Hierarchical Traceability of multimedia documents

Faten Chaabane

Republic of Tunisia
Ministry of Higher Education
and Scientific Research

University of Sfax
National Engineering School of Sfax

Graduate School
Sciences and Technologies

Thesis
Computer Systems Engineering (ISI)

Order N° : 279/16

THESIS

Presented at

National Engineering School of Sfax

in order to obtain the Degree of

DOCTORATE

in

Computer Systems Engineering

By

Faten Chaabane

HIERARCHICAL TRACEABILITY OF MULTIMEDIA DOCUMENTS

Defended on March 13, 2017, in front of the Examination Committee composed of

Chair	**Mr. Faiez Gargouri**	*Professor, ISIMS, University of Sfax, Tunisia*
Reviewer	**Mr. Walid Mahdi**	*Associate Professor, ISIMS, University of Sfax, Tunisia*
Reviewer	**Mme. Caroline Fontaine**	*Professor, CNRS, Lab-STICC-CID, Telecom Bretagne, France*
Examiner	**Mr. Afif Masmoudi**	*Professor, FSS, University of Sfax, Tunisia*
Supervisor	**Mr. Chokri Ben Amar**	*Professor, ENIS, University of Sfax, Tunisia*
Co-supervisor	**Mme Maha Charfeddine**	*Assistant Professor, ENIS, University of Sfax, Tunsia*
Guest	**Mr. William Puech**	*Professor, University Montpellier II – LIRMM, France*

Dedications

I dedicate this thesis:

To my parents *Mohamed* and *Bahija*,
Who by their sweetness and encouragement inspired me to achieve what they expected and desired.

To my beloved daughter *Khadija*,
She lights up my life every day.

To my husband *Slim*,
For the sacrifices deployed to me, for his prodigious advice, his understanding and endless support.

To my brother *Ahmed Amine* and my sister *Fadoua*
Who have been given me encouragement and moral support.

To my friend *Sourour*
Who has never failed to inspire me, even in the most difficult moments.

To my father in-law *Mustapha* and my mother in-law *Zohra*,
For their encouragement and their patience that resemble flowers crowning my career.

To my fallen sister in-law *Fatma*,
I wish you could have been there.

To my sisters in-law *Mouna* and *Mariem*, and my brother in-law *Mohamed Ali*
You always showed me the greatest respect.

To
All my friends and all those who felt one day to me a sense of respect and love.

Acknowledgments

The writing of this dissertation has been one of the most significant challenges in my academic life. Without the support and guidance of a number of people, this study would not have been completed. It is to them that I am greatly indebted.

I would like to express my deepest appreciation to Professor *Chokri Ben Amar*, my supervisor and director of *REGIM-Lab* (REsearch Groups on Intelligent Machines), to Doctor *Maha Charfeddine*, my co-supervisor and my friend whose expertise, understanding, and patience, added considerably to my graduate experience. They have provided me with incessant support in each of my steps during my thesis studies. They have always been the source of inspiration and motivation to me.

I am also indebted to Prof. *William Puech*, Professor at *LIRMM, Montpellier, France* for his welcome in Lirmm, his support, his human qualities, his willingness, his scientific rigor and breadth of his knowledge.

I would also like to thank Dr. *Caroline Fontaine*, CNRS full time researcher at both *the Lab-STICC CNRS lab* (co-leader of the SFIIS team), and *the ITI department* of *Telecom Bretagne* and Dr. *Walid Mahdi*, Senior Lecturer at *Higher Institute of Computer Science and Multimedia, Sfax University*, for accepting to be the reviewers of this thesis and for their relevant comments on my research work.

My gratitude goes to Prof. *Faiez Gargouri*, Professor and Director of the Higher Institute of Computer Science and Multimedia at *ISIMS-Sfax University*, for the honor he had accorded me for agreeing to be the committee chair of this thesis. My distinguished thanks go also to Prof. *Afif Masmoudi*, Professor at *FSS-Sfax*, for the valuable service to examine this work and to be a member of my committee.

I would like to sincerely thank all my colleagues in the *REGIM-Lab*: Hanene, Eya, Nesrine, Imen, Wiam, Mouna, Rim at *University of Sfax* for their helpful suggestions on many occasions.

Of course, I could not have achieved this purpose without making deep gratitude to teachers and researchers who have guided me in my quest for knowledge during my years of studies.

Last but not least, I could not have made a success of my work without my family; words can never express my gratitude to them for their patience, care, encouragements, and faith in what I am doing.

In short, and not to forget anyone, I take this opportunity to dedicate my thanks to all those who helped and encouraged me during my university studies.

Contents

CHAPTER 1

Introduction

Contents

1.1 Motivation

The ever growing of the digital era is tightly tied to the diversity of digital tools, the availability of Internet high speed connections and the encroachment of Peer to Peer networks. One key evidence is the affordability and the massive use of digital tools such personal computers, camcorders and mobile devices: smart phones and tablets. In fact, this digital invasion has really facilitated our daily lives and works but it has also deeply influenced the multimedia consumers' behaviors. Henceforth, it has enhanced the multimedia consumption in terms of multiplying shared media contents through multimedia distribution platforms, especially for *Video On Demand service*, *VOD* and even in terms of paid TV programs. According to the Future Market Insights, the *FMI* report, the global VoD service market was estimated at US 45.03 Billion of dollars in 2014 and is expected to expand at a CAGR [1] of 8.3 percent during the forecast period between 2016 and 2026. The major shift observed in users' preferences was the migration from the television's linear schedule to viewing content as per their preference and increasing the use of high speed Internet network even in emerging countries. One other

[1] Compound annual growth rate

example is, based on the digital TV report, the Pay TV subscriptions for 338 operators across 89 countries in the world will increase by 200 million from a collective 704 million in 2014 to 904 million by 2020, and according to a new report from Digital TV Research the number of subscribers to the NetFlix service [2], has reached 115 million in 2016 and 306 million across 200 countries by 2020. Unfortunately, the emerging digital technology phenomenon hides a volte-face. Henceforth, it is obliquely the origin of several unauthorized manipulations of digital content: multiple duplications, illegal redistribution and arbitrary modification. Indeed, the readiness of copying perfectly any type of digital content (image, text, video, audio file) to share it or to redistribute it encourages and leads to a more dangerous phenomenon which is the copyright violation, well-known as the digital piracy. As an example and according to studies published in ComScore and Nielsen for digital music market, the *IFPI* [3] estimates that 20 percent of fixed-line internet users are constantly accessing services providing copyright infringing music. Digital piracy is incessantly evolving and includes the distribution of unauthorized music through platforms such Tumblr and Twitter, unlicensed cyberlockers [4] and Bitorrent file sharing. IFPI estimates that in 2014 there were from billion music downloads via Bitorrent alone, the vast majority of which are infringing and this does not take into account other channels such as cyberlockers, linking sites and social networks.

Facing the piracy problem through the internet and Peer to Peer networks represent an important challenge especially to the software industry. In this context, it was necessary for media holders to find remedies to prevent the digital content from any type of fraud. Henceforth, the digital rights management systems, *DRM* have involved a set of measures to control the use of digital content [de Rosnay 2002]. More than one approach has succeeded in the literature to cope with that issue, the first one was attached to the cryptography community and proposed to encrypt the media content so that only the holder or the authorized users are able to access to the content [Cormen 2013]. The limit of this type of approach consists in the fact that once the decryption proceeds, the original version of the content can be easily accessed and its pro-

[2] Netflix is one of the most popular Internet television network including more than 75 million members in over 190 countries enjoying more than 125 million hours of TV shows and movies per day. Members can watch as much as they want, anytime, anywhere, on nearly any Internet-connected screen. Members can play, pause and resume watching, all without commercials or commitments.

[3] International Federation of the Phonographic Industry

[4] An online third-party service that offers file storing and sharing services

tection will be lost. The second proposed approach belongs to the multimedia forensics and its principle is to hide a watermark in the host signal (the media content) to ensure its protection and to check whether the media content is illegally redistributed [Thilagavathi *et al.* 2015].

The digital watermarking techniques belong to the multimedia forensics and consist in using the embedded watermark to identify the media owner. Furthermore, the change performed by the watermark insertion makes the copyright infringement harder but does not alter its quality. In the recent years, with the evolution of cloud and networks, the watermarking techniques can be easily circumvented by a group of experienced users who try to cooperate together to create and share illegally a new copy with unknown fingerprint. Henceforth, the watermarking has become insufficient when the target is not only the prevention but also the detection of colluders.

The digital fingerprinting techniques introduced the notion of collusion resilient secure fingerprints, they are also called "traitor tracing" approaches. By compensating the weaknesses of the watermarking techniques when considered the detection of the actors of a collusion trial, traitor tracing techniques were proposed not only to prevent media copyright infringement but also to detect who has contributed to the distribution of an illegal copy in multimedia distribution networks [Fontaine 2011]. In this thesis work, we are interested especially in traitor tracing process proposed for multimedia distribution platforms. In fact, such fingerprinting scheme consists of two basic components: the watermarking layer and the collusion-secure fingerprinting code where the latter is embedded in the digital content and has a structure which enables to trace actors of a piracy trial if it is discovered by the media holder [Charpentier *et al.* 2011]. In Figure 1.1, a generic example of traitor tracing in video distribution platform is depicted. In side (1), one principal actor is the media supplier whose role is to prepare a dictionary of identifiers $ID_{i,i\in\{1..k\}}$ to embed in sold releases before their distribution. Each identifier should be unique, specific to each legitimate user and should also allow the tracing back of users. These identifiers, named tracing codes are then embedded into media releases by a watermarking technique to ensure their owners' identification. In the case of collusion attack made by group of users, a suspicious copy similar to the distributed ones is discovered, this copy hides an identifier which is different from the other generated identifiers. The aim is then to analyze this video and to proceed by decoding the retrieved identifier, as shown in side (2) to retrieve at least one of collusion actors.

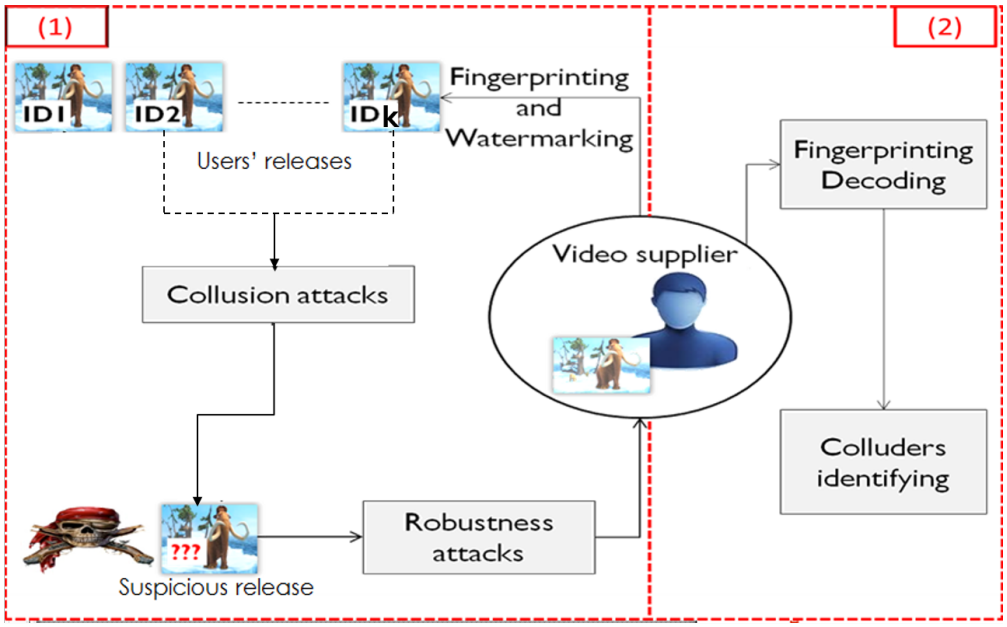


Figure 1.1: The general scenario of tracing process in a multimedia distribution platform.

1.2 Prior approaches and thesis goals

The general definition of "fingerprinting" consists in the inherent characteristics used to personalize an object from other similar ones [Wagner 1983]. In the history, several manipulations were proposed in fingerprinting field especially for security target:

As a first example, the fingerprinting was used in Neper logarithmic tables [Fontaine 2011]: John Neper has integrated an identification mechanism to protect the contents of algorithmic tables. This mechanism consists of a set of errors introduced to each table, which allows for differing a table from another, so if copies of one table are distributed, it is simple to identify the first owner of this table.

Another example of using the fingerprinting technique is the example of Margaret Thatcher who, after a press publication of confidential documents, has delivered different reports to her collaborators to find readily traitors.

Based on [Wagner 1983], the digital fingerprinting was firstly introduced in [Blakley *et al.* 1986]. When considered the cryptographic community, the fingerprinting was deeply studied especially in broadcast encryption system which allows a broadcaster to

encrypt the content so that only chosen users are able to decrypt the content. In this type of systems, a specific set of decryption keys is assigned to each decoder device [Liu 2005].
In this thesis report, we are interested in fingerprinting in general, which principle is to trace pirates by personalizing an object, even decoder devices. The notion of digital fingerprinting has been introduced by Wagner in [Wagner 1983] which consists in embedding a fingerprint in a document by using a watermarking technique, for copyright protection target. Recently, with the growing of the digital market and the important economic value of some digital content, it becomes basic to provide a relevant protection system where the media is prevented from any illegal redistribution. Therefore, the fingerprinting has been integrated into distribution process of digital contents. Indeed, it implies the presence of two essential components: a fingerprinting code and a tracing algorithm. The fingerprinting code is a specific message which is assigned to a unique user and embedded in each release of the media for identification and protection targets [Trappe *et al.* 2003]. Moreover, the tracing algorithm is a process whose role is to allow the retrieval back of traitorous users when considering the suspicious copy. Furthermore, another important part in a fingerprinting system is the embedding technique which seems to be relevant as it could be circumvented by several users to attack the content and create the pirated copy [He & Wu 2007]. In this context, the traitor tracing system should counter piracy trials which can be operated to the content or to the embedded codes [Chaabane *et al.* 2013]. Different fingerprinting techniques have been proposed in the literature [Boneh & Shaw 1998], [Hayashi *et al.* 2007]. Nevertheless, the majority of actual tracing work uses the well known Tardos code [Akashi *et al.* 2008]. But, although it has been proven that Tardos code provides a good compromise between the code length and the collusion size [Tardos 2003] [Peikert *et al.* 2003], it still requires improvement with regard to the complexity decoding. This weakness is unequivocal in case of multimedia distributing systems which involve a large size of audience and are expected to use a low complex pirates' retrieving process. Henceforth, the main concern which still requires researchers' interest is how to find a tracing strategy with fewer complexity costs for a Tardos-based tracing system in such applications. In some work, faced with a large number of users in multimedia distribution systems, some researchers [Wang *et al.* 2004],[He & Wu 2007] have assumed that users having the same geographic belonging have more probability to cooperate together in a collusion trial. Hence, they proposed to group users according to this criterion. The group-based choice

should provide a less complex Tardos decoding step in the accusation process. In fact, instead of parsing all the users' codewords, only the suspicious group is parsed. Although this strategy affords an interesting idea to minimize the accusation process costs [Wang *et al.* 2004], it was applied in case of independent signals only [Shahid *et al.* 2013].

Hence, the main objective of this thesis is to propose a multi-level hierarchical structure for Tardos fingerprints able to provide good detection rates even if the number of users is increasing. More precisely, instead of using the only geographic criterion proposed in [Wang *et al.* 2004], we studied the threat model in a VOD platform [Choi *et al.* 2012] to enlarge the criteria of the group-based construction step. Then, we chose to adapt a two-level tracing strategy which is closely attached to the group-based users' identifiers. While the majority of the existing fingerprinting schemes used image watermarking techniques [Shahid *et al.* 2013] [Ye *et al.* 2013] [Kuribayashi 2014], we chose to apply an original tracing scheme which enables a joint between the audio watermarking technique providing good robustness and inaudibility results and the two-level tracing code. This tracing scheme benefits from the hierarchical structure of fingerprints to provide a fast tracing process.

During the thesis work, more than one contribution has been elaborated. These contributions, detailed in the report, are summarized as follows:

- The construction of a multi-level hierarchical fingerprint: In this contribution, the main objective is to propose a group-based structure for users' identifiers. In fact, the resulting fingerprint is constructed according to a proposed multi-level hierarchy. The applied criteria are selected to be the most relevant ones in the multimedia distribution scenario. In a first step, it was interesting to study the users' behavior in a multimedia distribution platform. Then, according to statistical analysis in [Choi *et al.* 2012], we find that some users' characteristics are highly discriminative than others for the grouping step. Hence, the major novelty in this first contribution is the shifting from one constraint (the geographic one as mentioned in the literature) to several constraints which should fit the group construction and provide it a good refinement. The resulting multi-level hierarchical fingerprint, is compared to a non-hierarchical one and the experimental results have proven that grouping users according to more than one criterion contributes to ameliorate the Tardos tracing results, even for long users' codewords. Moreover, a generated

weight is assigned to each constructed group to refer to its "danger" degree, or rather to the probability of including colluders. Hence, these weights are also applied to accelerate the tracing process.

- The proposition of a two-level tracing strategy for hierarchical fingerprint based on combining the Boneh Shaw probabilistic collusion-secure fingerprinting code to the Tardos one. In this contribution, the main target was to provide a tracing strategy which is suitable to the hierarchical structure of the fingerprints and the group-based assumption. Eventually, the key idea was to use a two-level tracing strategy for the tracing process to focus on the group-based property of users' identifiers. Hence, a first level is applied to the Boneh Shaw code to select groups having higher probability to contain colluders, and a second level with Tardos code to trace guilty users only in selected groups. Although the choice of a two-level tracing strategy was proposed in literature in [Desoubeaux *et al.* 2012] and [Kuribayashi *et al.* 2008], the tracing strategy we propose has two interesting key characteristics: groups of users are not made randomly but are considered regarding the real scenarios of multimedia distribution applications, and a weight-based selection which refines efficiently the group search space to accelerate the tracing process.

- A proposal of optimizing fingerprinting code length based on a QR-code-based conversion. One primary concern of tracing schemes consists in overcoming the length of collusion-secure fingerprinting code. In fact, the application of a watermarking scheme to embed the fingerprinting code depends closely on its length. In case of multimedia of limited payload such as short audio files or images, the provided capacity is restricted to only a few bits, whereas collusion-secure fingerprinting codes typically require thousands of bits. Henceforth, the multimedia distribution applications which enables tracing process with fingerprinting codes is challenging. Thus, in case of our two-level code which consists of a parallel concatenation of two fingerprinting codes, adjusting the code length to the real requirements should significantly make our proposed system more appropriate to multimedia distribution applications. The choice of QR code conversion should be judicious since it enables not only to compensate the code length constraint but also it enhances the robustness to different types of signal processing attacks.

- A migration to a blind traitor tracing scheme based on hierarchical fingerprint is proposed to be closer to the real scenarios where guessing the number of traitors and the strategy they apply is harder. This type of migration was proposed previously in the literature to deal with the invariance of the Tardos decoder which seemed to have a conservative behavior whatever is the colluders' strategy. We propose in this contribution to check the efficiency of the proposed two-level tracing strategy in a blind scheme. When considering the existing estimation-based tracing approaches, an estimation step (based on the Expectation Maximization and Maximum a Posteriori algorithms) was appended as a preprocessing step to the tracing one. While the existing blind schemes were checked only for a simple Tardos-decoder, we propose in our thesis work to integrate the estimation step in our two-level tracing scheme. The resulting tracing scheme takes the advantage of the group-based property of fingerprints to provide more accurate and efficient detection process.

  The proposed contributions through this thesis work led to five communications in international conferences, to two accepted journal papers, as follows:

- Communications in international conferences

  1. [Chaabane *et al.* 2013]: F.Chaabane, M.Charfeddine, C.Ben Amar, "A survey on digital tracing traitors schemes", in Proceeding of the 9th International Conference on Information Assurance and Security, IAS 2013, Gammarth, Tunisia, December 4-6, 2013.
  2. [Chaabane *et al.* 2014]: F.Chaabane, M.Charfeddine, C.Ben Amar, "A Multimedia Tracing Traitors Scheme Using Multi-level Hierarchical Structure for Tardos Fingerprint Based Audio Watermarking", in Proceeding of the 11th International Conference on Signal Processing and Multimedia Applications, SIGMAP 2014, Vienna, Austria, 28-30 August, 2014.
  3. [Chaabane *et al.* 2015b]: F.Chaabane, M.Charfeddine, W.Puech, C.Ben Amar, "A QR-code based audio watermarking technique for tracing traitors", in the 23rd European Signal Processing Conference, EUSIPCO 2015, Nice, France, August 31-September 1-4, 2015

4. [Chaabane *et al.* 2015c]: F.Chaabane, M.Charfeddine, W.Puech, C.Ben Amar, "Towards a blind MAP-based traitor tracing scheme for hierarchical fingerprints", in 22nd International Conference on neural information processing, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015.
5. [Chaabane *et al.* 2015a]: F.Chaabane, M.Charfeddine, C.Ben Amar, " Clustering impact on group-based traitor tracing schemes", 5th International Conference on Intelligent Systems Design and Applications (ISDA), Marrakesh, Morocco, December 14-16, 2015.
6. [Chaabane *et al.* 2016b]: F.Chaabane, M.Charfeddine, W.Puech, C.Ben Amar, "An EM-based estimation for a two-level traitor tracing scheme", in The 2016 IEEE International Conference on Systems, Man, and Cybernetics , Budapest, October 2016, SMC 2016, Budapest, Hungary, October 9-12, 2016

- Communications accepted in international journals

1. [Chaabane *et al.* 2016a]: F.Chaabane, M.Charfeddine, C.Ben Amar, " Novel two-level tracing scheme using clustering algorithm" published in Journal of Information Assurance and Security, JIAS, January 2016.
2. [Chaabane *et al.* 2016c]: F.Chaabane, M.Charfeddine, W. Puech, C.Ben Amar, "A two-stage traitor tracing strategy for hierarchical fingerprints", published in Multimedia tools and Applications journal, MTAP, August 2016.

1.3 Thesis organization

The thesis report is organized in two essential parts:

Part I: Literature review

- In this part, we present the problematic tackled in this thesis work. We begin by detailing the background of the tracing traitor field: its history and its different components mainly: the watermarking technique, the collusion-secure fingerprinting code and the eventual threat channel. Moreover, we detail the different constraints that it should sur-

round to provide an efficient detection step. Finally, we provide a survey of state-of-the-art research works on tracing traitor field and essentially the hierarchical ones.

Part II: Contributions

- Chapter 3 presents the first contribution in this thesis work. In fact, as the fingerprint construction is the first step in a tracing scheme, we focus in this part in the multi-level hierarchical structure we propose for users' identifiers. According to the studied threat channel in multimedia distribution platform, we propose to construct groups of users having similar characteristics and then we assign a weight for each group of users. This weight should be a key constraint during the tracing process.

- Chapter 4 describes an essential contribution in this thesis work where the whole proposed tracing scheme is presented and we focus on the tracing process for which we propose a two-level tracing strategy which is chosen to be closely adapted to the hierarchical structure of the fingerprints.

- In chapter 5, we tackle a key point in the watermarking technique which is reducing codeword length to maintain good imperceptibility quality. Thus, we give an overview of the QR code and its characteristics and we detail on the different steps of the conversion we propose.

- In chapter 6, we study the impact of using an the expectation maximization and the maximum a posteriori algorithms to adjust the accusation functions of the Symmetric version of Tardos code in a two-level tracing scheme.

Finally, we close this thesis report with a summary of the elaborated contributions thesis and the most important results. Future works and perspectives are also presented.
Appendix 1 covers some important results on audio watermarking which confirm the efficiency of the applied audio watermarking technique for the different proposed fingerprints embedded into videos samples.
In Appendix 2, we join deep statistical analysis related to the most important characteristics of multimedia distribution applications: shared media and audience, provided by different sources. This study is fundamental in our choices throughout this thesis work.

Appendix 3 consists of experimental results related to the impact of clustering group identifiers on the detection rates of the proposed tracing system in Chapter 3.

Part I

Literature review

CHAPTER 2

Survey on Traitor tracing

Contents

2.1 Introduction

The easiness of using and manipulating digital media content has a volte face. In fact, although average users can simply be familiar with some manipulations such as a simple duplication, these manipulations can be dangerous with dishonest users whose target is illegal. Manipulating and duplicating digital media content via the Internet and Peer to Peer networks is available even to average users but can be used to unauthorized purposes with dishonest customers. Henceforth, facing the loss caused by unauthorized treatments and protecting the digital content become challenging to the media industry and research has led to different mechanisms of digital content protection.

The earlier approaches in the history were essentially steganography and cryptography which differ in principles and objectives. In fact, the cryptography consists in modifying a message to make it unreadable whereas steganography purpose [Anderson & Petitcolas 1998] is to insert a message in another to enable the secrecy of the transmitted data.

Compared to the other approaches, the digital watermarking is the most recent, since the 1990s

and several models were proposed for images, audio and video contents [Cox *et al.* 1996].
As collusion secure fingerprinting codes are codes inserted as watermarks in the media content and hence enhance the security side of the watermarking scheme, it will be interesting to introduce the different terms related to the context of protection media content before detailing the elaborate contributions of this thesis work.

2.2 The digital watermarking

The digital watermarking [Cox *et al.* 1999] is the technique of hiding information into a digital content (image, audio, video) to protect it from dishonest manipulations, it was proposed for many applications and essentially for the copyright protection one. The hidden information has many nominations: watermark, mark, or signature. The watermarking should respect several requirements, the most important points are known as magic triangle cotes. These requirements are:

- The imperceptibility: the operation of watermarking should be applied without altering the quality of the media and hence a simple user should not differentiate between a watermarked content and another one.

- The robustness: the embedded watermark should be resistant to different types of attacks.

- The capacity: the quantity of information hidden into the media content is one of the major requirements of a watermarking scheme because it is important to find a fair compromise between this quantity and the scheme robustness.

- The complexity: this requirement is tied to the real-time applications as well as it includes the computation time and the complexity of the system required to embed and detect the message watermark.

- The verification: It is related to the public availability of the watermarking algorithm and the secret key, i.e if the watermark can be detected only by a specific person who holds the secret key or it is detected publicly.

A watermarking technique is considered perfect if it finds a compromise between the different requirements, it should be, at the same time, robust, imperceptible and have a high capacity of insertion. Thus, there are many watermarking schemes proposed in the literature and many classifications to classify them [Charfeddine *et al.* 2012].

2.2.1 A brief state of the art of digital watermarking approaches

In the literature, more than one classification was proposed to classify the watermarking techniques. One of these classifications is made according to the transparency of the host signal, the watermarking technique is so blind [Yu & Sattar 2003], [Craver *et al.* 2006], [Zeng & Qiu 2008], [Charfeddine *et al.* 2010], [Kumar *et al.* 2011], [Pradhan *et al.* 2012], [Walia & Suneja 2013] or not [Saha *et al.* 2014].
Another classification is made according to the embedding position of the watermark in the host signal. It can be in the spatial domain [Laftsidis *et al.* 2003], [Lie & Chang 2006], [Martinez-Noriega *et al.* 2010] or in the transform domain like schemes using the DCT transform [Charfeddine *et al.* 2012], or the DWT transform [Pradhan *et al.* 2012], [Nematollahi *et al.* 2012].
While the watermarking was proposed as a suitable solution to the copyright infringement problem, it remains insufficient to the tracing traitors' target, where the issue is retrieving users who have contributed in the piracy operation. The resulting approach is consequently a joint between the watermarking technique and an anti collusion code able to identify each media release [Xie *et al.* 2008]. This class of approach is called the fingerprinting.

2.2.2 The general watermarking scheme

The general watermarking scheme consists of two main steps: the embedding (insertion) and the recovering (detection) of the watermark [Ye *et al.* 2013]. We will explain each step separately.

2.2.2.1 The watermark embedding step

As depicted in Figure 2.1, an owner with a key K embeds the mark W in the original document I. The watermarked document $I^{'}$ seems to be similar to the original document I with an embedded signature W. There are several domains to insert the watermark. In the next part; we present these different domains.

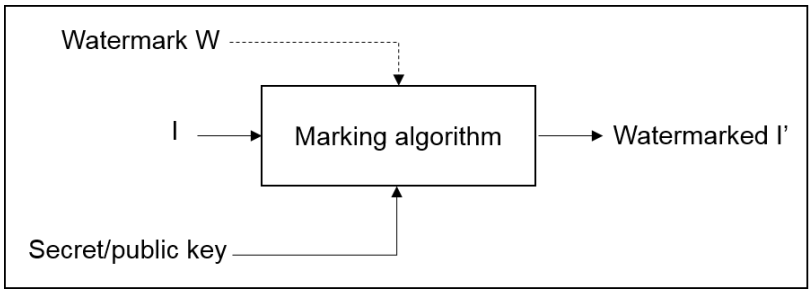


Figure 2.1: The watermark embedding process

The watermark insertion domain One key requirement to differentiate between watermarking techniques is the insertion domain: insertion domain with no transformation, the frequency domain and the multi-resolution one.

- The insertion domain without transformation: This domain depends on the type of watermarking scheme. For the image watermarking scheme, the insertion domain without transformation is the spatial one, it allows to deal perfectly with geometric attacks. For the audio watermarking schemes, this domain is the temporal one, it enables a high capacity of embedding.

- The frequency insertion domain: This domain involves the presence of a transformation step such as the DCT, Discrete Cosine Transform, or the DFT, Discrete Fourier Transform. It is the adopting space transformation in standards: JPEG for images or MPEG for video and Audio. Hence, it enables good robustness to compression attacks.

- the Multi-resolution insertion domain: The advantage of this embedding domain is the fact that it is used in recent compression standards such as JPEG 2000 or MPEG4.

2.2.2.2 The watermark detection step

For this step, as detailed in Figure 2.2, input parameters are the watermarked document I' and the key K (the same one used when inserting). The output of the detection D can be the detected mark W' or the result whether the mark W was found in watermarked document or not. The watermarking scheme can be blind (the original document I is not required for the recovering step) or not.

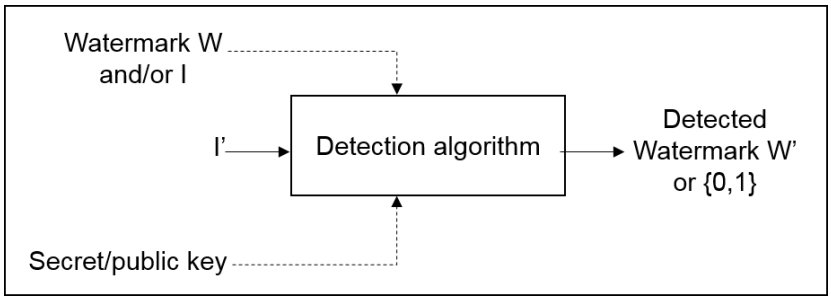


Figure 2.2: The watermark detection process

2.2.3 The watermarking applications

Digital watermarking has more than one application; it is essentially used for copyright protection but also for: content Authentication, broadcast monitoring, indexing, medical application, data embedding. In our thesis, we are interested in watermarking applied for fingerprinting. Indeed, hidden information, or the identifier, directly attached to user identification, is implanted in the media as a watermark. In case of copyright abuse or violation, this signature can help to trace the source of illicit copies and so recover traitors. For this reason, digital watermarking is one of the challenges of a tracing traitors' framework.

2.3 From watermarking to fingerprinting

The digital watermarking techniques consists in hiding information into a digital content. If a digital content in a multimedia distribution application is prone to collusion attack, the watermarking scheme is hence circumvented by this collusion actors. Henceforth, it becomes

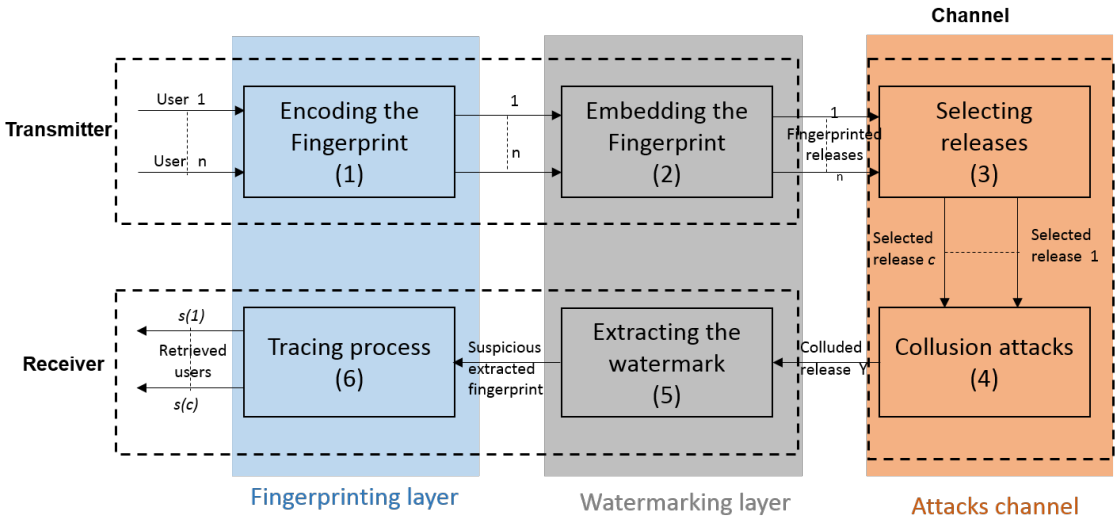


Figure 2.3: The general design of layered multimedia fingerprinting system.

insufficient when target is not only the prevention of the digital content but also the detection of colluders. In our thesis work, we are interested in watermarking applied for fingerprinting. Indeed, hidden information, or the identifier, directly attached to user identification, is implanted in the media as a watermark. In the case of copyright abuse or violation, this signature can help to trace the source of illicit copies and so recover traitors. For this reason, digital watermarking was one of the challenges of a tracing traitors' framework.

2.4 The general multimedia Fingerprinting scheme

From a practical standpoint, the whole multimedia fingerprinting system can be considered as a communication chain with a transmitter side, a channel and a receiver side. The transmitter and the receiver are the main parts of the tracing system while the collusion attacks are presented in the transmission channel.

2.4.1 The main components of a fingerprinting scheme

As numbered in Figure 2.3, the whole fingerprinting framework implies the presence of five essential steps as follows: the fingerprint encoding step, its embedding in the media release, a selection of some releases to participate in a collusion attack and to give rise to a colluded copy,

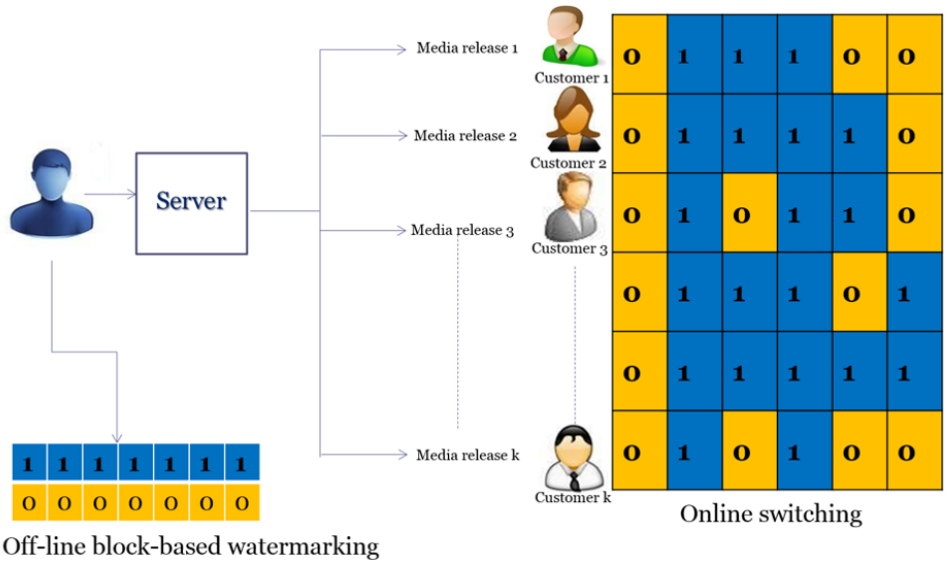


Figure 2.4: The general concept of offline-block based watermarking and online switching.

the extraction of the colluded code and the tracing process to trace back the colluders. We will detail each step separately:

2.4.1.1 The fingerprinting generation step

The fingerprinting generation step is closely attached to the users of the platform. Labeled by (1) in Figure 2.3, it includes the assigning, by the media distributor, of a unique codeword $fingerprint_{i,i\in\{1,\cdots,n\}}$ to each user before using it in the embedding layer to construct the fingerprinted releases $R(f(i)), i \in \{1,\cdots,n\}$. This unique identifier is essentially used to identify each media owner and to protect the media from any illegal treatment [Wagner 1983].
The fingerprinting step can be effectively applied if it is performed as shown in Figure 2.4: the content is divided into blocks and in each block one symbol is hidden. Serialization consists in combining master copies of the original one. These master copies are obtained by the embedding process made in advance offline which consists in embedding in each block the same symbol. It is assumed that the number of symbols is the same than the number of copies. Then, the only step made online is the combination of the different blocks derived from the master copies to yield the fingerprints. The efficiency of the serialization step depends in a first time on the embedding technique, the watermarking technique, whose relevance is shown when consid-

ered the several attacks the media content should survive during a sharing operation. Moreover, the tracing algorithm is also an important part of the fingerprinting layer, it is a process whose role is to allow the retrieval back of traitorous users when considering the extracted codeword from the colluded release Y. The fingerprinting layer depends on the efficiency of the tracing algorithm and even on its robustness to intentional attacks made by colluders. The diversity of the attacks and the large size of the audience in a multimedia distribution platform make the tracing task more complex. Hence, different designing of tracing algorithms have been proposed in the literature [Boneh & Shaw 1998], [Hayashi *et al.* 2007]. In Section 2.4.1.2, we detail more the watermarking concept and its different characteristics. In Section 2.5.1, we review some generalities about the tracing code, called also as anti-collusion code.

