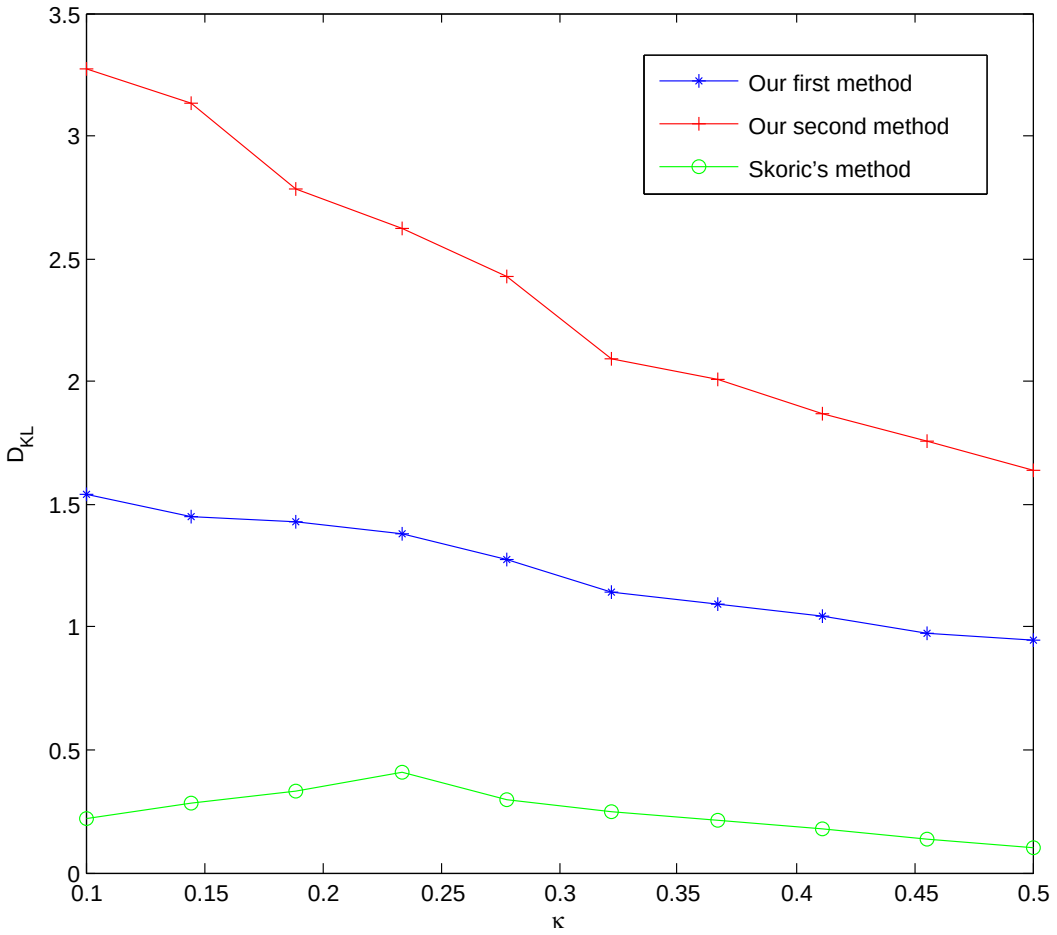

Figure 5.3: Kullback Leibler distance between the innocent's and colluder's scores pdfs against Dirichlet distribution shape parameter κ for the block exchange attack and Skoric's accusation method, the averaging fusion attack and our first accusation method, the averaging fusion attack and our second accusation method.

methods do not get a lot of gain for the robust and secure watermarking, due to its relatively weak robustness. Among the two proposed accusation methods, the second one is better, since the Kullback Leibler distances are twice bigger, for the first two watermarking techniques. All the four fusion attacks gave a comparative distances for these different watermarking techniques and method, except that the distances are slightly bigger after the moderated minority extreme attack.

Another practical issue is the value of the threshold Z. The following relationship holds for both methods:

$$\mu_I = \mu_{I,0}\,, \quad \mu_C \geq \mu_{C,0} \tag{5.10}$$

$$\sigma_I^2 \leq \sigma_{I,0}^2\,, \quad \sigma_C^2 \leq \sigma_{C,0}^2 \tag{5.11}$$

Therefore, if the length of the code is large enough to ensure required probabilities of false alarm and false negative when comparing the scores to Z for the pure fingerprinting attack such as the block exchange, then, this threshold will ensure even lower probabilities of errors for the fusion attacks thanks to the performances of our proposed methods. Consequently, our methods forces the collusion to reject the fusion attacks and to stick to the pure fingerprinting code attacks, for which fingerprinting codes have been designed. However, this statement is true only when the Gaussian assumption holds, *i.e.* for m large enough. Because the given Equation 5.9 to compute the Kullback Leibler distance is correct when the innocent's and colluder's scores follow the Gaussian distributions.

5.4 Chapter Summary

In this chapter, we proposed a complete multimedia fingerprinting system, which is based on the symmetric q-ary Tardos fingerprinting code, and the three proposed watermarking techniques in the last two chapters. This combination is realized through the *on-off* keying modulation. In Section 5.1, we give a detailed description for the whole multimedia fingerprinting system, which includes the fingerprint construction, the fingerprint embedding, the fingerprint detection, and the colluders' accusation.

The detection of multiple fingerprints in one attacked block gives birth to two extended accusation methods. This is thanks to the very robust watermarking

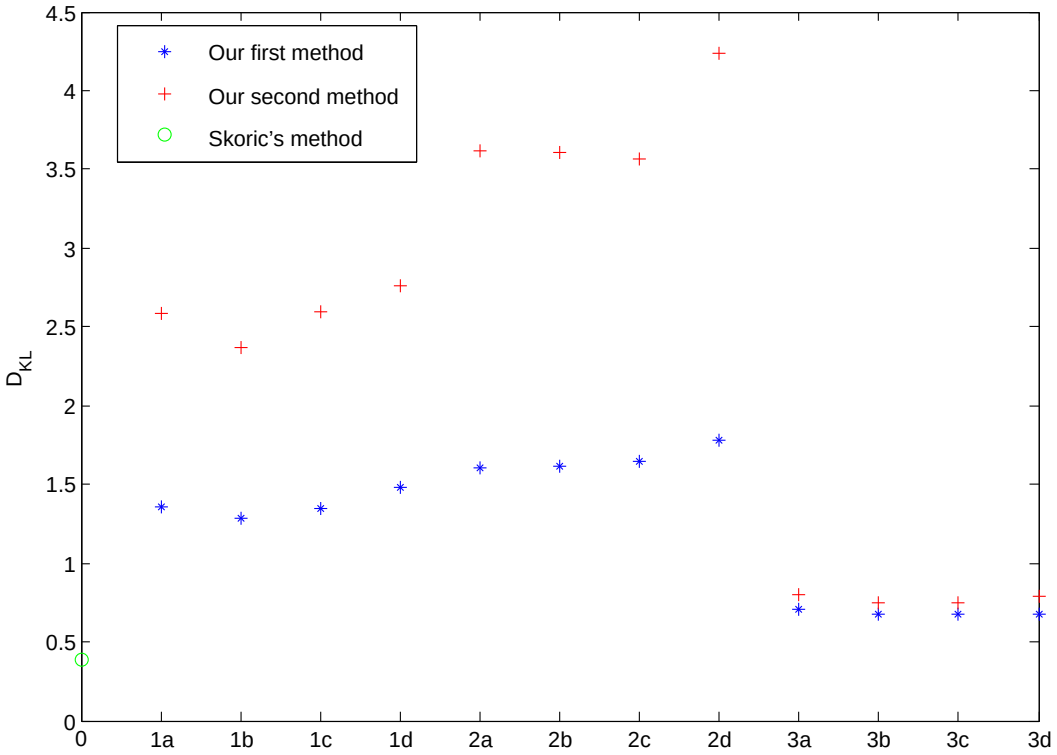

Figure 5.4: Kullback Leibler distances between the innocent's and colluder's scores pdfs for the Dirichlet distribution shape parameter $\kappa = 0.23$. In the x axis, '0' represents uniform exchange attack and Skoric's accusation method, '1' represents the AWC watermarking with Nv = 256 and Nc = 64, '2' represents the AWC watermarking with Nv = 1024 and Nc = 256, '3' represents the robust and secure watermarking. And for our proposed two method, we test four fusion attack: 'a' represents the averaging attack, 'b' represents the interleaving attack, 'c' represents the maximum attack, 'd' represents the moderated minority extreme attack.

techniques used in the multimedia fingerprinting system. We have made a detailed presentation of these two accusation methods in Section 5.2.

In Section 5.3, we evaluate the robustness of these watermarking techniques, and then assess the fingerprinting code. Finally, we evaluate the overall performance of the proposed system. The experimental investigations show the ingredients of our proposed multimedia fingerprinting system blend into a very good design, because it completely shuts down the fusion attacks. Following this strategy, the fusion attacks help more the accusation process than they delude it, and the colluders have no longer interest of using this class of attack. The collusion is then back to the pure fingerprinting code attacks, which is fully tackled by the fingerprinting code.

Chapter 6

Conclusions and Future Perspectives

In this dissertation, we have studied various aspects of multimedia fingerprinting for traitor tracing. Especially, we focused on the exploration of a robust and secure watermarking for this application; and finally proposed a complete multimedia fingerprinting system.

This thesis started from the study of the background and the state of the art of the multimedia fingerprinting field. We give a general framework of multimedia fingerprinting system, and analyze the various attacks such a system has to face. Furthermore, we also review the three historical approaches in the design of multimedia fingerprinting systems: the cryptography and coding approach, the signal processing approach, and the statistical approach. Contrary to these approaches, our goal in this thesis is to consider at the same time the design of the anti-collusion code and the design of the watermarking scheme that will embed the code into the content. This will provide us a global overview of the robustness of the fingerprint embedding and the efficiency of the accusation process.

As the first requirement to provide an efficient tracing is to preserve the presence of a reliable fingerprint, we need a very robust watermarking scheme. We then begin by focusing on the watermark layer. After the comparison of two main categories of watermarking techniques: spread spectrum watermarking and quantization based methods, we decided to explore the spread spectrum based watermarking schemes, focusing on a very robust zero-bit watermarking technique 'Broken Arrows'. However, even if this technique is very robust, it has

been attacked by Westfeld denoising attack, which is the worst robustness attack in the first episode of BOWS-2 contest. Hence, we propose in Chapter 3 two improved embedding variants AWC and BWC to further increase its robustness performance. These improvements perfectly prevent Westfeld denoising attack.