2.4.1.2 The fingerprint embedding step

Labeled by (2) in Figure 2.3, the fingerprint embedding step is launched once a customer has confirmed the media content purchase, it consists in assigning a unique fingerprint from the generated dictionary to this customer. This fingerprint is embedded by applying the watermarking insertion algorithm.

2.4.1.3 Collusion channel

The third main part in the fingerprinting system is the threat channel, as labeled by (3) and (4) in Figure 2.3, which refers to the set of attacks made by dishonest users. The objective is to yield a suspicious release of the digital content and to make the accusation step more complex. Different types of attacks were studied by researchers to prevent the fingerprinting system from piracy trials [Furon & Pérez-Freire 2009b], [Mathon *et al.* 2013].
Before detailing the notion of a collusion attack, it is primordial to introduce the Marking Assumption which was proposed by [Boneh & Shaw 1995] as an assumption on which a collusion trial depends in a practical standpoint.

The Marking Assumption The Marking assumption states that, if we see $(\vec{x}_{j1})_i = (\vec{x}_{j2})_i = \ldots = (\vec{x}_{j3})_c = s$ for the same position i and for a symbol s where $s \in \mathrm{Q}$, then it is assumed that the same symbol s is seen for all the actors of the collusion in the position i, which means

that any yielded codeword $\vec{y}$ should have also the symbol s in the position i, $\vec{y}_i = s$. More explicitly, the colluders in a collusion trial, by comparing their respective releases, are not able to change symbols in undetectable fingerprint positions. This assumption is based on the fact that once the fingerprint is hidden, it becomes impossible for colluders to differentiate between the fingerprinted blocks and the other blocks of the digital data. In this context, there are more than one model of attack which can be operated [Furon & Pérez-Freire 2009b]. In Table 2.1, an example of the Marking assumption is easily identified where symbols 0 and 1, respectively in positions 2 and 4 are imposed into the sequence y in the same positions. The collusion attacks

Table 2.1: Example of the Marking assumption for 3 binary codewords

Colluder1	0	**0**	1	**1**
Colluder2	1	**0**	1	**1**
Colluder3	0	**0**	0	**1**
y	0	**0**	1	**1**

are applied under the Marking assumption and other assumptions called "attack models" which describe all the possibilities y_i can have for a position i. These models range from the restricted model to the extended one.

- Restricted model: This model of attack says that y_i should be one of the colluders' symbol in the position i. This model is considered as too restricted and it leads to a deterministic accusation, where no accusation error is considered.
- Arbitrary model: This model says that $y_i \in$ Q. This model, compared to the restricted one, can be considered as more flexible and practical as colluders should be able to yield a symbol which belongs to the known alphabet. This model leads to a probabilistic accusation, to a weak traceability and hence to introduce the notion of error accusation probabilities.
- Extended model: In this last model, colluders have the ability to add an unreadable symbol in detectable positions [Guth & Pfitzmann 2000].

Indeed, a collusion attack is a secret agreement between colluders to construct a copy of the sold media in a way which makes difficult the tracing process. The colluders compare respectively

their different releases to establish a common strategy to yield the suspicious fingerprint. This strategy is called "collusion attack". In Section 2.4.1.3, we will present common collusion attacks described in the literature [Trappe *et al.* 2003].

Collusion attacks In this section, we detail possible collusion attacks made under the Marking Assumption. We remind that different classifications of collusion attacks were proposed in the literature [Trappe *et al.* 2003]. In one of these classifications, the attacks can belong to one of three main classes:

- Block exchange attack: In this type of scenario, as shown in Table 2.2, the symbol in the suspicious copy can be one of the symbols of colluders' codewords for a position. There are several block exchange attacks such as the majority/minority vote attacks, All-one, All-zero, the uniform one, and the Worst Case Attack (WCA) [Fontaine 2011].

Table 2.2: The Block exchange attack. Beyond the application of the marking Assumption in position 2 and 4, it is trivial to guess that the corresponding symbol in the suspicious copy is one of the colluders' symbols in the position *i*.

Colluder1	1	**0**	1	**1**
Colluder2	1	**0**	1	**1**
Colluder3	0	**0**	0	**1**
y	0	**0**	1	**1**

- Fusion attack: In this case of attack, as depicted in Table 2.3, the pirates mix their respective blocks to compute the corresponding block in the suspicious copy with one strategy: average, minimum, maximum, median, etc. This type of attacks can have an effect on both the fingerprinting and the watermarking schemes [Furon & Pérez-Freire 2009b] [Mathon *et al.* 2013].

- Individual signal processing: The suspicious copy is altered with a post processing made by colluders to erase the way to trace them, as shown in Table 2.4. This treatment can be, for example, a compression, noising, denoising, filtering. In the literature, it is considered that some attacks have a deeper influence on the accusation performance than others

Table 2.3: The fusion attack. It is difficult to guess which symbol can be extracted from the pirated copy in positions 1 and 2, and it depends strongly on the robustness of the applied watermarking technique. In positions 2 and 4, the Marking assumption is respected.

Colluder1	1	**0**	1	**1**
Colluder2	1	**0**	0	**1**
Colluder3	0	**0**	0	**1**
y	1	**0**	0	**1**

[Furon & Pérez-Freire 2009b]. Consequently, it is interesting to study the stability of the tracing scheme over different types of attacks.

In this context, the whole fingerprinting system has more than one constraint related to its different layers, the media distributor should find a good conjunction between the watermarking layer and the fingerprinting layer to counter piracy operations which can be operated to the content or to the embedded codes [Chaabane *et al.* 2013].

Table 2.4: The individual signal processing attack. It is difficult in this case to guess which symbol can be extracted from the pirated copy in positions 2 and 3, and it depends strongly on the robustness of the applied watermarking technique.

Colluder1	1	**0**	1	**1**
Colluder2	1	**0**	0	**1**
Colluder3	0	**0**	0	**1**
y	?	**0**	?	**1**

2.4.1.4 The fingerprint detection step

The fingerprint detection step, labeled by (5) in Figure 2.3, consists in detecting the watermark embedded in the suspicious release. By applying the watermarking detection algorithm, the extracted watermark is transformed to the fingerprinting code. Then, a comparison to all of the assigned fingerprints are made. If the extracted fingerprint is close to one fingerprint, then

the guilty user is trivially identified and the process is completed, otherwise, the fingerprint is yielded by a collusion attack and the tracing process is needed.

2.4.1.5 The tracing process

The principle of tracing process, labeled by (6) in Figure 2.3, is to analyze the fingerprint code extracted from the suspicious copy, by means of other requirements: the totality of assigned fingerprints, the fingerprinting parameters and available information from the generation step. The tracing process may retrieve at least one of the actors of a collusion trial responsible of the suspicious copy.

The next part reviews related work on different fingerprinting approaches proposed in the literature. It is important to notice that in a first time the fingerprinting layer and the watermarking one were studied separately as two disjoint layers. From a theoretical standpoint, a fingerprinting system consists of a set of techniques to trace back the origin of a signal among a set of possible sources. The nature of this signal depends on the corresponding application. But, one key requirement of the fingerprinting system is the robustness to collusion attacks made by a set of traitors. The nature of these attacks can be guessed by analyzing and comparing the different sources.

The first concept of fingerprinting was introduced in [Wagner 1983] and was designed especially for copyright protection or for preserving a content privacy character purposes. It was used in applications where a digital content should be shared legally or a confidential document is distributed to a specific number of users [Fontaine 2011], [Boneh & Shaw 1995]. This characteristic is tied to the serialization step which consists in embedding a unique and specific code in the media release to identify it [Chaabane *et al.* 2013]. This step is important to discover any illegal distribution which is the main goal of a traitor tracing scheme. Whatever is the application of the fingerprinting, three principal actors are operating in this system:

- The video supplier or the seller: the actor who constructs and embeds the fingerprint in every sold release. Called also the media merchant, he should detect, in the case of an illegal redistribution of the media content, who, among the set of buyers, has contributed to construct the suspicious release which seems to be identical to the original copy.

- The buyer: or the customer is the user of the multimedia distribution platform, he can be a person or an entity which has the authorization to access to the media.
- The colluder, called also "opponent" by [Wagner 1983] can proceed solely or with others in order to construct a copy similar to the original one with unknown fingerprint in order to redistribute it illegally. In the fingerprinting systems, the colluder is considered as a traitor.

2.4.2 Types of fingerprinting schemes

[Wagner 1983] has proposed more than one classification for the fingerprinting when considered one of its main characteristics.

One first classification: based on the object nature [Wagner 1983]: In this context, the fingerprinting is thus:

- Digital fingerprinting: the object is a file or a digital content.
- Physical fingerprinting: the fingerprint depends on the inherent characteristics of the object.

A second classification: based on the type of the fingerprint [Wagner 1983]:

- Discrete fingerprint which consists of a set of values (binary values, or n-ary where $n \geq 2$.
- Continuous fingerprint where there is no limit to the number of fingerprint values.

A third classification: based on the detection process [Wagner 1983] [Schäfer 2016]:

- Perfect fingerprinting: in this case, it is assumed that any altered object can not be recognized and thus the detection becomes trivial otherwise.
- Deterministic fingerprinting: the detection is possible in all the cases with no error, called also "catch-all" fingerprinting class.
- Probabilistic fingerprinting: the detection is made with probability of making errors.

A fourth classification: based on the relation between the fingerprint and the object [Wagner 1983]:

- The fingerprint is one of the inherent parts of the object.
- The fingerprint is added to the object.
- The fingerprint is made by applying modification to some parts of the object.
- The fingerprint is made by omitting one of the parts of the object.

Recently, with the evolution of the fingerprinting field, some other classifications can be taken into account and hence, we add:
A classification based on the fingerprinting mode:

- Static fingerprinting: in this type of scenario, the fingerprints are defined and only one round of accusation process is made [Schäfer *et al.* 2010].
- Dynamic fingerprinting: in contrast with the static one, the set of fingerprints can be changed dynamically. In fact, more than one round of accusation process is made and hence, the fingerprints are changed to frame efficiently the traitors [Fiat & Tassa 1999], [Laarhoven *et al.* 2013].

Another classification tied to the environment of the fingerprinting scheme :

- Symmetric fingerprinting: in this type of scheme, the distributor is the only responsible of generating fingerprints and identifying traitors [Pfitzmann & Schunter 1996].
- Asymmetric fingerprinting: in this case of scheme, a third actor, named "arbiter", other than the distributor or the buyer takes part in the generation and the identifying process. The merchant or the distributor holds the original copy without any fingerprints and each buyer holds its specific fingerprinted copy. This scheme was proposed by [Pfitzmann & Schunter 1996] and was designed to enable the merchant to convince the arbiter that he is able to discover the buyer of a fingerprinted release.

- Anonymous asymmetric fingerprinting: was proposed later by [Pfitzmann & Waidner 1997] as an alternative to carry out an anonymous fingerprinting scheme, where the merchant is not able to guess the relation between a buyer and a sold release but nevertheless he can discover traitors in case of piracy trial.

Another classification tied to the fingerprint storage [Desoubeaux 2013]

- Fingerprinting made in advance: the fingerprinted releases are prepared in advance and stored to be used later in the distribution.
- Fingerprinting are prepared in the same time as the distribution and hence no storage is necessary here.

According to the studied properties of fingerprinting schemes, we are especially interested in **static**, **probabilistic** and **symmetric** fingerprinting systems using **modification** process to add **discrete** fingerprints to **digital** contents. This type of schemes can be observed in multimedia distribution platforms expected to integrate a tracing process for illegal redistribution trials [Schäfer 2016] [Desoubeaux 2013].

2.4.3 Applications of fingerprinting

In this part, we give an overview of the main applications of fingerprinting schemes to emphasize the relevance of combining watermarking scheme to collusion secure fingerprinting codes. According to [Schäfer 2016], the different applications of fingerprinting whose purpose is the protection of the copyright are as follows:

2.4.3.1 The promotion of a digital product

Before launching a new digital product such as: music album, film, audio book, TV program or video game, a promotional phase is dedicated to journalists to evaluate and rate the content. This phase should, hence, include a mechanism of protection to prevent the content from any trial of unauthorized diffusion.

2.4.3.2 The online shop application

This application enables an end-customer authentication process. Indeed, the digital product is purchased and transmitted via Internet.

2.4.3.3 The digital cinema application

It is also an end-customer authentication process which includes a step of local cinema authentication.

2.4.3.4 Video game application

It represents the Boxed video games applications including the promotion application, the online shop and even the digital cinema with a phase proper to the game development [Berchtold *et al.* 2013].
As the research point addressed for this thesis is related to collusion secure fingerprinting codes, the focus is on the deployment of these codes for multimedia distribution applications, defined previously as online shop applications. We turn now our attention to the proposed techniques in the literature in the fingerprinting field.

2.5 Tracing traitors from cryptographic point of view

In the literature, several fingerprinting techniques were proposed [Chaabane *et al.* 2013]. One trend work was attached to the cryptographic and coding orientations and has proposed to improve the properties of tracing codes [Boneh *et al.* 2010] migrating from the deterministic fingerprinting approaches, [Trappe *et al.* 2003] to the probabilistic ones. The purpose of the majority work has still been oriented in proposing fingerprinting codes with good detection rates and fair length, although the number of users and pirates is increasing, [He & Wu 2007].

2.5.1 Collusion-Secure fingerprinting Codes

We have introduced the properties and objectives of the digital watermarking which is applied for customer authentication which is one of the main interesting points in our thesis work. Then, regarding the limitations of the watermarking scheme in case of collusion attacks, it is imperative to define the collusion secure fingerprinting codes.
Collusion secure-fingerprinting codes can be split into deterministic or probabilistic one. We will detail each class separately:

2.5.1.1 Deterministic codes

From a mathematical point of view, the different notions in the fingerprinting field can be formulated as follows:
First, it is assumed that the data is represented by a string of symbols belonging to an alphabet Q of size q, $\mathrm{Q} = \{0,1,\ldots,q-1\}$. In the binary case, q=2 and hence $\mathrm{Q} = \{0,1\}$. So, to apply the traitor tracing process, we begin by embedding fingerprints in the original data. These fingerprints, called also codewords should have a length m over Q. Then, the fingerprint construction step consists in assigning for each user a unique fingerprint which presents a vector from Q^m, where $\vec{x}_j$ denotes the vector for a user j. The resulting fingerprint matrix X is a $n \times m$ matrix with n the number of users and l the codeword length. $\mathrm{X} = (\vec{x}_j)_i$ where $\vec{x}_{ji}$ the symbol in the position i of the codeword $\vec{x}_j$.
An eventual collusion made by a set $\langle \vec{x}_{j1},\ldots,\vec{x}_{jc} \rangle$ has a size c and is able to yield a codeword $\vec{y}$, with $\vec{y} \notin X$.
The collusion secure fingerprinting codes were classified in the literature according to namely two basic properties: the colluders' detection and the non accusation of innocent users [Boneh & Shaw 1998], [Staddon *et al.* 2001], [Encheva & Cohen 2002], [Yagi *et al.* 2007]. It is important to mention that these proposed classes are not useful all the time since some existing codes could not belong to any class. In this part, we present the four essential classes of codes:

- Frameproof code: No yielded collusion can involve an innocent user, hence it is the best way to prevent trivial accusation. In fact, $C = \{\vec{x}_1,\ldots,\vec{x}_n\}$ is called c-frameproof when

for $c \geq 1$, no collusion can involve a codeword $\vec{x} \in C$ whose owner is not in the collusion.
Let give the example of the code $C = \{100, 010, 001\}$,
C is 3-frameproof. In fact, the 3-size collusion could not involve an innocent user because it consists of the all users' codewords, whereas the 2-size collusion is not able for all cases to construct the third codeword as detailed below:
$\langle 100, 010 \rangle = \{100, 010, 110\}$ does not involve the codeword 001.
$\langle 100, 001 \rangle = \{100, 001, 101\}$ does not involve the codeword 010.
$\langle 010, 001 \rangle = \{010, 001, 011\}$ does not involve the codeword 100.

- C-secure frameproof code: Two disjoint collusions of at most size c can not generate the same codeword.
  Let give the example of the code $C = \{111, 100, 010, 001\}$,
  C is 2-secure frameproof. In fact, for the 2-size collusion,
  a first collusion $c1 = \langle 111, 100 \rangle$ contains the codewords $\{1**\}$, whereas another collusion $c2 = \langle 010, 001 \rangle$ contains the codewords $\{0**\}$ which means that the two disjoint collusions *c1* and *c2* are not able to generate a common codeword.
  Similarly, a second example of collusion $c3 = \langle 111, 010 \rangle$ contains the codewords $\{*1*\}$ whereas the collusion $c4 = \langle 100, 001 \rangle$ contains the codewords $\{*0*\}$, *c3* and *c4* are also not able to generate a common codeword.
  For the third case also, collusion $c5 = \langle 111, 001 \rangle$ contains the codewords $\{**1\}$ whereas the collusion $c6 = \langle 100, 010 \rangle$ contains the codewords $\{**0\}$ which means that the two disjoint collusions *c5* and *c6* are not able to generate a common codeword.

- Identifiable Parent Property code: No collusion of size at most c can generate a codeword which has not at least one guilty parent [Barg & Kabatiansky 2004].
  Let give the example of $C = \{0000, 0111, 0222, 1012, 1120, 1201, 2021, 2102, 2210\}$, a code of length 4 over an alphabet of size $q = 3$, this code as described in [Blackburn *et al.* 2007] is a 2-IPP code, known also as the Tetracode or the ternary Hamming code of length 4. This is a linear error-correcting code so that two codewords have a Hamming distance at least 3. Now let give the example of a collusion of size 2 which yields a codeword $\vec{y}$. Then, since the two codewords cooperating in the collusion have a common symbol on exactly one position, the codeword y also should have a common

symbol with both codewords on the same position. Moreover, $\vec{y}$ has common symbols with one of them on the remaining three positions. Hence, one of these two codewords must match the yielded codeword on at least 3 positions, giving it a Hamming distance of at most 1 to the forgery. So the result of the tracing algorithm in terms of Hamming distance, should always accuse exactly this guilty user. Henceforth, this code is a 2-IPP code.

- Traceability code: C is considered as a c-traceability code if for any collusion of size at most c and any codeword $\vec{y}$ yielded by this collusion, the user whose codeword has the smallest Hamming distance to the codeword $\vec{y}$ is always accused [He & Wu 2005].

Regarding the properties of collusion secure fingerprinting codes, it is interesting to split them also to two families [Xie 2010]: codes with strong traceability and codes with weak traceability.

Weak traceability vs strong traceability

- Strong traceability: Having a code with strong traceability is a very hard requirement because it means that the code should never accuse an innocent user [Staddon *et al.* 2001]. The Error Correcting codes were proposed as codes with strong traceability because they considerate that the probability of accusing an innocent is null. But the first problem with these codes is that they are too long and the tracing operation becomes impossible when the number of colluders is more than 2 colluders [Kiayias & Yung 2001]. That is why Reed Solomon codes were proposed as tracing codes because they represent large alphabets.

- Weak traceability: This type of traceability introduces that a code is not perfectly traceable, it uses two error probabilities [Barg & Kabatiansky 2011] [Boneh & Shaw 1998] :
  ε_1: called also false positive probability: the probability of accusing falsely an innocent, this probability is in the order of 10^{-6}.
  ε_2: called also false negative probability: the probability of missing pirates (false negative), it is in the order of 10^{-1}, we tolerate the inability to catch a guilty user.

Several codes with weak traceability were proposed in the literature. According to [Schäfer 2016], only the ε_1 rate is required for the fingerprint generation process [Tardos 2003]

[Skoric *et al.* 2007] [Simone & Škorić 2012].

When considering the legal target of tracing traitor field, it is undoubted that deterministic codes provide a solid assessment to this type of applications. But from the practical standpoint, probabilistic codes are more suitable for real scenarios since they provide shorter code length with a fair probability of error [Schäfer 2016]. Henceforth, all contributions elaborated within this thesis work are based on probabilistic codes.

2.5.1.2 Probabilistic codes

The first model of probabilistic codes was proposed by Boneh and Shaw in [Boneh & Shaw 1998], this model uses a binary alphabet and a code length m=$c^4 \log(\frac{n}{\varepsilon_1}) \log(\frac{1}{\varepsilon})$.

(a) **The BS with replication scheme:** The Boneh Shaw code with replication scheme was proposed in [Boneh & Shaw 1998], it applies the Marking assumption and is represented by the couple *(n,d)* where *n* is the number of the codewords and *d* is the replication value. The code length is defined by the equation:

$$M_t = (n-1) \times d. \tag{2.1}$$

In this case, the BS code is binary, thus the alphabet is equal to $\{0,1\}$. The result of the code generation is a matrix of (n-1) column types of the form $1 \quad 1 \quad \cdots \quad 0 \quad 0$ replicated d times and suiting the rule that the i^{th} user has a 0 in the (i-1) columns and 1 in the others. The BS code with replication scheme is considered as a n-secure binary code with a code length $m = O(c^4 \log(\frac{n}{\varepsilon_1})$ and an error probability ε, when an accusation failure of a tracing function is bounded by ε. The distributor must keep secret transitions from colluders and hence has to perform a secret permutation of the columns of the code BS(n, d) before hiding the user codeword inside the multimedia content. When considering the variation of the number of bits of 1 between blocks of different types, this randomized approach enables the distributor to prove that a user is probably a guilty one. Thus, the performance of the accusation depends only on the value of the replication parameter d:

$$d = 2n^2 log\frac{2n}{\varepsilon}. \tag{2.2}$$

The secret permutation of columns in the code BS(n, d), as shown in Figure 2.5 is applied after the code generation to enhance the security of the code by preventing colluders from predicting the position of embedded fingerprint symbols.

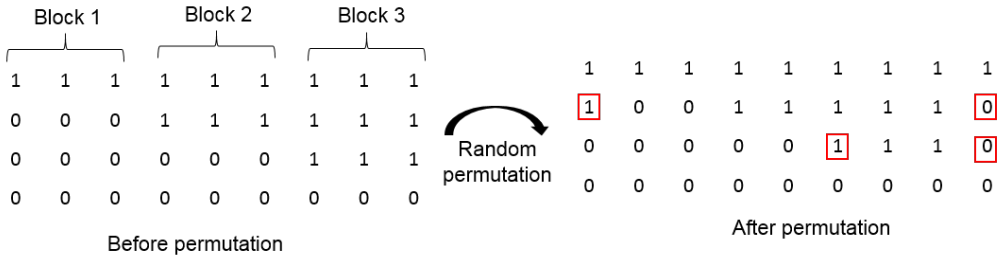


Figure 2.5: BS(4,3) code generation before and after a random permutation.

Although, The BS codes have marked an interesting change by introducing the error probabilities, they are still far from the limit bound of code length required by [Peikert *et al.* 2003].

(b) **The Tardos code:** Another probabilistic code called the Tardos code has been proposed in [Tardos 2003] and has attracted increasing interest due to its short code length with an adjustment of the error probability [Peikert *et al.* 2003]. In fact, it is a fully randomized binary code, with some new parameters:

- *X:* $n \times m$ matrix, as given in Figure 2.6, with n is the users' number and c is the collusion size.

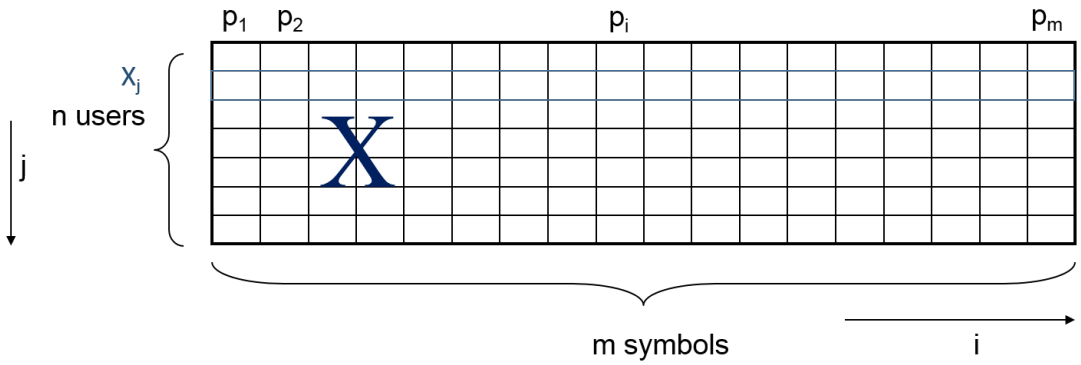


Figure 2.6: *(n×m)* matrix for Tardos code with n number of users and m the codeword length.

- A codeword X_{ji} is assigned to each user j, $i \in \{1, \cdots, m\}$. Let p_i a set of independent and identical probabilities generated randomly from $[t, 1-t]$, with $t = \frac{1}{300c}$, $0 < t\prime < \frac{\pi}{4}$,$\sin^2 t\prime = t$, $p_i = \sin^2(r_i)$, $r_i \in [t\prime, \frac{\pi}{2-t\prime}]$.

- ε_1, ε_2 are respectively the false positive and false negative probabilities and m is the code length. According to the current state of the art and to the asymptotic parameters given by [Laarhoven & de Weger 2011];

$$m = \frac{1}{2}\pi^2 c^2 \ln(\frac{1}{\varepsilon_1}). \tag{2.3}$$

The tracing algorithm of the Tardos code consists of three steps as follows: the initialization, the construction and the accusation steps. Initially, codewords' matrix X_{ji} is constructed with Prob[X_{ji}=1] = p_i. The score $S_{j,j\in\{1\ldots n\}}$ is computed in Eq.2.4 related to each user j as follows:

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

The score S_j is computed according to the symmetric accusation functions defined by [Skoric *et al.* 2007] in Eq.2.5 and Eq.2.6 as follows:

$$g(1,1,p) = g(0,0,1-p) = \sqrt{\frac{(1-p)}{p}}, \tag{2.5}$$

$$g(1,0,p) = g(0,1,1-p) = -\sqrt{\frac{p}{(1-p)}}. \tag{2.6}$$

Z is the threshold parameter [Laarhoven & de Weger 2011] defined by:

$$Z = \pi c(\ln\frac{1}{\varepsilon_1}). \tag{2.7}$$

The output of the Tardos tracing algorithm includes a user j if $S_j > Z$ with a false positive error $< \varepsilon_1$.

We have presented the different classes of collusion secure fingerprinting codes in this section and in the next section, we will review some former works from cryptographic point of view. We will begin by the first approach proposed in the literature which is the deterministic approach, especially approach based on Error Correcting codes. And then, we turn our attention to fingerprinting schemes based on Tardos code which has prompted a spate of interest in the literature.

2.5.2 Fingerprinting schemes based on error correcting codes

Many research has been conducted on proposing Error Correcting Codes [Lin *et al.* 2009], ECC, as tracing codes to deal essentially with the problem of attacked codes [Tomas-Buliart *et al.* 2009]. As mentioned in [Chor *et al.* 1994], an ECC code C having the parameters (m,n,d) is a c-tracing code if its minimal distance $d > m(1-c^{-2})$ (with n the number of codewords of length m and minimal Hamming-based distance $d = \min_{x,y\in C}(d(x,y))$) [Chor *et al.* 1994].

The Reed Solomon codes, called also RS codes, due to their small minimal distance, have been widely studied in fingerprinting field [Ma & Ding 2006]. In [Tomas-Buliart *et al.* 2009], authors propose an improved fingerprinting code design based on combining a convolutional code to a probabilistic code, the Boneh Shaw code, BS code. This design was proposed to solve the problem of false positive error detected in the scheme of [Zhu *et al.* 2006] assessed for only 2 colluders. The ECC was proposed essentially to reduce the error accusation probability by modifying the Viterbi decoding algorithm. Shaathun et al. [Schaathun 2008] have also proposed to construct a fingerprinting scheme based on concatenating a random ECC to the BS code to enhance the robustness of the system against copy-paste collusion attacks. Despite the provided good detection rates, the proposed design suffers from a long embedded code. Authors in [Schäfer *et al.* 2010] have proposed a Hamming-based constrained tracing code which provides a fair accusation rate with minimal error rate against a collusion size of at most 2. In [Lin *et al.* 2009], the proposed design is a concatenation of an orthogonal binary code with a large-distance Reed-Solomon code. The ECC-based forensic code is constructed to enhance the resistance to post-processing collusion attacks and has a good detection for a collusion size less than 20. Moreover, authors in [Tomas-Buliart *et al.* 2008] has proposed to use Turbo codes to the BS code to survive some weaknesses of BS code: The codeword length has been reduced by the use of Turbo code as outer code and the improvement of the tracing process runtime by a Maximum Likelihood Decoding algorithm.

When considering the majority ECC-based fingerprinting schemes proposed in the literature, we notice that despite their fast and deterministic tracing process, they still suffer from the weaknesses of requiring long codeword for limited small collusion size and few number of users. Hence, this type of approach is limited when regarding the real scenarios of multime-

dia distribution platforms which involve a huge number of connected users and consequently an important number of pirates [Schäfer 2016] [Desoubeaux 2013]. However, probabilistic fingerprinting schemes are the most suitable for applications in practice, since the permitted error probability provides shorter fingerprints. Hence, we will review some probabilistic fingerprinting schemes based on Tardos code since actual tracing work agreed upon the fact that the well-known Tardos code provides a good compromise between the code length and the collusion size [Furon *et al.* 2009].

2.5.3 Fingerprinting schemes based on Tardos code

Despite its fair compromise between the code length and the detection rate, the Tardos code still requires improvement with regard to its complexity decoding. This weakness is unequivocal in the case of multimedia distributing systems which involve a large size of audience and are expected to use a low complex pirates' retrieving process. The fingerprinting schemes based on Tardos code have witnessed a flurry of research efforts. This part surveys the current state of the art in traitor tracing approaches based on Tardos code. Some approaches focus on applying shifts to the Tardos code itself. Whereas, few other attempts have been made to ameliorate its tracing process by using a group-based property.

2.5.3.1 Tardos-based approaches proposing to improve its accusation functions

In [Furon *et al.* 2008], a study of Tardos code has been provided to justify with statistical point of view the accusation functions and the p_i distribution. Since the paper of [Furon *et al.* 2008], more than one work was proposed to improve the accusation process of Tardos code. According to [Tardos 2003], the false negative probability and the false positive one are tightly tied by the relation $\varepsilon_2 = \varepsilon_1^{\frac{c}{4}}$. In traitor tracing field, it is assumed that a large value of false negative probability is tolerated compared to false positive one.
In [Blayer & Tassa 2008] and [Furon *et al.* 2008], it was proposed to consider the two probabilities separately and hence to reduce the code length. Moreover, although in the original version of Tardos code [Tardos 2003], only "1" symbols are considered, in [Skoric *et al.* 2007], a symmetric version of Tardos decoder was proposed by considering the "0" symbols and according

to that, the resulting code length was reduced to the half of the original length. In addition, due to the continuity of the Tardos distribution, generating the Tardos code words can be hard and applying an approximation of this distribution can reduce the security and the performance of the code.

To solve this problem, [Nuida *et al.* 2007] has proposed an improvement to the Symmetric version of the Tardos code by a distribution discretization. This assumption has been used in [Simone & Skoric 2011] to study the error probability in case of some collusion attacks and hence to find suitable experimental solutions for analyzing this probability, even for non-binary Tardos code.

In [Laarhoven & de Weger 2011], it was proposed to combine the symmetric accusation functions of [Skoric *et al.* 2007] to the analysis of [Blayer & Tassa 2008] to reduce significantly the code length to up to four times shorter than the code length in [Blayer & Tassa 2008] and to two times than in [Skoric *et al.* 2007]. A proposal design which allows to achieve the theoretical bound code length. Even for large collusion size, it provides good results and a code length of 4.93% of the original version.

In other set of work, the notion of joint decoding was introduced where the Tardos score is not computed from a user but for a set of c users. In [Amiri & Tardos 2009], this joint decoding has provided optimal detection rate but suffers from hard complexity decoding step because all possible combinations of c among n users. To solve this problem, [Meerwald & Furon 2012] has proposed to use an iterative decoder based on side information of previous found users.

2.5.3.2 The group-based tracing approaches

In tracing traitor, it is undoubted that one crucial requirement is the efficiency and the detection performance of the tracing code.

- The main principle of group-based tracing approaches

  According to [Wang *et al.* 2004], the group-based tracing scheme mainly consists in grouping users having common characteristics according to the assumption that they have more probability to cooperate together. Henceforth, this type of scheme may reduce the Tardos users' search space and consequently the complexity of its decoding step.

  As a first example, in [Akashi *et al.* 2008], a two-level hierarchical structure was pro-

posed for the fingerprint generation process. Assigning users to a group was made randomly without considering any relationships inter or in groups. The main target was to minimize the complexity of the Tardos tracing decoding process.
In [Hamida *et al.* 2011], the idea was to focus on the hierarchical structure proposed by [Wang *et al.* 2004], not for independent Gaussian signals but for Tardos-based fingerprints. Despite the good detection rates proven in this technique, the main weakness was the important length of the tracing code. In [Ye *et al.* 2013], authors analyze users' relationships in a social network to construct a multi-level hierarchical fingerprint for digital content diffusion through the internet and Peer to Peer networks. The resulting fingerprint of each user is the combination of Boneh Shaw code as an outer code and the Tardos code as an inner one. However, this work was tested to only the majority vote attacks which can not be considered to be the worst case attack [Furon & Pérez-Freire 2009b].
In the same context, and according to the identifier multi-level hierarchical structure we proposed previously, it was necessary to find a suitable tracing strategy which enables a first group selection and then user accusation in the retrieved group. In this part, we give a brief overview on the existing tracing techniques proposing on a two-level tracing strategy for group-based fingerprints. Mainly, these techniques are collected from two different tracing schemes: the static scheme and the dynamic one. While only one run of the tracing process is made in the static scheme with a known code length and a set of retrieved users, in a dynamic scheme, the tracing process is made on several runs and once a user is accused, he will be disconnected from the system. In the dynamic scheme, the code length, the number of colluders and even the number of connected users are changing periodically [Laarhoven & de Weger 2011] according to each round of tracing.