Based on these results, we then focus in Chapter 4 on the security of the watermarking technique. As there are two known security attacks on Broken Arrows scheme, published by Bas and Westfeld, we propose to counter both of them. Thanks to our counterattacks, these security attacks are no longer threats. But the cost is a small loss in robustness and an additional computational complexity. Finally, in order to go further and address some worst unpublished threats and get a higher security level, we proposed some modifications, and introduced a novel secure watermarking scheme. This scheme is based on *a contrario* decision test and a constrained maximization embedding method. However, we face once again the same trade-off between security and robustness in this scheme, but we believe that this scheme has some serious potential in multimedia fingerprinting.

We finally proposed in Chapter 5 a complete multimedia fingerprinting system, which is based on the symmetric q-ary Tardos fingerprinting code, and the three above proposed watermarking techniques via on-off keying modulation. Thanks to the very good robustness of the underlying watermarking techniques, we can detect multiple fingerprints in one attacked block, this giving birth to two extended accusation methods. With the extended accusation methods, we evaluate the overall performance of the proposed multimedia fingerprinting system. The experimental investigations show the ingredients of our proposed multimedia fingerprinting system blend into a very good design, because the fusion attacks, which are usually considered as very tricky to face, help more the accusation process than they delude it. Hence, with this design, usually tackled attacks as block exchange are still very well managed, and usually non tackled fusion attacks are also efficiently countered.

Despite the work carried out in this thesis, there are still several interesting research directions which are worth to be further explored.

1. We have met several times the trade-off between security and robustness for our proposed watermarking techniques. It seems to be inherent to any

watermarking scheme, but until now, to our best knowledge, there is no unifying framework to study this problem. We think this study would be an interesting direction if we want to get a breakthrough.

2. Our proposed robust and secure watermarking scheme is not perfectly secure. It is only secure against second order statistics analysis tools. However, some high order statistics might leak information on the secret space. Therefore, the issues now turn to be how many contents and computing power a high order analysis requires to work accurately. We believe that it is significantly more demanding, but it is necessary to quantify this gap in the future.

3. The watermark detector outputs binary decision, which indicates the presence or absence of the watermarks (this includes potential multiple detections). The *a contrario* decision test of the proposed robust and secure watermarking can indeed provide a probability of the presence of a given symbol, this might allow a soft output bringing more information for the accusation step. This idea is also the subject of our next work.

4. In this thesis, we tried to design a robust and secure watermarking for multimedia fingerprinting, as it can resist certain robustness attacks, security attacks, and fusion attacks. However, there are certainly some other attacks exited. Such as, the de-synchronization attack [90], which is caused by intentional or non-intentional geometric transformation of the host signal; and stable neighboring frames attack [91], in which the pirate exchange the perceptually similar frames to puzzle the fingerprint detection process. We should also test our proposed multimedia fingerprinting system by these attacks and propose some countermeasures. So our work is not finished, the game between the designer and the attacker continues.

5. We have not studied some rare fingerprinting schemes such as asymmetric fingerprinting [92], anonymous fingerprinting [93], dynamic traitor tracing [94]. So we are not sure that the proposed watermarking techniques in this thesis also works well for these rare schemes, it is interesting to verify it in future work.

Appendix A

Résumé Français (Version Longue)

A.1 Introduction

L'avancée des technologies des télécommunications et la vulgarisation des réseaux à haut débit ont facilité la distribution et le partage de documents multimédia. Elle a aussi en parallèle facilité la duplication illimitée des documents, leur modification arbitraire, ou encore leur redistribution illégale. Cette dernière viole inévitablement la propriété intellectuelle des ayants droit des documents multimédias concernés, et il est nécessaire aujourd'hui de disposer de moyens de protection adaptés, pour leur conserver cette facilité de diffusion, tout en limitant les risques de piratage. De nombreux travaux ont été menés depuis une vingtaine d'années sur ce sujet, mêlant principalement des techniques venant de la cryptographie et/ou du traitement du signal, comme le chiffrement, la signature, le tatouage, etc. On arrive aujourd'hui à une certaine maturité dans la conception de tels systèmes de protection, mais il reste encore de nombreux défis à relever.

Le traçage de documents multimedia est une des solutions apportées pour protéger les droits d'auteur. Elle vise à personnaliser les documents au moment

de leur délivrance, en y insérant de manière imperceptible, grâce à une technique de tatouage, des messages spécifiques afin d'identifier leurs utilisateurs. Ainsi, en cas de redistribution illégale de ces documents, on espère pouvoir remonter la piste des utilisateurs qui sont à l'origine de cette redistribution. Toutefois, les pirates ne sont pas stupides : afin d'éviter d'être accusés, ils emploient une grande variété d'attaques pour supprimer ces messages. L'objectif de cette thèse est de concevoir un tel système, qui résiste aux attaques, qu'elles soient intentionnelles ou non. Ce système doit être : 1) robuste aux manipulations génériques classiques de traitement du signal (compression, filtres, etc) qui tentent de lessiver le message caché tout en préservent une qualité acceptable pour l'utilisateur ; 2) sûr contre des attaques ayant trait à la sécurité, attaques plus ciblées et dédiées, dont l'objectif est de retrouver les clés secrètes du système en observant beaucoup de contenus tatoués ; 3) résistant aux attaques par collusion, au cours desquelles les utilisateurs malveillants ou pirates mélangent leurs copies pour forger un contenu non perceptuellement équivalent, mais non protégé. La conception d'un schéma de traçage de documents multimédia repose ainsi sur un schéma de tatouage, qui va cacher les messages de manières les plus robuste et sûre possibles, et un code anti-collusion, qui va déterminer la structure-même des messages que l'on va cacher, et qui va permettre de contrer les attaques par collusion, remontant aux origines du message contenu dans le document forgé par les pirates.

Dans cette thèse, nous adressons ces trois types d'attaques, au travers des deux couches du systèmes, à savoir tatouage et code anti-collusion. Nous nous appuyons sur un schéma de tatouage appelé "Broken Arrows", qui a été mis à l'épreuve en 2007-2008 durant le concours BOWS-2. Nous nous intéressons tout d'abord à l'amélioration de sa robustesse, puis de sa sécurité. Puis nous montrons comment l'associer à un code anti-collusion, ici un code de Tardos, pour contrer les attaques par collusion et construire un système complet et efficace de traçage de documents multimédia.

A.2 État de l'Art

Depuis le premier article écrit par Wagner en 1983 [6], beaucoup de travaux ont été publiés. Ils traitent principalement des attaques que les pirates peuvent imaginer, de la conception de codes anti-collusion, ou encore de la combinaison entre le code de traçage et certaines techniques de tatouage classiques. Dans cette section, nous présentons brièvement le cadre général de ces études et résumons des progrès significatifs réalisés ces dernières années.

Un schéma de traçage de documents multimédia complet ressemble à une chaîne de communication, avec un émetteur qui génère le message d'identification et le cache avec une technique de tatouage, un canal qui modélise les attaques que le document risque de subir après sa distribution, et un récepteur qui va extraire le message caché afin de confondre le(s) fraudeur(s). Un tel schéma peut également être considéré comme composé de trois modules : un module de codage (code anti-collusion), un module de transmission (tatouage) et un canal d'attaques, comme le montre la Figure 2.1.

Nous allons tout d'abord traiter des diverses attaques qui peuvent affecter le système. Une compréhension approfondie de ces attaques est essentielle pour concevoir un système de traçage de documents multimédia efficace. Ces attaques peuvent être classées en trois catégories, en fonction de leurs objectifs. 1) Les attaques qui ciblent le module de tatouage peuvent être classées en deux familles : les attaques liée à la robustesse du schéma, et les attaques liées à sa sécurité. Pour les contrer, il faut utiliser un schéma de tatouage robuste et sûr. 2) Les attaques qui ciblent le module de codage anti-collusion sans chercher à mettre en défaut le module de tatouage, mélangent les différentes copies que les utilisateurs ont reçues d'un même document, par exemple en intercalant des morceaux provenant des unes et des autres. Ces mélanges ont été modélisés de différentes manières, dont la plus populaire et la plus ancienne est la "Marking Assumption" [11]. Pour les contrer, il faut utiliser un code anti-collusion efficace. 3) Les attaques par fusion fusionnent les copies pour forger le document pirate, par exemple en les moyennent pixel à pixel. Elles sont faciles à réaliser par les pirates, mais difficiles à appréhender pour le concepteur du système, car leurs conséquences

sur les messages cachés sont difficiles à prévoir car les messages ne se retrouvent plus mélangés comme précédemment. Pour les contrer, il faut un schéma de tatouage suffisamment robuste pour permettre, malgré la fusion, de retrouver des informations fiables sur les messages impliqués dans la forge du document, puis avoir un code anti-collusion efficace. Contrairement aux deux précédentes familles d'attaques, qui peuvent être testées facilement module par module, on ne peut envisager d'étude de la robustesse á ce troisième type d'attaques sans regarder le système dans son ensemble. Zhao et al. ont donné une description assez complète d'attaques non linéaires de ce type dans [7] [8], tandis que H. G. Schaathun a introduit plusieurs attaques par collusion pour des systèmes de traçage de documents multimédia reposant sur des techniques de tatouage utilisant l'étalement de spectre [41] [9]. Aujourd'hui, ces attaques par fusion restent l'enjeu principal lors de la conception de systèmes de traçage de documents multimédia.