Dynamic tracing schemes including two-level tracing code: The tracing scheme proposed in [Tassa 2005] is a dynamic hybrid scheme. It proposed in a first time to combine the Boneh Shaw code [Boneh & Shaw 1998] and the Fiat and Tassa code [Tassa 2005] in order to provide an efficient bandwidth guaranteed by the former code and an efficient time offered by the latter code. In fact, this scheme proceeds by applying in a first step a tracing round to a set of time segments, and then applying a second tracing algorithm to frame dishonest users. Although, this scheme has proven a fair bandwidth, its weakness

has still been the inefficient run-time of the Boneh Shaw scheme which is closely tied to its codeword length. As an improvement, authors in [Tassa 2005] propose to use the Tardos code [Tardos 2003] instead of the Boneh Shaw code. This choice is based on the fact that it has been proven that Tardos code allows to reach the bound of the shortest code length with an adjustment of the error probability [Peikert *et al.* 2003]. In the same context of dynamic tracing schemes, the main idea in [Wu & Liew 2012] was to adapt the Tardos code in a dynamic scheme. Indeed, with regard to the important complexity of this code $O(n \times m)$ where n is the number of users and m is the codeword length, it seems to be difficult, if not infeasible to use Tardos code in tracing systems operating in real time and involving a large number of users. Hence, authors propose to reduce the complexity of the tracing process to $O(\log(n))$ by concatenating the Tardos code to an error correcting code. The main contribution in this work is that the obtained code can efficiently reduce the complexity of the tracing algorithm in a dynamic scheme where successive tracing rounds are needed, but it seems to be not useful for static scenarios where only one tracing round is made.

Static tracing schemes including two-level tracing code: We turn now our attention to the tracing schemes operating in the static scenario. The first work was proposed by [Kuribayashi *et al.* 2008], and whose principle was to apply a two-layered Tardos tracing strategy, called TT in the sequel. In fact, users' codewords are partitioned in a set of groups in order to reduce the computational costs of the Tardos parsing of all the users' codewords. The tracing process is made as follows: a first group accusation level is applied and then a second accusation level is applied to trace users within the accused groups. While this work has proven good detection results, it has some weaknesses. In fact, for the decoding step, the authors in [Kuribayashi *et al.* 2008] consider only the users in suspicious groups, which is not successful in all the times. For instance, the Tardos code does not guarantee that all the scores of guilty users are above the accusation threshold but that at least one score is satisfying the condition. Hence, an accused group by the first code could contain a colluder which is not framed by the second code. Another two-level tracing code in the static field was proposed in [Desoubeaux *et al.* 2012]. It consists in a joint of two codes: Boneh and Shaw with repli-

Table 2.5: Example of constructing a TT codeword

	X_j^g	X_j^u
X_j	$0\,1\cdots 1$	$1\,1\cdots 1$

cation scheme, BS (with alphabet cardinality q_{BS}) to each group of users, and Tardos code (with alphabet cardinality q_T) to each user. Compared to [Kuribayashi *et al.* 2008], instead of constructing two separated codes, the idea is to append them to prevent mainly traitors from identifying the positions of each code in their codewords and also from co-operating together.

When considering the fingerprinting field from cryptographic point of view, the fingerprinting layer is the most important part in the tracing scheme, especially the code design part. In the history, the ECC were the first tracing codes proposed in the literature with a deterministic accusation and too long codeword which makes their effectiveness very limited from practical point of view. Then, several trials of other tracing codes were proposed. The real evolution in fingerprinting field was tied with the probabilistic codes: The BS code and the Tardos code.

2.6 Tracing traitors from signal processing point of view

One other trend of fingerprinting schemes has considered that it will be more interesting to not separate between the watermarking and the fingerprinting layers. This research orientation has strived in finding a satisfying joint between them to have a resulting tracing scheme able to survive essentially fusion collusion attacks [Wang *et al.* 2003].

In this type of approach, compared to the ECC-based fingerprinting schemes, the main requirement is the distribution of fingerprints and not the distance between them. Henceforth, fingerprints are constructed as orthogonal Gaussian signals [Desoubeaux 2013] [Fontaine 2011].

The pioneer work proposed as a secure and robust watermarking scheme based on Gaussian orthogonal signals was proposed by [Cox *et al.* 1996]. In fact, the spread spectrum-based watermarking technique was applied in more than one fingerprinting work in the literature [Wang *et al.* 2005] [Kilian *et al.* 1998] [He & Wu 2005]. In spite of their good detection rates,

the main limitations have still been the complexity of the fingerprint decoding step which is in $O(n \times l)$ (where n the number of fingerprints and l is the signal size) and the limitation of the code achievable rate [Furon *et al.* 2009]. Henceforth, several researchers were oriented in improving the detection capacity and reducing the complexity. In [Wang *et al.* 2004] and [Hayashi *et al.* 2007], authors have focused on the study of the collusion resilient watermarking technique. Recently, in [Desoubeaux *et al.* 2011], the particularity of the proposed tracing approach is that it embeds the Tardos code to detect the orthogonal zero-bit image watermark of traitorous user. Despite the satisfying tracing results shown with this technique, the robustness to the worst case collusion attack was not assessed.

In [Kuribayashi 2012], the author proposed to improve the traceability of the spread spectrum-based fingerprinting scheme by using an iterative accusation process combined to an operation of interference removal. The iterative tracing step is a group-based accusation process which allows to reduce iteratively the number of fingerprints. Combined with the interference removal, the proposed system should be able to resist to more colluders.

While, the majority of the proposed fingerprinting schemes are based on image watermarking techniques [Desoubeaux *et al.* 2011], [Hayashi *et al.* 2007] and [He & Wu 2007], only few fingerprinting approaches using audio watermarking techniques were proposed in the literature [Cha & Kuo 2007], [Qureshi *et al.* 2015]. So far, in [Cha & Kuo 2007], the detailed approach has proposed to use orthogonal spreading followed by the Inverse Fourier transform codes in a time audio watermarking scheme. Indeed, the robustness of the proposed codes has been checked only against the average and the pre-average collusion attacks for a small number of users. Additionally, in [Qureshi *et al.* 2015], a Peer to Peer platform for multimedia distribution was presented integrating a tracing traitor process.

In this scheme, a Quantization Indexed Modulation (QIM) based audio watermarking technique was proposed to embed Nuida's codes [Nuida *et al.* 2007] in audio contents. The robustness to collusion attacks was proven against some attacks and for a few number of traitors. Moreover, authors consider that to preserve a good quality of the digital content, the proposed system can provide a robustness to a limited collusion size.

In practical point of view, focusing on the watermarking technique as the spread spectrum techniques in the fingerprinting technique should provide good robustness to fusion attacks but

suffers from the increasing complexity tied to the number of considered users in a multimedia distribution platforms [Hayashi *et al.* 2007] [Kuribayashi *et al.* 2008].

2.7 Conclusion

This chapter has concentrated on a survey of existing traitor tracing approaches proposed especially for multimedia distribution. These approaches are given according to the cryptographic or the signal processing point of view. In one side, several schemes focused on the fingerprinting layer by proposing different tracing codes constructions. In the other side, other schemes enhanced the watermarking layer and proposed to use orthogonal gaussian signals as fingerprints. From this study, both the two types of approaches have limitations regarding the context of multimedia distribution platforms which involves great number of users. The major limitations are tied to the complexity, the code length and the accusation rates. Consequently, developing a traitor tracing scheme suited to the content distribution context has been considered as a promising issue and a real challenge to be investigated. Different open questions and problems discussed in this chapter remain unresolved. Henceforth, research on traitor tracing field is still ongoing.

Part II

Contributions

CHAPTER 3

Generating a multi-level hierarchical fingerprint for traitor tracing scheme

Contents

3.1 Introduction

The starting point in this thesis work is to propose a suitable structure to the embedded fingerprints to ensure an efficient and fast tracing process in multimedia distribution platforms involving great number of users. One main challenge of the existing Tardos-based tracing approaches was to face the decoding complexity and the computational costs of the tracing process.

Hence, the tracing scheme we propose is a group-based scheme which enables to construct groups of users according to a multi-level hierarchy, a choice which can be justified by the fact that the closer is the group construction, the more reduced is the Tardos search space, and hence the faster is the tracing process. The fingerprinting step consists of two essential steps: the hierarchy construction and the generation of the corresponding fingerprints. These multi-level hierarchical fingerprints are then embedded into the media content using an audio watermarking technique. The proposition of these fingerprints is designed to deal with the problem of the complexity and the computational costs of the tracing step.

In this chapter, we detail the different steps of the fingerprinting process and show how the hierarchical structure of fingerprints can influence positively the detection process.

3.2 The proposed Multi-level fingerprint generation step

The majority of the traitor tracing field agree upon the fact that the whole fingerprinting system includes three main steps: the fingerprint generation step, their embedding into the media content and the tracing process [Liu 2005]. According to related work, we have focused on rising to the challenge of improving robustness results and accusation rates of Tardos code. Thus, we try to reduce its complexity computation by proposing a multi-level hierarchical fingerprint. Then, we embed it by an original robust watermarking technique.

Group-based schemes using a two-level tracing accusation For multimedia distribution platform integrating a Tardos serialization system, one key constraint in the tracing scheme is the required computational costs to compute the scores S_j of all considered users, which is around $O(n \times m)$ operations, where n is the number of users and m is the codeword length. We present

in Figure 3.1 the illustration of the fingerprinting system we adopt in our thesis work. We will try to detail each improved part separately. The proposed approach consists in generating multi-

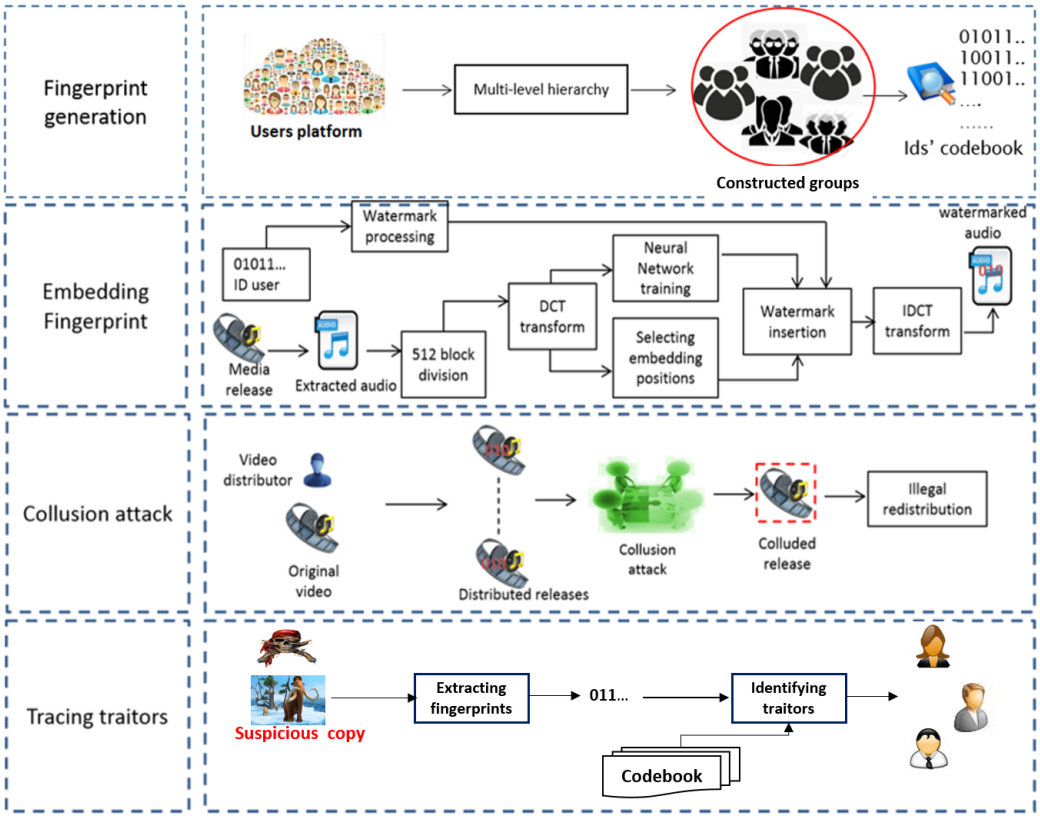


Figure 3.1: Illustration of the general fingerprinting framework.

level hierarchical fingerprint according to the studied hierarchy. The result of the generation step as shown in Figure 3.2 is a codebook of codewords; a set of identifiers.

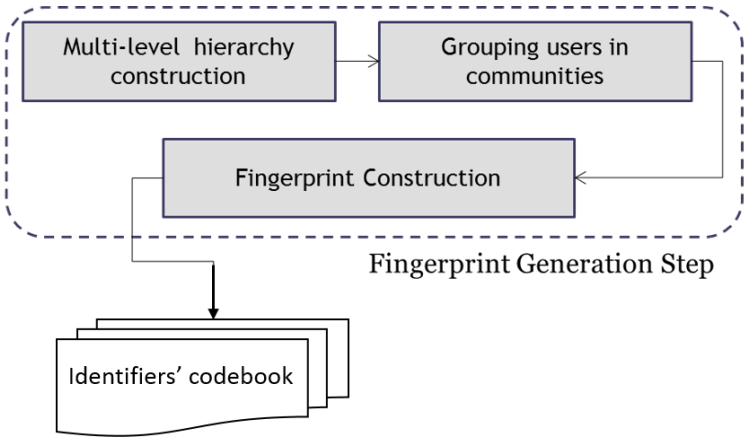


Figure 3.2: Details of the fingerprint generation step.

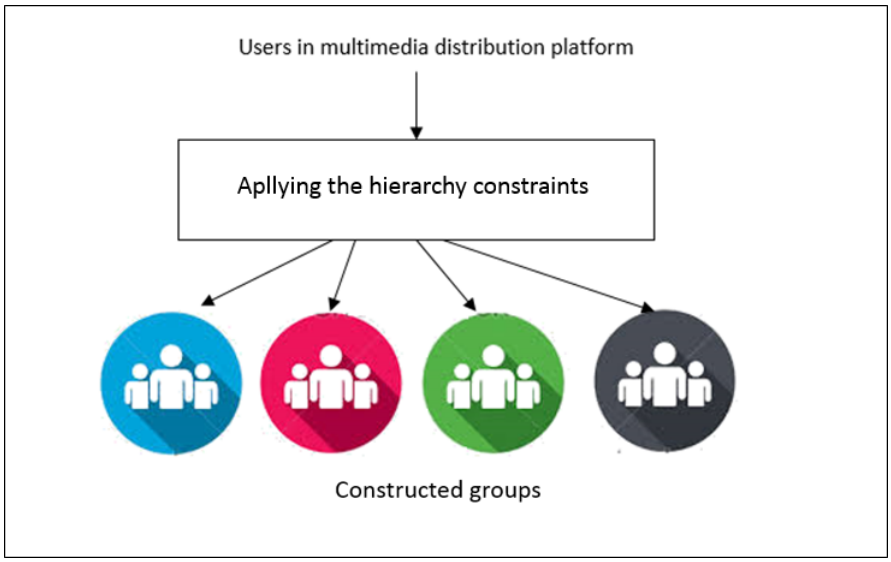


Figure 3.3: Applying group-based properties to construct groups of users.

Reducing the search space of dishonest users by assigning a user to a specific group as depicted in Figure 3.3 represents a suitable solution to face the Tardos accusation costs. The user assignment to a group can be used to counter different types of coalitions: temporal, geographic, social, etc. In the hierarchy, each chosen constraint corresponds to a level. We will detail each selected constraint separately.

3.2.1 Temporal constraint

It is the main distinguishing criterion in our hierarchy. To the best of our knowledge, there has not been taken into account previously in the literature. This constraint is tied to the time a video stays accessed by viewers. The frequency of the access to a video in a multimedia distribution platform depends on its popularity. In the beginning, when a video belonging to the Top 10 is added to the platform, users are very curious and the frequency accessing is very important, this behavior changes later to decrease significantly over time. According to [Liu *et al.* 2014], the viewer interest decreases from 100% to less than 10% during a 4-month period. Henceforth, in the time level, we focus on two phases of respectively two months duration and we consider each period separately. We assume that the first period for a video since its adding to the platform is very important or even decisive in the tracing process. This phase can be considered as the most period of time where a video can be prone to pirate attack.

3.2.2 Geographic constraint

This constraint was taken into account in more than one previous work in the literature [Wang *et al.* 2004], [Hayashi *et al.* 2007], [Hamida *et al.* 2011] because of its relevant significance in grouping users. In fact, this assumption relates that two users belonging to the same geographic place are more able to collude together than with other users from other regions. According to that, we have studied in [Chaabane *et al.* 2013], the degree of software piracy in different countries and based on the BSA report [1], we conclude that digital piracy in general is more important in some countries than in others. According to that, we split the geographical constraint to two levels: continents and countries, since piracy rate can be totally different in two countries from the same continent. Let give the example of Japan in which the piracy rate is about 19% whereas in China it exceeds 38% [2].

3.2.3 Social constraints

We have enhanced our study with statistics shared by the NPD, National Purchase Diary Group, known for its consumer market research. The NPD has studied the media traffic in one of the most popular multimedia distribution service, Netflix, and has shown that audience behavior changes depending on the age and the gender. The highest rate of users in a week is noted with persons under 15 years old age. This study demonstrates also that men are less interested in this type of services than women.

Thus, according to this study, we embrace a multi-level hierarchy in the fingerprint generation step. Each criterion, as depicted in Figure 3.4, is represented by a level in the hierarchy.

The first level is the time level where we assume that the most important period for a video life in a VOD platform is about 4 months [Choi *et al.* 2012], [Liu *et al.* 2014]. Hence, we consider two groups of two-month-duration: in the first one, users' curiosity is moderately important and increases gradually to reach the maximal audience interest and in the second one it decreases to reach the minimal bound in the fourth one.

A second level in the hierarchy represents the geographic criterion where we propose to divide platform users to two essential regions: Zone_A where the piracy phenomenon is very impor-

[1] http://globalstudy.bsa.org/2013/downloads/studies/2013GlobalSurvey_Study_en.pdf

[2] http://globalstudy.bsa.org/2013/downloads/studies/2013GlobalSurvey_Study_en.pdf

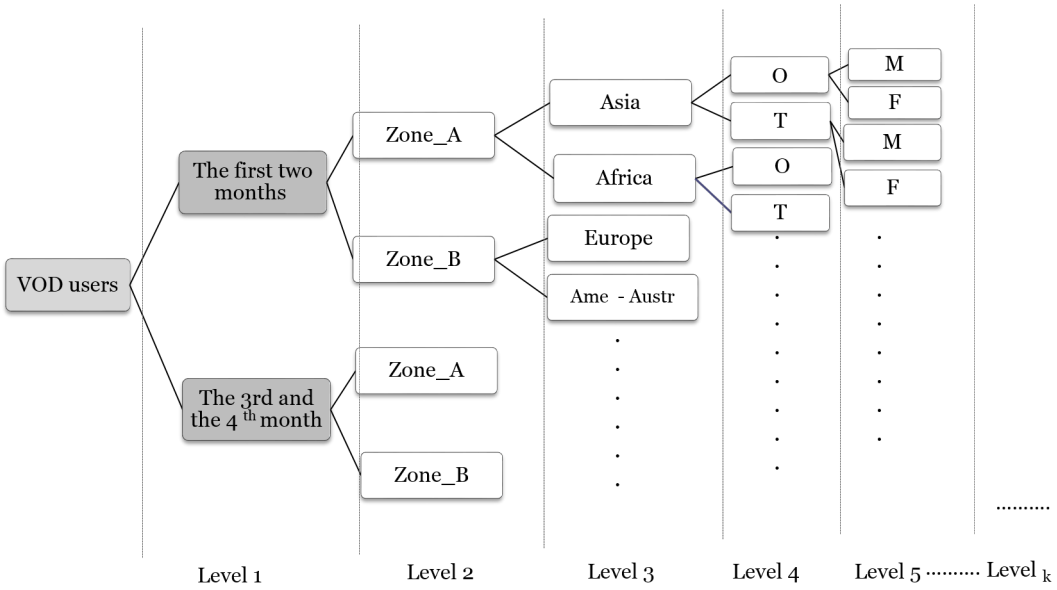


Figure 3.4: The proposed multi-level hierarchy for the fingerprint.

tant, especially in Asia and Africa continents, and Zone_B where the piracy phenomenon is less important especially in Europe, America and Austria.

We divide each continent to two groups of the main large countries in the third level.

In the last levels, we are interested in social criteria, such as the age and the gender ones, etc.

Once the hierarchy is fixed, we construct groups of users so that users having the same characteristics belong to the same community and have more probability to collude together in a forgery attempt than with users belonging to other communities. The resulting fingerprint for each user is the concatenation of his community identifier concatenated to his personal identifier which is encoded with Tardos code.

$$Final_{identifer} = id_{level1} + id_{level2} + .. + id_{levelk} + personal_{id} \tag{3.1}$$

The multimedia distribution context does not require that the tracing step should be performed in real time. The whole operations of decoding and accusation are made offline.

3.3 The tracing process using multi-level hierarchical fingerprint

One key requirement in the fingerprinting system is the tracing process: when the supplier detects a copy with unknown fingerprint Y, he tries to trace back colluders by analyzing the extracted fingerprint Y, retrieving its similarity to a group identifier and hence tracing individual colluders. The tracing is performed here by the Tardos code. As shown in Figure 3.5, group selection is based on computing Hamming distance between the ID_{group} of the suspicious copy and the other groups' identifiers. The selected groups are the closest to the extracted one; namely groups having the smallest Hamming distance to the ID_{group} of the suspicious copy. Then, the tracing process continues with the Tardos tracing step, the score S_j is computed per user only in selected groups. The user whose score exceeds the threshold Z [Laarhoven & de Weger 2011] is thus accused. The detection rate of the proposed system is then computed to check its efficiency.

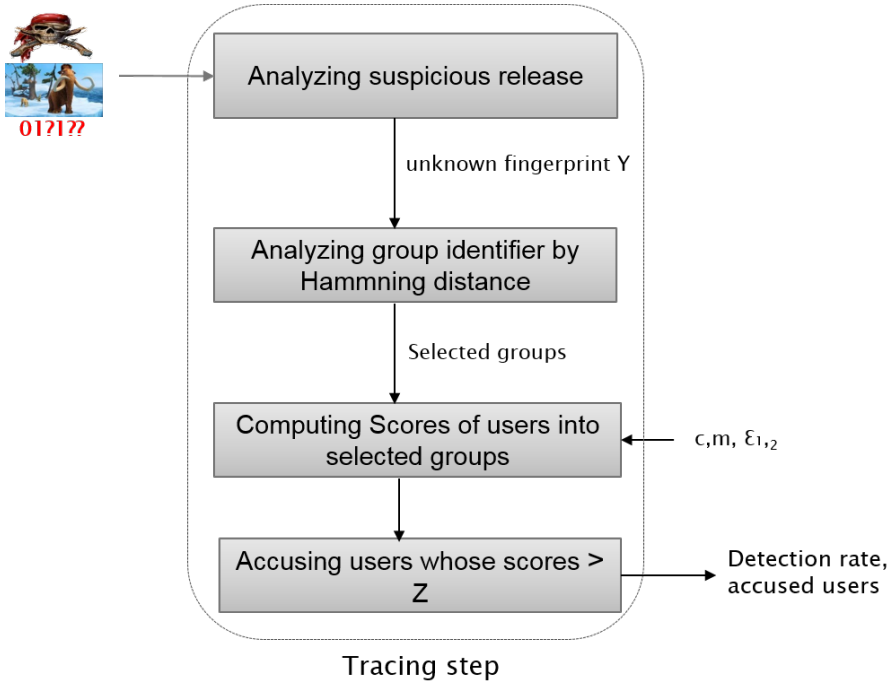


Figure 3.5: Details of the tracing step.

3.4 Experiments and discussion

The real challenge of a traitor tracing scheme is to cover the gap between theoretical and practical results. Optimizing the fingerprinting code parameters and preserving the robustness even if the collusion size increases are the most important requirements of a traitor tracing scheme. In this contribution, we propose to construct a multi-level hierarchical fingerprint whose structure should improve the Tardos accusation process in terms of time consumption and tracing performance. To validate this multi-level structure, we propose to evaluate its impact on the tracing process when considering different collusion attacks and varying collusion size. In all the experiments, we focus on varying the number of levels by comparing the multi-level hierarchical fingerprint to the non-hierarchical one. We propose, thus, to generate 1000 users' codewords with 5 colluders in a first example and 8 colluders in the second experiment. In Table 3.1, we give an illustration example for each tested collusion attack in this part.

3.4.1 Evaluation of the multi-hierarchical fingerprint versus the non hierarchical one in terms of CPU time consumption

It is undeniable that one key constraint in the Tardos-based tracing scheme is the required computational costs to parse all users' codewords and compute their corresponding scores before the accusation.

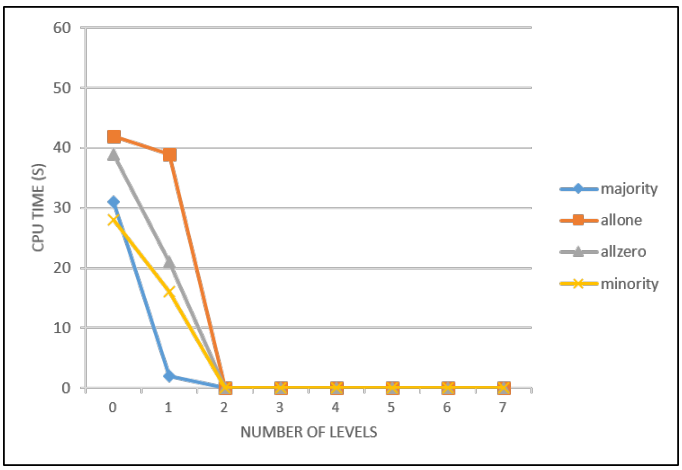


Figure 3.6: CPU time consumption by hierarchical structures for different collusion attacks and for collusion size c=5

Table 3.1: Exemples for tested collusion attacks.

Attack strategy	Example
Majority vote attack	0 1 0 1 0 0 1 1 1 **1 1 0**
Minority vote attack	0 1 0 1 0 0 1 1 1 **0 0 1**
All one attack	0 1 0 1 0 0 1 1 1 **1 1 1**
All zero attack	0 1 0 1 0 0 1 1 1 **0 0 0**

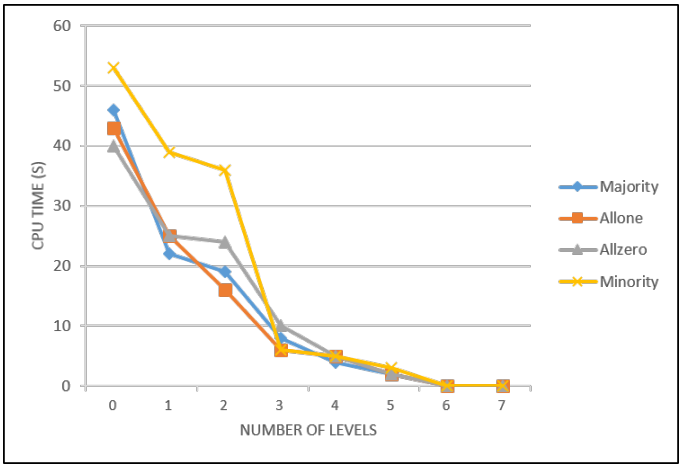


Figure 3.7: CPU time consumption by hierarchical structures for different collusion attacks and for collusion size c=8

To show the outperformance of the proposed multi-level hierarchy when addressing the time consumption during the accusation process, we propose to compute the required CPU time to trace colluders for different collusion attacks and two collusion sizes c=*5* and c=*8*. The time consumption in computed for varying hierarchical structures, ranging from hierarchy of one level to hierarchy of seven levels. The non-hierarchical structure is referred by a number of levels equal to zero. As illustrated in Figures 3.6 and 3.7, the time consumption is reduced significantly when the number of hierarchical levels is increasing. For the non-hierarchical structure, it exceeds 30s for the majority vote attack for the two collusion size, nevertheless it is close to 0.1s since the second level for c=*5* and below 2s since the fourth level. Regarding the payoff in time when using the hierarchical structure, it exceeds 95% and 60% since the second level for the two curves. Now, when regarding the collusion size, it is clear that the tracing process spends more time to retrieve 8 colluders than for 5 colluders. Indeed, for c=*8*, the CPU time becomes close to 0.1s in the sixth level. The impact of the collusion attack on the required CPU time is also an important point. In fact, from the two figures, we notice that some attacks such as the Minority vote attack has deeper effect on the time than the others. This can be proved by the fact that time values are the most important for this attack. However, with the hierarchical structure, this required time is also reduced significantly.
When studying the CPU time criterion, we notice that the hierarchical structure provides an important reduction in the time consumption of the tracing process. This can be explained by the fact that according to this structure, users are grouped together and hence the search space of the Tardos code is reduced to only the selected groups.

3.4.2 Evaluation of the multi-hierarchical fingerprint versus the non hierarchical one in terms of positive false alarm

Now, another important point to evaluate the tracing performance is the positive false alarm probability, the probability of accusing falsely an innocent.

As shown in Figures 3.8 and 3.9, the probability of falsely accusing innocents is also reduced when the number of levels is increasing. For the non-hierarchical structure, the pfa is very important and exceeds 0.5 for the majority of collusion attacks. However, it is closer to 0

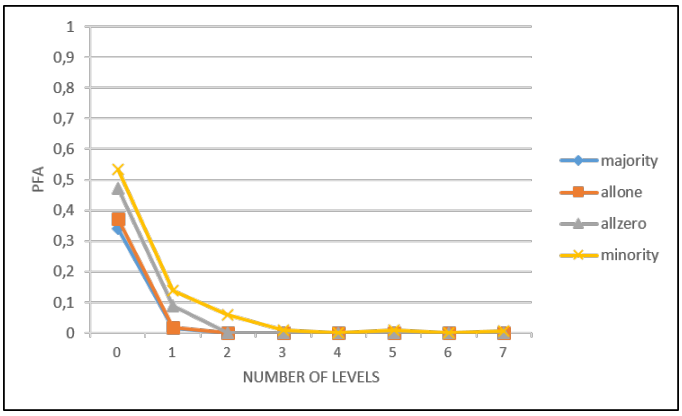


Figure 3.8: Probability of false positive of hierarchical structures for different collusion attacks and for collusion size c=5

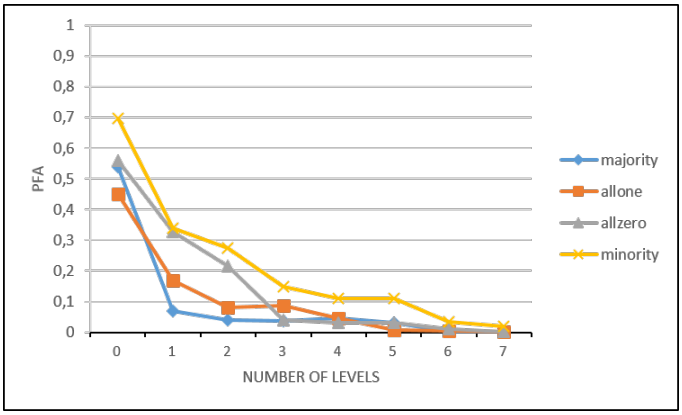


Figure 3.9: Probability of false positive of hierarchical structures for different collusion attacks and for collusion size c=8

since the third level for c=5 and the sixth level for c=8. When studying the pfa rates, we also notice that some collusion attacks have deeper effect on the accuracy of the detection rates than others. In our experiments, the minority attack has the highest values of pfa. These values are also reduced with hierarchical structure. The behavior of the pfa curve can be explained by the fact that the hierarchical structure keeps more accuracy to the accusation process. Parsing groups susceptible to contain colluders prevents from accusing falsely innocent users.

The whole experimental assessments were made with the proposed assumption that users belong to the same group. In case of having users in different groups, the proposed tracing algorithm is executed for all the selected groups (having a minimal Hamming distance). Retrieving one colluders is fairly sufficient to stop the tracing process. Otherwise, the tracing process con-

tinues to parse all the selected groups. If users belong to all the groups of the hierarchy, the tracing process is executed for all the groups because they have the similar Hamming distance, and hence we are in the worst case (which is in contrast with our assumption).

In this context, in a last part of this chapter, we focus on the group-based property of our hierarchy. In fact, this property is applied randomly to groups of fingerprints; the groups are constructed with no constraints. In [Yong *et al.* 2014], group of users are constructed by applying a clustering algorithm to the group identifier.

Data clustering process in traitor tracing context can be assimilated to an unsupervised data analysis process whose goal is to partition unlabeled users' identifiers into groups of similar characteristics, called clusters. In this part, we propose to study the users' grouping step by involving a clustering step to construct our multi-level hierarchical fingerprint. The only requirement is the codeword length of users' fingerprints which is constrained by the applied watermarking technique used in the tracing scheme [Charfeddine *et al.* 2010]. We make the studied scheme and its experimental part in the Appendix3.

3.4.3 Comparison of the proposed fingerprinting approach to other hierarchical techniques

Compared to non hierarchical fingerprints, the experimental assessments prove that the multi-level hierarchical fingerprints have good robustness to collusion attacks and are able to provide good detection rates in fewer time.

Now, when regarding the existing hierarchical tracing approaches, it is obvious that they belong to different classes: code-based tracing classes and signal-based ones. Hence, we propose to report the most interesting experimental results able to allow us to compare them to the proposed approach. Looking at the different results reported in Table 3.2, we compare the tracing complexity of respectvely [Wang *et al.* 2004], [Akashi *et al.* 2008], [Hamida *et al.* 2011] and [Ye *et al.* 2013]. The smallest decoding complexity is shown for both the proposed technique and [Ye *et al.* 2013].

Now, if we compare the experimental results related to the detection rate for [Ye *et al.* 2013], we notice that the majority vote attack is the only assessed collusion attack. We use the same experimental parameters proposed in [Ye *et al.* 2013]. We set the number of users n to 10^3, the

Table 3.2: Comparative tracing complexity results to some existing hierarchical techniques

Hierarchical technique	Tracing complexity
[Wang *et al.* 2004]	$O(N \times l)$
[Akashi *et al.* 2008]	$O(\mathrm{c}_g \times \sqrt{N} \times l)$
[Hamida *et al.* 2011]	$O(\mathrm{c}_g \times \mathrm{n}_g \times m)$
[Ye *et al.* 2013]	$O(\mathrm{m})$
The proposed technique	$O(\mathbf{m})$

number of groups to 8 and the number of colluders to 5. Compared to the CPU tracing time value of the [Ye *et al.* 2013] required to retrieve the 5 colluders whose value is about 10^{-1} s, the CPU time value required by the proposed technique is close to 10^{-2} s for a 3-hierarchical structure (the number of levels is tied to the number of groups).

In this section, we have detailed the first contribution we propose in traitor tracing field. We have constructed a multi-level hierarchical fingerprint in order to apply a group-based tracing process which has proven a significant reduction of tracing and computational costs of the Tardos code even compared to other hierarchical existing tracing approaches.

3.5 Conclusion

In this chapter, we presented the group-based fingerprinting system we propose for traitor tracing in multimedia distribution platform. The construction of fingerprints is based on a multi-level hierarchy where each level corresponds to a constraint inspired from the threat channel in the platform. The aim from this construction is to reduce the search space of the Tardos code and hence to reduce the complexity of the Tardos decoding step even in case of great number of users. We performed a detailed analysis of the proposed system performance according to two criteria: the robustness to collusion attacks and the tracing time criterion. The proposed fingerprinting system was evaluated for different collusion sizes. We also assigned an important consideration to the comparison of the performance of the proposed hierarchical system with non hierarchical one. In case of having more than one selected group to parse, we

have proposed to use a clustering step to ensure the group-based property which has improved a little bit the tracing results. In a future work, we propose to ensure the group selection step by using a collusion secure fingerprint. This selection should improve more the accuracy accusation of the Tardos code.

CHAPTER 4

A two-stage traitor tracing scheme for hierarchical fingerprints

Contents

4.1 Introduction

The multimedia traitor tracing field involves the embedding of a collusion secure fingerprint in the host signal to retrieve and prevent any multimedia content fraud. Trendy work aims at providing a tracing system which offers a good protection of the digital content and an efficient tracing process. These challenges depend on reducing the length of the embedded fingerprint and the complexity of the accusation process. Furthermore, addressing these issues becomes more and more relevant in media distribution applications involving an important number of users. In this chapter, we propose to improve our fingerprinting system by applying a two-stage tracing strategy which combines two probabilistic tracing codes: Boneh Shaw with replication scheme and Tardos codes. The tracing strategy consists in a two-stage decoding strategy which is applied to the multi-level hierarchical fingerprint. Hence, we decode in two successive levels: we accuse groups in a first level with Boneh Shaw code and then we use the Tardos code to accuse users only into the retrieved groups. This strategy, improved by the concept of weighted groups, enables a good colluders' detection rate. It provides an efficient reduction of the computational costs of the tracing step compared to other two-stage tracing systems proposed in the literature. According to [Desoubeaux *et al.* 2012] and in order to survive the problem of the important code length when considering a two-level tracing scheme, we use a 4-cardinality alphabet to construct the resulting fingerprint, BST-H, which combines the two tracing codes: Boneh Shaw and Tardos. BST-H is then embedded using a DCT-based audio watermarking technique to evaluate its robustness to different types of attacks. The applied watermarking technique is an existing technique [Charfeddine *et al.* 2012] which has proven good robustness against different types of audio Stirmark attacks, we apply it to embed the proposed two-level code to evaluate it in terms of inaudibility and robustness. We further review some related work proposing two-level accusation strategies in multimedia fingerprinting systems. Then, we show the experimental results designed to assess the tracing performance of the whole framework according to different criteria.

4.2 The improved fingerprinting system based on the two-level tracing strategy

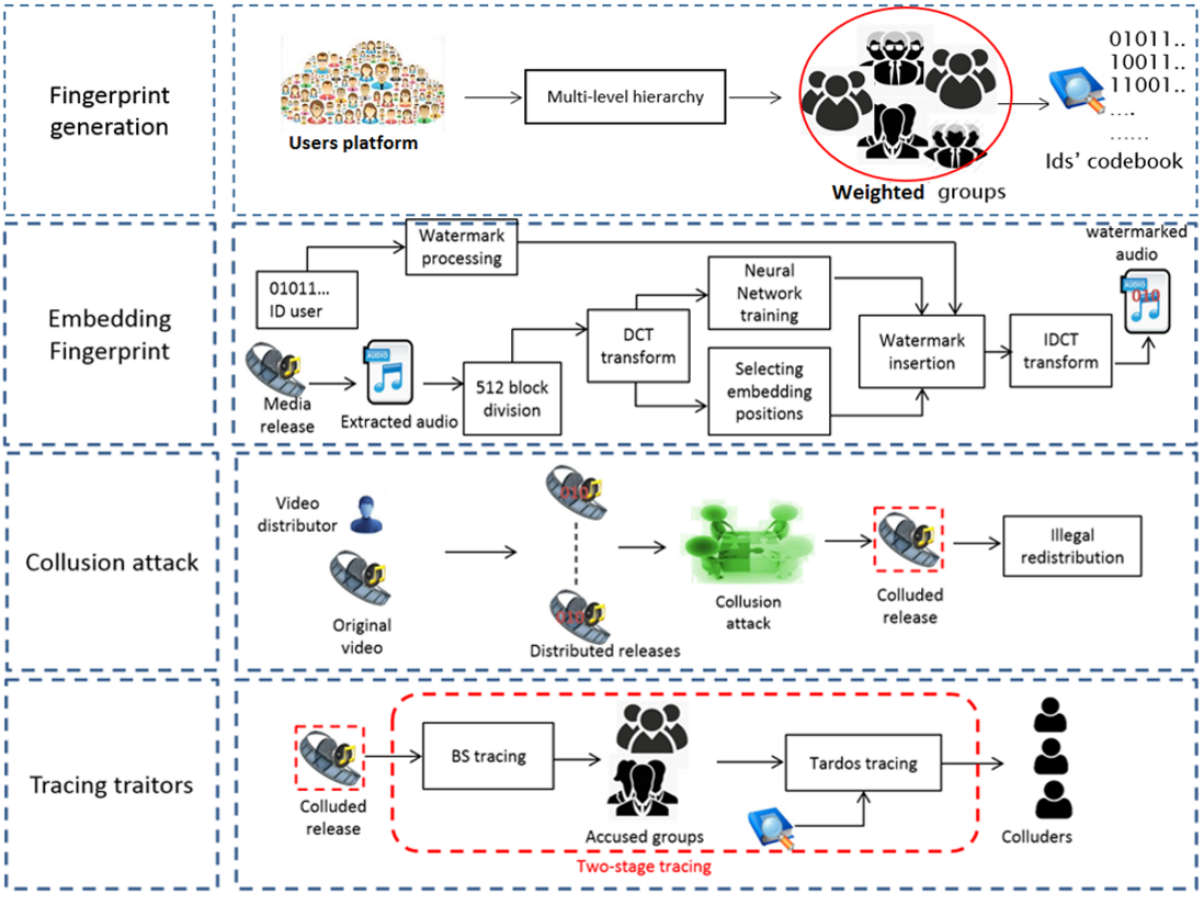


Figure 4.1: The multimedia distribution application including a traitor tracing system

Before illustrating the proposed tracing strategy, we present the proposed fingerprinting framework applied to a multimedia distribution platform. The proposed fingerprinting system has a twofold purpose. Indeed, in a first time, we aim at constructing a multi-level hierarchical fingerprint as shown in Figure 4.1. This construction consists of four parts: the fingerprint generation step, the embedding step, collusion attacks and the traitor tracing process. In this section, we make a description of each part separately.

4.2.1 The proposed two-level tracing step

Compared to [Desoubeaux *et al.* 2012] and [Kuribayashi *et al.* 2008], the proposed tracing strategy has two particularities: instead of randomly forming group of users, we group them according to the hierarchy we proposed in [Chaabane *et al.* 2014] whose different levels are relevant in group construction step. Moreover, after computing groups' scores with the BS code, we attribute to each score a weight which is generated by the hierarchy and indicates the degree of the piracy rate. For instance, we assume that some groups have higher level of piracy rates than others according to the different criteria of our proposed hierarchy. Hence, groups having the highest weights should be traced above all because they have more probability to contain traitors. To face the problem of the codeword length due to the sequential concatenation of the two codes, authors in [Desoubeaux *et al.* 2012] propose to increase the alphabet cardinality of the obtained code, called BST, to q_{BST}, given that $q_{BST} = q_{BS} \times q_T$ [Desoubeaux *et al.* 2012]. In Table 2.5 and Table 4.1, two examples of constructing a user codeword X_j are given. The first one consists in the construction of a TT codeword from two binary codes: X_j^g (as a group identifier) and X_j^u (as a user identifier) and the latter is a user codeword X_j of 4-cardinality alphabet from respectively two binary codes: X_j^g and X_j^u . In the proposed technique, instead of sequentially combining the two identifiers, group identifier and personal identifier, to get the final codeword X_j and so falling in the trap of the codeword length as the case of [Kuribayashi *et al.* 2008], we double the alphabet cardinality of the obtained code as proposed in [Desoubeaux *et al.* 2012] by using equal lengths to the two codes. The constructed code consists in symbols embedded in the same position inside the content. We illustrate the principle of a two-level tracing code in Figure 4.2. Indeed, it consists in com-

Table 4.1: Example of constructing a codeword from 4-cardinality alphabet

X_j^g	0	.	.	**0**	1	**1**
X_j^u	1	.	.	**0**	0	**1**
X_j	2	.	.	**0**	1	**3**

bining two tracing codes: the C_g code is assigned to identify the group and C_u code is assigned to identify a user into his group. The corresponding coding function is the combination $C_g \circ C_u$ which generates the totality of $n_g \times n_u$ users, with n_g is the number of groups and each group

contains exactly n_u users. In the following, we denote the Boneh Shaw code by BS, the Tardos

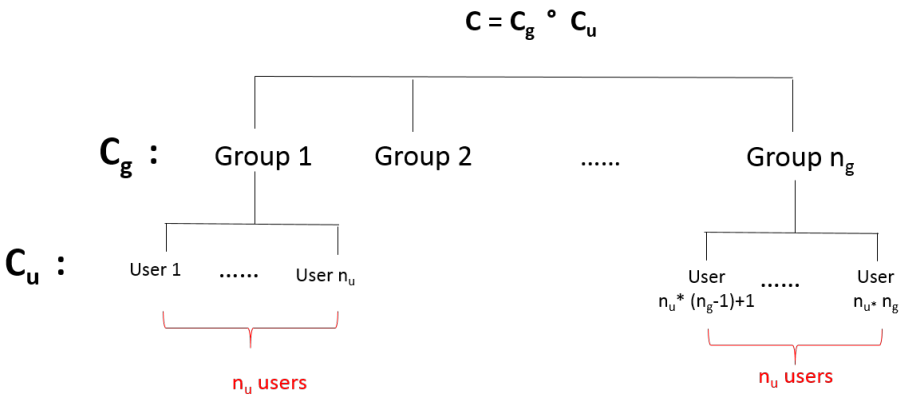


Figure 4.2: The general concept of a two-level tracing code [Desoubeaux *et al.* 2012].

by T, the combination of the two codes in [Desoubeaux *et al.* 2012] by BST and the proposed code by BST-H. The output of the Tardos tracing algorithm includes a user j if $S_j > Z$ with a false positive error $< \varepsilon_1$. In the proposed work, we adopt the two-level accusation strategy as in [Desoubeaux *et al.* 2012] but we improve the second level with a weight-based selection.

The principle of the proposed two-stage tracing strategy: In view of the above, the two-level fingerprint BST-H consists in applying successively two codes: BS with replication scheme BS(n_g, d) applied in a first time to frame a guilty group and then the Tardos code T(n, c, ε_1) in a second level. BS code is applied to n_g groups. In the first step, the weight W_{g_i} of each d-length block is computed. It is the difference of 1's bits number between $(k-1)\times d$ and $k \times d$. Let $w = (w_1, \cdots, w_{n_g})$ the weight vector and σ_k^g the score related to a group k. We remind that the tracing algorithm of the BS proposed in [Boneh & Shaw 1998] proceeds as follows:

1. $\sigma_k^g = w_1$ for the first group,
2. $\sigma_k^g = w_k - w_{k-1}$ for $k \in \{2, \cdots, n_g\}$,
3. $\sigma_k^g = d - w_n$ for $k = n_g$.

The main target in [Desoubeaux *et al.* 2012] was to make a simple group decoding process. Indeed, when the suspicious sequence Y is detected, the scores of the groups are decreasingly sorted and then analyzed in this order until retrieving one colluder. In the proposed approach,

we aim to enhance the accusation process by using the hierarchical structure of the fingerprint, and so we proceed differently. In fact, we add another weight h_k according to the hierarchy knowing that $\sigma\prime_k^g = h_k\sigma_k^g$. Using the weights generated from the hierarchy should enable a faster accusation and more accurate decoding results process than in [Desoubeaux *et al.* 2012]. The Tardos accusation is then applied to decoded groups to retrieve at least one colluder. Our tracing algorithm is summarized by the Algorithm.2. Compared to the joint decoding princi-

Algorithm 2 Tracing algorithm.

$[\mathrm{n}_d] =$ **BST-H**(c,n_g, h, ε_1, n_u)

input : c: number of colluders, n_g: number of groups, h: hierarchy weights, ε_1: false positive probability, n_u: number of users

output: n_d: number of detected colluders

begin

/* First level accusation: BS accusation */

if $(W_1 > 0)$ **then**

$first\_gr = 1$ **else if** $(W_{n_g-1} < d)$ **then**

$first\_gr = n_g$

end

else

for $2 \leq i < n_g - 1$ **do**

/* a loop for remaining groups */

$dw(i) = |W_{i+1} - W_i|$

end

end

end

end

Sort dw

/* Second level accusation: Tardos accusation */

$dw\prime \leftarrow hierarchy(h, dw)$

$n_d \leftarrow Tardos(first\_gr, dw\prime, n_u)$

ple which consists in computing the users' scores and assigning iteratively weights to groups

[Meerwald & Furon 2012], the proposed tracing scheme is closely attached to the hierarchy we proposed in [Chaabane *et al.* 2014], and the weight-based decision is based on the weights generated from the hierarchy. For the security side, two secret keys are used in the tracing scheme: the embedding positions in the audio chunks and the probability sequence p of the Tardos code. Another security point attached to the two-level structure is the number of the embedded symbols which makes the cooperative between colluders difficult.

4.2.2 The considered threat channel

In this part, we detail the different collusion attacks we will apply in our experimentation. We have chosen to study different types of collusion attacks and even the worst one to analyze the performance of the proposed fingerprinting system. According to [Furon & Pérez-Freire 2009b], a collusion strategy made by a collusion of size c is defined by a vector Θ with $\Theta = (\Theta_0, \cdots, \Theta_c)$. Y_j is the random variable which represents the binary symbol in a position in the pirated release:

$$\Theta(k) = P(Y = 1 / \sum_{j \in C} Y_j = k). \tag{4.1}$$

According to the Marking assumption, $\Theta(0) = 0$ and $\Theta(c) = 1$. In this work, we apply:

- The interleaving attack defined by:

$$\forall k \in [c+1], \Theta(k) = k/c. \tag{4.2}$$

- The Majority attack defined by:

$$\Theta(k) = \begin{cases} 0 & \text{if } k \in [0, c/2[, \\ k/c & \text{if } k = c/2, \\ 1 & \text{if } k \in]c/2, c]. \end{cases} \tag{4.3}$$

- The WCA attack defined by:

$$\Theta(k) = \arg\min_{\Theta} \{R(\Theta)\}. \tag{4.4}$$

R is the achievable rate of tracing code [Furon & Pérez-Freire 2009b].

- The averaging attack is defined by:

$$\Theta_i(k) = (\sum_{i,k \in c} X_{ik})/c. \tag{4.5}$$

X_{ik} is the user codeword of length m.

4.3 The audio watermarking technique

Embedding collusion secure fingerprint codes with a robust watermarking technique has necessarily impacts in a tracing scheme, mainly against some types of robustness attacks. In our tracing approach, we propose to use a DCT-based audio watermarking technique described in details in [Charfeddine *et al.* 2010]. As depicted in Figure 4.3 and Figure 4.4, this watermarking technique was proposed by [Charfeddine *et al.* 2010] and is essentially based on the DCT transform. The DCT is applied to each 512-length block of the audio stream. The resulting watermark is of size pq. After that, a random selection of pq indexes is made to choose the blocks to be watermarked. Furthermore, the watermark symbols are embedded in the Middle Frequencies, MF, band of each chosen block. To enhance the security side of the watermarking technique, a Back Propagation Neural Network, BPNN, is trained and simulated to select the most appropriate embedding position. To obtain the watermarked audio chunk, an IDCT transform is applied to the modified blocks. For the detection step, as shown in Figure 4.4, it consists in the inverse of the embedding step to extract the watermark.

4.4 Experiments and Evaluation

In this section, we will evaluate our fingerprinting system using the proposed improved strategy BST-H. We show in the first part the tracing results in terms of group-based decoding performance and the robustness against different types of attacks. In a second part, we test the robustness and the inaudibility of the applied watermarking technique against some audio Stirmark attacks. According to [Choi *et al.* 2012] and to the study we made in [Chaabane *et al.* 2014], the most required videos in a multimedia distribution platform are films, TV reality programs, political speeches, sport competitions and music clips. Thus, we

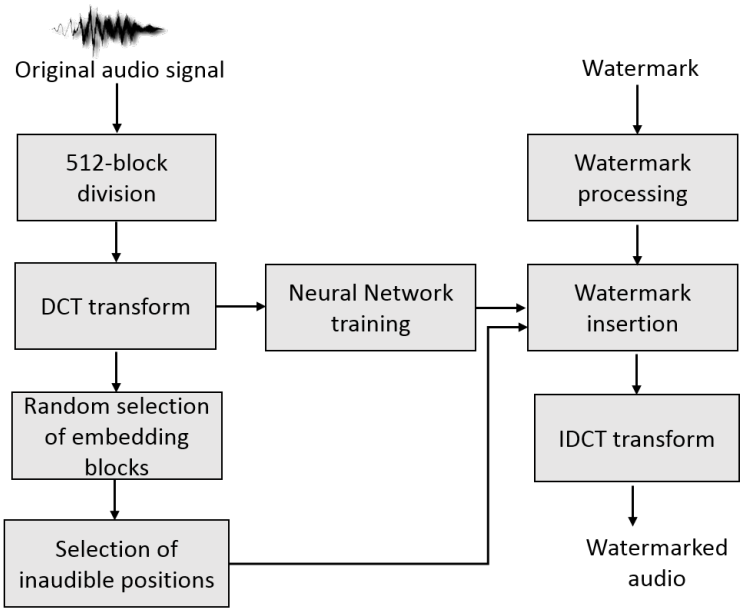


Figure 4.3: The embedding step of the DCT-based audio watermarking technique[Charfeddine *et al.* 2010]

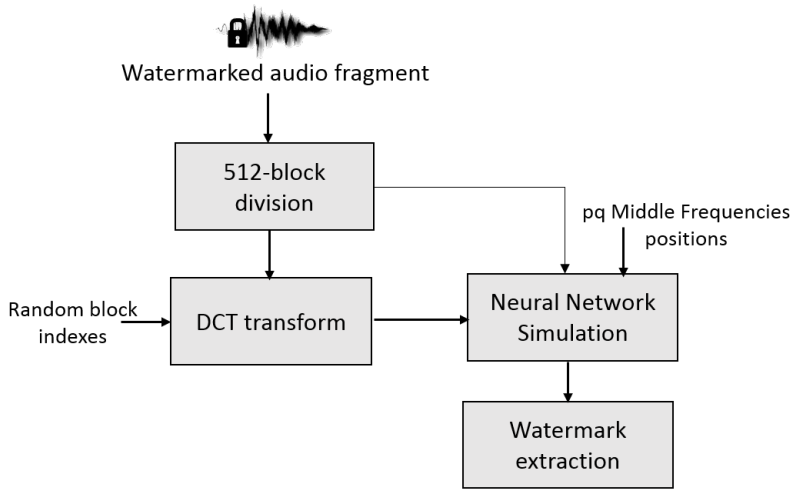


Figure 4.4: The detection step of the DCT-based audio watermarking technique[Charfeddine *et al.* 2010]

have tested different types of videos as shown in Figure 4.5. In Table 4.2 and Table 4.3, we present details of these used video files and extracted audio samples. We have carried out our tests in Matlab on a workstation with i-7 processor at 2.7 GHz and 16 Gb of RAM.

Table 4.2: Properties of all experimental video samples

video files	Time length (min:s)	Format	Resolution	Frame rate (fps)
match	11:39	AVI	320×240	25
TV	6:57	AVI	320×240	25
song	4:53	AVI	320×240	25
film	00:24	AVI	720×576	25
clip2	3:45	AVI	144×192	24
sport	13:56	AVI	640 ×360	29.97
politic	1:28	AVI	640 ×360	29.97
film2	1:55	AVI	640 ×360	23.97
prog2	3:42	AVI	640 ×360	29.97

Table 4.3: Properties of all experimental audio samples

Time length (s)	20s
Format	wav
Bits per sample	16
Sample rate (KHz)	44.1
Channel mode	mono

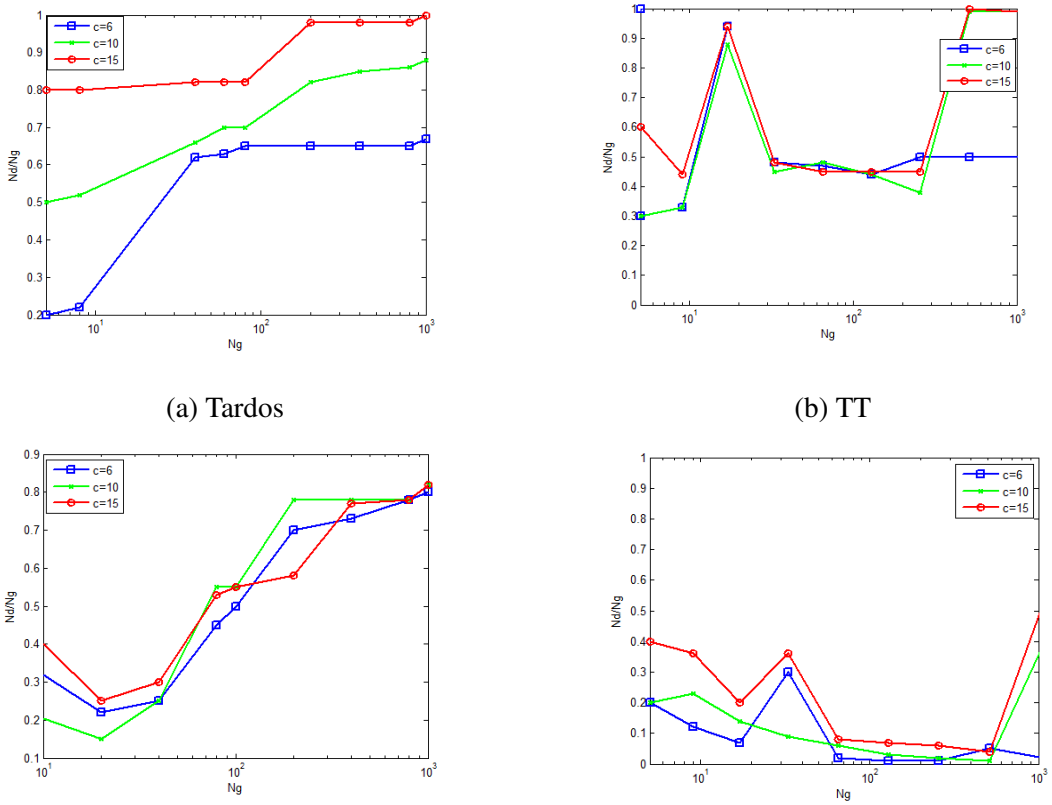
Figure 4.5: Snapshots samples of experimented videos.

4.4.1 Tracing results

In this section, we give experimental results obtained with the improved strategy of BST code. In fact, we show the tracing results against different collusion attacks. In order to prove the outperformance of the system in terms of tracing rates, we compare it to the two-level existing strategies using group-selection technique [Desoubeaux *et al.* 2012], BST, and [Kuribayashi *et al.* 2008], TT, and also to the original Tardos code, T, [Tardos 2003] applied here with randomly selecting the groups. We use the same experimental parameters proposed in [Desoubeaux *et al.* 2012] as shown in Table 4.4. We set the number of users n to 10^5, the code length m is equal to 2048 symbols, we vary the collusion sizes c in $\{6, 10, 15\}$. In the following experimental results, the results obtained with the proposed strategy are in bold. In this part, we show the tracing results of the proposed BST-H compared to the TT [Kuribayashi *et al.* 2008] and BST [Desoubeaux *et al.* 2012]. The proposed BST-H should provide less complexity in the decoding step in $O(m)$ compared to TT code which is in $O(m \times \mathrm{n}_g)$. It should also offer an optimization in group selection compared to the BST code by using the hierarchy. We are

Table 4.4: Expectation of bad decoded groups for T, BST, BST-H and TT, m=2048, c=2, 6, 10 and 15

d	2	4	8	16	32	64	128	256	512
$\mathbf{n}_g$	1025	513	257	129	65	33	17	9	5
T c=2	678.2	342.72	169.62	84.61	40.67	20.52	9.92	0	0
BST c=2	353.45	89.88	15.59	1.7	0.04	0	0	0	0
BST-H c=2	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
TT c=2	0	0	0	0	0	0	0	0	0
T c=6	876.87	433.31	215.44	104.22	49.74	22.68	9.73	3.6	0.15
BST c=6	827.51	378.72	174.95	69.88	24.69	6.18	0.68	0.01	0
BST-H c=6	**1.37**	**1.93**	**1.4**	**1.1**	**0.93**	**0.3**	**0.25**	**0.12**	**0**
TT c=6	511	255.5	127	56	30	14	15	2	0.5
T c=10	902.71	447.2	219.1	107.5	43.7	31	9.5	4.5	1.5
BST c=10	827.3	401.39	203.25	71.2	36.95	8	3.25	2	0.75
BST-H c=10	**385**	**5**	**4.75**	**4.01**	**3.19**	**2.12**	**1.54**	**1.12**	**0**
TT c=10	1020	508	98	56.75	30.5	14	14	2	0.5
T c=15	994.25	471.2	251.21	122	44.6	31	12.5	7	2
BST c=15	194	406.26	205.7	76	38.25	25	6.67	4.25	1.5
BST-H c=15	**513**	**19**	**15.33**	**7.2**	**3.3**	**10**	**1.51**	**1.3**	**0**
TT c=15	1020	511	115	57	31	16	14	2	1



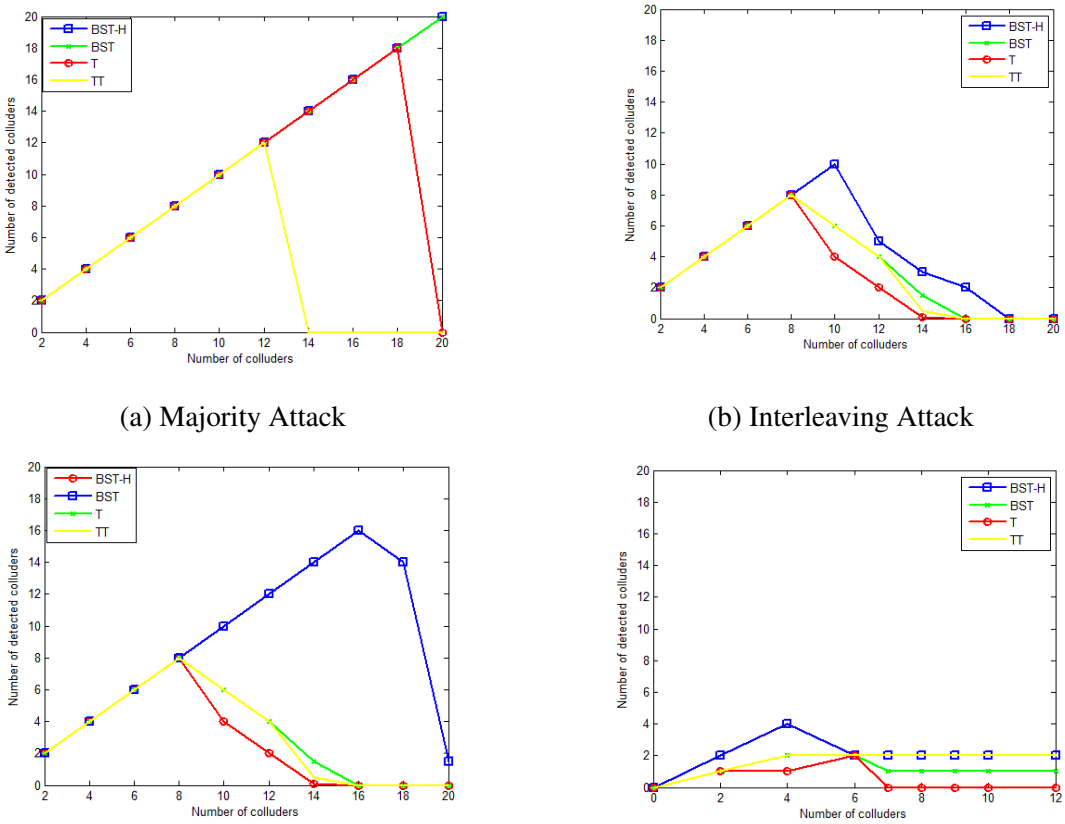
(a) Tardos

(b) TT

(c) BST

(d) BST-H

Figure 4.6: Expectation values of decoded groups computed by different codes to trace c colluders.



(a) Majority Attack

(b) Interleaving Attack

(c) Averaging Attack

(d) WCA Attack

Figure 4.7: Average detected colluders for BST-H, BST, Tardos and TT when varying collusion size.

interested mainly in three criteria: the group selection for different collusion size (6, 10 and 15), the traceability and the time consumption taken by each strategy to find colluders.

4.4.1.1 The group selection criterion

In this part, we show that assigning groups can have some limitations. In fact, the decoding rate performance is presented in Table 4.4 as the error of the number of bad decoded groups, i.e. the groups not including colluders. We give the expectation of this error for 1000 trials of the tracing algorithm. It is undeniable that the expectation of the error increases with the collusion size and decreases with the replication factor d. For *c=2*, the decoding error is null. However for the other sizes, it could be better to decode more groups and accept errors than to have best performance in expectation. Collusion size increases the error for the four codes but the decoding performance is still interesting for BST-H which reduces efficiently the users' search space by almost 38 percent. It is outstanding that compared to the TT, BST-H gives good performance and improves efficiently the performance of BST. The worst performance is given by the Tardos code in the Interleaving scenario. According to [Meerwald & Furon 2012], the Tardos decoding is quite fast for $n_u = 10^6$, and m=1024. Moreover, BST-H takes the advantage of the selection group to manage a great number of users. Hence, the tracing scheme we propose can scale to a number of users $n = 10^6$ per group and so the totality of $n = 10^9$ given $n = n_u \times n_g$ and n_g is enhanced up to $n_g = 10^3$ in our experiments. In Figure 4.6, we present the expectation values of the ratio n_d/n_g required by Tardos, TT, BST code and BST-H codes for different size of collusion. The value of n_d/n_g computes the ratio between the number of the decoded and the total number of groups and defines the proportion of users required to retrieve *c* colluders. When this ratio is close to 1, the tracing code needs to decode the totality of groups to trace all the colluders.

In Figure 4.6(a), the ratio n_d/n_g increases when the collusion size increases. An evident behavior which is explained by the Tardos decoding step which needs more groups to decode when c increases. In Figure 4.6(b), TT requires to parse more than 60 percent of groups for *c=20* to retrieve the colluders for n_g more than 20. In Figure 4.6(c) and (d) the ratio n_d/n_g for BST-H is lower than for BST when the number of n_g is important. We notice that despite the important

size of collusion and the increasing number of groups, we do not require with our BST-H code to decode more than 50% of the totality of users' number.

4.4.1.2 Decoding performance

In this section, we evaluate the traceability of the four techniques in terms of the averaged values of, respectively, the detected colluders and the false negative probability. As shown in Figure 4.7, we applied four attacks to show the impact of different types attacks on the decoding performance of the tracing code. As studied in [Furon & Pérez-Freire 2009b] and shown in Figure 4.7(d), the WCA attack has a deeper influence on the decoding performance than the others attacks. Faced with the four attacks, The proposed BST-H has the best decoding performance and guarantees to decode at least two colluders even for the worst case attack. In Figure 4.8, we vary the collusion size and we compare the different techniques in term of false negative error probability. According to [Kuribayashi *et al.* 2008], [Desoubeaux *et al.* 2012] and our assessment, we observe that despite the fact that the false negative probability increases when the collusion size is important, the BST-H code has the least probability of missing guilty users.

4.4.1.3 The tracing time criterion

In Figure 4.9, we show the mean of the tracing time taken by the three strategies: BST-H, BST and TT for different number of groups. The proposed BST-H has the shortest tracing time compared to the others (less than 1s for n_g less than 10^3).

4.4.2 Robustness and inaudibility results

We use two major criteria in this experimental part: *NC*, the Normalized Cross Correlation, which value reflects the similarity between the original watermark and the detected one, and the *SNR*, the Signal to Noise Ratio, which is an objective measure to show the quality of the audio after the insertion step.

$$SNR = 10 \times \log \left(\frac{\sum_i Y_i^2}{\sum_i \left(Y_i - Y'\right)^2} \right) \tag{4.6}$$

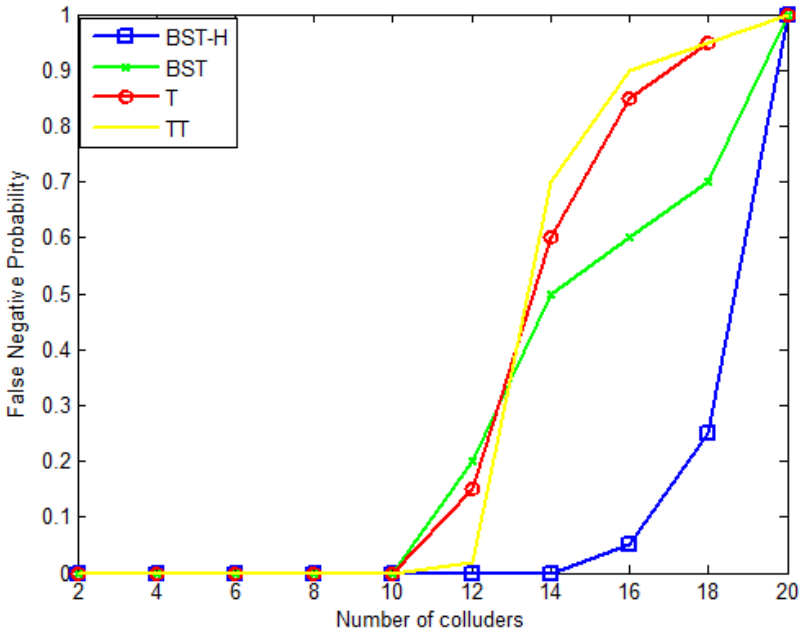


Figure 4.8: False Negative Probability of by BST-H, BST, Tardos and TT when varying collusion size.

Y and Y' are respectively the original audio and the watermarked one.

$$NC = \frac{\sum_{x=1}^{N} W(x) \cdot W'(x)}{\sqrt{\sum_{x=1}^{N} W(x)^2} \cdot \sum_{x=1}^{N} W'(x)^2} \tag{4.7}$$

W and W' are respectively the original watermark and the detected one.

We remind that Tardos parameters m and c are tied by the equation below:

$$m = 2\Pi^2 c^2 \left[\ln \frac{1}{\varepsilon_1}\right] \tag{4.8}$$

4.4.2.1 The evaluation criteria

To check the robustness and the inaudibility of the proposed fingerprinting system, we used the two major criteria in this experimental part: the NC and SNR measures to show the quality of the audio after the insertion step.

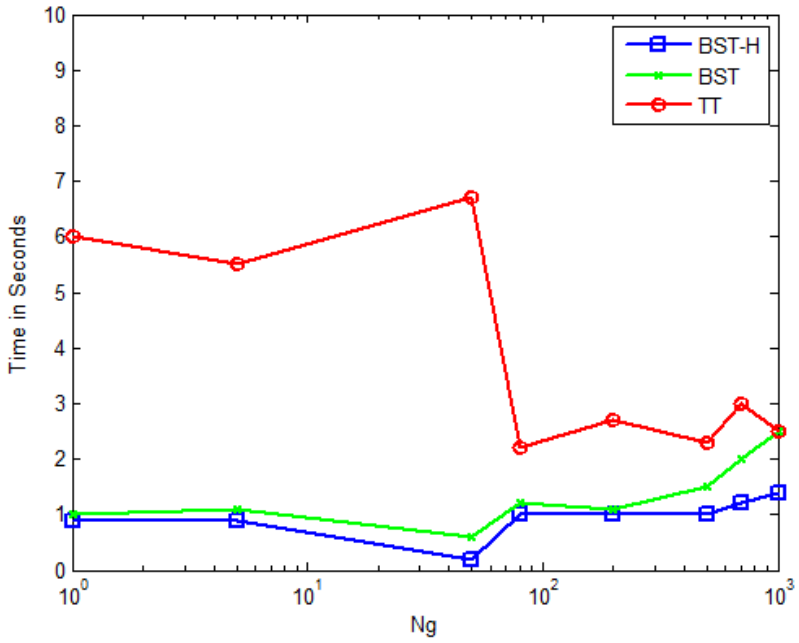


Figure 4.9: Comparison of tracing time taken by BST-H vs BST and TT codes

4.4.2.2 The Robustness results

In this part, different experiments were conducted to show the robustness of the proposed system against different audio Stirmark attacks for the four video samples. As shown in Figure 4.10, the NC values, above 0.9 for the majority of attacks, prove that the embedded watermark is not altered and so the watermarking technique has a good robustness. In Figure 4.11, another experiment was conducted to evaluate the robustness of the watermarking technique against compression attack. Although we use different MP3 compression rates: 64, 96 and 128 Kbps, the value NC is close to 1 for the four videos, and hence we obtain good robustness results.

4.4.2.3 The inaudibility results

Checking the inaudibility of the audio watermarking technique is performed according to subjective and objective tests [Neubauer & Herre 1998]. These tests are reported in Appendix 1, we keep in this chapter only SNR values.

According to the IFPI, the International Federation of the Phonographic Industry, a watermarked audio signal should have an SNR value more than 20 dB [Charfeddine *et al.* 2012].

(a)for "Match.avi" (b)for "tv.avi" (c)for "song.avi" (d)for "film.avi" (e)for "clip2.avi" (f)for "sport.avi"

Figure 4.10: Stirmark attacks for the embedded fingerprint for different videos samples.