Dans la littérature, on distingue plusieurs approches pour la conception d'un schéma de traçage de documents multimédia. Il y a tout d'abord l'approche des chercheurs en cryptographie et codes correcteurs d'erreurs, qui a donné lieu à de très nombreuses publications et fait abstraction de la couche d'insertion pour se focaliser sur la conception du code anti-collusion. Ces premiers travaux remontent à [6] (en 1983) et [10] (en 1986). Plus tard, Boneh et Shaw ont introduit un modèle formel d'étude, dit de la "Marking Assumption", dans lequel ils supposent que les éléments du document qui sont communs à tous les membres de la collusion resteront inchangés dans le document forgé. Dans ce cadre, ils ont également introduit des propriétés utiles à la mesure de la capacité à tracer des codes anti-collusion, et ont proposé un code binaire qui les satisfait [11]. Ce code a été amélioré par la suite par Yacobi pour application à des signaux multimédias [12]. Beaucoup de travaux se sont placés dans ce modèle, dont [13] et [14], qui ont introduit dans ce cadre de nouvelles propriétés liées aux capacités de traçage des codes, comme la propriété de parent identifiable (IPP) ou encore le code traçable (TA), plus fortes mais aussi plus contraignantes que celles initialement proposées par Boneh et Shaw. Les codes IPP ou TA ne peuvent pas accuser un innocent, et sont dits à traçabilité forte. Les codes qui peuvent parfois accuser un innocent, avec une probabilité non nulle mais maîtrisée sont dits à traçabilité faible. Mais

la traçabilité forte n'est pas accessible dans la pratique [15] [16], et on est donc contraints d'utiliser dans des schémas opérationnels des codes à traçabilié faible. Ainsi, tous les codes mentionnés dans cette thèse relvent de cette catégorie, et peuvent parfois accuser un innocent, avec une probabilité d'erreur bornée.

Plus récemment, une autre approche a été développée en parallèle par les chercheurs en traitement du signal, et notamment à l'université du Maryland. Wang et al. ont utilisé tout d'abord les signaux mutuellement indépendants (ou orthogonaux) comme les signaux de traçage pour identifier les pirates [17]. En raison de la charge de calcul au décodage, Trappe et al. ont employé plus tard un code modulant des signaux indépendants pour construire les signaux redondants [18]. En outre, afin de rendre leur système de traçage de documents multimédia plus efficace, certaines stratégies accessoires ont été introduites [19] [20] [21]. Enfin, certaines applications de traçage de documents multimédia ont été explorées [22] [23].

En parallèle, une percée a été effectuée en 2003 lorsqu'un statisticien, G. Tardos, a publié un nouveau code anti-collusion probabiliste binaire dont la longueur a un ordre de grandeur théoriquement minimum [24], et qui plus est est simple à mettre en œuvre et efficace. Par la suite, Skoric et al. ont étendu ce code de Tardos original de binaire à q-aire [25]. D'autres améliorations qui été proposées, afin de réduire la longueur du code de Tardos [26] [27] [28] [29], d'optimiser la mémoire d'utilisation [30], ou encore pour améliorer les fonctions accusations [31] [32].

Mais la plupart des publications se focalisent sur la conception du code anti-collusion, laissant à d'autres le soin de concevoir la couche d'insertion. Or, si l'on souhaite obtenir un schéma opérationnel, il faut ajuster les deux couches au mieux, et cela ne peut être obtenu qu'avec une vision globale du système, sans rien laisser de côté. C'est cette voie que nous avons suivie, choisissant au mieux les primitives d'insertion et de traçage, et les adaptant au besoin pour les articuler efficacement. Dans les sections suivantes, nous allons montrer comment choisir une technique de tatouage adéquate, ici issue de l'amélioration du schéma de tatouage "Broken Arrows", puis nous montrerons comment l'articuler avec un code de Tardos lui aussi amélioré pour construire un schéma de traçage de documents multimédia complet et efficace.

A.3 Tatouage Robuste pour Le Traçage de Documents Multimédia

N'importe quel schéma de traçage de documents multimédia que l'on souhaite robuste et sûr doit s'appuyer sur une technique de tatouage robuste et sûre pour insérer les codes d'identification des utilisateurs. Toutefois, concevoir une technique de tatouage robuste et sûre est un travail difficile, puisque la robustesse et la sécurité sont deux concepts très différents [62] [63] [64]. Ce sont deux aspects indispensables mais parfois contradictoires du tatouage. Afin de choisir une technique de tatouage appropriée, nous comparons la robustesse et la sécurité de deux familles principales de tatouage: les méthodes basées sur l'étalement de spectre et les méthodes basées sur la quantification. Certains travaux antérieurs montrent que les méthodes basées sur la quantification dépassent en performance les méthodes basées sur l'étalement de spectre dans le sens des critères traditionnels d'évaluation de tatouage : la distorsion et la robustesse. Toutefois, elles ont des niveaux de sécurité relativement faible. Nous avons donc choisi de nous concentrer sur les techniques par étalement de spectre. Cayre et Bas ont proposé deux nouvelles modulations de tatouage dans le cadre de 'Watermarked-Only-Attack' (uniquement les contenus tatoués sont disponibles pour l'attaquant), appelées "tatouage naturel" et "tatouage circulaire" [78]. Ces schémas considèrent la sécurité comme une priorité, et puis évaluent la robustesse. Mais la perte en robustesse est assez grande par rapport aux techniques robustes récentes, et pour cette raison nous avons choisi de regarder dans une autre direction. Nous avons préféré nous appuyer sur une technique de tatouage zéro-bit très robuste appelée 'Broken Arrows', qui, puis nous nous sommes attachés à améliorer encore sa robustesse, puis sa sécurité. Dans cette section, nous nous concentrons sur la robustesse, la sécurité étant traitée dans la section suivante.

'Broken Arrows' [33] a été conçu pour le deuxième concours de 'Break Our Watermarking System' (BOWS-2) [34]. Ses performances en termes de la robustesse et d'imperceptibilité sont très bonnes en comparaison à l'état de l'art. Comme par ailleurs il a été intensivement mis à l'épreuve pendant le concours de BOWS-2 et qu'il a bien résisté, il constitue une base de travail solide. Toutefois,

'Broken Arrows' a été attaqué par A. Westfeld [39] dans le premier épisode du concours de BOWS-2. Westfeld a conçu une attaque spécifique, qui peut être considéré comme un processus de débruitage. Cette attaque est principalement basée sur l'estimation de l'amplitudes du coefficient d'ondelette en fonction des coefficients de son voisinage par une régression linéaire. Notre objectif ici est de fournir une version améliorée de 'Broken Arrows', robuste contre cette attaque spécifique, tout en restant robuste aux attaques génériques habituelles.

Afin de renforcer la robustesse de 'Broken Arrows', nous proposons deux directions d'amélioration : (i) équilibrer les coefficients d'ondelette de trois sous-bandes dans le même niveau de transformation (BWC) et (ii) calculer la moyenne du coefficient d'ondelette avec ses quatre voisins dans la même sous-bande (AWC). La première amélioration consiste à corréler les coefficients des trois sous-bandes dans le même niveau de transformation d'ondelette (voir Figure 3.2). Intuitivement, ce type d'insertion améliore la dépendance entre les sous-bandes des coefficients d'ondelette du signal de tatouage. Cette solution est détaillée dans la sous-section 3.3.1. La deuxième solution améliore la robustesse du schéma en prenant en compte la dépendance entre les coefficients voisins (voir Figure 3.3). L'idée principale est inspirée directement par l'attaque de débruitage de Westfeld. Nous remplaçons tous les coefficients d'ondelette du masque visuel par une moyenne de cinq coefficients : lui-même et ses quatre voisins locaux. De cette façon, le signal de tatouage peut modifier les signes des coefficients des signaux hôtes. Comme BWC, cette méthode AWC est également une solution efficace pour faire face à l'attaque de débruitage de Westfeld. Elle est détaillée dans la sous-section 3.3.2.

Pour évaluer ces amélioration, nous avons utilisée la base des 2000 images de taille 512×512 de BOWS-2, et les mêmes conditions de test que dans [33], soit avec un PSNR souhaité de 43dB. Dans nos simulations, trois stratégies différentes sont comparées : l'insertion proportionnelle originale BA de Broken Arrows, les insertions proportionnelles de BWC et d'AWC. Si nous considérons par exemple l'image "sheep", le PSNR commenté est 43 dB, le PSNR réel d'image tatouée est 42.88 dB pour BA, 42.88 dB pour BWC et 42.81 dB pour AWC. Nous savons qu'avec un PSNR supérieur à 40 dB, les amplitudes des coefficients d'ondelette

du signal de tatouage sont presque tous inférieurs à 1 pour BA, donc BA conserve tous les signes des coefficients d'ondelette. En fait, conformément à l'essai pour l'image "sheep", seulement 0.76% des coefficients d'ondelette voient leur signe modifié par le processus d'insertion BA. Par conséquent, nous pouvons dire que le signal de tatouage est alors uniquement caché dans l'amplitude. Mais les normes du signal de tatouage des deux techniques améliorées sont plus grandes, et le tatouage modifie certains signes des coefficients d'ondelette. Dans notre simulation, 2.36% des coefficients d'ondelette ont changé leur signe après l'insertion proportionnelle BWC, et 2.16% des coefficients d'ondelette ont changé leur signe après l'insertion proportionnelle AWC. Par conséquent, le signal de tatouage est non seulement caché dans les amplitudes des coefficients d'ondelette, mais également dans leurs signes.

De manière générale, sur les 2000 images, le PSNR réel des images tatouées est compris entre 42.5 dB et 43 dB. Comme BA, les distorsions sont invisibles pour presque toutes les images lors de l'utilisation des méthodes d'insertion BWC ou AWC. Nous appliquons tout d'abord le même benchmark sur les images tatouées que celui qui avait été appliqué dans [33] : un certain nombre d'attaques, principalement composées des compressions JPEG et JPEG 2000 à des facteurs de qualité variés, de filtrage passe-bas, d'effacement de sous-bandes ondelette et un algorithme de débruitage simple. La probabilité de détecter la marque est donnée pour la moyenne du PSNR des images attaquées (voir Figure 3.9). L'impact sur la probabilité de détection est intéressant : chaque technique d'insertion a son avantage pour résister à des attaques différentes, mais la performance globale de l'insertion proportionnelle BWC est moins bonne que celle des deux autres techniques.

Nous avons ensuite évalué la robustesse des trois techniques d'insertion (BA, BWC et AWC) contre l'attaque de débruitage de Westfeld. Pour obtenir un résultat comparable avec l'expérience ci-dessus, nous gardons les mêmes conditions de test et utilisons les mêmes 2000 images. Les PSNRs des images attaquées vont de 19.9 à 46.2 dB. Ce résultat est presque identique à celui de Westfeld (de 19.7 à 45.0 dB). Nous calculons les pourcentages des images attaquées avec succès pour un PSNR moyen donné (voir Figure 3.10). Les résultats expérimentaux montrent que, pour BA, l'attaque de débruitage de Westfeld est vraiment puissante.

Elle réussit à 100% lorsque le PSNR est inférieur à 30 dB, et même si son efficacité diminue lorsque le PSNR croît, elle réussit avec 40% des images attaquées lorsque le PSNR est environ de 35 dB. Au contraire, pour BWC, l'attaque de débruitage de Westfeld ne fonctionne pas du tout : le pourcentage des images attaquées avec succès est de 0% pour tous les PSNRs. Et pour AWC, l'attaque de débruitage de Westfeld fonctionne très peu pour les images qui ont un PSNR de 26 à 32 dB.

Une de nos deux techniques d'insertion améliorées est suffisante pour faire face à l'attaque de débruitage de Westfeld. Toutefois, un peu de robustesse est perdue contre certaines attaques classiques, notamment pour la méthode d'insertion proportionnelle BWC. Par conséquent, afin d'éviter l'attaque de débruitage de Westfeld ainsi que les autres, nous devons faire un compromis, et la méthode d'insertion proportionnelle AWC semble être le meilleur choix.

A.4 Tatouage Sûr pour Le Traçage de Documents Multimédia

Dans la dernière section, nous avons sélectionné 'Broken Arrows' comme technique de tatouage pour notre système de traçage de documents multimédia, et proposé des améliorations pour renforcer sa robustesse contre l'attaque de débruitage de A. Westfeld [39]. Toutefois, des failles de sécurité ont été découvertes pendant le troisième épisode de BOWS-2, au cours duquel les participants pouvaient observer beaucoup d'images tatouées avec la même clé secrète. Discutées [5], elles empêchent clairement l'utilisation du 'Broken Arrows' original dans le cadre du traçage de documents multimédia. Aussi, nous proposons ici des contre-mesures pour améliorer les niveaux de sécurité et contrer ces deux attaques.

Comme nous l'avons mentionné dans la dernière section, la méthode d'insertion proportionnelle AWC est efficace pour empêcher l'attaque de débruitage de A. Westfeld, tout en conservant une bonne robustesse contre les attaques habituelles. Mais son impact sur les niveaux de sécurité n'a jamais été examiné plus avant. Cependant, comme l'attaque de regroupement de A. Westfeld, qui était la plus efficace lors du troisième épisode de BOWS-2 [34], repose en partie sur l'attaque de

débruitage de A. Westfeld évoquée plus haut, AWC devrait empêcher l'estimation des signaux de tatouage, neutralisant l'attaque de regroupement, qui ne peut plus faire une bonne classification des signaux de tatouage. Afin de confirmer cette idée, nous avons reprogrammé l'attaque de regroupement de A. Westfeld sous Matlab, dans les mêmes conditions. Nous avons mesuré sa précision par l'information mutuelle ajustée (voir Figure 4.1). Pour la technique d'insertion BA, l'information mutuelle ajustée est de 0.7 ; cela signifie que les groupes estimés sont très similaires aux vrais groupes, et donc l'attaque de regroupement réussit à estimer les cônes secrets avec une bonne précision. Toutefois, ce classifieur ne fonctionne pas avec notre méthode d'insertion améliorée AWC, puisque l'information mutuelle ajustée est inférieure à 0.05. La technique d'insertion proportionnelle AWC est donc effectivement une solution efficace pour bloquer l'attaque de regroupement de A. Westfeld.

L'attaque d'estimation de sous-espace de P. Bas n'utilise pas l'attaque de débruitage de A. Westfeld. Cependant, nous savons que cette attaque d'estimation de sous-espace par des techniques de type PCA, via l'implémentation OPAST, est basée sur le fait que l'insertion modifie la distribution des puissances du signal dans l'espace secret. Cela laisse un indice pour divulguer cet espace. Par conséquent, nous proposons un outil pour mesurer la différence de puissance entre une direction sélectionnée comme un support de cône secret et une direction quelconque. Nous proposons un réglage des paramètres du système pour réduire cette différence. Nous définissons une distance chordale carrée normalisée : SCD_{norm}, où $SCD_{norm} = 0$ signifie que l'espace estimé est égal à l'espace secret, et $SCD_{norm} = 1$ signifie que les sous-espaces sont orthogonaux et que, par conséquent, l'attaque a échoué. Le résultat expérimental montre que (voir Figure 4.4), pour la technique d'insertion initiale, SCD_{norm} est décroissant avec le nombre d'observations, et l'estimation converge très rapidement. Cela confirme les résultats de P. Bas [5]. Toutefois, pour la technique d'insertion AWC avec les nouveaux paramètres, SCD_{norm} diminue très lentement : même après 3.10^4 observations, SCD_{norm} est toujours très proche de 1 (plus de 0.9), ce qui montre que l'attaque ne peut plus estimer les sous-espaces efficacement.

Nous appliquons à nouveau le même benchmark que dans l'article original de BA pour examiner l'incidence sur la robustesse causée par les nouvelles

améliorations proposées. Le résultat montre que la robustesse est légèrement diminuée. En d'autres termes, notre contre-mesure donne une grande amélioration des niveaux de sécurité en sacrifiant un peu de robustesse.

Jusqu'à présent, nous avons évoqué BA comme une technique d'insertion zéro-bit, indépendemment de tout scénario. Dans l'application de traçage de documents multimédia, BA est utilisée en conjonction avec un code anti-collusion, plus précisément un code de Tardos en version q-aire. Par conséquent, nous avons adapté la technique de tatouage à la modulation 'on-off keying'. L'espace secret est décomposé en q sous-espaces : chacun entre eux rassemble les directions de cônes secrets associées à un symbole. Étant donné que les symboles insérés sont distribués uniformément, toutes les directions des sous-espaces secrets ont la même probabilité de servir comme porteuse du cône secret. De cette façon, la répartition de la puissance du signal de tatouage dans l'espace secret est beaucoup plus uniforme qu'avant, et le niveau de sécurité de la technique de tatouage est encore renforcé.

En première conclusion, on peut dire qu'améliorer le niveau de sécurité nous a coûté un peu en robustesse par rapport à la technique de BA originale, et les processus d'insertion et de détection sont ralentis d'un facteur 4. Dans l'ensemble, et compte tenu du gain en sécurité, cette amélioration est d'ores et déjà satistfaisante. Néanmoins, nous proposons d'aller plus loin afin d'anticiper sur d'autres attaques potentielles. Il est par exemple possible pour le pirate d'utiliser des implémentations de PCA plus puissantes que OPAST, et de collecter plus d'images tatouées pour divulguer l'espace secret.

Nous proposons donc de repartir de notre première amélioration, et de la renforcer pour contrer des attaques basées sur des statistiques de second ordre, en restituant une distribution de puissance parfaitement uniforme. Ces changements comprennent 1) l'introduction d'un critère de sécurité, 2) un processus d'insertion mis en œuvre comme une maximisation de la robustesse, sous les contraintes de la perception et de la sécurité, et 3) une détection de tatouage basée sur un critère de décision *a contrario*. Commençons par la détection *a contrario*. Nous savons que BA projette beaucoup de coefficients d'ondelette de l'image dans un sous-espace secret de dimension très faible, ce qui rend le vecteur projecté Gaussien. Partant de ce fait, nous proposons un paradigme très différent, basé sur la détection *a*

contrario en statistiques. La détection valide une des deux hypothèses H_0 ou H_1 selon certaines observations. Le cadre est le suivant : le modèle statistique est supposé sous hypothèse H_0 avec une distribution p_{H_0}, mais l'alternative H_1 est beaucoup trop large pour soutenir un tel modèle, et/ou ce qui se passe sous H_1 n'est pas bien connu. Par conséquent, le détecteur évalue la probabilité d'avoir ces observations sous p_{H_0}, et si cet événement est peu probable il décide H_1. Nous appliquons exactement la même idée à notre application : pour les images naturelles originales, nous supposons que le vecteur projecté a une distribution Gaussienne. La détection du tatouage est déclenchée quand ce vecteur n'a pas cette distribution typique. Par conséquent, l'insertion de tatouage équivaut à rendre le vecteur projecté non typique autant que possible au regard de p_{H_0}. Discutons maintenant comment ce concept est mis en œuvre. Nous supposons ici que nous avons extrait un vecteur Gaussien $\mathbf{v}_X$ à partir du contenu hôte, et qu'un PSNR souhaité contrôle l'insertion de tatouage. Nous avons besoin de trois fonctions pendant l'insertion: 1) une fonction de qualité, qui suppose que le PSNR souhaité impose une contrainte sur la puissance du signal de tatouage pour contrôler la distorsion perceptuelle, 2) une fonction de sécurité, qui assure qu'une attaque de type PCA échouera en assurant que la puissance du signal tatoué dans l'espace secret, en moyenne, est égale à celle qui figure dans l'espace complémentaire, 3) une fonction de Gaussianité qui est en fait la fonction de test d'Anderson-Darling, utilisée pour décider si le vecteur secret suit la distribution Gaussienne ou non. A l'insertion, nous utilisons cette dernière comme fonction objective, les deux premières fonctions étant prises comme contraintes. L'insertion de tatouage est une maximisation sous contraintes : elle recherche le vecteur de tatouage qui maximise cette fonction objective sous les deux fonctions de contrainte. A notre connaissance, c'est la première fois que la sécurité est prise en compte de manière aussi formelle lors de la conception d'un algorithme d'insertion de tatouage.

Comme la qualité perceptuelle et le niveau de sécurité du système de tatouage proposé sont assurées lors de la génération du signal de tatouage, nous évaluons ici seulement sa robustesse. Nos expériences montrent sa bonne robustesse contre une attaque par ajout de bruit (voir Figure 4.8). Par ailleurs, nous évaluons les

performances de robustesse avec le même benchmark qu'auparavant (voir Figure 4.9). Les résultats montrent que, en général, la robustesse cette nouvelle technique de tatouage proposée est plus faible que celle des précédentes. Pour une attaque donnée, la probabilité de bonne détection des images attaquées est plus ou moins la même que celle des techniques précédentes. Cependant, cette technique est beaucoup moins fiable que les précédentes parce que la probabilité de fausse alarme ici est ici de 1.10^{-2} alors que celles des précédentes étaient à 3.10^{-6}. C'est le prix à payer pour avoir un excellent niveau de sécurité. Enfin, nous estimons sa robustesse contre les attaques de collusion dans le cas binaire ($q = 2$) : attaque par moyennage, attaque par entrelacement, attaque "maximum", attaque "minimum" et attaque "uniforme" (voir Table 4.1). Les résultats montrent que toutes ces attaques permettent une détection double avec une grande probabilité (94%), ce qui améliore considérablement les performances de l'algorithme de traçage [88]. Nous pensons que ce schéma est particulièrement adapté à certaines applications comme le traçage de document multimédia, car alors la probabilité de fausse alarme n'est plus un problème. Par ailleurs, le test de décision *a contrario* peut aussi fournir une probabilité de la présence d'un symbole donné dans le tatouage, et fournir une information non plus dure (présence ou absence), mais souple, apportant plus d'information pour l'étape d'accusation du code anti-collusion.