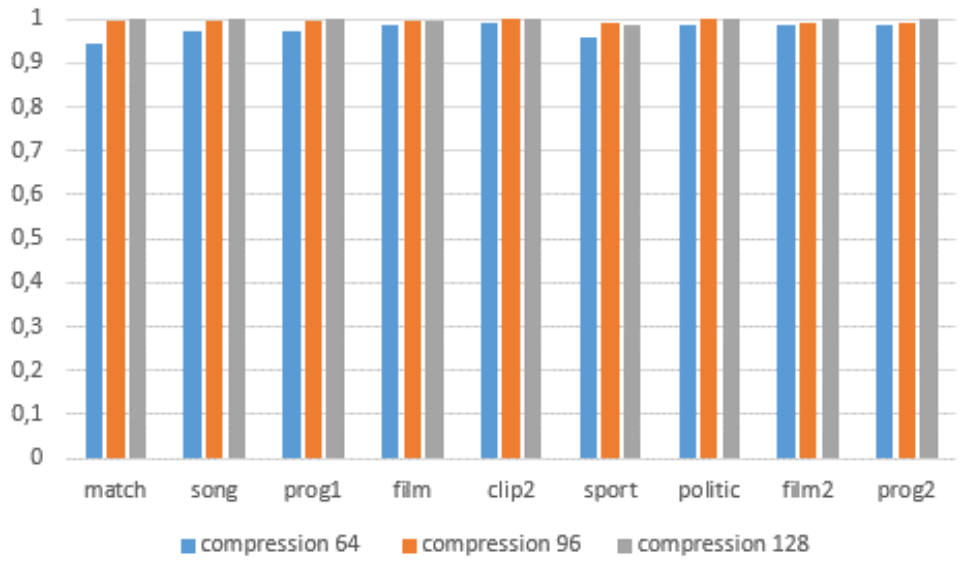


Figure 4.11: Robustness against different compression rates: 64, 96 and 128 Kbps for "match.wav" , "song.wav", "program.wav" and "film.wav"

Table 4.5: Inaudibility results for our video sequences

video title	Extracted audio file	SNR(dB)
"Clip"	"song.wav"	**52.1**
"match"	"match.wav"	**38.57**
"Prog1"	"prog1.wav"	**48.41**
"film"	"film.wav"	**53.02**
"clip2"	"clip2.wav"	**57.21**
"sport"	"sport.wav"	**58.59**
"politic"	"politic.wav"	**49.35**
"film2"	"film2.wav"	**57.35**
"prog2"	"prog2.wav"	**53.01**

Thus, all obtained SNR values exceed 20 dB which proves that the watermarking technique has good inaudibility results.

4.4.3 Study of the security of the audio watermarking scheme

One key perspective when dealing with traitor tracing scheme is to study the impact of the robustness and security constraints of the data-hiding technique on the attack strategy of the colluders and the accusation process [Mathon *et al.* 2013]. In this part, we propose to study and evaluate the security level of the audio watermarking we use to embed our fingerprints. The security of a watermarking technique, as mentioned, is referred mathematically to the estimation of embedded symbols by colluders for each position i in the codeword of length m with an error ε. It is assumed that if the ε value is closer to 0, colluders have more probability to guess embedded symbols in their releases and to execute the worst possible attack, the watermarking is not sure. Nevertheless, if ε exceeds 0.5, colluders do not know if the embedded symbols are 0 or 1 and the scheme is considered sure, colluders have not thus a grand choice of attacks .

In the embedding step of the adopted watermarking technique, as described in [Charfeddine *et al.* 2010], after applying a DCT transform to audio blocks, a sequence of pq positions is randomly generated where pq represents the watermark size. Then, given these pq positions, we proceed by selecting pq blocks and the watermark is inserted in MF of each block. Hence, from a mathematical point of view, the probability that a colluder is able to guess the embedding position (i.e., a secret key) inside his codeword is very small (about $\frac{1}{pq}$). In fact, colluders cannot succeed to watermark a block without knowing this secret key. Moreover, to collude together, two colluders or more should know their respective symbols in a same position i. Hence, the probability of occurring of these two events is very small (about $\frac{1}{pq} \times \frac{1}{pq}$). According to [Charfeddine *et al.* 2010], the capacity of the audio watermarking scheme exceeds 1024 bits for only a 20s duration audio chunck. Henceforth, the event of guessing embedding symbols and cooperating in a fusion attack should be a rare event. The probability of error estimation is important (about $1 - (\frac{1}{pq})^c$) and exceeds eventually 0.5 which prove that our audio watermarking scheme is fairly sure.

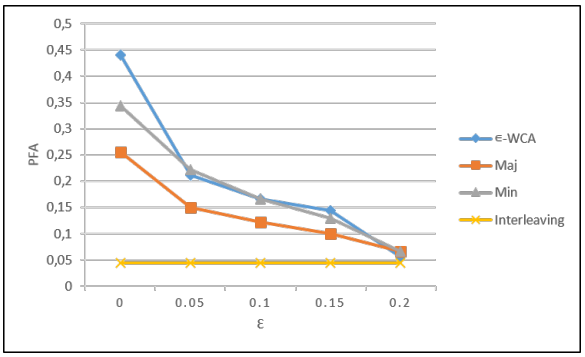


Figure 4.12: Estimation of pfa for different collusion attacks depending on ε for c=3, m=300, $\tau = 80$.

4.4.3.1 Impact of security constraints on the accusation process

We propose to evaluate the security level of our audio watermarking scheme by introducing the $\varepsilon - WCA$, the $\varepsilon-$ Worst Case Attack, where the collusion strategy $\Theta_{\varepsilon-WCA}$ minimizes the achievable rate with regard to the error estimation rate. In this part of experiment, due to the heavy computational tests, we choose only "match.wav" as an audio sample.

Table 4.6: Values of $\Theta_{\varepsilon-WCA}$ depending on ε for c=3 and c=4, for c=2 $\Theta_{\varepsilon-WCA} = (0.0.51.)$.

	$c=3$	$c=4$
$\varepsilon = 0$	(0. 0.651 0.349 1.)	(0. 0.487 0.5 0.513 1.)
$\varepsilon = 0.05$	(0. 0.726 0.274 1.)	(0. 0.543 0.5 0.457 1.)
$\varepsilon = 0.1$	(0. 0.830 0.170 1.)	(0. 0.620 0.5 0.379 1.)
$\varepsilon = 0.15$	(0. 0.982 0.018 1.)	(0. 0.734 0.5 0.266 1.)
$\varepsilon = 0.2$	(0. 1. 0. 1.)	(0. 0.908 0.5 0.091 1.)
$\varepsilon > 0.2$	(0. 1. 0. 1.)	(0. 1. 0.5 0. 1.)

This collusion attack was proposed by [Mathon *et al.* 2013] to check the tracing performance of a tracing scheme when considering its security level. An example of this attack for c=3 and c=4 is defined by [Mathon *et al.* 2013] in Table 4.6.

As shown in Figure 4.12 and 4.13, we estimate the probability of false alarm,*pfa*, depending on ε for four collusion attacks: the majority vote attack, the minority vote attack, the $\varepsilon - WCA$

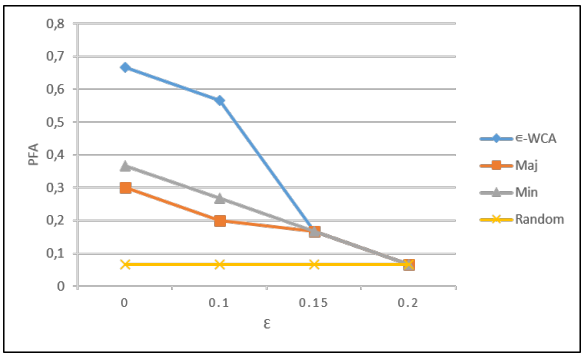


Figure 4.13: Estimation of pfa for different collusion attacks depending on ε for c=4, m=300, $\tau = 80$.

attack and the interleaving one for a threshold τ fixed to 80 here. This estimation is based on 1000 trials using the algorithm of rare events proposed in [Furon *et al.* 2009]. For the two collusion size, we notice that the highest values of *pfa* are obtained with the $\varepsilon - WCA$ attack with a small reduction for the other strategies. So for a small value of error, the $\varepsilon - WCA$ attack can be considered the worst case attack that colluders can achieve. For an $\varepsilon \geq 0.2$, all strategies converge to behave as an interleaving attack. This can be explained by the fact that once the error estimation is great, colluders do not have only one choice to collude which is exchanging their blocks.

4.4.3.2 Evaluation of both security and robustness attacks

In this part, we propose to estimate the *pfa* depending on ε after a colluding strategy which is the $\varepsilon - WCA$ attack and a robustness attack. The chosen audio Stirmark attacks are addnoise and MP3 compression attack at 96 Kbps rate. The addNoise attack is classified by the StirMark Benchmark for Audio SMBA as an Add/Remove attack [Petitcolas *et al.* 1998]. And hence, both the two selected robustness attacks have a deep effect on the quality of the audio. As depicted in Figure 4.14, the estimated *pfa* is decreasing with the increasing of ε. Nevertheless, the AddNoise attack applied after the colluding strategy $\varepsilon - WCA$ is more efficient than the compression and $\varepsilon - WCA$ attacks. For a value of ε close to 0.5, *pfa* is reduced significantly which can be explained by the security level of the scheme. Henceforth, for a non-sure watermarking scheme, in case of combining a collusion attack to a robustness attack, it can be beneficial to a colluder, which is not the case with our adopted audio watermarking technique.

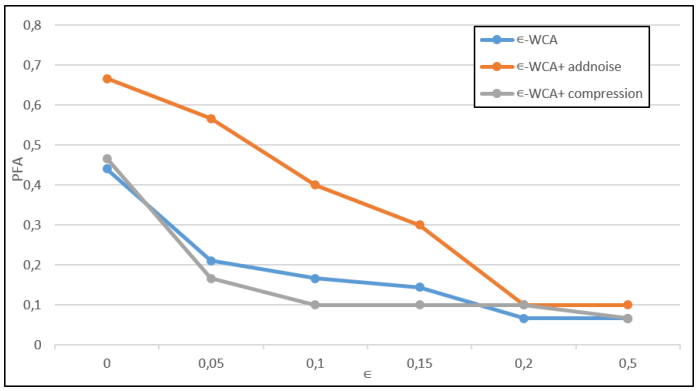


Figure 4.14: Estimation of *pfa* for robustness attacks occuring after $\varepsilon - WCA$ attack depending on ε for c=4.

In this part of the experimental tests, we have tested the robustness and the inaudibility of the watermarking technique we used in the proposed fingerprinting system to embed our proposed two-level fingerprint in the audio stream extracted from the video content. Firstly, we show, that for different video contents, we have good robustness against Stirmark Audio attacks and MP3 compression attacks. Furthermore, the watermarking technique has shown good inaudibility results for the four tested videos. Finally we evaluate and check the security of our watermarking technique from a theoretical and practical points of view.

4.5 Conclusion

The majority of multimedia distribution applications through the internet has a serious problem which is the ability to trace back pirates in illegal media manipulations. In this chapter, we have proposed a suitable system for hierarchical fingerprint able to provide good detection rates despite the great number of users and colluders. Our proposed system is based on an improved two-level fingerprint and a weighted group selection. To validate the fingerprinting system, a set of experiments were realized to prove its tracing performance and its satisfying robustness against different types of attacks. The proposed system has also provided an efficient reduction of complexity by reducing the group search space in a first time and the users' search space in a second time.

CHAPTER 5

A proposal of optimizing fingerprinting code length based on a QR-code-based conversion

Contents

5.1 Introduction

Handling a great number of users and surviving different types of attacks present fundamental challenges of the majority fingerprinting systems in the tracing traitor field. In this work, the proposed technique consists in embedding a fingerprint, a QR code in the audio stream extracted from the media release. Using the QR-code provides several advantages as supporting a large amount of information in a compact format end damage resiliency. The aim is not only improve the two-stage tracing strategy by reducing the complexity computation, but also to enhance the security side of the proposed technique by the preprocessing treatment before generating the QR-code. In this chapter, we propose a QR-based watermarking technique as an improvement to the two-stage tracing strategy proposed in Chapter 4. The QR-code has the ability to support a large size of information in a reduced space which will require less embedding time and less complexity computation. Firstly, we detail the different steps of the improved tracing system. Then, we present the different experimental tests we carry out to validate the performance in terms of embedding time and robustness to different types of attacks.

5.2 The proposed tracing system with QR code-based audio watermarking technique

We continue with the improved two-stage-based tracing strategy which consists in a combination between the Boneh Shaw for group selection and the Tardos code to trace users into weighted groups. The two codes have the same length m. We remind that according to this strategy, each user belongs to a group and has consequently a group identifier concatenated to his personal identifier. To survive the problem of the code length and inspired by [Desoubeaux *et al.* 2012], we proposed to enlarge the alphabet size from a binary alphabet to a 4-length cardinality one. Although this technique affords to reduce the codeword length from $2 \times m$ to m, it still requires an important embedding time. The improved coding technique consists in converting Boneh Shaw-Tardos-H code to a QR-code. This format should provide not only less embedding time but also a good robustness to attacks due to its internal channel coding based on Reed Solomon code.

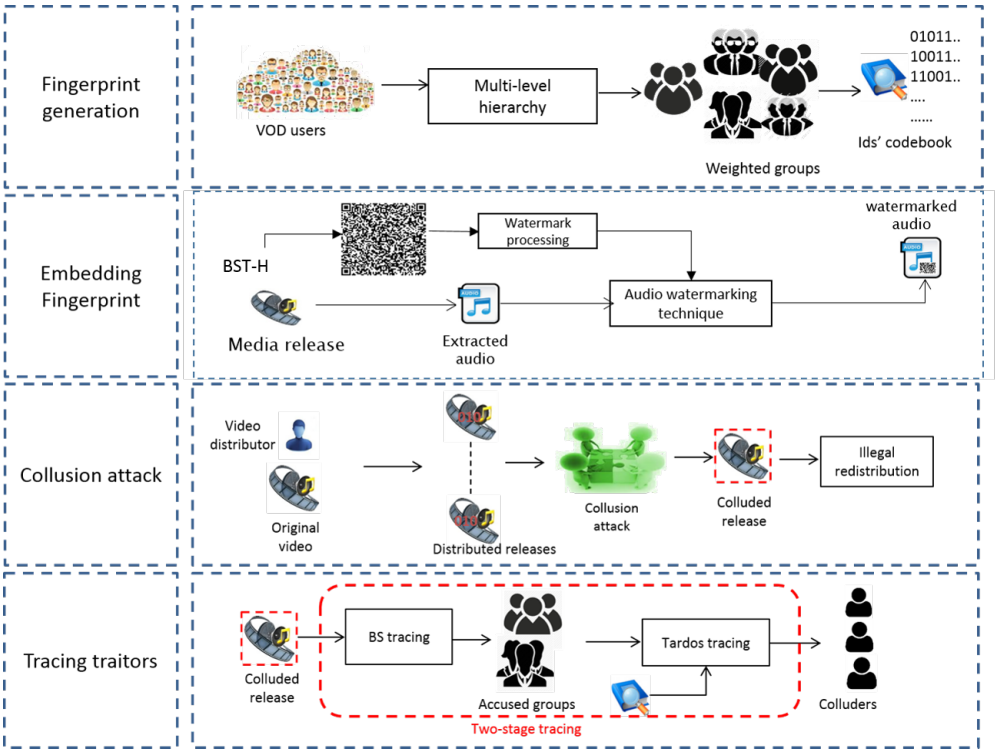


Figure 5.1: The proposed fingerprinting scheme with the QR-code-based watermarking technique.

5.2.1 The QR code: definition and features

In this chapter, we propose to improve the fingerprinting scheme by proposing a QR-code-based audio watermarking technique. Before detailing the different steps of the watermarking technique, we give some information about the QR code.

5.2.1.1 Definition of a QR code

The Quick Response code, known as QR code, is a code which was proposed by Denso Wave Incorporated in Japan in 1994 as a fingerprint adapted for all types of commercial products. It is a machine-readable optical label which contains in general information about the product. Compared to the 1D-barcode in which the information is encoded in one direction, the information in QR code is encoded horizontally and vertically and hence enables it to be readable in all directions.

The QR Code is a 2-D matrix code that transmits data information by the arrangement of its

black and white elements, called "QR-code modules" in columns and rows, respectively in horizontal and vertical directions. Each module of a QR Code, as shown in Figure 5.2 has a specificity and has a binary value which makes it machine-readable.

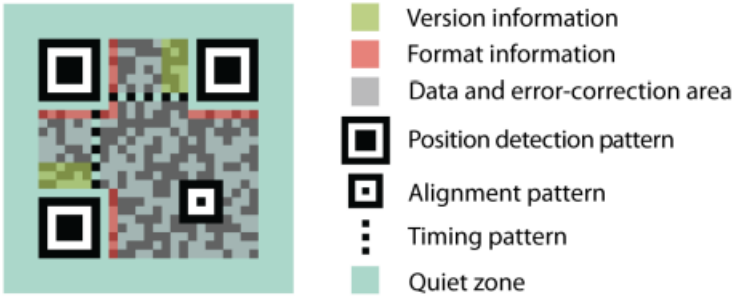


Figure 5.2: Specificity of QR-code modules.

5.2.1.2 QR code features

The QR code has more than one characteristics:

High capacity of storage: It is one of the main characteristics of a QR code. In fact, it presents the amount of quantity of information a QR code can encode. Compared to other codes, the QR code is able to support an important amount of information of different types: numeric, binary, alphanumeric, etc.

Figure 5.3: Example of QR-code converting 300 alphanumeric characters.

The small size The QR code has a compact size, it enables to store the same amount of data contained in a 1-D barcode in only one-tenth the space.

Error correcting capacity The QR Code is characterized by a powerful error-correction capability based on Reed-Solomon codes added to the original data. This allows a QR Code symbol to be read even if it is altered. In fact, a QR Code symbol can be decoded even if up to 30% of the data is altered. This recovering operation depends on the chosen error-correction level. Four levels of error correction, as reported in Table 5.1, are available for QR codes. The size of the QR code depends on the error correcting level. In fact, the higher is the level, the greater is the error correction.

Error-Correction Level	Amount of Correction
L	7%
M	15%
Q	25%
H	30%

Table 5.1: The four Error Correction levels of QR codes.

QR code versions: QR Codes have 40 different symbol versions ranging from 21 x 21 modules for version 1 to 177 x 177 modules for version 40. When moving from a version to another, 4 additional modules per side are added to the QR code and hence 16 additional modules. So, the higher is the version, the greater is the mount of stored information.

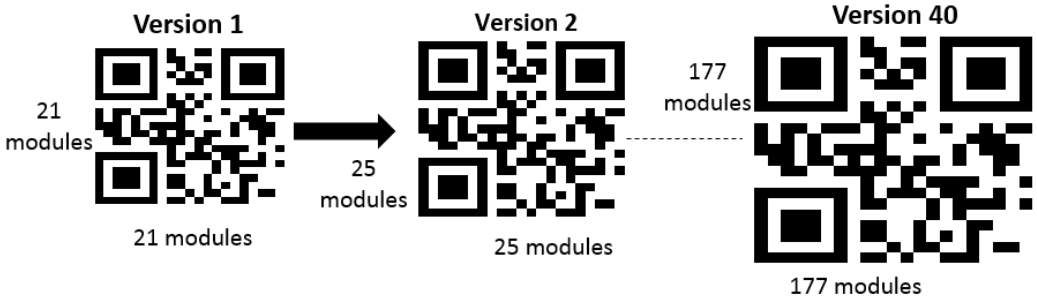


Figure 5.4: The different versions of QR code.

It is important to mention that the maximum amount of data that can be stored in QR code depends on type of characters, its version and the required error-correction level.

5.2.2 The watermark preprocessing step

The identifier, the Boneh Shaw-Tardos-H code, is the starting-point of the watermark preprocessing process. As depicted in Figure 5.5, this step includes:

- the generation of the QR-code: the Boneh Shaw-Tardos-H code is converted to a QR-code.
- The QR-code has four templates which do not contain any encoded information but are easily identifiable and are used essentially in the detection of the watermark as shown in Figure 5.6. Three among these templates are similar and called Template 1, the fourth one is called Template 2 as shown in Figure 5.6. The QR-code, then, is divided to k overlapped blocks of $(\frac{m}{k}+1)*(\frac{m}{k}+1)$ size. In this step, k depends on two requirements: the length of the embedded code and the payload of the watermarking technique.
- The scrambling process: in this level, the Arnold Transform is applied to the original blocks to obtain k blocks scrambled A_p times. To retrieve descrambled blocks, the parameter A_p should be known by the video distributor.
  We can define the Arnold transform as follows:
  Let (X,Y) be pixel of square digital image, N is the height or width of the processed square image, the move to another pixel (X',Y') is obtained as follows:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (\mod N) \quad (5.1)$$

  Then, different blocks are then inserted sequentially in equal fragments of the audio stream extracted from the video.

In the next section, we detail the embedding step.

5.2.3 The watermark embedding step

The adopted watermarking technique is the same audio watermarking technique proposed in [Charfeddine *et al.* 2010]. The different steps of the QR-code-based embedding process are

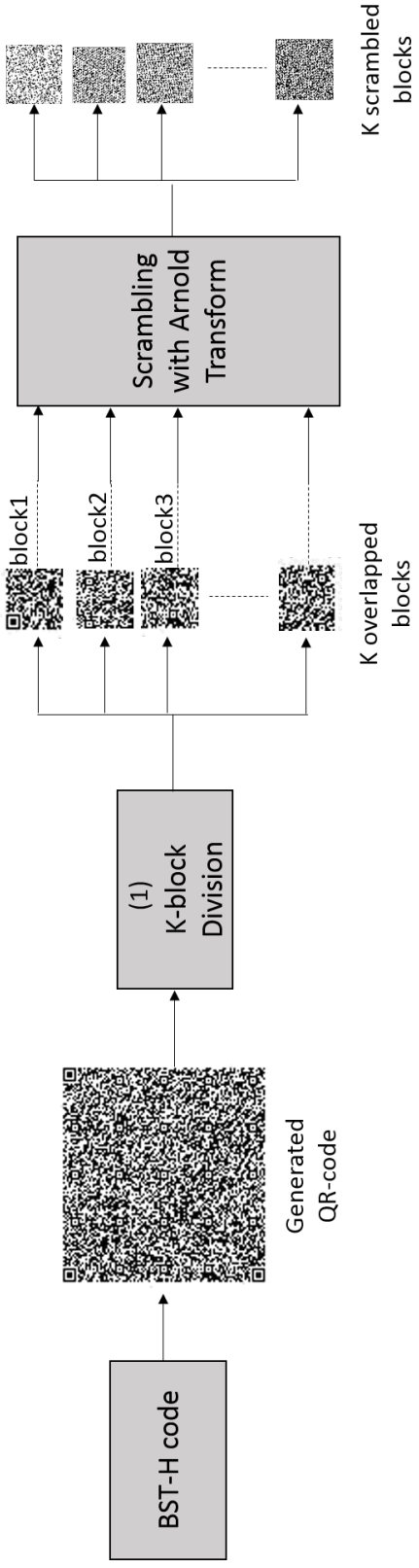


Figure 5.5: The watermark preprocessing step.

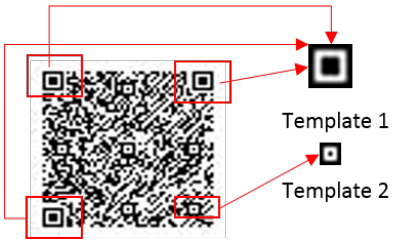


Figure 5.6: The four identifiable template in a QR-code

presented in Figure 5.7 .

In fact, the audio stream is extracted from the media copy and then divided into 512-size independent blocks. A DCT transform is then applied to each block. In the same time, after the watermark pre-processing step, we obtain k scrambled blocks $Sc_{i,i\in\{1..k\}}$ having a size of pq. After that, pq randomly selected indexes are chosen from the original blocks. The watermark symbols are embedded in Middle Frequencies MF band of each block, a choice which was proven by a study made in [Charfeddine *et al.* 2010]. The Back Propagation Neural Network, BPNN, is trained and simulated to choose the best embedding position. The watermarked audio is obtained by applying the IDCT transform.

It is relevant to indicate that each scrambled block is embedded in a fragment $f_{i,i\in\{1..k\}}$ of the audio stream. The length of the audio required to embed all the watermark is thus equal to $k \times f_k$. We add the A_p: the parameter of the Arnold transform which is saved as a secret watermarking key to retrieve blocks of the QR-code before the scrambling operation.

5.2.4 The watermark detection step

The adopted watermarking technique is a blind watermarking scheme and so to retrieve the watermark Sc_i', we need only the pq MF positions, the random indexes, the A_p and the coordinates of Template 1. This step is presented in Figure 5.8, it consists in the inverse of the embedding one.

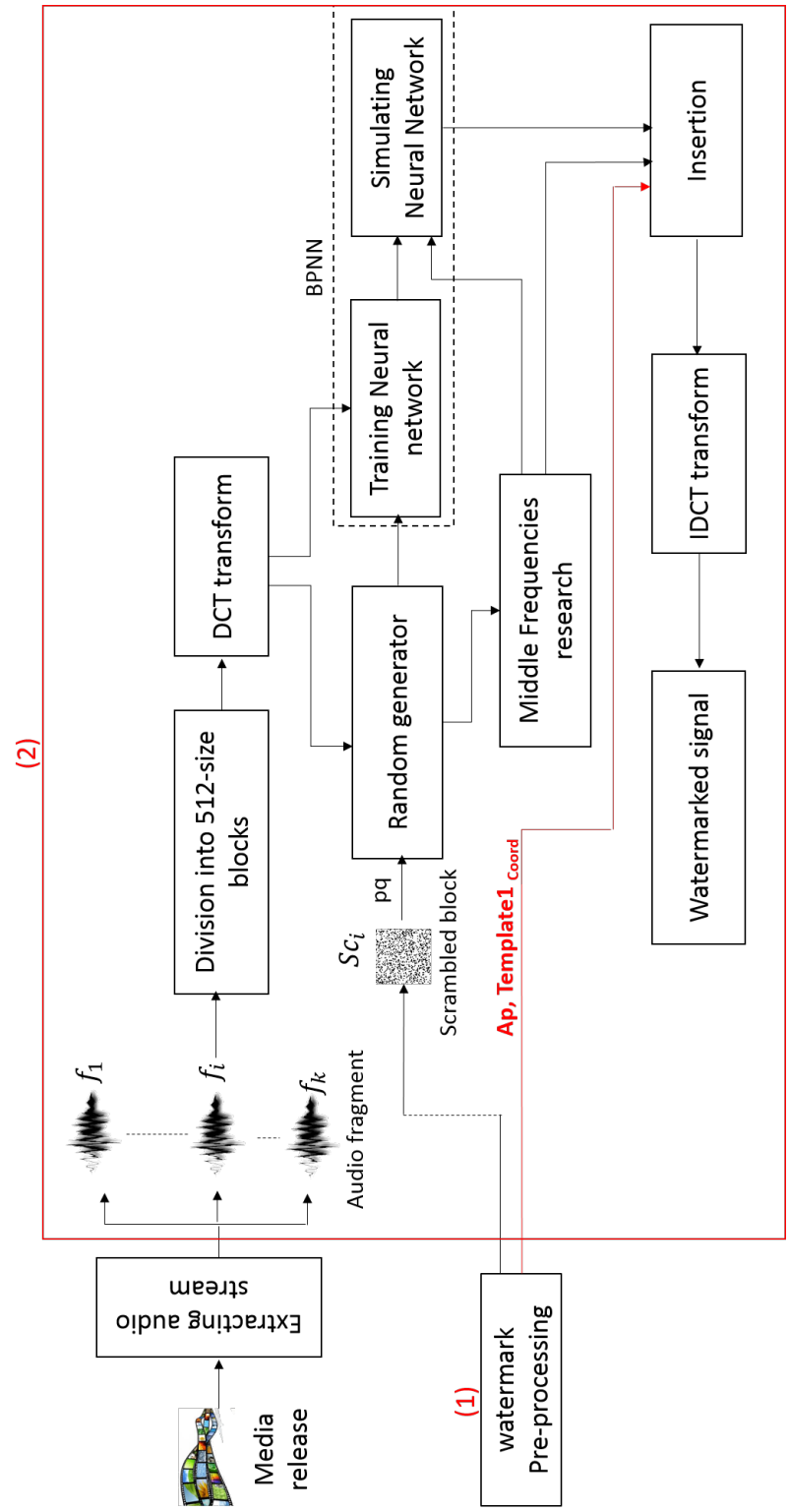


Figure 5.7: The watermark embedding step

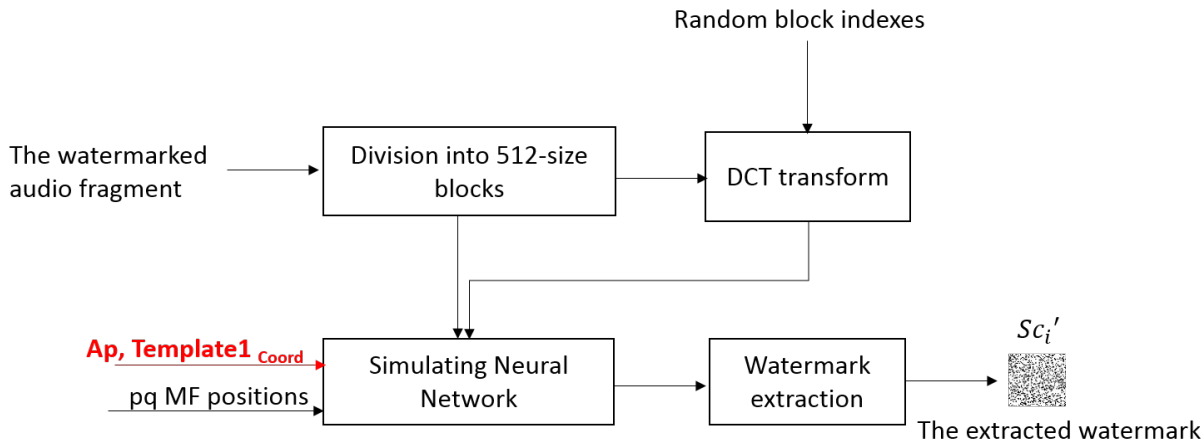


Figure 5.8: The watermark detection step

5.2.5 The descrambling step

In this step, we proceed by the inverse of the Arnold transform to each detected block Sc_i'. Knowing the number of iterations A_p of the scrambling operation, we obtain k descrambled blocks DSc_i'.

5.2.6 The matching step

Matching the different descrambled blocks $DSc_{i,i\in\{1..k\}}$' to retrieve the detected QR-code should be difficult when the digital content is attacked, which can damage the embedded watermark. We propose in this step, as presented in Figure 5.10, to use two measures, the *NCC*, Normalized Cross Correlation, to retrieve the first detected blocks DSc_1' and then the *SAD*, the Sum of Absolute Differences, to gather the remaining overlapped blocks. The proposed matching algorithm is summarized as follows:

1. Read the image of Template 1 and the image of the descrambled block $DSc_{i,i\in\{1..k\}}$'
2. Iterate Template 2 over DSc_i' and compute the *NCC* matrix
3. The coordinates where the value of *NCC* is the largest correspond to the best similarity and so to the required position as shown in Figure 5.9.

In this case, f is the descrambled block $DSc_{i,i\in\{1..k\}}$' and g is the template image in the original QR-code.

4. Given that the coordinates of Template 1 are known, we compare them to the saved ones and according to that we obtain the first block in the detected watermark.

5. Iterate DSc_1' over the remaining $DSc_{i,i\in\{2..k\}}$' and compute the *SAD*. The block which is overlapped to the first block has the coordinates where the lowest *SAD* value is obtained. Save the block position coordinates and iterate 4 for the other $DSc_{i,i\in\{3..k\}}$'.

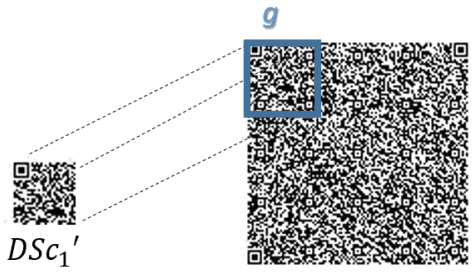


Figure 5.9: Computing NCC for block matching.

6. According to the obtained coordinates, combine the detected blocks and construct the retrieved watermark.
   After detecting the QR watermark, it is possible to check the validity of the watermarking technique.

Both NCC and SAD are measures of similarity:

- The **NCC** is defined by:

$$NCC(f,g) = C_{fg}(\widehat{f},\widehat{g}). \tag{5.2}$$

  where

$$\widehat{f} = \frac{f-\widehat{f}}{\sqrt{\Sigma(f-\overline{f})^2}}, \widehat{g} = \frac{g-\widehat{g}}{\sqrt{\Sigma(g-\overline{g})^2}}.$$

- The Sum of Absolute Differences, **SAD**, is calculated as shown in Eq.5.3 by subtracting or removing pixels within a square neighborhood between the reference image I_1 and

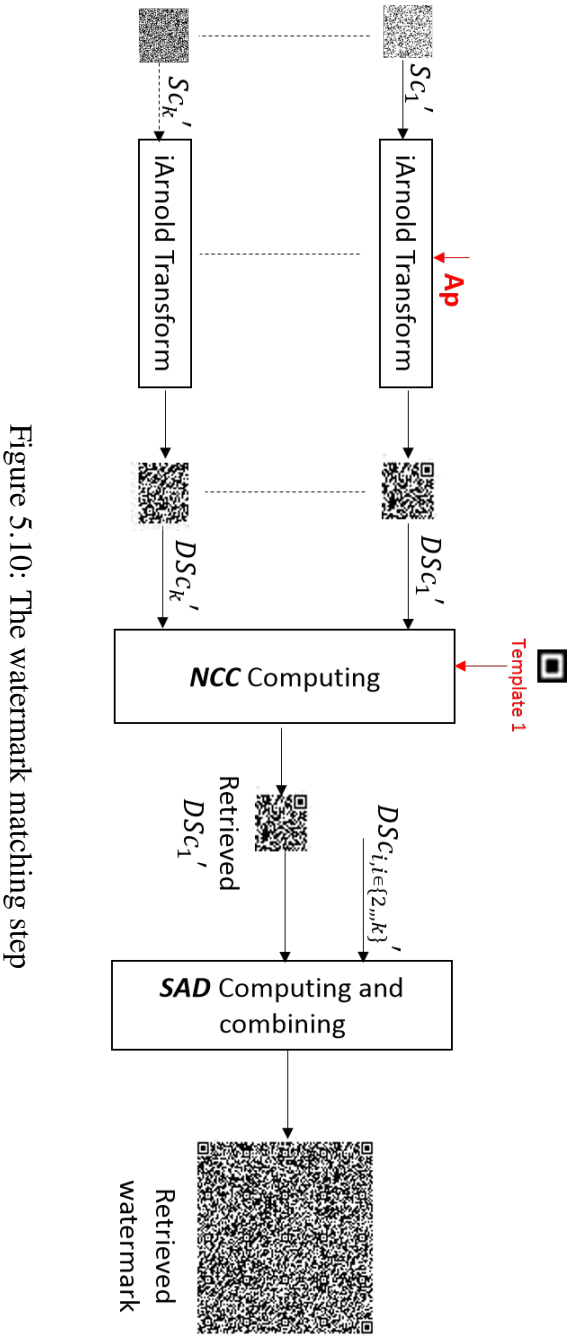


Figure 5.10: The watermark matching step

the target image I_2 followed by the aggregation of absolute differences within the square window W.

$$SAD = \sum_{(i,j)\in W} |I_1(i,j) - I_2(x+i, y+j)|. \tag{5.3}$$

5.3 Experiments and Evaluation

In this section, we evaluate the proposed technique according to different criteria: the QR-code capacity which implies a high watermarking payload and less embedding time when varying the code length m and the robustness to the almost collusion attacks and to other attacks tied to the audio signal. To have good watermarking robustness and inaudibility results, we choose the same parameters required by the audio watermarking technique proposed in [Charfeddine *et al.* 2010]. All the extracted audio signals are divided into fragments of 20 s and are sampled in 44.1 KHz with 16 bits/sample. In each audio fragment, we insert a scrambled block which dimension should not exceed 32×32 bits. To support the various code length values, we choose the suitable QR-code version to guarantee a good embedding capacity.