A.5 Un Système Complet, Sûr et Robuste de Traçage de Documents Multimédia

Dans cette section, nous proposons un système complet de traçage de documents multimédia, basé sur le code de Tardos en version q-aire symétrique, introduit par Skoric et al. [25], et les trois techniques de tatouage présentées dans les deux sections précédentes. Cette combinaison est réalisée par une modulation de type *on-off keying*. En résumé, le système complet de traçage de documents multimédia comprend la construction du code, l'insertion des symbole, puis la détection de ces symboles, et enfin l'accusation de colluders. En particulier,

le contenu multimédia est divisé en blocs consécutifs, chaque bloc permettant de cacher un symbole de l'identifiant. L'accusation s'effectue après calcul des scores des utilisateurs testés, scores obtenus à l'aide de fonctions d'accusations, appliqués à leurs identifiants et au mot extrait du contenu piraté. Plus le score d'un utilisateur est élevé, plus il a de chance d'être coupable. Dans un premier temps, nous avons utilisé les mêmes fonctions d'accusation que Skoric et al.

Toutefois, d'après les résultats expérimentaux de la détection de symboles, et grâce à la très grande robustesse des techniques de tatouage utilisées, plusieurs symboles peuvent être détectés dans un bloc attaqué. Comme la fonction d'accusation originale de Skoric ne profite pas de toute l'information des symboles décodés, nous proposons deux amélioration pour en tirer profit. Dans la première méthode, nous prenons tous les symboles détectés d'un bloc comme les opérandes dans la somme du score. De cette manière, tout se passe comme si la longueur du code avait augmenté. Comme nous le savons, plus long est le code de traçage, plus fiable est le processus d'accusation. Par conséquent, cette méthode proposée est plus fiable que l'originale. Dans la deuxième méthode, nous gardons le même score que Skoric, mais remplaçons la probabilité du symbole détecté par la somme des probabilités de tous les symboles détectés dans le bloc attaqué. La justification de cette méthode est qu'elle diminue la variance des scores des colluders : quel que soit leur symbole dans la liste des symboles détectés, ils reçoivent la même pénalisation.

Dans nos expériences, nous évaluons tout d'abord la robustesse des trois techniques de tatouage contre des attaques de type fusion, considérée comme difficiles à contrer. Nous appliquons quatre attaques de collusion typiques : 1) l'attaque par moyennage, 2) l'attaque par entrelacement, 3) l'attaque "maximum", et 4) l'attaque de minorité extrême modéré (MMX). Puis, les images attaqués sont encore dégradées par une compression JPEG avec un facteur de qualité $Q = 20$. Les résultats expérimentaux montrent que pour les deux premières techniques de tatouage, les résultats sont très similaires et la dernière est un peu meilleure. Lorsque la taille de collusion ℓ est petite (1 ou 2), ces quatre attaques ont une performance similaire, la probabilité de détection est très bonne (~ 0.98). Toutefois, lorsque le mélange prend plus de deux images tatouées, leurs performances deviennent moindres, parce que la force du signal de tatouage est plus petite

quand plusieurs images sont mélangées. Pour $\ell = 4$, plus de la moitié du temps, nous ne pouvons plus détecter un signal de tatouage dans les images attaquées. Les résultats de la dernière technique de tatouage sont assez différents de ceux des deux précédentes : lorsque ℓ est petite (1 ou 2), la probabilité de détecter aucun symbole est beaucoup plus grande. Mais cette technique de tatouage a un avantage : avec l'augmentation de la taille de la collusion ℓ, la probabilité de ne détecter aucun symbole diminue très lentement. Cela signifie que cette technique de tatouage a un certain potentiel dans la conception de traçage de documents multimédia, en particulier lorsque ℓ est grand.

Nous évaluons ensuite la performance de notre code de Tardos symétrique q-aire. Les attaques considérées dans cette partie sont limitées à la couche de codage et ne concernent pas la couche de tatouage, comme par exemple l'attaque par échange de blocs, qui simplement choisit pour un bloc donné une des versions tatouées détenues par les pirates. Nous calculons ensuite les espérances et les variances des scores des innocent et ceux des colluders. Ces scores sont considérés comme Gaussiens, et nous calculons la distance de Kullback Leibler entre leurs deux distributions. Collectant ces outils et indicateurs, nous évaluons alors la performance globale du système complet de traçage de documents multimédia. Nous conservons les mêmes conditions de test que pour l'évaluation du code de traçage, et prenons en compte l'effet apporté par la couche de tatouage. Nous étudions d'abord la probabilité que nos méthodes donnent des scores différents de celui du processus d'accusation de Skoric. Une condition nécessaire est qu'on détecte plusieurs symboles différents cachés dans un même bloc attaqué. Le résultat expérimental montre que ce cas se présente avec une probabilité supérieure à 0.57 si la taille de collusion $c \geq 3$. Si nous prenons en considération la bonne robustesse des techniques de tatouage contre les attaques de type fusion, le résultat montre que la probabilité de ne détecter aucun symbole augmente lentement avec c. Même pour $c = 20$, nos méthodes restent pertinentes pour plus de 75% des blocs pour les deux premières techniques de tatouage, et 66% des blocs pour la troisième technique de tatouage.

En outre, nous comparons les performances de nos deux nouvelles méthodes d'accusation avec la méthode du Skoric. Afin de simplifier le test, nous testons

tout d'abord un seul cas : la technique de tatouage AWC et l'attaque par moyennage ; les résultats expérimentaux montrent que l'espérance du score d'un colluder est beaucoup plus élevée que pour la méthode de Skoric; et de plus les variances des scores (des innocents et des colluders) sont plus petites. Les performances globales mesurées par la distance de Kullback Leibler confirment que la collusion n'a aucun intérêt à utiliser la fusion d'images. Pour évaluer les performances globales des différentes techniques de tatouage contre les différentes attaques, nous calculons les distances de Kullback Leibler entre les pdfs des scores des innocents et des colluders. Les résultats montrent que grâce à nos méthodes d'accusation, les distances de Kullback Leibler après toutes les attaques de fusion sont plus élevées que pour l'attaque par échange de blocs avec la méthode d'accusation de Skoric. Par conséquent, les attaques de fusion aident plus le processus d'accusation que la collusion, et les collders n'ont plus aucun intérêt à utiliser ce type d'attaques. Ainsi, nos méthodes forcent la collusion à rejeter les attaques par fusion et à se contenter des autres attaques qui, elles, sont naturellement très bien gérées par les codes anti-collusion.

A.6 Conclusion

Cette thèse porte sur la conception d'un tatouage sûr et robuste pour le traçage de documents multimédia, et propose un schème complet de traçage de traîtres. L'insertion des identifiants des utilisateurs s'effectue grâce à une version améliorée de la technique de tatouage zéro-bit très robuste 'Broken Arrows'. Nous proposons plusieurs variantes de cette amélioration, pour encore accroître sa robustesse, puis sa sécurité. Ces amélioration permettent notamment de contrer les attaques de sécurité connues, telles que l'attaque de regroupement de A. Westfeld et l'attaque d'estimation de sous-espace de P. Bas. En outre, afin d'anticiper d'autres menaces et obtenir un niveau de sécurité encore plus élevé, nous allons plus loin en formalisant les critères garantissant sa sécurité. Nous obtenons ainsi non seulement un schéma qui se comporte bien, mais aussi une estimation de sa sécurité plus formelle. Notre dernière amélioration repose sur un test

de décision *a contrario* et une méthode d'insertion avec maximisation sous contraintes. Sa sécurité est garantie au regard des critères établis. Enfin, nous proposons un schéma complet de traçage de documents multimédia, basé sur les codes anti-collusion de Tardos en version q-aire symétrique, et les trois des techniques du tatouage proposées ci-dessus avec une modulation numérique "tout ou rien". Comme les techniques de tatouage proposées sont très robustes, nous pouvons détecter plusieurs symboles dans un seul bloc attaqué. Nous tirons parti de ce gain d'information lors de l'accusation, en proposant deux nouvelles méthodes d'accusation. Les études expérimentales montrent que notre conception est de bonne qualité, car elle dissuade complètement les pirates d'utiliser des attaques par fusion, qui étaient jusque là tant redoutées des concepteurs de schéma de traçage. Les pirates sont alors obligés de se rabattre sur des attaques plus simples, que les codes anti-collusion gèrent bien par nature.

Appendix B

Annexe 1 for Chapter 5

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.02	0.06	0.92	0	0
ℓ=3	0.30	0.28	0.23	0.19	0
ℓ=4	0.74	0.19	0.05	0.01	0.01

Table B.1: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=256 and Nc=64, and averaging attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.02	0.07	0.91	0	0
ℓ=3	0.29	0.29	0.24	0.18	0
ℓ=4	0.73	0.20	0.05	0.01	0.01

Table B.2: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=256 and Nc=64, and interleaving attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.02	0.06	0.92	0	0
ℓ=3	0.32	0.30	0.23	0.15	0
ℓ=4	0.79	0.16	0.04	0.01	0

Table B.3: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=256 and Nc=64, and maximum attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.02	0.07	0.91	0	0
ℓ=3	0.17	0.22	0.28	0.33	0
ℓ=4	0.52	0.28	0.12	0.06	0.02

Table B.4: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=256 and Nc=64, and moderated minority extreme attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.01	0.04	0.95	0	0
ℓ=3	0.18	0.20	0.21	0.41	0
ℓ=4	0.52	0.21	0.13	0.07	0.07

Table B.5: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=1024 and Nc=256, and averaging attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.01	0.04	0.95	0	0
ℓ=3	0.17	0.21	0.21	0.41	0
ℓ=4	0.51	0.24	0.12	0.07	0.06

Table B.6: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=1024 and Nc=256, and interleaving attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.01	0.04	0.95	0	0
ℓ=3	0.21	0.21	0.22	0.36	0
ℓ=4	0.60	0.21	0.10	0.06	0.03

Table B.7: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=1024 and Nc=256, and maximum attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0	1.00	0	0	0
ℓ=2	0.01	0.04	0.95	0	0
ℓ=3	0.08	0.13	0.19	0.60	0
ℓ=4	0.24	0.22	0.19	0.17	0.18

Table B.8: The conditional probabilities $P(D|\ell)$ for AWC watermarking with Nv=1024 and Nc=256, and moderated minority extreme attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0.14	0.83	0.03	0	0
ℓ=2	0.26	0.29	0.44	0.01	0
ℓ=3	0.41	0.29	0.21	0.09	0
ℓ=4	0.41	0.36	0.19	0.04	0

Table B.9: The conditional probabilities $P(D|\ell)$ for robust and secure watermarking, and averaging attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0.14	0.83	0.03	0	0
ℓ=2	0.27	0.29	0.43	0.01	0
ℓ=3	0.40	0.30	0.23	0.07	0
ℓ=4	0.40	0.36	0.20	0.04	0

Table B.10: The conditional probabilities $P(D|\ell)$ for robust and secure watermarking, and interleaving attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0.14	0.83	0.03	0	0
ℓ=2	0.27	0.29	0.43	0.01	0
ℓ=3	0.41	0.30	0.21	0.08	0
ℓ=4	0.43	0.31	0.19	0.07	0

Table B.11: The conditional probabilities $P(D|\ell)$ for robust and secure watermarking, and maximum attack.

P	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
ℓ=1	0.14	0.83	0.03	0	0
ℓ=2	0.26	0.29	0.44	0.01	0
ℓ=3	0.41	0.29	0.22	0.08	0
ℓ=4	0.29	0.63	0.07	0.01	0

Table B.12: The conditional probabilities $P(D|\ell)$ for robust and secure watermarking, and moderated minority extreme attack.

Appendix C

Annexe 2 for Chapter 5

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.62	0.37	0	0
3	0.03	0.48	0.48	0.01	0
4	0.05	0.41	0.51	0.03	0
5	0.07	0.37	0.52	0.04	0
6	0.10	0.34	0.52	0.04	0
10	0.15	0.28	0.50	0.07	0
15	0.20	0.26	0.46	0.08	0
20	0.24	0.25	0.43	0.08	0

Table C.1: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=256 and Nc=64, and averaging attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.63	0.36	0	0
3	0.03	0.49	0.47	0.01	0
4	0.05	0.42	0.51	0.02	0
5	0.07	0.37	0.52	0.04	0
6	0.09	0.35	0.52	0.04	0
10	0.15	0.29	0.50	0.06	0
15	0.20	0.27	0.46	0.07	0
20	0.23	0.25	0.44	0.08	0

Table C.2: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=256 and Nc=64, and interleaving attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.62	0.37	0	0
3	0.03	0.48	0.48	0.01	0
4	0.06	0.41	0.51	0.02	0
5	0.08	0.37	0.52	0.03	0
6	0.10	0.34	0.52	0.04	0
10	0.16	0.29	0.50	0.05	0
15	0.21	0.26	0.46	0.06	0
20	0.25	0.25	0.43	0.07	0

Table C.3: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=256 and Nc=64, and maximum attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.63	0.36	0	0
3	0.02	0.48	0.48	0.02	0
4	0.03	0.41	0.51	0.05	0
5	0.05	0.36	0.53	0.06	0
6	0.06	0.33	0.53	0.08	0
10	0.10	0.27	0.52	0.11	0
15	0.13	0.25	0.49	0.13	0
20	0.15	0.24	0.46	0.15	0

Table C.4: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=256 and Nc=64, and moderated minority extreme attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0	0.62	0.38	0	0
3	0.02	0.46	0.49	0.03	0
4	0.03	0.39	0.52	0.06	0
5	0.04	0.34	0.54	0.08	0
6	0.06	0.31	0.53	0.10	0
10	0.10	0.25	0.51	0.14	0
15	0.13	0.23	0.47	0.16	0.01
20	0.15	0.21	0.45	0.18	0.01

Table C.5: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=1024 and Nc=256, and averaging attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.63	0.36	0	0
3	0.03	0.49	0.47	0.01	0
4	0.05	0.42	0.51	0.02	0
5	0.07	0.37	0.52	0.04	0
6	0.09	0.35	0.52	0.04	0
10	0.15	0.29	0.50	0.06	0
15	0.19	0.27	0.46	0.07	0
20	0.23	0.26	0.43	0.08	0

Table C.6: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=1024 and Nc=256, and interleaving attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.62	0.37	0	0
3	0.03	0.48	0.48	0.01	0
4	0.06	0.41	0.51	0.02	0
5	0.08	0.37	0.52	0.03	0
6	0.10	0.34	0.52	0.04	0
10	0.16	0.29	0.50	0.05	0
15	0.21	0.27	0.46	0.06	0
20	0.25	0.25	0.43	0.07	0

Table C.7: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=1024 and Nc=256, and maximum attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.01	0.63	0.36	0	0
3	0.02	0.48	0.48	0.02	0
4	0.03	0.41	0.51	0.05	0
5	0.05	0.36	0.53	0.06	0
6	0.06	0.33	0.53	0.08	0
10	0.10	0.27	0.52	0.11	0
15	0.13	0.25	0.49	0.13	0
20	0.15	0.24	0.46	0.15	0

Table C.8: The conditional probabilities $P(D|c)$ for the AWC watermarking with Nv=1024 and Nc=256, and moderated minority extreme attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.19	0.62	0.19	0	0
3	0.22	0.52	0.25	0.01	0
4	0.24	0.47	0.27	0.02	0
5	0.26	0.44	0.28	0.02	0
6	0.27	0.42	0.28	0.03	0
10	0.30	0.37	0.29	0.04	0
15	0.32	0.36	0.28	0.04	0
20	0.33	0.34	0.28	0.05	0

Table C.9: The conditional probabilities $P(D|c)$ for the robust and secure watermarking, and averaging attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.19	0.61	0.19	0.01	0
3	0.22	0.52	0.25	0.01	0
4	0.24	0.47	0.27	0.02	0
5	0.26	0.44	0.28	0.02	0
6	0.27	0.42	0.29	0.02	0
10	0.30	0.38	0.29	0.03	0
15	0.32	0.36	0.28	0.04	0
20	0.33	0.35	0.28	0.04	0

Table C.10: The conditional probabilities $P(D|c)$ for the robust and secure watermarking, and interleaving attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.19	0.61	0.19	0.01	0
3	0.23	0.52	0.24	0.01	0
4	0.25	0.47	0.26	0.02	0
5	0.26	0.44	0.28	0.02	0
6	0.27	0.42	0.28	0.03	0
10	0.31	0.38	0.28	0.03	0
15	0.32	0.35	0.28	0.04	0
20	0.34	0.34	0.27	0.05	0

Table C.11: The conditional probabilities $P(D|c)$ for the robust and secure watermarking, and maximum attack.

c	$D=0$	$D=1$	$D=2$	$D=3$	$D=4$
2	0.19	0.61	0.19	0.01	0
3	0.22	0.52	0.25	0.01	0
4	0.24	0.47	0.27	0.02	0
5	0.26	0.44	0.28	0.02	0
6	0.27	0.43	0.28	0.02	0
10	0.29	0.39	0.29	0.03	0
15	0.30	0.38	0.27	0.04	0
20	0.32	0.38	0.26	0.04	0

Table C.12: The conditional probabilities $P(D|c)$ for the robust and secure watermarking, and moderated minority extreme attack.

References

[1] `http://en.wikipedia.org/wiki/PPLive`. 2

[2] `http://en.wikipedia.org/wiki/PPStream`. 2

[3] `http://www.havocscope.com/movie-piracy-market-value/`. 2

[4] S. Mooney W. Rosenblatt, W. Trippe. *Digital Rights Management: Business and Technology*. John Wiley & Sons, Inc., November 2001. 3

[5] P. Bas and A. Westfeld. Two key estimation techniques for the broken-arrows watermarking scheme. In *Proc. of 11th ACM Multimedia and Security Workshop, Princeton, USA*, September 2009. 4, 8, 14, 59, 60, 61, 62, 63, 72, 128, 129

[6] N. R. Wagner. Fingerprinting. In *Proc. of the IEEE Symposium on Security and Privacy*, pages 18–22, Oakland, California, USA, 1983. 5, 18, 122, 123

[7] H. Zhao, M. Wu, Z. J. Wang, and K. J. R. Liu. Performance of detection statistics under collusion attacks on independent multimedia fingerprints. *IEEE Transactions on Image Processing*, 14, Issue: 5:646–661, May, 2005. 5, 17, 23, 123

[8] H. Zhao, M. Wu, Z. J. Wang, and K. J. R. Liu. Forensic analysis of nonlinear collusion attacks for multimedia fingerprinting. *IEEE Transaction on Image Processing*, 14, no.5:646–661, May 2005. 5, 17, 18, 123

[9] H. G. Schaathun. Novel attacks on spread-spectrum fingerprinting. *EURASIP Journal of Information Security*, Vol. 2008, Article ID 803217:15 pages, 2008. 5, 18, 123

[10] C. Meadows, G. R. Blakley, and G. B. Purdy. Fingerprinting long forgiving messages. *Lecture Notes Computer Science*, 218, Jan. 1986. 5, 123

[11] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *IEEE Trans. Inform. Theory*, 44:1897–1905, September 1998. 5, 11, 16, 18, 19, 21, 24, 26, 92, 122, 123

[12] Y. Yacobi. Improved boneh-shaw content fingerprinting. *Lecture Notes Computer Science*, 2020:378391, 2001. 6, 123

[13] D. To, R. Safavi-Naini, and Y. Wang. A 2-secure code with efficient tracing algorithm. *Lecture Notes Computer Science*, 2551:149162, 2002. 6, 123