5.3.1 Embedding time

In Table 5.2, we show how the proposed technique takes the least embedding time compared to respectively three fingerprinting systems [Tardos 2003, Kuribayashi *et al.* 2008, Chaabane *et al.* 2016c].

In this case, the code length is around $m = 2048$ and the number of users is equal to 10^4. We choose the *version* 20 and the level H as error correcting level. This version offers to convert 2061 numeric symbols into a 97×97 QR-code dimension. In Figure 5.11, we vary the embedded code length m and we compare the embedding time required to insert a fingerprint in a video for the original Boneh Shaw- Tardos-H proposed in [Chaabane *et al.* 2016c] and the QR-based improved technique. While the embedding time for the proposed technique is around few minutes, it is over 25 min for m $> 10^4$ which can not be allowed for some video sequences in multimedia distribution context. The QR-code version depends on embedded fingerprint code length and hence, we show in this part of experiments that the required number of audio fragments to embed the watermark is tightly tied to its code length and hence to its version

Tracing strategy	**required clip duration**
Tardos [Tardos 2003]	$\approx$ 20 min
Tardos-Tardos [Kuribayashi *et al.* 2008]	$\approx$ 40 min
Boneh Shaw-Tardos-H [Chaabane *et al.* 2016c]	$\approx$ 21 min
The proposed technique	5 min 20s

Table 5.2: Comparison of existing tracing strategies in term of embedding time required to insert a code length $m = 2048$.

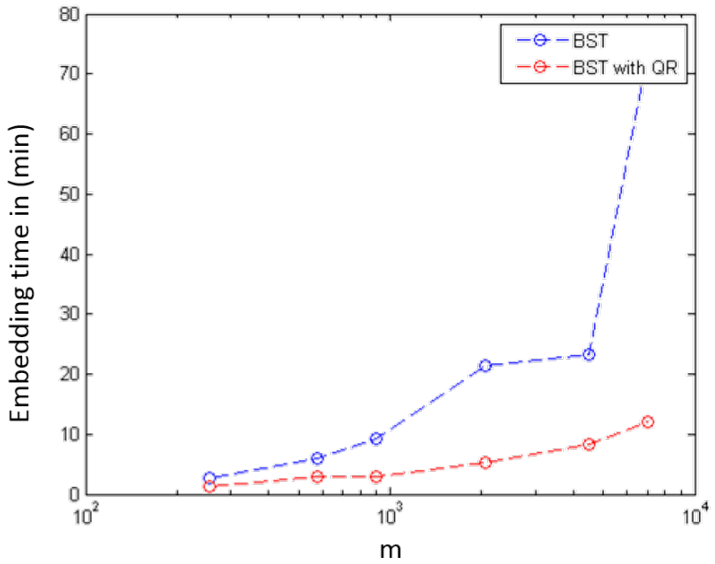


Figure 5.11: The embedding time for the proposed technique vs [Desoubeaux *et al.* 2012]

and error correction level. In Table 5.3 we show the associated QR-code version used for each length of m represented in Figure 5.12.

code length	**64**	**128**	**256**	**512**	**1024**
QR-code version	2	4	7	10	16
Error correction level	L	M	M	M	M
QR-code size	25×25	33×33	45×45	57×57	81×81

Table 5.3: The associated QR code characteristics used for experimental values of m.

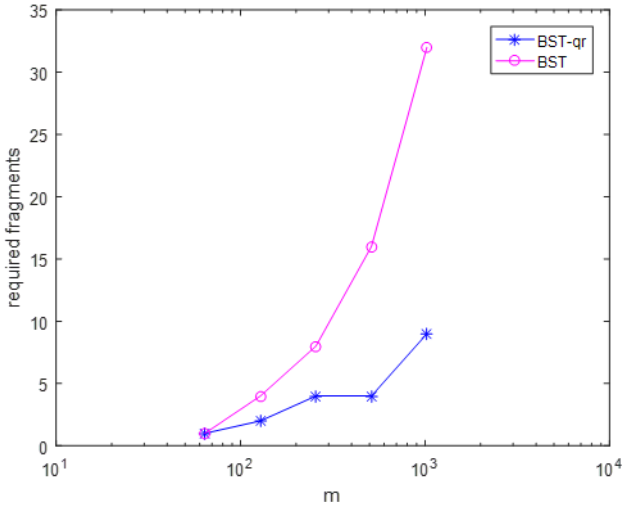


Figure 5.12: The required number of audio fragment to embed fingerprint of $m$for the proposed technique vs [Desoubeaux *et al.* 2012]

5.3.2 Tracing results

The fingerprinting technique is a group-based tracing strategy, Boneh Shaw-Tardos-H code. This technique has reflected, due to a group selection, a high tracing rates. We choose a Tardos length m equal to 5330 symbols which is tied to a number of users equal to 10^7, ε_1 set to 10^-6 and c the number of colluders equal to 2. To embed the fingerprint, the suitable version of QR-code is version 35 with L as an error correcting level, this version affords to carry

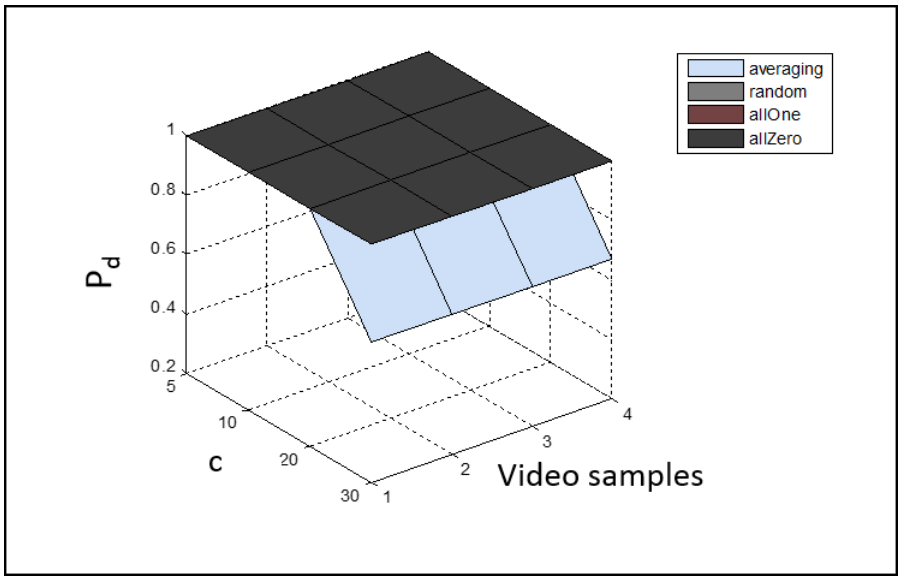


Figure 5.13: Robustness to different types of collusion attacks.

5529 numeric characters. In Figure 5.13, we show that the proposed technique affords a good detection probability P_d of more than 20 colluders against several collusion attacks.

5.3.3 Watermarking robustness and inaudibility

In Table 5.4, we calculate the SNR values for four different tested video samples. We notice that all these values exceed 20 dB and so guarantees good inaudibility results.

To verify the watermarking robustness of the proposed technique, we are interested only by audio attacks and so we carry out a set of audio Stirmarkaudio attacks : noising, filtering and MP3 compression for different rates: 128, 96 and 64 Kbps.
As depicted in Figure 5.15, *NC* values are close to 1 and reveal good robustness results which is enhanced by the error correcting capability of the QR-code.

5.4 Conclusion

In this chapter, we have proposed a QR-based watermarking technique in tracing traitors' for multimedia distribution platforms. The construction of the fingerprint is based on the QR-code and thus provides the possibility to embed a great amount of information and so to trace

Video name (.avi)	SNR values(dB)
match	52.5765
sport	35.8021
film	49.03
prog$_t v$	28.45
film2	40.79
song1	48.73
song2	38.47
politic	36.42
prog2	36.4384

Table 5.4: Inaudibility results obtained with the different tested videos.

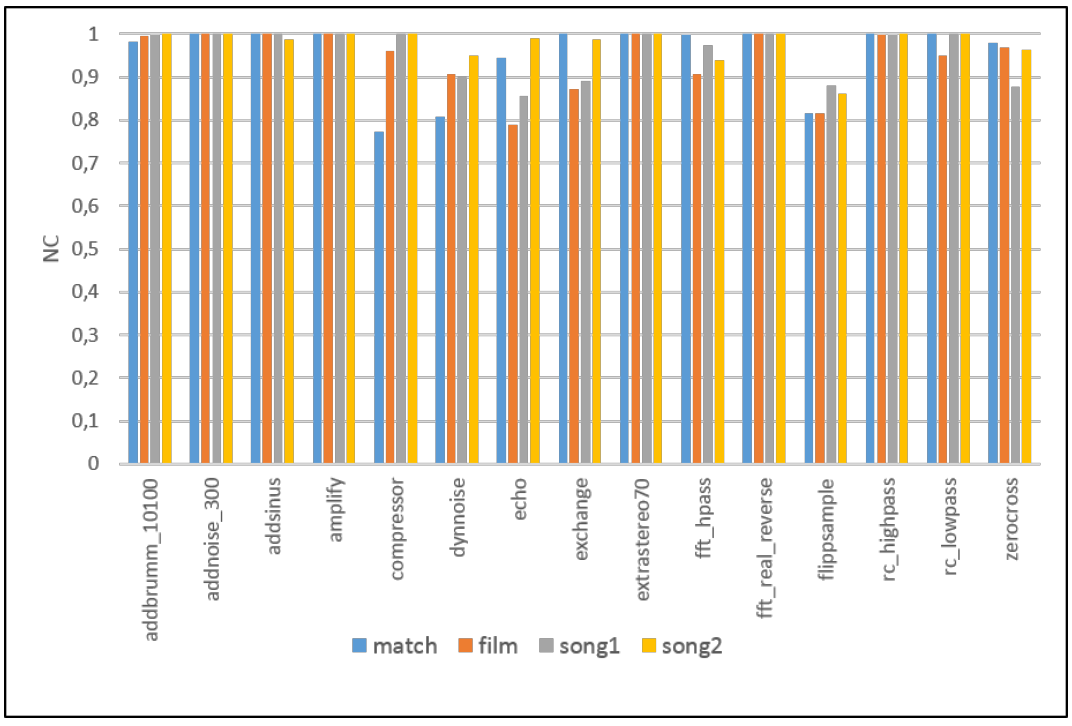


Figure 5.14: Robustness to several Stirmarkaudio attacks for four audio samples.

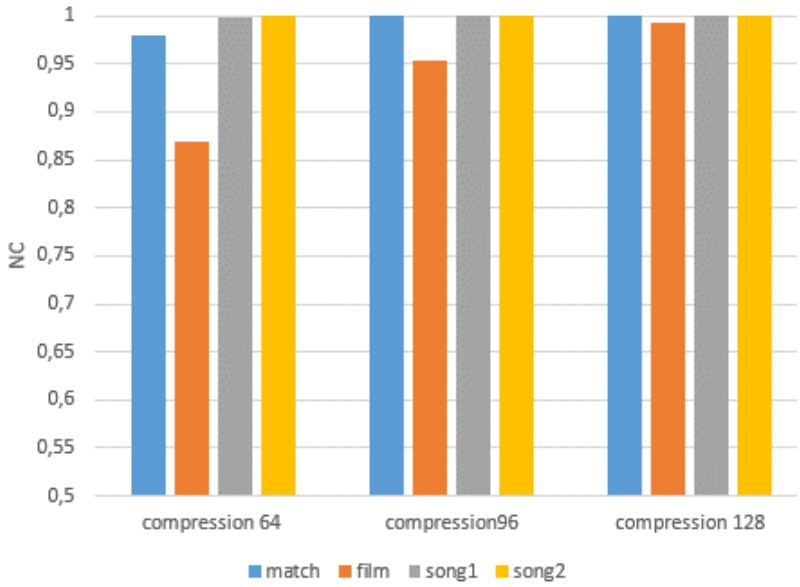


Figure 5.15: Robustness of four audio samples to different MP3 compression rates: 64, 96 and 128 Kbps.

a great number of users in a less embedding time without altering the media quality. The security side of the technique is also improved by the proposed fingerprint preprocessing step. To validate the fingerprinting system, a set of experiments were realized according to several criteria: embedding time, inaudibility and robustness to collusion and other audio attacks. This technique was proposed for static tracing schemes where the fingerprint is constructed in the distributor side and tracing colluders is made after diffusing all the copies.

CHAPTER 6

Towards a blind traitor tracing scheme for hierarchical fingerprint

Contents

6.1 Introduction

In multimedia distribution platforms, one of the main challenges is to provide an efficient and accurate tracing process despite the increasing of the collusion size. When considering

simple decoding of Tardos probabilistic codes as in [Tardos 2003] and [Skoric *et al.* 2007], it is undoubted that this type of decoders provides affordable computations in practice and allows to reach the bound of the shortest code length with an adjustment of the error probability. Nevertheless, their performance is considered suboptimal because of their agnostic behavior and conservative accusation regardless the collusion strategy. Some previous work proposed to estimate the collusion channel and thus to tune the Tardos accusation functions.

In this chapter, we explore and we evaluate the impacts of two existing algorithms in our group-based tracing scheme to deal with the computational costs and the invariance of the Tardos accusation performance.

The tracing scheme we propose benefits from a twofold accusation process. Indeed, in a first time, it is based on a two-level tracing strategy which consists in tracing guilty groups in a first level with the Boneh Shaw tracing code and in retrieving at least one colluder in accused groups with Tardos code in the second level. This strategy has reduced efficiently the decoding process of the Tardos code.

The main shift we propose in the second level is to apply the estimation algorithm to be tightly tied to collusion yielded by colluders and hence to choose the more efficient algorithm to find the more accurate Tardos accusation functions.

The performance of the resulting tracing scheme is evaluated according to different criteria and promising results have been achieved when compared to the existing tracing schemes proposed in the literature.

6.2 Applying an estimation algorithm in traitor tracing scheme

When considering the invariance of the Tardos code, more than one decoder was proposed in the litertaure. One key work was based on the Expectation Maximization algorithm [Charpentier *et al.* 2009], [Furon & Pérez-Freire 2009a].

In a first part of this work, we will study the efficient impact of introducing the EM iterative decoding in two-level tracing scheme on the accusation process and we will compare the tracing performance to those of the Tardos code and the iterative EM with the simple

Tardos decoder [Furon & Pérez-Freire 2009a]. Authors in [Pérez-Freire & Furon 2009] and [Desoubeaux *et al.* 2013] agree upon the fact that despite the outperformance of the iterative EM-based tracing algorithm in terms of accusation rates, it remains not suitable for scenarios involving a great number of users because of its computational complexity (about *O(nmc)* for each iteration).

According to that, in the second part of this work, we turn our attention to an other type of blind decoder, the Maximum a Posteriori-based decoder which principle is to use a Maximum a Posteriori decision rule to trace colluders under the Marking Assumption without any estimation to the collusion channel [Desoubeaux *et al.* 2013], [Kuribayashi 2014]. The latter choice can be proved by the fact that for this thesis work, the main challenge is to address the problem of traitor tracing in applications involving large audience.

In Figure 6.1, we show the modified two-level tracing scheme where the estimation step is added after the group selection by BS and during the Tardos accusation process.

We aim in this work at analyzing the impact of using an estimation-based step in a two-level tracing scheme which was not studied previously in the literature.

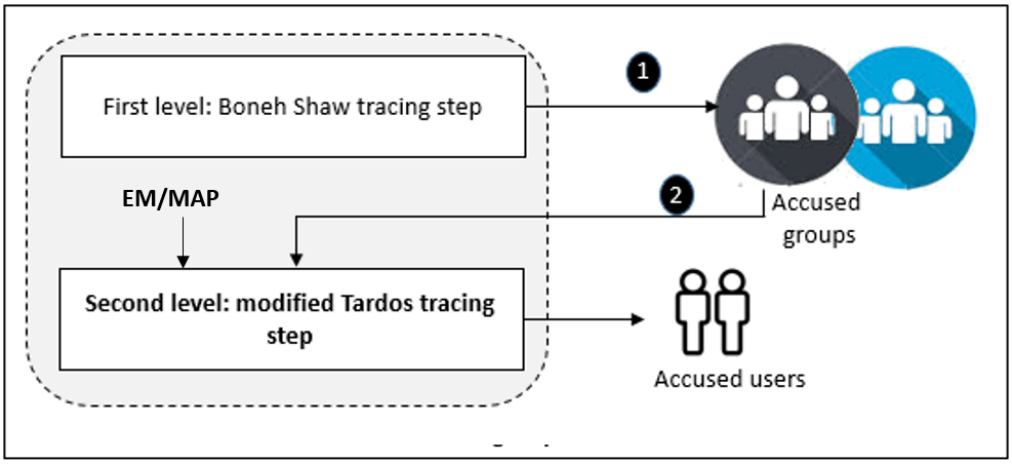


Figure 6.1: Applying the estimation algorithm in the two-level tracing scheme

6.3 The Expectation Maximization decoding strategy applied in tracing scheme

The EM-based estimation algorithm of the Symmetric version of Tardos accusation functions [Skoric *et al.* 2007] was proposed firstly by [Charpentier *et al.* 2009], it consists in estimating iteratively the collusion channel and adjusting according to that the accusation functions to have an accurate detection performance. In this estimation, it is assumed that the same collusion strategy is applied in all positions of the colluders' identifiers. Let define the collusion strategy Θ by the set of probabilities:

$$\left\{ \mathbb{P}(Y_i = 1 \mid \sum_i = \sigma_i), \sigma_i = 0..c \right\}_{i=1..m} \tag{6.1}$$

with $\sum_i = \sum_{j \in c} X_{ji}$ the random variable presenting the colluders' identifiers having 1's bits in position i. The EM-based estimation as depicted in Figure 6.2 consists of four steps:

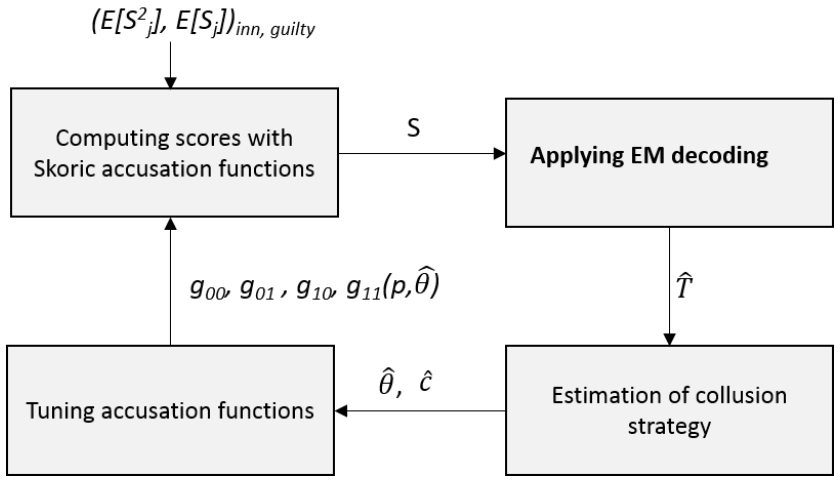


Figure 6.2: The principle of the EM estimation algorithm

- The initialization step: users' scores are computed according to the Symmetric version of Tardos accusation functions as proposed by Skoric in [Skoric *et al.* 2007]. The output of this step is a set S of users' scores including guilty users' scores and innocent users' ones.

- The EM decoding step is applied to the S to separate between the distributions of innocent and guilty users' scores. In this step, it is assumed that users' scores are distributed

according to Gaussian distributions as respectively: for innocent users, $S_j \sim \mathcal{N}(0,m)$ and for guilty ones $S_j \sim \mathcal{N}(2\pi^{-1}mc^{-1}, m(1-4\pi^{-2}c^{-2}))$. Hence, the inputs of this step are respectively the first and second moment $\mathbb{E}(S_j)$ and $\mathbb{E}(S_j^2)$ of innocent and guilty users' distributions. The vector $\widehat{T}$ presents the probability of S_j to be a colluder score.

- The collusion strategy estimation: Given the resulting vector $\widehat{T}$, the estimation of the collusion size $\widehat{c}$ is computed with $\widehat{c} = \left\lceil \sum_{j=0}^{n} \widehat{T}_j \right\rceil$. When considered the codewords of users having the highest probabilities $\widehat{T}_j$, the estimation of the collusion strategy $\widehat{\Theta}$ is made. In fact, for each position i, the σ_i is computed as the sum of the codewords X_{ji} in the estimated vector $\widehat{T}$. Then, for each possibility of σ_i, a mean of corresponding Y is computed.

- Tuning the accusation functions step: according to [Charpentier *et al.* 2009], the kullback-Leibler distance between distributions of colluders and innocents scores (respectively $\mathcal{N}_{coll}$ and $\mathcal{N}_{inn}$) is defined by:

$$D_{KL}(\mathcal{N}_{coll}, \mathcal{N}_{inn}) = \frac{1}{2}(m\tilde{\mu_{coll}}^2 - \log(\tilde{v_{coll}}) + \tilde{v_{coll}} - 1) \tag{6.2}$$

  with $\tilde{\mu_{inn}} = 0$ and $\tilde{v_{inn}} = 1$ as mentioned in [Furon *et al.* 2008]. Authors in [Charpentier *et al.* 2009] have performed several calculations to maximize the kullback-Leibler distance to separate better between the innocents' and colluders' distributions and thus to tune the different accusation functions: $g_{00}, g_{01}, g_{10}, g_{11}$.

Now, in case of our two-level tracing strategy, we focus firstly on the EM-based estimation of the Tardos accusation functions which were detailed above, and we will apply them in our two-level tracing scheme. As depicted in Figure 6.1, applying the two-level accusation process should ameliorate the tracing performance of a simple Tardos decoder. Furthermore, selecting groups with the Boneh Shaw code in a first time guarantees a detection of at least one group. Hence, applying the EM algorithm in the second Tardos accusation level only on the selected group should not only reduce the research space and the time efficiency but it provides also more accuracy to the accusation process. The MAP-based decoder has been also proposed as a blind decoder to address the invariant behavior of the Tardos decoder

6.4 The MAP-based blind decoder for our two-level tracing scheme

The first work proposing the MAP algorithm to improve Tardos accusation step was [Desoubeaux *et al.* 2013].

Before detailing this probabilistic model, it is important to remind some notations which will be mentioned throughout this part as follows: X is a $\{0,1\}^{m\times n}$ matrix representing users' codewords of length m where X_j is the embedded information in the sold release related to the user j. According to this, let s a $\{0,1\}^n$ vector which assigns $s_j = 1$ to each user selected as a colluder, and otherwise $s_j = 0$.

Typically, it is assumed that the i^{th} symbol in the suspicious release depends only on the 1s' number in each position i in colluders' codewords. Obviously, a vector t $\in \{0,\ldots,c\}^m$ is constructed, representing the 1s' number encountered in all colluders' codewords for each position i, $i \in \{1,\ldots,m\}$. Thus, $t = Xs$.

The probability of generating y from colluders' codewords consists essentially in a conditional probability as follows:

$$\mathbb{P}(\mathbf{y}|\mathbf{t, G}) = \prod_i \mathbb{P}(y_i|t_i, \mathbf{G}), \tag{6.3}$$

where $\mathbf{G}$ denotes the collusion channel. It is assumed that $\mathbf{G}$, is a $m \times c$ matrix. The only requirement for g_{ik} elements is to respect the Marking Assumption [Skoric *et al.* 2007], for $k \in \{0,c\}$ by following: $\forall i, g_{i0} = 1 - g_{ic} = 0$. Computing a score related to each user is based on the Neyman-Pearson theorem [Desoubeaux *et al.* 2013], which states that to decide about his pertaining in the collusion trial, the optimal score σ_j^{NP} is as follows:

$$\sigma_j^{NP} = \frac{\mathbb{P}(y \mid x_j, s_j = 1, \mathbf{G}, \mathbf{p}, c)}{\mathbb{P}(y \mid x_j, s_j = 0, \mathbf{G}, \mathbf{p}, c)}. \tag{6.4}$$

When addressing the problem of missing some information about the collusion channel, it is judicious to estimate the vector of accusation s from the codeword y. Furthermore, in practice, it was agreed upon that selecting at least one colluder is sufficient [Chor *et al.* 1994].

We are interested in the blind decoder proposed by [Desoubeaux *et al.* 2013], the MAP-based decoder, which was proposed to provide a tradeoff between two crucial requirements in a tracing scheme: the accusation accuracy and the computational costs.

The MAP-based decoder is based on defining two non-informative priors, $\mathbb{P}(G)$ and $\mathbb{P}(c)$, which are respectively tied to the strategy **G** and the collusion size **c**. To derive the decoder, and according to Bayesian statistics, authors exploit a joint probability in order to marginalize the unknown quantities as follows:

$$\mathbb{P}(\mathbf{y, t, X, p, s}) = \sum_{c}(\int \mathbb{P}(\mathbf{y, t, X, p, s}|c, \mathbf{G})\mathbb{P}(G)d\mathbf{G})\mathbb{P}(c). \tag{6.5}$$

From Eq.6.5, the likelihood ratio σ_j^{MAP}, defined previously in Eq.6.4 can be computed as follows:

$$\sigma_j^{MAP} = \frac{\mathbb{P}(s_j = 1|\mathbf{y}, \mathbf{x}_j, \mathbf{p})}{\mathbb{P}(s_j = 0|\mathbf{y}, \mathbf{x}_j, \mathbf{p})} = \frac{\mathbb{P}(\mathbf{y}, \mathbf{x}_j, \mathbf{p}, s_j = 1)}{\mathbb{P}(\mathbf{y}, \mathbf{x}_j, \mathbf{p}, s_j = 0)} \tag{6.6}$$

Let c, the discrete random variable describing the collusion size varying from 1 to *cmax* and which distribution suits a uniform law so that:

$$\mathbb{P}(c) = \frac{1}{c_{max}} \tag{6.7}$$

In [Desoubeaux *et al.* 2013], the authors have performed several calculations which have lead to a calculable result of the likelihood ratio.

Figure 6.3 shows that selecting groups with the Boneh Shaw code is made in a first time, and then the MAP algorithm is applied in the second Tardos accusation level only on the selected group and the output is σ_j^{MAP}.

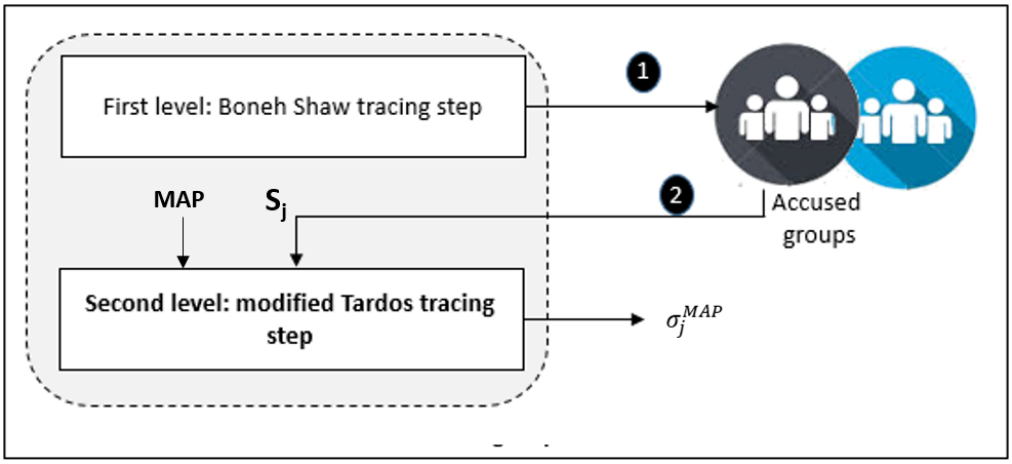


Figure 6.3: Details of the two-layered MAP-based tracing scheme

6.5 Experimental results

Despite the fact that the EM-based iterative estimation can not be suitable for the multimedia distribution platform, we show some experimental results to focus on its impact on tracing performance of our two level scheme in comparison to the other Tardos decoders. Then, we exhibit the different tests performed with the MAP decoder.

6.5.1 The evaluation of the EM-based two-layered tracing scheme

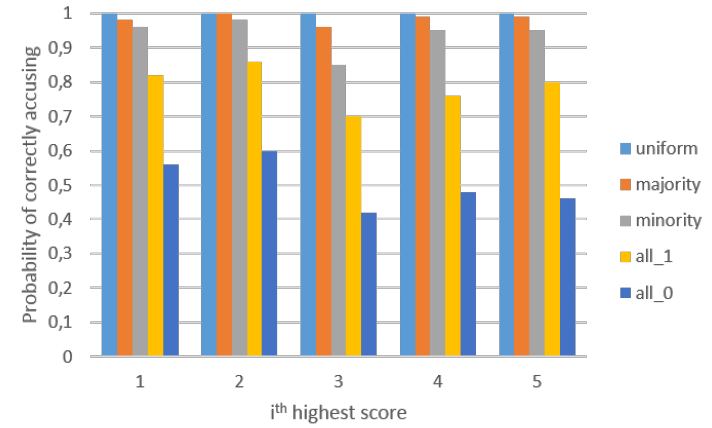
To test the traceability of the EM-based decoder in the proposed two-level tracing scheme, we assess it according to different criteria: the time detection and the detection rates for different collusion attacks. We compare the proposed two-level tracing scheme, called EM-BST, to two other schemes: the scheme with EM estimation for the simple Tardos decoder, EM-T [Charpentier *et al.* 2009], and the scheme using the Symmetric version of Tardos code [Skoric *et al.* 2007], called T in the sequel. The set of tested attacks are defined as follows:

- The uniform attack:
$$\mathbb{P}(Y = 1 \mid \sum = \sigma) = \frac{\sigma}{c}. \tag{6.8}$$
- The Majority attack:
$$\mathbb{P}(Y = 1 \mid \sum = \sigma) = 1 \quad \text{if} \quad \sigma > \frac{c}{2}, \quad 0 \quad else. \tag{6.9}$$
- The Minority attack:
$$\mathbb{P}(Y = 1 \mid \sum = \sigma) = 1 \quad \text{if} \quad \sigma > \frac{c}{2}, \quad 1 \quad else. \tag{6.10}$$
- The All-One attack:
$$\mathbb{P}(Y = 1 \mid \sum = \sigma) = 1 \quad \text{if} \quad \sigma \neq 0. \tag{6.11}$$
- The All-Zero attack:
$$\mathbb{P}(Y = 1 \mid \sum = \sigma) = 0 \quad \text{if} \quad \sigma \neq c. \tag{6.12}$$

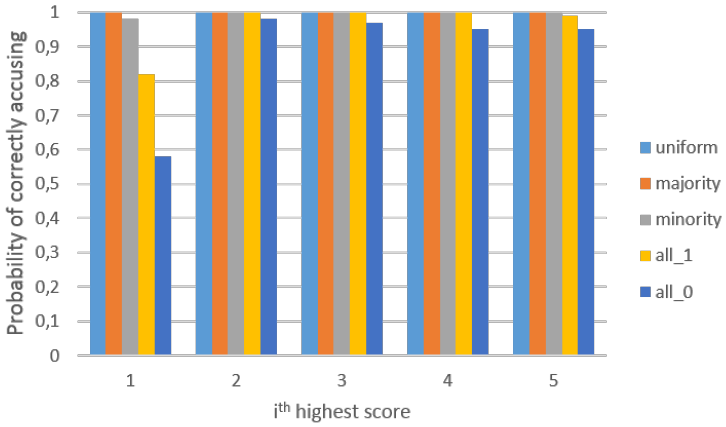
6.5.1.1 Stability against different collusion strategies

In this part, we evaluate the detection performance of the three tracing schemes: T, EM-T and EM-BST for different collusion strategies and for two different collusion size as shown in Figure 6.4 and Figure 6.5. The aim is to check the efficiency of the accusation functions when the collusion size c and the strategy θ are unknown and are estimated with the EM algorithm. Figures 6.4(a) and 6.5(a) show the Tardos invariant behavior regardless the collusion strategy θ. The detection performance of the Symmetric version of the Tardos is quite similar for the different collusion attacks but it decreases significantly when the collusion size increases, which can be explained by the code length which is kept for c =8 and seems to be insufficient for Tardos to provide good detection results [Tardos 2003].
Nevertheless, according to Figures 6.4(b) and(c) and Figure 6.5(b)and (c), for the proposed group-based EM-BST, compared to the the EM-T with the Tardos simple decoder, the detection performance is improved for the two collusion sizes. It is undoubted that the estimation is closely tied to the collusion attack and hence some collusion attacks have deeper impact on the detection (are harder to be estimated) than others. When considered the detection results given with the proposed EM-BST, the improvement can be explained by the impact of applying a first accusation level on groups and thus reducing the search space in the EM estimation.

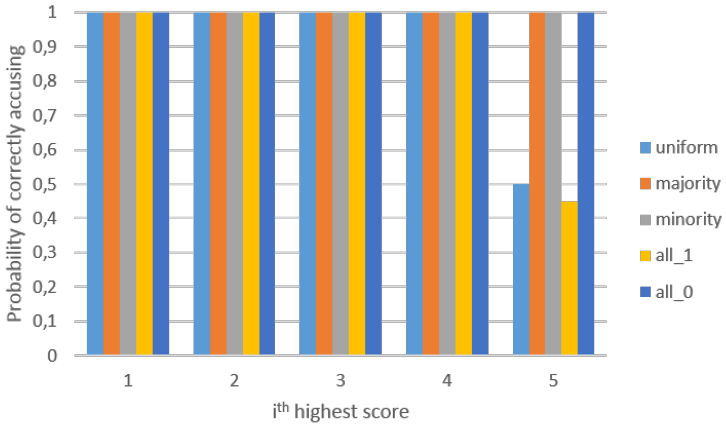
In the context of our thesis work, the EM iterative algorithm can not be adopted with regard to computational complexity which makes it intractable when considering a huge number of users. Moreover, the false alarm probability is not bounded for any size collusion. In addition, for computational simplification target, it is assumed that the collusion strategy is constant for all positions i. Hence, the issue from this experimental assessment is not to select the EM decoder for our tracing scheme but mainly to focus on the impact of estimating collusion channel on Tardos behavior. In the same context, using the MAP-based blind decoder should be relevant in our two-level tracing scheme because of its threshold-based accusation and the general assumption on the collusion channel in the sense that all possible strategies are considered at each position i.



(a) Tardos

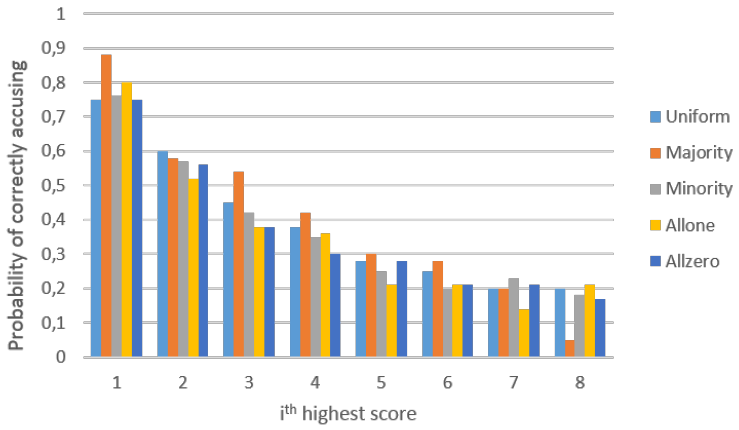


(b) EM-Tardos

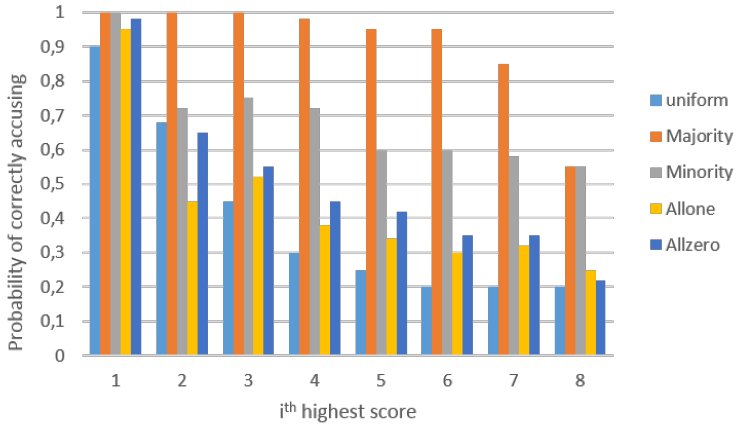


(c) EM-BST

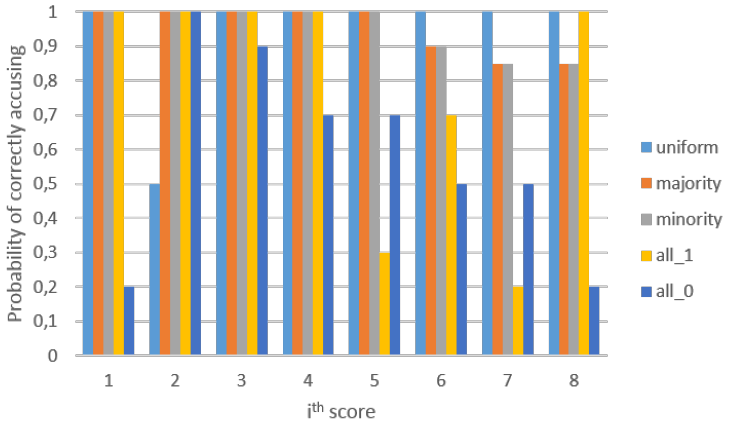
Figure 6.4: Probability of correctly accusing the ith highest score for the proposed group based scheme EM-BST vs T and EM-T with i $\in$ 1..5, m = 1000 and c=5.