[14] A. Barg, G. R. Blakley, and G. A. Kabatiansky. Digital fingerprinting codes: problem statements, constructions, identification of traitors. *IEEE Trans. Inform Theory*, 49(4):852–865, apr 2003. 6, 19, 123

[15] J. R. Staddon, D. R. Stinson, and R. Wei. Combinatorial properties of frameproof and traceability codes. *IEEE Trans. Inform. Theory*, 47:1042–1049, mar 2001. 6, 124

[16] H. D. L. Hollmann, J. H. van Lint, J-P. Linnartz, and L. M. G. M. Tolhuizen. On codes with identifiable parent property. *Journal of Combinatorial Theory*, 82:121–133, 1998. 6, 20, 124

[17] Z. J. Wang, M. Wu, H. Zhao, W. Trappe, and K. J. R. Liu. Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation. *IEEE Transaction, on Image Processing.*, 14, no.6:804–821, June 2005. 6, 22, 23, 124

[18] W. Trappe, M. Wu, Z. J. Wang, and K. J. R. Liu. Anti-collusion fingerprinting for multimedia. *IEEE Trans. on Signal Processing*, 51(4):1069–1087, April 2003. Special Issue on Signal Processing for Data Hiding in Digital Media and Secure Content Delivery. 6, 24, 25, 124

[19] Z. J. Wang, M. Wu, W. Trappe, and K. J. R. Liu. Group-oriented fingerprinting for multimedia forensics. *EURASIP Journal on Applied Signal Processing,Special Issue on Multimedia Security and Rights Management*, vol.2004:14:pp.2153–2173, October 2004. 6, 25, 26, 124

[20] S. He and M. Wu. Joint coding and embedding techniques for multimedia fingerprinting. *IEEE Transaction. on Infomation Forensics and Security*, 1, no. 2:231–247, June, 2006. 6, 25, 26, 124

[21] S. He and M. Wu. Adaptive detection for group-based multimedia fingerprinting. *IEEE Signal Processing Letters*, 14, Issue: 12:964–967, December, 2007. 6, 26, 124

[22] H. Gou and M. Wu. Data hiding in curves with applications to map fingerprinting. *IEEE Transaction on Signal Processing*, 53:3988–4005, Octobre. 2005. 6, 124

[23] S. He and M. Wu. Collusion-resistant video fingerprinting for large user group. *IEEE Transactions on Information Forensics and Security*, 2, Issue: 4:697–709, December, 2007. 6, 26, 124

[24] G. Tardos. Optimal probabilistic fingerprint codes. In *Proc. of the 35th annual ACM symposium on theory of computing*, pages 116–125, San Diego, CA, USA, 2003. ACM. 6, 27, 28, 30, 31, 124

[25] B. Skoric, S. Katzenbeisser, and M. Celik. Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes. *Designs, Codes and Cryptography*, 46(2):137–166, February 2008. 6, 28, 30, 93, 106, 124, 132

[26] K. Nuida, M. Hagiwara, H. Watanabe, and H. Imai. Optimal probabilistic fingerprinting codes using optimal finite random variables related to numerical quadrature. arXiv:cs/0610036v2, dec 2006. 6, 30, 31, 124

[27] K. Nuida, S. Fujitsu, M. Hagiwara, T. Kitagawa, H. Watanabe, and K. Ogawa. An improvement of tardos's collusion-secure fingerprinting

codes with very short lengths. In *Proceedings of 17th International Symposium on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes(AAECC-17), Bangalore, India*, December, 2007. 6, 30, 31, 124

[28] K. Nuida, S. Fujitsu, M. Hagiwara, T. Kitagawa, H. Watanabe, K. Ogawa, and H. Imai. An improvement of discrete tardos fingerprinting codes. *Designs, Codes and Cryptography*, 52, Issue 3:339 – 362, 2009. 6, 30, 31, 124

[29] O. Blayer and T. Tassa. Improved versions of tardos' fingerprinting scheme. *Designs, Codes and Cryptography*, 48, Issue 1:79–103, 2008. 6, 30, 124

[30] K. Nuida, M. Hagiwara, H. Watanabe, and H. Imai. Optimization of tardos's fingerprinting codes in a viewpoint of memory amount. In *Proceedings of 9th Information Hiding (IH 2007), Saint Malo, France*, June, 2007. 6, 30, 31, 124

[31] T. Furon, A. Guyader, and F. Cérou. On the design and optimization of tardos probabilistic fingerprinting codes. In *Information Hiding 2008, Santa Barbara, California, USA*, May 2008. 6, 31, 124

[32] A. Charpentier, F. Xie, T. Furon, and C. Fontaine. Expectation maximisation decoding of tardos probabilistic fingerprinting code. In *Proc. of SPIE on Media Forensics and Security XI, San Jose, California, USA*, January 2009. 6, 7, 32, 124

[33] T. Furon and P. Bas. Broken arrows. *EURASIP Journal on Information Security*, 2008. 7, 13, 38, 40, 48, 52, 66, 74, 77, 81, 82, 86, 125, 126, 127

[34] BOWS-2. http://bows2.gipsa-lab.inpg.fr/, 2007. 7, 14, 38, 55, 62, 125, 128

[35] F. Xie, T. Furon, and C. Fontaine. Better security levels for 'broken arrows'. In *Proc. of SPIE Electronic Imaging on Media Forensics and Security XII, San Jose, California, USA*, January, 2010. 8

[36] F. Xie, T. Furon, and C. Fontaine. Towards robust and secure watermarking. In *Proc. of 12th ACM Multimedia and Security Workshop, Rome, Italy*, September, 2010. 8

[37] F. Xie, T. Furon, and C. Fontaine. On-off keying modulation and tardos fingerprinting. In *Proc. of 10th ACM Multimedia and Security Workshop, Oxford, UK*, September 2008. 8

[38] F. Xie, C. Fontaine, and T. Furon. Un schéma complet de traçage de documents multimédia reposant sur des versions améliorées des codes de tardos et de la technique de tatouage. In *GRETSI 2009, 22e Colloque en Traitement du Signal et des Images, Dijon, France*, September, 2009. 8

[39] A. Westfeld. A regression-based restoration technique for automated watermark removel. *Proc. of 10th ACM Multimedia and Security Workshop, Oxford, UK*, September 2008. 13, 43, 55, 59, 60, 126, 128

[40] F. Cayre, C. Fontaine, and T. Furon. Watermarking attack: Security of wss techniques. In I. Cox, T. Kalker, and H.-K. Lee, editors, *Proc. of Int. Workshop on Digital Watermarking*, volume 3304 of *Lecture Notes in Computer Science*, pages 171–183, Seoul, Corea, oct 2004. IWDW'04, Springer-Verlag. Best Paper Award. 14

[41] H. G. Schaathun. Attack analysis for he&wu's joint watermarking/fingerprinting scheme. *IWDW, Springer Lecture Notes in Computer Science*, 3304, December 2007. 18, 26, 123

[42] H. Guth and B. Pfitzmann. Error and collusion secure fingerprinting for digital data. *In Information Hiding'99, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, New York*, 1768:134–145, 2000. 19

[43] R. Safavi-Naini and Y. Wang. Collusion-secure q-ary fingerprinting for perceptual content. In Springer-Verlag, editor, *Proc. Security and Privacy in Digital Rights Management, SPDRM'01*, volume 2320 of *Lecture Notes in Computer Science*, pages 57–75, 2001. 19, 95

[44] D. R. Stinson and R. Wei. Combinatorial properties and construction of traceability schemes and frameproof codes. *SIAM Journal on Discrete Mathematics*, 11:41–53, 1998. 20

[45] M. Fernandez and M. Soriano. Soft-decision tracing in fingerprinted multimedia content. *IEEE Multimedia*, 11(2):38–46, 2004. 20, 21, 100

[46] V. Guruswami and M. Sudan. Improved decoding of reed-solomon and algebraic-geometry codes. *IEEE Trans. Inform. Theory*, 45:1757–1767, sep 1999. 20, 100

[47] A. Silverberg, J. R. Staddon, and J. Walker. Application of list decoding to tracing traitors. *IEEE Trans. Inform. Theory*, 49:1312–1318, may 2003. 20, 100

[48] M. Fernandez and M. Soriano. Identification of traitors using a trellis. In *Proc. Information and Communications Security*, volume 3269 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004. 21

[49] Z. J. Wang, M. Wu, H. Zhao, W. Trappe, and K. J. R. Liu. Resistance of orthogonal gaussian fingerprints to collusion attacks. In *Proc. of Int. Conf. on Acoustics, Speech and Signal Processing*, pages 724–727, Hong Kong, April 2003. IEEE ICASSP'03. 22

[50] I. Cox, J. Kilian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing*, 6(12):1673–1687, December 1997. 23, 36

[51] H. Zhao, M. Wu, Z. J. Wang, and K. J. R. Liu. Nonlinear collusion attacks on independent fingerprints for multimedia. In *Proc. of Int. Conf. on Acoustics, Speech and Signal Processing*, Hong Kong, April 2003. 23

[52] Y. Wu. Linear combination collusion attack and its application on an anti-collusion fingerprinting. In IEEE, editor, *Proc. of Int. Conf. Acoustics, Speech, and Signal Processing, ICASSP'05*, volume II, pages 13–16, Philadelphia, USA, mar 2005. 25

[53] S. He and M. Wu. Performance study on ecc-based collusion-resistant multimedia fingerprinting. In *Proc. of Conferences on Information Sciences and Systems (CISS'04)*, 2004. 25

[54] S. He and M. Wu. Group-oriented joint coding and embedding technique for multimedia fingerprinting. In *SPIE Conference on Security, Watermarking and Stegonography, San Jose, CA*, January 2005. 25, 26

[55] W. S. Lin, S. He, and J. Bloom. Binary forensic code for multimedia signals: resisting minority collusion attacks. In *Proc. of SPIE on Media Forensics and Security XI, San Jose, California, USA*, 2009. 26

[56] C. Peikert, A. Shelat, and A. Smith. Lower bounds for collusion-secure fingerprinting. In *Proceedings of the 14th annual ACM-SIAM symposium on Discrete algorithms, Baltimore, Maryland, USA*, January, 2003. 27

[57] B. Skoric, T. Vladimirova, M. Celik, and J. Talstra. Tardos fingerprinting is better than we thought. *IEEE Transactions on Information Theory*, 54, Issue: 8:3663–3676, August, 2008. 30