(a) Tardos



(b) EM-Tardos



(c) EM-BST

Figure 6.5: Probability of correctly accusing the i^{th} highest score for the proposed group based scheme EM-BST vs T and EM-T with i $\in$ 1..8, m = 1000 and c=8.

6.5.2 The evaluation of the MAP-based two-layered tracing scheme

To show the efficiency of the proposed two-layered strategy accusation, we assess it in terms of detection rates for different collusion attacks and different code lengths. Another important criteria is the time estimation which computes the time required by the detector to estimate accurately the collusion channel. The aim of the proposed MAP-based detector for hierarchical fingerprints is to find a trade off between the detection runtime and the accusation rate, by taking less time for a good accusation rates. To estimate the decoder performance, we used the Monte Carlo Estimation algorithm proposed by [Furon *et al.* 2009], and we compared the proposed detector, the MAP-based detector for hierarchical fingerprints, to the agnostic decoder, and also to to the non-layered detector, called respectively MAP-H, T and MAP in this part. Each run of Monte Carlo estimation consists in 10^3 trials.

We remind that the Monte Carlo simulation consists in generating a set of N new codewords $\left\{\tilde{x_j}\right\}_{j=1}^{N}$, and a codeword y. One estimation consists in $\tilde{\mathbb{P}}(S > z) = N^{-1}\left|\{j \mid G(x_j, p, \tilde{y}) > z\right|$.

6.5.2.1 Results of stability against different collusion strategies

In this part, we used ROC curves to compare the Symmetric Tardos decoder, the MAP-based decoder in hierarchical context and the MAP decoder proposed by [Desoubeaux *et al.* 2013] in terms of different criteria. We evaluate the stability of the three decoders against three collusion attacks: minority, coin flip and WCA attacks. The 'maph', 'map' and 'tardos' state respectively for the three decoders. The 'min', 'cf' and 'wca' state respectively for the Minority vote attack, the coin flip' and 'WCA' ones. The fingerprint code is of length m=1000, the maximum number *cmax* of colluders is equal to 10 and the number of colluders is c=6. In Figure 6.6, compared to the two other detectors, the Tardos decoder varies a little bit for the two strategies. Although, it provides a good stability whatever is the collusion attack, it loses in detection accuracy especially for the minority attack, where we notice a large difference with the two other decoders. For all the strategies, we show that the proposed decoder improves the performance of the non-layered MAP-based decoder and provides a good performance and an accurate accusation rate. We notice that the WCA has the deepest effect on the performance of the three decoders, even for the proposed decoder.

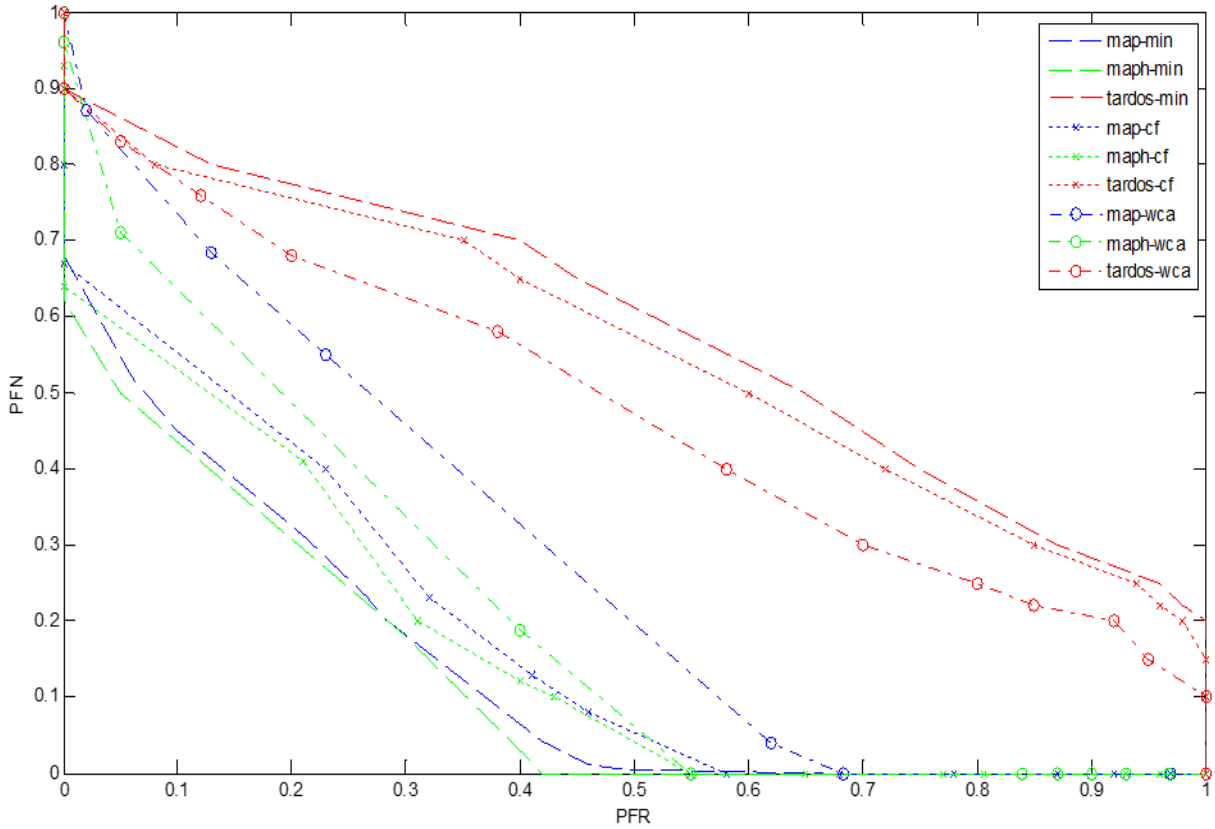


Figure 6.6: ROC curves of the MAP hierarchical decoder, the MAP decoder and the Tardos decoder $c = 6$ and $n = 1000$ users for different collusion strategies

6.5.2.2 Results when varying the code lengths

In Figure 6.7, we evaluate the performance of the MAP-based decoder for hierarchical fingerprints, the simple MAP decoder and the Tardos code when varying the code length (m=300 and m=800) against the WCA attack. We choose to evaluate this attack because it has been proven as the worst attack for the three decoders. We use the logarithmic scale for the probability of false negative and the probability of false alarm. We notice that the proposed decoder has the least detection errors compared to the other decoders, especially to the Tardos code. In addition, when increasing the code length, the performance of our decoder is increasingly outstanding.

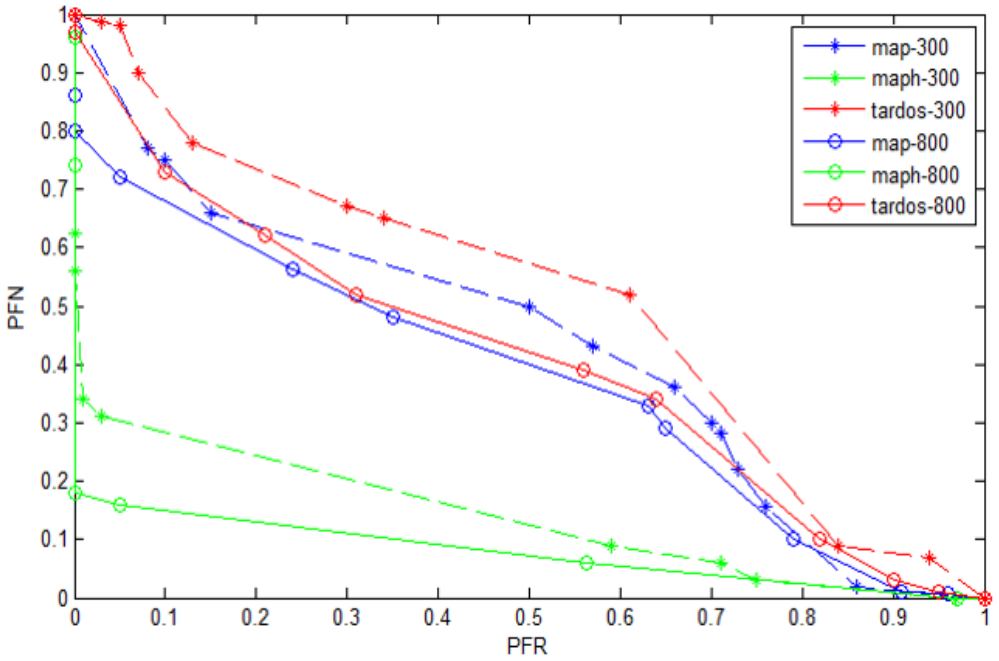


Figure 6.7: ROC curves of the MAP hierarchical decoder, the MAP decoder and the Tardos decoder for $c = 6$ and $n = 1000$ users, for different code lengths m

6.5.2.3 Time estimation

The aim of the majority traitor tracing systems is to provide an accurate accusation of colluders despite the great number of users. In this part of the experimentation, we compared the time taken by the three decoders to estimate accurately the colluders. The experimental results shown in Table 6.1 are averaged values of 10^2 trials for different number of users. We show that the proposed technique reduces significantly the time estimation of the non-layered MAP decoder which can be explained by the group-based strategy. The MAP estimation is applied only in the selected group. Although, the Tardos code has a satisfying time accusation, the gap between the proposed detector and the other decoders increases as the code length is important. The proposed decoder provides a good tradeoff between the runtime estimation and the accusation rates.

Table 6.1: Time estimation for different number of users (in sec)

n	*Tardos* [Skoric *et al.* 2007]	*MAP* [Desoubeaux *et al.* 2013]	$MAP-H$
10^3	8	830	5
10^4	600	1791	64
10^5	1800	31552	500

6.6 Conclusion

In this chapter, the main target was to propose a blind tracing system which corresponds better to real scenarios of fingerprinting system where there is a lack of information about the collusion size and strategy and where the Tardos accusation behavior remains invariant. Previous approaches in the literature proposed to use an estimation step to address the problem of the invariant behavior of the Tardos code against collusion attacks. We opt to the MAP-based blind decoder and apply it in our two level tracing scheme for several reasons related to accusation mechanism and our thesis challenges. Compared to the existing simple decodes, our two-level system takes the advantage of the group selection accusation step to have more accurate and efficient accusation.

CHAPTER 7

Conclusion and future perspectives

The research work of this thesis concerns the traitor tracing context in multimedia distribution platform. This type of applications which involves a great number of connected users need to include a tracing process to prevent media content from copyright infringement and fraud operations. Tracing traitors, or retrieving fraudulous actor in a piracy trial is constrained by important complexity and computational cost which increase with the increase of the number of audience and the number of colluders (the actors in a collusion or piracy attack).

The state of the art study has provided us a critical and comparative analysis of the existing approaches and stood out major challenges to be addressed to substantially enhance the accusation rate during the tracing process. The well-known Tardos code provides good detection rates and fair response to the major requirement of a tracing system by being close to the shortest fingerprint code length bound and allowing good detection rates. We continue with the Tardos code by improving its decoding step. The group-based property of our fingerprints offers to reduce the search space of the Tardos code and hence it contributes in enhancing the performance of the tracing process. An interesting characteristic of our fingerprint is its belonging to a hierarchy we propose to construct based on the most features which characterize a digital consumer in a multimedia distribution platform.

In this work, a two-level tracing strategy is adapted to the multi-level hierarchical fingerprints we use to embed in audio stream extracted from video content. In this report, we detailed and checked our contributions in light of the existing approaches in the literature. This chapter summaries these contributions. Then, we outline in Section 7.1 the future work to achieve this thesis.

7.1 Summary of contributions

This summary consists of the main important contributions proposed in this thesis work:

- **Traitor tracing in multimedia distribution platforms: A survey:** In Chapter 2, we give an overview on the existing traitor tracing approaches in the literature from two points of view: the cryptographic point of view and the signal processing one. Starting by studying the state-of-the-art in traitor tracing context allows us to notice that approaches based on joining the two watermarking and the fingerprinting layers in one layer are not guaranteed to be effective in multimedia distribution platforms handling million of users. Furthermore, despite the deterministic accusation results of approaches based on Error correcting codes, they remain limited in case of great size collusion attacks and the important code length they require. Hence, multimedia distribution platforms via the Internet and Peer to Peer networks, in case of including a tracing process, require appropriate choice of the tracing code and the watermarking technique.

- **Generating a multi-level hierarchical fingerprint for traitor tracing scheme:** Chapter 3 presents the standpoint of this thesis work, this first contribution consists in orienting firstly our thesis work to be considered as a group-based fingerprinting approach which proposes essentially to construct multi-level hierarchical fingerprints. Before choosing the different levels of the hierarchy, and as there is no explicit work proposing hierarchical discrete fingerprints, we deeply studied the consumers' profiles to choose the more crucial criteria which allow us to group users in the most accurate way. The decoding step of the Tardos-based tracing process has been significantly improved in terms of tracing time and computational costs. Instead of parsing all users' codewords, only selected groups are parsed to trace users.

  We performed a twofold experimentation: in one set of experiments, we evaluate the proposed system in terms of robustness to collusion attacks and tracing time with comparison to the non hierarchical fingerprinting system.

- **A two-stage traitor tracing scheme for hierarchical fingerprints:** In Chapter 4, we continue with improving the Tardos tracing process by proposing a suitable two-level tracing strategy. The accusation is made in two successive levels: in the first level, the

selection of groups is made with BS code, and in the second level, tracing users is made by the Tardos code in only selected groups. This strategy, improved by the group weights generated from the hierarchy, enables a more accurate and faster colluders' detection rate.

We assigned an important consideration to the comparison of the performance of the proposed two-level tracing system with that of state-of-the-art systems.

- **A proposal of optimizing fingerprinting code length based on a QR-code-based conversion:** One key requirement in a fingerprinting system, especially in the watermarking layer, is the fingerprint length which should not alter the quality of the media content. We propose in Chapter 5 to reduce our fingerprint length (which is a parallel concatenation of two codes: BS and Tardos) to reduce the embedding time required for the insertion step. So, we propose to convert our 4-ary fingerprint to a QR code, which provides us a compact fingerprint size and more robustness due to its error correcting code. The algorithm we propose to embed fingerprint, based on scrambling the different blocks of the QR code, enhances the security side of the watermarking layer.

  We compare the tracing performance of the proposed fingerprinting system in terms of embedding time and robustness to different attacks, to the other tracing schemes.

- **Towards a blind traitor tracing scheme for hierarchical fingerprint:** One key challenge to address when applying traitor tracing scheme in real scenarios is to provide an efficient and accurate tracing process despite the lack of information about the colluders' strategy and collusion size. Indeed, the original Tardos tracing performance is considered as suboptimal because of its agnostic behavior and conservative accusation regardless the collusion strategy. In the literature, some approaches propose to use on estimation algorithms: EM and MAP for simple Tardos decoder to estimate the collusion channel. According to that, we propose to validate the efficiency of the two algorithms in our two-level fingerprinting scheme in terms of robustness to collusion attacks, accusation time and error probability of accusation. We opt then to the MAP decoder which is the most suitable algorithm for our fingerprinting system.

7.2 Future perspectives

During our thesis work, more than one perspective tied to our proposed contributions have drawn our attention and deserve to proceed research.

- To survive the problem of the code length and inspired by [Desoubeaux *et al.* 2012], we proposed to enlarge the alphabet size of our codewords from a binary alphabet to a 4-length cardinality one. Then, we proposed to convert these codewords to QR-code to reduce the proposed codeword length and because the embedding technique requires to downscale the fingerprint to a binary alphabet. The perspective idea here is to adapt the watermarking technique or to propose a watermarking technique able to embed our fingerprints with higher alphabets and hence to study the eventual constraints as the watermarking complexity, its efficiency and its robustness to new designs of attacks proposed in the same context of higher alphabets.

- A second perspective is to extend our work to a dynamic traitor tracing context. In fact, merging to dynamic scheme should require additional constraints as time constraint and adaptive fingerprint dictionary which should be updated at every tracing round. We should study and check if our two-level tracing strategy is efficient in such constrained scheme.

- Another interesting idea is to propose fingerprints with varying code length m. The variation of the code depends on its ability to be a colluder or to belong to a suspicious group of users. And here the fingerprint generation step is made according to the proposed hierarchy. Moreover, using variable value of m should prevent more collusion attacks and discourage colluders to collaborate. In the accusation process, estimating the code length m using an estimation algorithm is required.

- It will be interesting to propose a whole collusion secure fingerprinting scheme for other application than multimedia distribution platform. One promised sceanrio is the video game scenario proposed by [Schäfer 2016]. Considering the media types and formats available in video games, joining different watermarking techniques respectively for each

media type and benefiting from a higher payload for embedding fingerprints deserve a deep research.

CHAPTER 8

Appendix 1

8.1 The digital audio watermarking

Several approaches were striving in finding a good compromise between respecting the human auditory system (HAS) and embedding the watermark into the audio signal without altering the signal quality [Bhat K. *et al.* 2008], [Wang *et al.* 2011], [Lanxun *et al.* 2007]. In fact, due to the reduced capacity of the audio signal which is lower than for the image, embedding additional information in audio content seems to be more hard than in images.

8.1.1 The IFPI requirements for the audio watermarking efficiency

The IFPI, International Federation of the Phonographic Industry, has studied the efficiency requirements of digital watermarking and has added some other conditions: to check that the embedded watermark does not alter the original audio signal quality after the watermarking process, the value of the SNR of the watermarking algorithm should exceed 20 dB. Additionally, the hidden watermark must be able to resist to different types of attacks: the most common audio processing attacks and the illicit removal of the watermark as long as the quality of the audio signal is quite good. Moreover, IFPI assigns that the computation time of the watermark insertion and extraction processes depends on the watermarking application.

8.1.2 The audio watermarking attacks

Any manipulation of the audio signal can affect the hidden watermarks. Some attacks have deeper effect than others depending on the manner the audio will be used. For example, in case of detection of radio transmission of commercials, the watermarking scheme should be

robust against normalization, compression, de-noiser, filtering attacks, etc.. Furthermore, in case of copyright protection application, when sharing the audio content via Internet, the main attack will be the loss compression like MP3 at high compression rates. Henceforth, it was assumed that audio watermarking applications should use a standard benchmark tool to check their audio watermarking robustness results. The most popular audio benchmark tool is the StirMark Benchmark for Audio SMBA.

8.2 Comparison of the adopted watermarking technique to other existing techniques

8.2.0.1 Comparison to other audio watermarking techniques

This part is related to experimental results in Chapter 6. In this part, due to the diversity of used watermarks and audio signals, we propose to report results from existing techniques and to compute the average of the NC and SNR of the different watermarks and audio signals for all compared techniques. In Table 8.3, we report details of the different samples used in these techniques.

- Comparative inaudibility results: We compute the average value of the SNR of different audio signals respectively in the scheme we propose and in the other existing schemes. As depicted in Table 8.2, compared to the SNR values of the different existing techniques, the SNR value of the used technique for the different audio sequences extracted from the tested videos (48.02 dB) is the highest one.

- Comparative of MP3 compression results: We compare in this part the robustness to the MP3 compression of the used technique to the other existing techniques by using the NC measure. As shown in Table 8.1, we notice that the robustness to the MP3 compression of the adopted technique has the highest NC values.

- Comparative of audio Stirmark attacks results: We compare in this part the robustness to some audio Stirmark attacks of the used technique to the other existing techniques by using the average value of the NC measure. Looking at the different results as depicted

in Table 8.4, the adopted technique has good robustness against the add noise attack, the normalized attack, the zero cross attack compared to the other techniques.

Table 8.1: Comparative MP3 compression results to some existing audio watermarking techniques

Algorithm	128 Kbps	96 Kbps	64 Kbps
The used technique	**1**	**0.99**	**0.97**
Quantization of wavelet coefficients [Bhat K. *et al.* 2008]	X	X	0.84
Reduced singular value decomposition [Wang *et al.* 2011]	0.94	0.93	0.92
Mean-quantization of wavelet coefficients [Lanxun *et al.* 2007]	X	X	0.77
Support Vector regression [Xu *et al.* 2007]	0.96	X	X

Table 8.2: Comparative Inaudibility results to some existing audio watermarking techniques

Algorithm	SNR(dB)
The used technique	**48.02**
Quantization of wavelet coefficients [Bhat K. *et al.* 2008]	21
Reduced singular value decomposition [Wang *et al.* 2011]	24.3
Mean-quantization of wavelet coefficients [Lanxun *et al.* 2007]	37.9

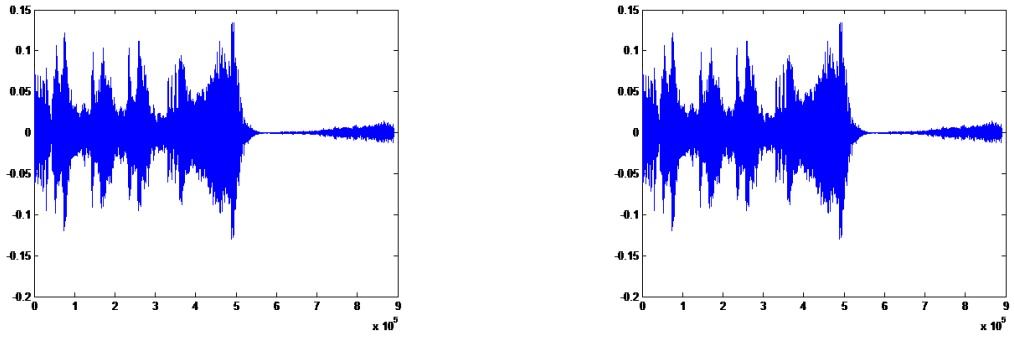
Table 8.3: Details of audio samples used in some existing techniques

	Audio watermarking technique		
	[Bhat K. *et al.* 2008]	[Wang *et al.* 2011]	[Lanxun *et al.* 2007]
Time length (s)	X	X	16
Format	wav	wav	wav
Bits per sample	16	32	16
Sample rate (KHz)	44.1	48	44.1
Channel mode	mono	X	X

Table 8.4: Comparative of some audio Stirmark attacks to existing audio watermarking techniques

Attack	Adopted technique	[Bhat K. *et al.* 2008]	[Wang *et al.* 2011]	[Lanxun *et al.* 2007]
Add noise	**0.95**	X	X	X
Normalized	**0.94**	X	X	X
Zerocross	**0.91**	X	X	X
LowPass filtering	**0.91**	X	0.77	X
HighPass filtering	**0.92**	X	0.9	X

8.3 Some experimental results related to the audio watermarking technique

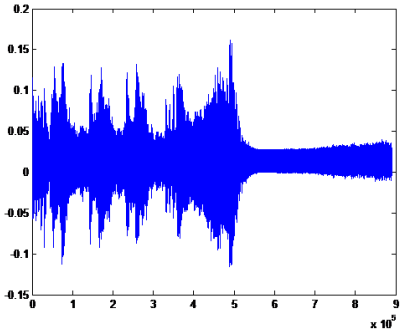


(a) Original audio signal

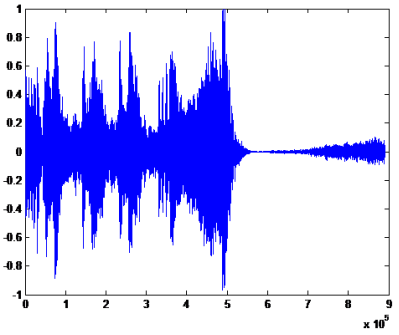
(b) Fingerprinted audio signal

Figure 8.1: Original audio signal vs fingerprinted audio signal.

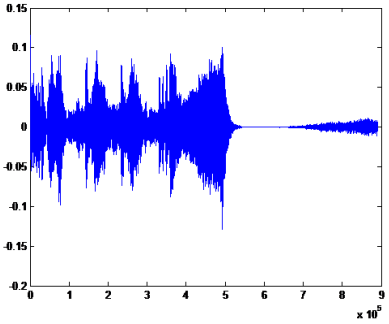
We show in Figure 8.1, a short portion of an original audio signal extracted from "film.avi" and that of the corresponding fingerprinted signal. In Figure 8.2 and Figure 8.3, we show the attacked fingerprinted signals respectively with some Audio Stirmarkaudio attacks and collu-



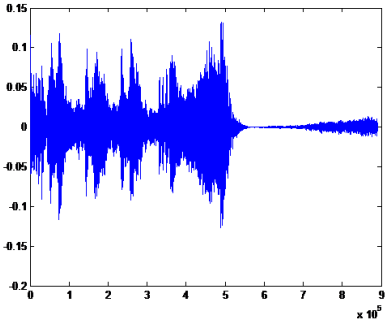
(a) Add noise attack



(b) Normalized attack

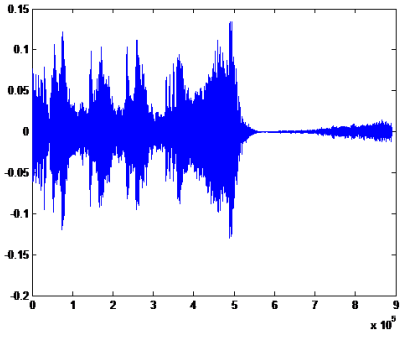


(c) High pass filtering attack

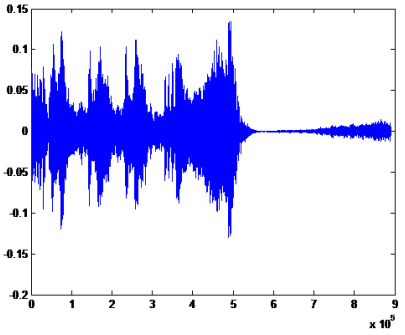


(d) Low pass filtering attack

Figure 8.2: Portions of the attacked audio signal with audio Stirmark attacks.



(a) Averaging attack



(b)WCA attack

Figure 8.3: Portions of the attacked audio signal with collusion attacks.

sion ones. It is clear that audio Stirmark attacks (Normalized, High pass and low pass filtering attacks) have deeper effect on the audio signal than other attacks.

8.3.1 Inaudibility results of the audio watermarking technique

Subjective tests: This type of listening tests are important to perceptual quality assessment, since the final judgment is provided by human acoustic perception. In this test, ten listeners are provided with the original and the watermarked audio signals and are asked to report the differences between the two signals, using a five-point subjective grade, SDG, as shown in Table 8.5. The average SDG scores obtained with our scheme are shown in Table 8.6. These

Table 8.5: Subjective and Objectives difference grades

SDG	ODG	Impairements' description	Quality
5	0	Imperceptible	Excellent
4	-1	Perceptible but not annoying	Good
3	-2	Slightly annoying	Fair
2	-3	Annoying	Poor
1	-4	Very Annoying	Bad

Table 8.6: Average SDG and ODG scores obtained with the different tested audio files

Audio File	Average SDG	ODG
"song.wav"	4.9	-0.22
"match.wav"	4.55	-0.65
"program.wav"	4.85	-0.88
"film.wav"	4.7	-0.45
"clip2.wav"	4.9	-0.30
"sport.wav"	4.1	-0.66
"politic.wav"	4.8	-0.1
"film2.wav"	4.3	-0.66
"prog2.wav"	4.65	-0.83

high SDG scores prove that our scheme provides good inaudibility of the watermark in the audio signals.

Objective tests: The target of objective measurement algorithms is to replace the subjective listening tests by modeling the listening behavior of human beings. The objective measurement metric called objective difference grade, *ODG*, could not always correlate closely to the subjective listening tests results [Keiler 2006]. Henceforth, a final judgment with regard to the audio quality has to be based on subjective listening tests [Cvejic *et al.* 2007]. The *ODG* value is the output variable obtained from perceptual evaluation of audio quality (PEAQ) measurement algorithm specified in ITU-R BS.1387 (International Telecommunication Union-Radiocommunication Sector) [Thiede *et al.* 2000]. It corresponds to the subjective grade mentioned in human based audio tests. The ODG value should range from 0 to -4 (respectively from imperceptible to very annoying) as explained in Table 8.6. To compute the ODG between original and watermarked audio signals the EAQUAL software program (Evaluation of Audio Quality) [development] based on ITU-R BS.1387 is used. Table 8.6 shows obtained results of objective tests and we notice that all the obtained ODG scores are between -1 and 0 which proves that it is hard to differentiate between the original and the watermarked audio signals.

In Table 4.5, other objective tests are made by computing the SNR measures for each listed audio sample.

8.3.2 Efficiency time criterion

Regarding the time efficiency criterion, we remark the lack of this information in the majority of the existing audio watermarking techniques. Hence, Table 8.7 summarizes the computational time of the embedding and detection steps of the watermarking technique we apply in this work.

Table 8.7: Efficiency time

Audio file	Embedding time (in second)	Detection time (in second)
"match.wav"	52	27
"song.wav"	24.5	20.47
"prog1.wav"	36.5	25
"film.wav"	32	30
"clip2.wav"	52	24
"sport.wav"	40	35
"politic.wav"	42	41
"film2.wav"	47	41
"prog2.wav"	42	42

CHAPTER 9

Appendix 2

9.1 Study of digital consumer behaviour

The digital revolution which characterizes this era has undoubtedly facilitated our daily lives and works by providing a diversity of digital tools: digital TV, laptops, mobile phones, web cams, etc. Furthermore, this revolution has led to a significant change of consumer behavior and a spate of interest to a massive multimedia consumption. This consumption includes different multimedia treatments: copying, sharing, etc, which can be authorized or unauthorized treatments in several cases. In this study, we aim to report some analysis and statistics provided by different consumer market researches allover the world showing the recent digital consumer characteristics. Regarding this study, it was interesting in this thesis work to be tied to the real scenarios and to base our contributions on it.

9.1.1 Favorite devices for digital consumers

According to the 2016 Accenture digital consumer survey, 91% of consumers now own a smartphone and 56% plan to buy a smartphone in the next 12 months. From the 2015 Accenture digital consumer survey, as reported in Figure 9.1 , when accessing different types of digital content, the laptops are the most preferred device for consumers.

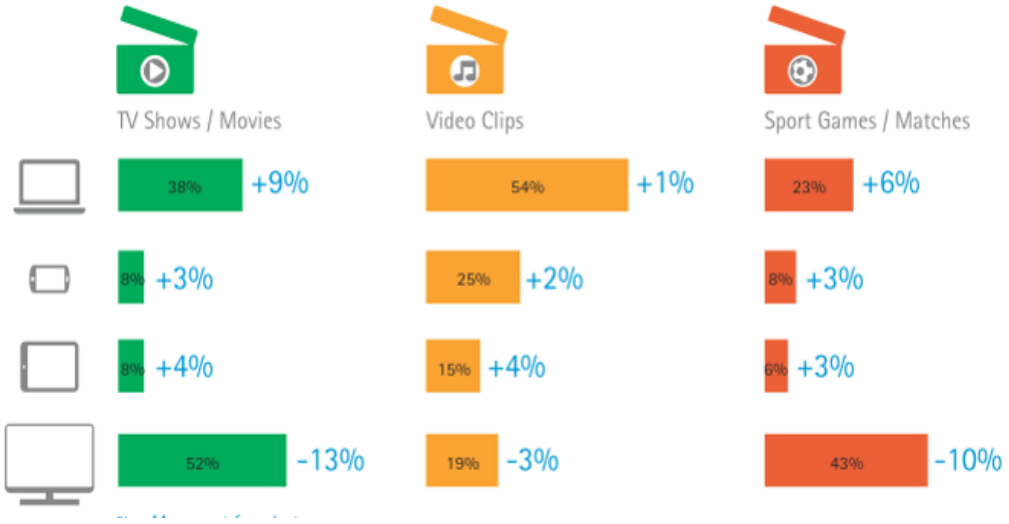


Figure 9.1: Favorite devices for digital consumers.

9.1.2 Characteristics of digital consumers

One key standpoint in our thesis work was to analyze the digital consumers' characteristics in general and especially in multimedia distribution platforms. According to NPD group [1], about 40% of households with a connected TV streamed Netflix content during the period âĂŤ a percentage that topped 50% among consumers age 18 to 24. Twenty-one percent of connected-TV owners said they migrated from using over-the-top (OTT) video services on the computer and now watch on the TV instead. Now, nearly 25% of 18- to 24-year-olds stream Netflix on a laptop or PC. Among all age demos, 14% of respondents to NPD's connected intelligence survey watch Netflix on a laptop or PC, 13% on a tablet and 8% on a mobile phone. Interestingly, Netflix use on connected devices is higher among women than men. As shown in Figure 9.2, reported from 2013 Accenture digital consumer survey, more than 90 percent of consumers prefer watching video content over the Internet, which includes movies, TV programs, videos on demand. Furthermore, this consumption is more and more frequent and increasing and the most significant growth is evident in high frequency categories, as depicted in Figure 9.3, those watching videos daily or three to five times per week.

[1]The NPD Group, Inc. (formerly National Purchase Diary) is a market research company.The NPD group operates in 20 countries, interviews 12 million consumers a year, and monitors consumer purchase data from over 165,000 stores.

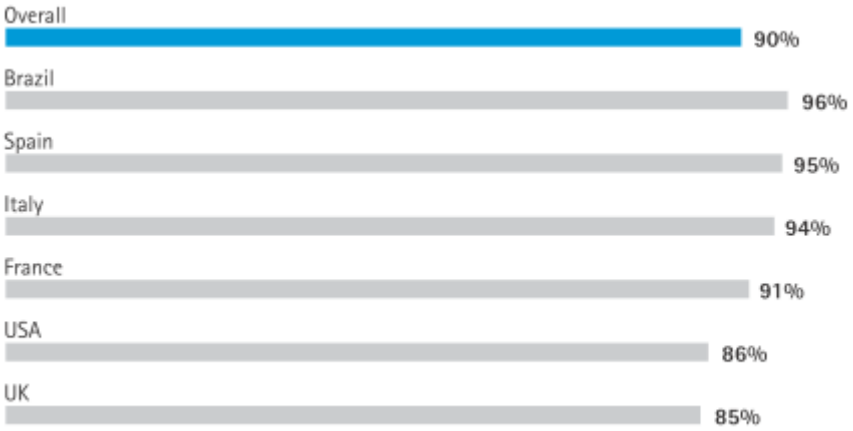


Figure 9.2: Video online consumption allover the world.