[58] M. Hagiwara, G. Hanaoka, and H. Imai. A short random fingerprinting code against a small number of pirates. In *Proceedings of 16th International Symposium on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC-16), Las Vegas, NV, USA*, February, 2006. 31

[59] E. Amiri and G. Tardos. High rate fingerprinting codes and the fingerprinting capacity. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009, New York, NY, USA*, January, 2009. 31

[60] N. P. Anthapadmanabhan, A. Barg, and I. Dumer. On the fingerprinting capacity under the marking assumption. *IEEE Transactions on Information Theory - Special Issue on Information-theoretic Security*, 54, No. 6:2678–2689, Jun 2008. 31

[61] T. Furon, L. Pérez-Freire, A. Guyader, and F. Cérou. Estimating the minimal length of tardos code. In *Proc. of the 11th Information Hiding Workshop, Springer-Verlag, vol. LNCS, Darmstadt, Germany*, June, 2009. 31

[62] F. Cayre, C. Fontaine, and T. Furon. Watermarking security: Theory and practice. *IEEE Trans. Signal Processing*, 53(10):3976 – 3987, oct 2005. 35, 125

[63] T. Furon. A survey of watermarking security. In M. Barni, editor, *Proc. of Int. Work. on Digital Watermarking*, volume 3710 of *Lecture Notes on Computer Science*, pages 201–215, Sienna, Italy, september 2005. Springer-Verlag. 35, 125

[64] L. Pérez-Freire, P. Comesaña, J. R. Troncoso-Pastoriza, and F. Pérez-González. Watermarking security: a survey. *Transactions on Data Hiding and Multimedia Security I*, 4300:41–72, October 2006. 35, 125

[65] P. Moulin and A. Ivanovic. The zero-rate spread-spectrum watermarking game. *IEEE Trans. on Signal Processing*, 51(4):1098–1117, April 2003. Special Issue on Signal Processing for Data Hiding in Digital Media and Secure Content Delivery. 36

[66] H. S. Malvar and D. A. F. Florêncio. Improved spread spectrum: A new modulation technique for robust watermarking. *IEEE Transactions on Signal Processing*, 51:898–905, April 2003. 36

[67] B. Chen and G. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Trans. on Information Theory*, 4:1423–1443, May 2001. 36, 37

[68] I. Cox, M. Miller, and A. McKellips. Watermarking as communication with side information. *Proc. of the IEEE*, 87(7):1127–1141, July 1999. 37

[69] M. H. M. Costa. Writing on dirty paper. *IEEE Trans. on Information Theory*, 29(3):439–441, May 1983. 37

[70] J. Chou, S. Pradhan, and K. Ramchandran. Turbo coded trellis-based constructions for data embedding: channel coding with side information. In *Proc. of the 35th Conf. on Signals, Systems and Computers*, Asilomar,CA,USA, November 2001. 37

[71] M. Miller, G. Doërr, and I. Cox. Applying informed coding and informed embedding to design a robust, high capacity watermark. *IEEE Tran. Image Processing*, 13(6):792–807, jun 2004. 37

[72] P. Moulin and R. Koetter. Data hiding codes. *Proceedings of the IEEE*, dec 2005. 37

[73] L. Pérez-Freire, F. Pérez-González, T. Furon, and P. Comesaña. Security of lattice-based data hiding against the known message attack. *IEEE Trans. on Information Forensics and Security*, 1((4)):421–439, dec 2006. 37

[74] L. Pérez-Freire. *Digital Watermarking Security*. PhD thesis, University of Vigo, Spain, 2008. 37

[75] L. Pérez-Freire and F. Pérez-González. Spread spectrum watermarking security. *IEEE Transactions on Information Forensics and Security*, 4(1):2–24, March 2009. 37, 61

[76] L. Pérez-Freire and F. Pérez-González. Security of lattice-based data hiding against the watermarked only attack. *IEEE Transactions on Information Forensics and Security*, 3(4):593–610, December, 2008. 37

[77] P. Bas and G. Doërr. Practical security analysis of dirty paper trellis watermarking. In T. Furon, F. Cayre, G. Doërr, and P. Bas, editors, *Information Hiding: 9th international workshop*, volume 4567 of *Lecture Notes in Computer Science*, Saint-Malo, June 2007. Springer Verlag. 37

[78] F. Cayre and P. Bas. Kerckhoffs-based embedding security classes for woa data hiding. *IEEE Transactions on Information Forensics and Security*, 3:1–15, 2008. 37, 125

[79] B. Mathon, P. Bas, and F. Cayre. Practical performance analysis of secure modulations for WOA spread-spectrum based image watermarking. In *Proceedings of the 9th ACM workshop on Multimedia & security*, pages 237–244, New York, NY, USA, 2007. 38

[80] B. Mathon, P. Bas, F. Cayre, and B. Macq. Optimization of natural watermarking using transportation theory. In *ACM Multimedia and Security Workshop*, Princeton , NJ, USA, Jun 2009. 38

[81] M. Miller and J. Bloom. Computing the probability of false watermark detection. In A. Pfitzmann, editor, *Proc. of the third Int. Workshop on Information Hiding*, pages 146–158, Dresden, Germany, September 1999. Springer Verlag. 40

[82] A. V. Knyazev and M. E. Argentati. Principal angles between subspaces in an a-based scalar product: Algorithms and perturbation estimates. *SIAM Journal on Scientific Computing*, Volume 23, Issue 6:2008 – 2040, 2002. 61

[83] N. X. Vinh, J. Epps, and J. Bailey. Information theoretic measures for clusterings comparison: Is a correction for chance necessary? In *the 26th International Conference on Machine Learning (ICML'09), Montreal, Canada*, 2009. 64

[84] T. Furon. A constructive and unifying framework for zero-bit watermarking. *IEEE Trans. Information Forensics and Security*, 2(2):149–163, jun 2007. 77

[85] P. Comesa na, M. Barni, and N. Merhav. Asymptotically optimum embedding strategy for one-bit watermarking under gaussian attacks. In *Proc. of SPIE-IS&T Electronic Imaging, SPIE*, San Jose, CA, USA, jan 2008. 77

[86] A. Désolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. *IEEE Trans. on Pattern Analysis and Machine Learning*, 25(4):508–513, April 2003. 77

[87] G. R. Shorak and J. A. Wellner. *Empirical Processes With Applications to Statistics*. Wiley, 1986. 80

[88] L. Pérez-Freire and T. Furon. Blind decoder for binary probabilistic traitor tracing codes. In *Proceedings of First IEEE International Workshop on Information Forensics and Security*, pages 56–60, London, UK, December 2009. IEEE WIFS'09. 89, 132

[89] M. Gursoy, H. Poor, and S. Verdú. On-off frequency-shift keying for wideband fading channels. *EURASIP Journal on wireless communications and networking*, 2006(ID 98564):15 pages, 2006. 95

[90] M. Barni. Shedding light on some possible remedies against watermark desynchronization: a case study. In *Proceedings of SPIE Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 106–113, January 2005. 118

[91] C. De Roover, C. De Vleeschouwe, F. Lefebvre, and B. Macq. Robust video hashing based on radial projections of key frames. *IEEE Transactions on Signal processing*, pages 4020–4038, 2005. 118

[92] B. Pfitzmann and M. Schunter. Asymmetric fingerprinting. In LNCS 1070, editor, *Advances in Cryptology*, pages 84–95, Berlin, Germany, 1996. EUROCRYPT'96, Springer-Verlag. 118

[93] B. Pfitzmann and M. Waidner. Anonymous fingerprinting. In LNCS, editor, *Proc. of Eurocrypt'97*, pages 88–102, Konstanz, Germany, May 1997. Springer-Verlag. 118

[94] R. Safavi-Naini and Y. Wang. Sequential traitor tracing. *IEEE trans. Inform. Theory*, 49(5):1319–1326, may 2003. 118

Résumé

Cette thèse porte sur la conception d'une technique de tatouage sûr et robuste dans le contexte du traçage de documents multimédia, et propose un système complet du traçage de traîtres. Ces travaux s'appuient sur la technique de tatouage zéro-bit robuste `Broken Arrows', dont nous proposons des améliorations afin de la rendre plus robuste, notamment à l'attaque de débruitage de A. Westfeld, et plus sûre. Sa sécurité est renforcée au regard des attaques connues et publiées, telles que l'attaque de regroupement de A. Westfeld et l'attaque d'estimation de sous-espace de P. Bas. Par ailleurs, nous étendons sa sécurité en considérant des attaques non publiées. Nous proposons ainsi une nouvelle technique de tatouage sûr, basé sur un test de décision `a contrario' et une insertion avec maximisation sous contraintes d'imperceptibilité et de sécurité. Nous proposons dans le dernier chapitre un schéma complet de traçage de documents multimédia, basé sur les codes de Tardos en version q-aire symétrique et les techniques du tatouage améliorées mentionnées plus haut. Comme les techniques du tatouage sont très robustes, nous pouvons détecter plusieurs symboles en un seul bloc attaqué, ce qui nous permet de proposer deux méthodes d'accusation étendues pour le code de Tardos. Les études expérimentales montrent les bonnes performances de notre schéma de traçage, y compris face à des attaques jusqu’alors mal gérées comme la fusion de documents.

Abstract

This thesis focuses on the design of a robust and secure watermarking technique in the context of multimedia fingerprinting for traitors tracing; and proposes a complete multimedia fingerprinting system. Our work builds on a robust zero-bit watermarking technique `Broken Arrows', and we propose two improvements to make it more robust, especially, to prevent A. Westfeld's denoising attack. Its security aspect is also strengthened in view of some known and published security attacks, such as A. Westfeld's clustering attack and P. Bas's subspace estimation attack. Furthermore, we extend its security by considering some unpublished attacks, introducing a novel secure watermarking scheme, which is based on `a contrario' decision test and a maximization embedding method constrained by imperceptibility and by security. Finally, we propose a complete multimedia fingerprinting system, which is based on the symmetric q-ary Tardos fingerprinting code, and these improved watermarking techniques. As these watermarking techniques are very robust, we can detect multiple fingerprints in one attacked block; this allows us to propose two extended accusation methods for the Tardos code. The experimental investigations show the good performance of our design; it can even deal with the previously poorly managed attacks like the fusion of documents.