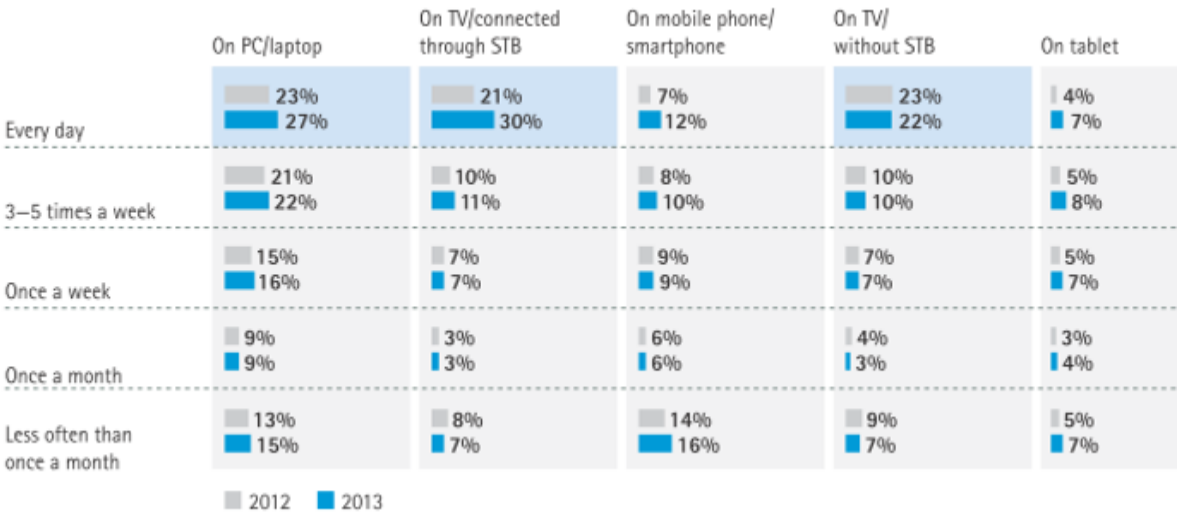
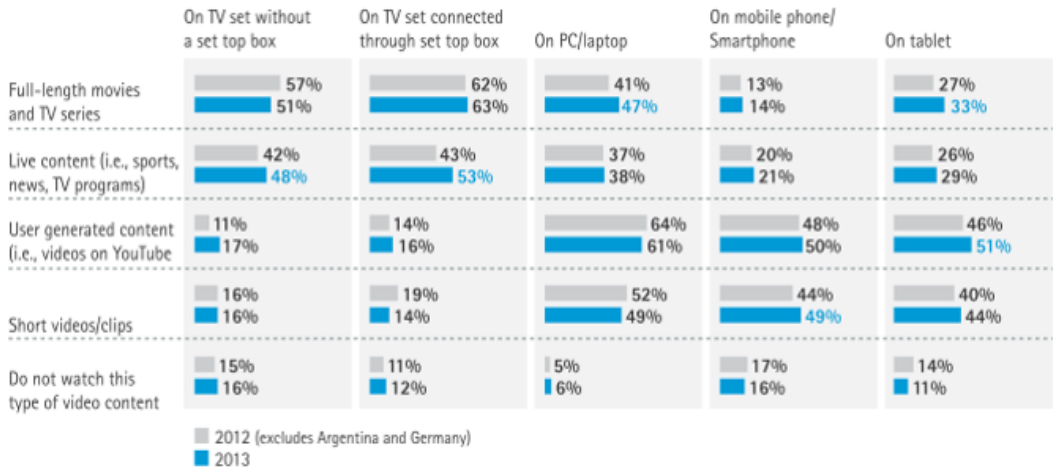


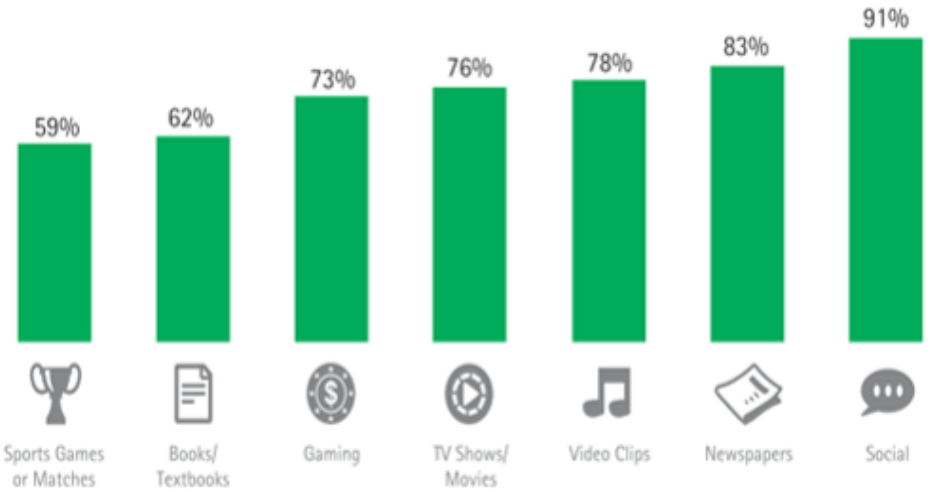
Figure 9.3: Video online consumption allover the world.

9.1.2.1 The most requested video content

In Figure 9.4, reported from 2013 Accenture digital consumer survey, the most requested video content are video clips, films, TV programs, sports, videos on Youtube, clips, etc. According to these statistics, we have collected our experimental video samples among the different types.



(a) for 2012-2013



(b) for 2015

Figure 9.4: The most requested content for video online consumption for 2012 and 2013.

9.1.2.2 Social characteristics

The digital consumer age range According to the 2015 Accenture digital consumer survey, and as shown in Figure 9.5, the most interested age range in video distribution platforms is the teenager one, ranging from 18-old to 34-old year, for an increasing frequency of PC accessing rate percentage exceeding 44%, compared to less than 24% for old people exceeding 55-old year.

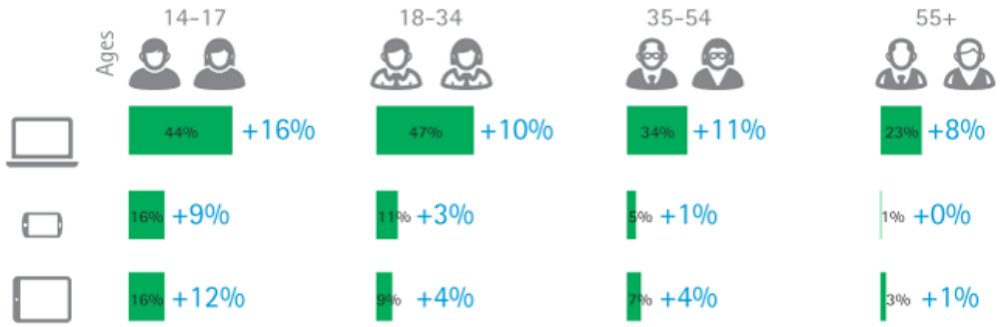


Figure 9.5: Frequency of accessing digital content by age range.

9.1.2.3 Geographic characteristics

Accessing to digital video content is more important in one country than in another. According to the World Economic Forum, the digital piracy rate depends on geographic belonging. It is very important, as shown in Figure 9.7, in North European, Latin American, Asian and African countries which can be explained as detailed in Figure 9.8 by the lack of laws to protect intellectual property from infringement in such countries. As an example, we can see that the piracy rate in Japan is less than 19% which is tied to a strong intellectual property protection of 6. However, in Brazil, the important piracy rate of about 50% is due to a a weak intellectual property protection of 3.3 which in less the average value of 3.8.
According to this study, we have chosen the most important characteristics of digital consumers to construct the multi-level hierarchy of users' fingerprints. Users having similar profiles are grouped together.

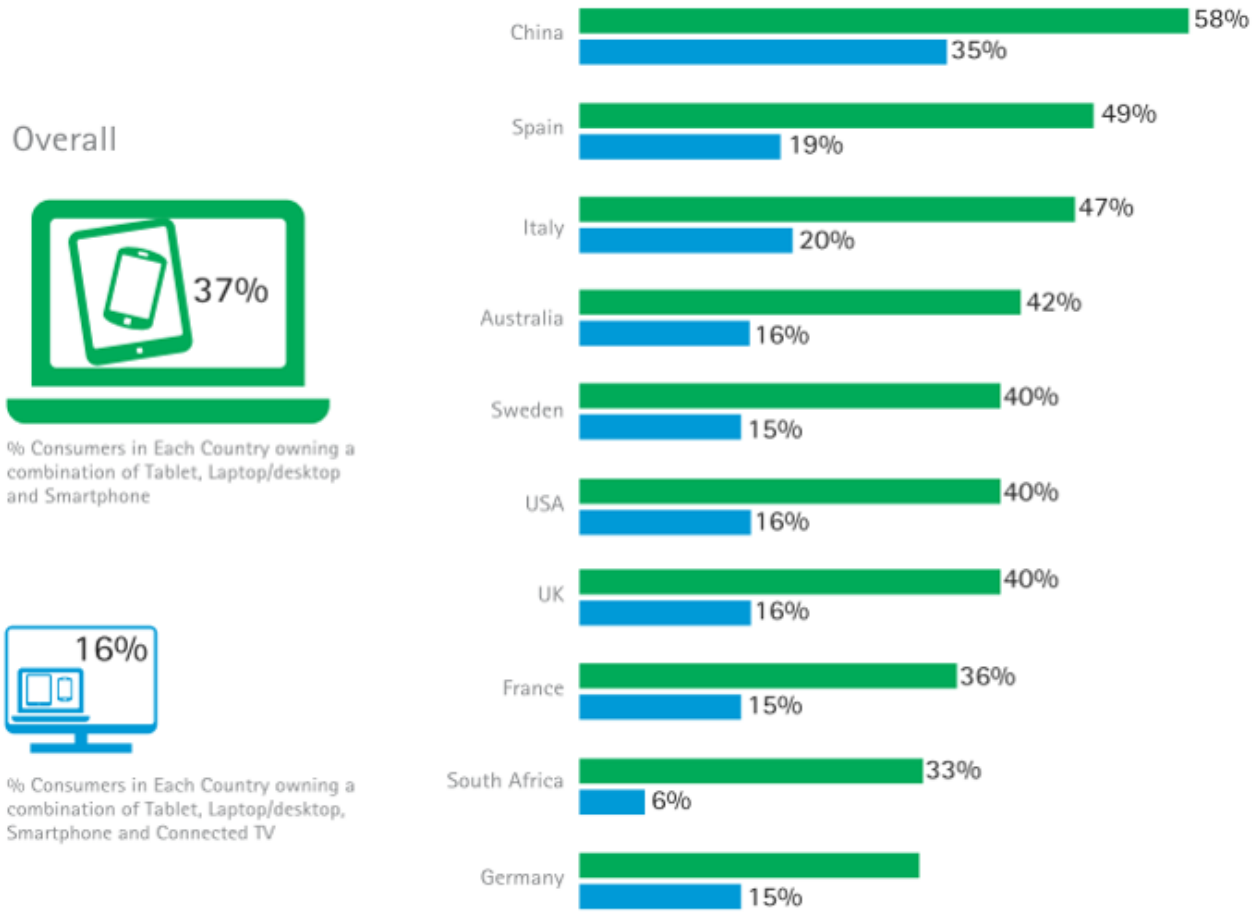


Figure 9.6: Accessing digital content by geographic belonging.

RANK	COUNTRY/ECONOMY	VALUE
1	United States	18
2	Japan	19
3	Luxembourg	20
3	New Zealand	20
5	Australia	21
6	Austria	22
7	Denmark	23
7	Sweden	23
9	Belgium	24
9	Finland	24
9	Germany	24
9	Switzerland	24
9	United Kingdom	24
14	Canada	25
14	Netherlands	25
14	Norway	25
17	Israel	30
18	Singapore	32
19	Ireland	33
20	Czech Republic	34
20	South Africa	34
22	France	36
22	United Arab Emirates	36
24	Slovak Republic	37
25	Korea, Rep.	38
25	Taiwan, China	38
27	Hungary	39
28	Portugal	40
29	Puerto Rico	42
30	Hong Kong SAR	43
31	Malta	44
32	Slovenia	45
32	Spain	45
34	Cyprus	47
34	Estonia	47
34	Italy	47
37	Iceland	48
38	Qatar	49
39	Brazil	50
39	Saudi Arabia	50
41	Poland	51
42	Colombia	52
42	Croatia	52
44	Bahrain	53
44	Latvia	53
44	Lithuania	53
47	Malaysia	54
47	Mexico	54
49	Mauritius	55
50	Jordan	57
51	Kuwait	58
52	Chile	59
52	Costa Rica	59
54	India	60
54	Oman	60
54	Turkey	60
57	Egypt	62
57	Greece	62
57	Romania	62
57	Russian Federation	62
61	Bulgaria	63
62	Macedonia, FYR	65
62	Peru	65
64	Morocco	66
65	Uruguay	68
66	Argentina	69
66	Philippines	69
66	Serbia	69
69	Lebanon	71
69	Thailand	71
71	Panama	72
72	China	74
72	Honduras	74
72	Kazakhstan	74
75	Albania	75
75	Dominican Republic	75
75	Tunisia	75
78	Senegal	77
79	Kenya	78
79	Montenegro	78
81	Bolivia	79
81	Botswana	79
81	Guatemala	79
84	Côte d'Ivoire	80
84	El Salvador	80
86	Nigeria	81
86	Vietnam	81
86	Zambia	81
89	Cameroon	82
89	Nicaragua	82
91	Sri Lanka	83
91	Ukraine	83
93	Indonesia	84
93	Paraguay	84
95	Algeria	85
95	Azerbaijan	85
95	Pakistan	85
98	Armenia	86
99	Bangladesh	87
99	Yemen	87
101	Venezuela	88
102	Libya	89
103	Georgia	90
103	Moldova	90
105	Zimbabwe	91
n/a	Angola	n/a
n/a	Barbados	n/a
n/a	Bhutan	n/a
n/a	Burkina Faso	n/a
n/a	Burundi	n/a
n/a	Cambodia	n/a
n/a	Cape Verde	n/a
n/a	Chad	n/a
n/a	Ethiopia	n/a
n/a	Gabon	n/a
n/a	Gambia, The	n/a
n/a	Ghana	n/a
n/a	Guinea	n/a
n/a	Guyana	n/a
n/a	Haiti	n/a
n/a	Iran, Islamic Rep.	n/a
n/a	Jamaica	n/a
n/a	Kyrgyz Republic	n/a
n/a	Lao PDR	n/a
n/a	Lesotho	n/a
n/a	Madagascar	n/a
n/a	Malawi	n/a
n/a	Mali	n/a
n/a	Mauritania	n/a
n/a	Mongolia	n/a
n/a	Mozambique	n/a
n/a	Myanmar	n/a
n/a	Namibia	n/a
n/a	Nepal	n/a
n/a	Rwanda	n/a
n/a	Seychelles	n/a
n/a	Suriname	n/a
n/a	Swaziland	n/a
n/a	Tajikistan	n/a
n/a	Tanzania	n/a
n/a	Timor-Leste	n/a
n/a	Trinidad and Tobago	n/a
n/a	Uganda	n/a

Figure 9.7: Piracy rate allover the world.

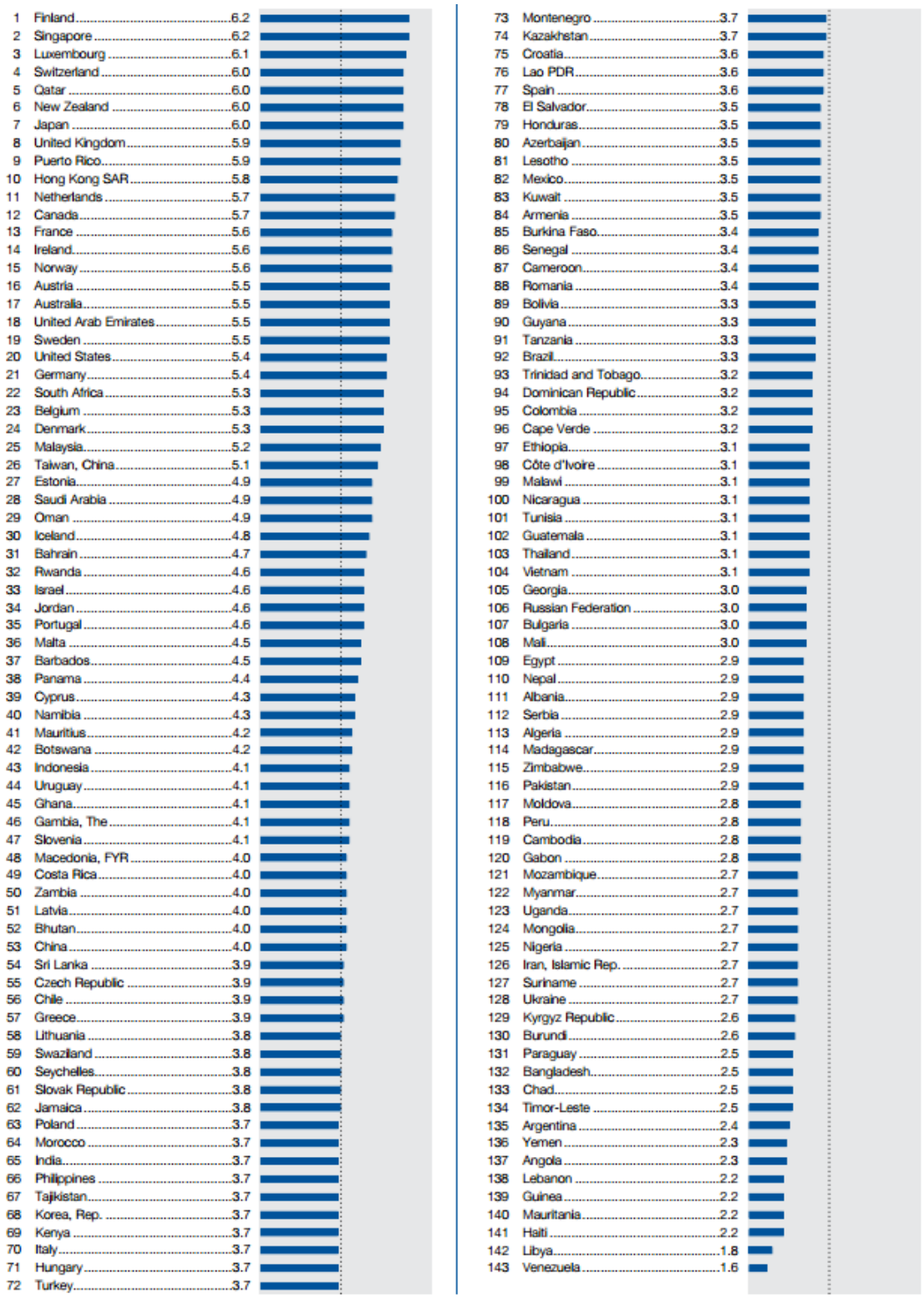


Figure 9.8: Intellectual property protection rate allover the world ranging from 1 for extremely weak protection to 7 for extremely strong one.

CHAPTER 10

Appendix 3

10.1 Clustering impact on multi-level hierarchical fingerprints

According to [Wang *et al.* 2004], several researchers in the tracing traitor field agree upon the fact that constructing a group-based fingerprint should reduce the computational costs of the tracing process. The group-based fingerprinting approach has been used previously in some works. As an example, in [Desoubeaux *et al.* 2012], the group-based property was applied randomly to groups of fingerprints; the groups are constructed with no constraints. Another example, in [Yong *et al.* 2014], group of users are constructed by applying a clustering algorithm to the group identifier.

Data clustering process in traitor tracing context can be assimilated to an unsupervised data analysis process whose goal is to partition unlabeled users' identifiers into groups of similar characteristics, called clusters. In this part, we propose to involve a clustering step, as shown in Figure 10.1 to construct our multi-level hierarchical fingerprint. The only requirement is the codeword length of users' fingerprints which is constrained by the applied watermarking technique used in the tracing scheme [Charfeddine *et al.* 2010]. The widely used k-means, hierarchical clustering, SOM and FCM are suggested as the clustering algorithms to generate the groups' identifiers [Yong *et al.* 2014] [Fahad *et al.* 2014].

10.1.1 How to cluster users' fingerprints?

To show off the impact of using a group-based fingerprints in the tracing process, we compare between different clustering algorithms, namely K-means, the Fuzzy C-means, the Hier-

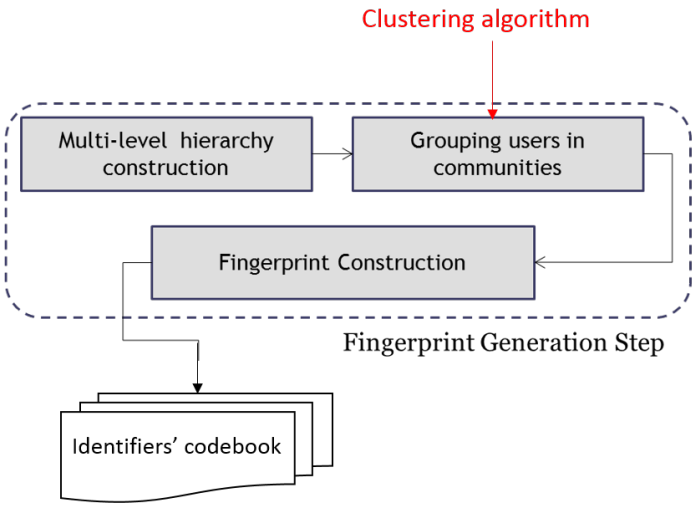


Figure 10.1: The fingerprint generation step including clustered groups.

archical Clustering and the SOM clustering algorithm [Panda *et al.* 2012]. Hence, we remind the clustering principle of each algorithm separately:

- The K-means algorithm: in the context of group-based clustering, the algorithm consists in computing in an iterative way the Euclidean distance between each user fingerprint and the center of the group. For each iteration, groups are obtained by minimizing the computed distance. When there is no change, the algorithm converges and we get the final K groups of users' fingerprints.

- The Fuzzy C-means algorithm [Bezdek *et al.* 1984]: *FCM* Initially, the matrix of fingerprints is constructed. The most important inputs of the FCM clustering algorithm are respectively: the normalized fingerprints' metrics, the required number of clusters and the number of the iterations. The output is the the matrix of the membership of each fingerprint in each cluster.

- The hierarchical clustering algorithm, *HC*: as assessed in [Johnson 1967], applying the HC algorithm in constructing groups consists in starting by assigning each user fingerprint to a cluster. Then, the closest pair of fingerprints is computed to merge into one cluster. The outputs of this clustering algorithm are the groups of fingerprints.

- The Self Organizing Map algorithm: *SOM*, the input consists in users' fingerprints and the self-organizing map computed with the number of groups and fingerprints. From the

output matrix, we obtain the column vector whose *ID* is equal to the member *ID*. The output is the the matrix of the belonging of each fingerprint to a cluster.

Once the clusters are obtained, the traitor tracing process starts. The retrieved fingerprint is treated to check the detection success.

10.1.2 How to trace a suspicious user in a group-based tracing process?

According to the Tardos-based tracing process detailed in Section 3.3, for the K-means, the distance between the retrieved fingerprint and the center of each cluster is computed to add it in the corresponding group. For the other clustering algorithms, the belonging of the retrieved fingerprint to a group is obtained by applying a second clustering round to the set of fingerprints. We assume that the detection performance is successful once the matching between the retrieved group and the original cluster is obtained. In Section 10.1.3, we present the evaluation of the impact of the group-based fingerprinting system we propose.

10.1.3 Evaluation of the clustering algorithm in the fingerprinting system

As far as it is known, the group-based fingerprinting system for discrete approaches has not been taken into account previously in the literature until the system we propose.
Hence, It is important to evaluate the performance of the tracing process according to two relevant criteria: the robustness to collusion attacks and the capacity of the group-based fingerprinting system. We cluster the users' fingerprints to obtain the groups of fingerprints. In the following, according to the adopted DCT-based audio watermarking technique , we choose to construct 1000 fingerprints and then we cluster them to 10 clusters by using the different clustering algorithms (Kmeans, fcm, HC and SOM).

10.1.3.1 Robustness to collusion attacks

In this part, we show the robustness of the group-based fingerprinting systems under collusion attacks : the averaging attack, the all-one, the all-zero and the random attack [Trappe *et al.* 2003].

We have evaluated the proposed fingerprinting system in two cases: the robustness to collusion attacks into the same group and the robustness between different groups.

10.1.3.2 Robustness to collusion attack into the same group

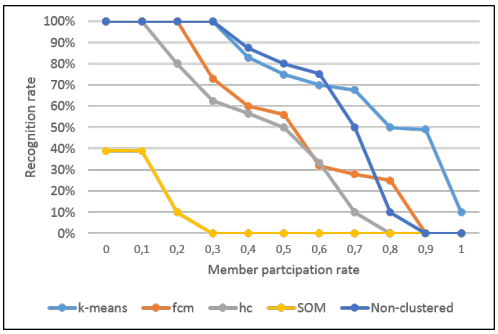
According to Figure 10.2, we observe that, compared to non-clustered group-based fingerprint, the clustered one, especially for K-means has better recognition rate, even for increasing collusion size. Furthermore, although the group recognition rate decreases with the increase of the member participation rate, K-means has the best behavior and a satisfying group recognition rate which is close to 100 percent when the member participation rate is below 30 percent. As studied in [Furon & Pérez-Freire 2009b], some collusion attacks have deeper impact on the performance of a tracing system than others. In our work, we show that the Random and the AllOne attacks have deeply influenced the detection rates of the four clustering algorithms. In fact, according to the Figure 10.2(b) and Figure 10.2(c), the recognition rate decreases clearly for the FCM, the HC and the SOM clustering algorithm. The K-means, however, has still the best rates compared to the other clustering algorithms and guarantees a recognition rate above 10 percent even if the member participation rate is 100 percent which provides the detection of at least one suspicious user fingerprint.

10.1.3.3 Robustness to collusion attacks between different groups

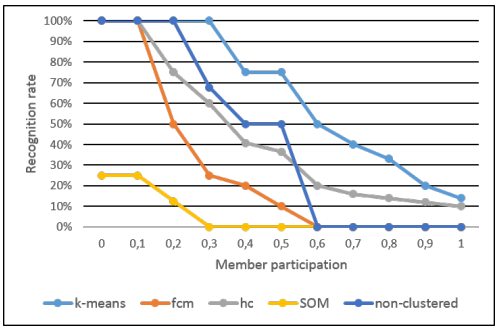
According to the experiments conducted in Sect.10.1.3.2, we have noticed that K-means which has provided the best recognition rate compared to the other clustering algorithms. Hence, we propose to test it for collusion including more than one group. As depicted in Figure 10.3, the group recognition rate of K-means decreases clearly with the increase of the group participation in the collusion attack which can be explained by the distance-based clustering process.

Another important point is to compare the group detection capacity of the proposed technique to other group-based existing techniques. Moreover, due to the diversity of experimental assessments, we propose to report the main important and available experimental results in respectively [Wang *et al.* 2004], [He & Wu 2007], [Hamida *et al.* 2011] and [Ye *et al.* 2013].

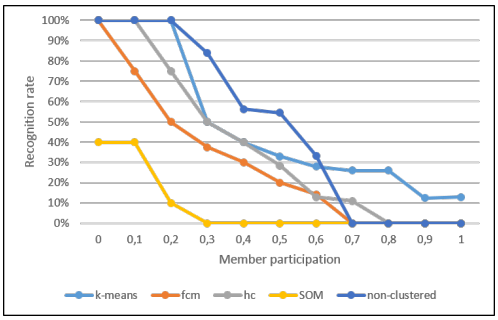
Looking at the different results reported in Table 10.1, we notice that, for all the group-based



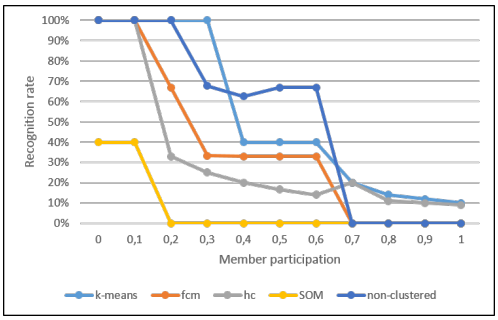
(a) Averaging attack



(b) Random attack



(c) All-one attack



(d) All-zero attack

Figure 10.2: Robustness of the group-based fingerprinting system to collusion attacks into the same group.

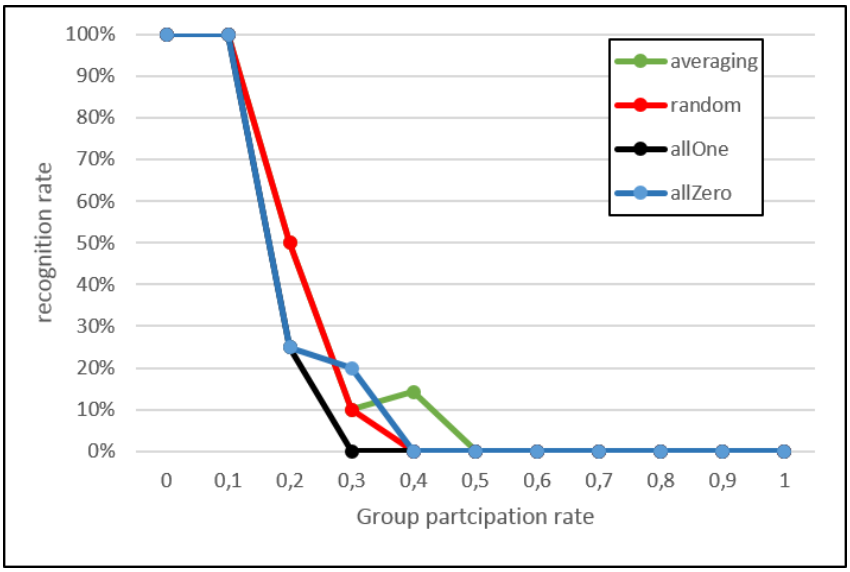


Figure 10.3: Robustness of the group-based fingerprinting system to different attacks: Averaging, Random, AllOne, AllZero

techniques, the group detection capacity under the averaging collusion attack decreases with the increase of the group participation rate, called Group rate. Moreover, the proposed technique based on K-means algorithm shows good detection rate compared to the other existing techniques. Furthermore, we have proved previously that the group-based technique we propose provide a good compromise between the detection rate, the CPU time tracing and the decoding complexity.

Table 10.1: Comparative group detection capacity results to some existing group-based techniques

Technique	Group rate=0.1	Group rate=0.2	Group rate=0.25	Group rate=0.3
[Wang *et al.* 2004]	0.5	X	X	X
[He & Wu 2007]	$\leq$0.9	$\leq$0.35	X	X
The proposed technique	**1**	**0.5**	**0.45**	**0.1**

10.1.3.4 The capacity of the group-based fingerprinting system

One other relevant criterion to evaluate the group-based fingerprinting system is its capacity, which means the maximal number of groups it can support to maintain a good group recognition rate (above 50 percent [Yong *et al.* 2014]).

In fact, the real challenge of the majority of the multimedia distribution systems including a traitor tracing process is to handle the great size of fingerprints. Hence, in our context, we propose to compare between the four clustering algorithms to find which one provides the largest group capacity. According to Tab.10.2, especially for the K-means clustering algorithm, the group capacity is constrained by the applied collusion attack, it decreases with the Random attack. This is still interesting (above 50 groups) compared to the other clustering algorithms: FCM, HC and SOM for which the capacity is not very interesting and does not exceed 20 groups.

In this part of the thesis work, we have considered the robustness results of the proposed group-based fingerprinting scheme to different types of collusion attacks for the two scenarios: between different groups and into the same group, for the four clustering algorithms. We have also studied the group capacity of the four clustering algorithms under collusion attacks. We

Table 10.2: The group capacity of the different clustering algorithms

	K-means	FCM	HC	SOM
Averaging attack	100	3	2	2
Random attack	50	3	2	2
AllOne attack	100	20	0	2
AllZero attack	100	20	0	2

notice that compared to the performance of the other clustering algorithms, the K-means has shown a satisfying tracing rates.

Bibliography

[Akashi *et al.* 2008] N. Akashi, M. Kuribayashi and M. Morii. *Hierarchical construction of Tardos code*. In Information Theory and Its Applications, 2008. ISITA 2008. International Symposium on, pages 1–6, 2008. (Cited on pages 5, 40, 60 and 61.)

[Amiri & Tardos 2009] Ehsan Amiri and Gábor Tardos. *High rate fingerprinting codes and the fingerprinting capacity*. In Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2009, New York, NY, USA, January 4-6, 2009, pages 336–345, 2009. (Cited on page 40.)

[Anderson & Petitcolas 1998] Ross J. Anderson and Fabien A. P. Petitcolas. *On the limits of steganography*. IEEE Journal on Selected Areas in Communications, vol. 16, no. 4, pages 474–481, 1998. (Cited on page 16.)

[Barg & Kabatiansky 2004] Alexander Barg and Gregory Kabatiansky. *A Class of I.P.P. Codes with Efficient Identification*. J. Complex., vol. 20, no. 2-3, pages 137–147, April 2004. (Cited on page 33.)

[Barg & Kabatiansky 2011] Alexander Barg and Grigory Kabatiansky. *Digital Fingerprinting under and (Somewhat) beyond the Marking Assumption*. In Information Theoretic Security - 5th International Conference, ICITS, pages 202–205, 2011. (Cited on page 34.)

[Berchtold *et al.* 2013] Waldemar Berchtold, Marcel SchÃd'fer, Huajian Liu, FÃąbio Touceira Takahashi, AndrÃl' Schmitz, Sascha Zmudzinski, Martin Steinebach and Jonas Wieneke. *Video game watermarking*, 2013. (Cited on page 31.)

[Bezdek *et al.* 1984] James C. Bezdek, Robert Ehrlich and William Full. *FCM: The fuzzy c-means clustering algorithm*. Computers & Geosciences, vol. 10, no. 2, pages 191 – 203, 1984. (Cited on page 144.)

[Bhat K. *et al.* 2008] Vivekananda Bhat K., Indranil Sengupta and Abhijit Das. Information systems security: 4th international conference, iciss 2008, hyderabad, india, december 16-20, 2008. proceedings, chapter Audio Watermarking Based on Quantization

in Wavelet Domain, pages 235–242. Springer Berlin Heidelberg, 2008. (Cited on pages 127, 129 and 130.)

[Blackburn *et al.* 2007] Simon R. Blackburn, Tuvi Etzion and Siaw-Lynn Ng. *Prolific Codes with the Identifiable Parent Property*. IACR Cryptology ePrint Archive, vol. 2007, page 276, 2007. (Cited on page 33.)

[Blakley *et al.* 1986] G. R. Blakley, C. Meadows and G. B. Purdy. Fingerprinting long forgiving messages, pages 180–189. Springer Berlin Heidelberg, Berlin, Heidelberg, 1986. (Cited on page 4.)

[Blayer & Tassa 2008] Oded Blayer and Tamir Tassa. *Improved versions of Tardos' fingerprinting scheme*. Designs, Codes and Cryptography, vol. 48, no. 1, pages 79–103, 2008. (Cited on pages 39 and 40.)

[Boneh & Shaw 1995] Dan Boneh and James Shaw. *Collusion-Secure Fingerprinting for Digital Data (Extended Abstract)*. In Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings, pages 452–465, 1995. (Cited on pages 23 and 27.)

[Boneh & Shaw 1998] Dan Boneh and James Shaw. *Collusion-Secure Fingerprinting for Digital Data*. IEEE Transactions on Information Theory, vol. 44, no. 5, pages 1897–1905, 1998. (Cited on pages 5, 23, 32, 34, 35, 41 and 67.)

[Boneh *et al.* 2010] Dan Boneh, Aggelos Kiayias and Hart William Montgomery. *Robust fingerprinting codes: a near optimal construction*. In Proceedings of the 10th ACM Workshop on Digital Rights Management, Chicago, Illinois, USA, October 4, 2010, pages 3–12, 2010. (Cited on page 31.)

[Cha & Kuo 2007] B. H. Cha and C. C. J. Kuo. *Design of Collusion-Free Hiding Codes using MAI-Free Principle*. In Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on, volume 2, pages II–145–II–148, April 2007. (Cited on page 44.)

[Chaabane *et al.* 2013] Faten Chaabane, Maha Charfeddine and Chokri Ben Amar. *A survey on digital tracing traitors schemes*. In 9th International Conference on Information

Assurance and Security, IAS 2013, Gammarth, Tunisia, December 4-6, 2013, pages 85–90, 2013. (Cited on pages 5, 8, 26, 27, 31 and 53.)

[Chaabane *et al.* 2014] Faten Chaabane, Maha Charfeddine and Chokri Ben Amar. *A Multimedia Tracing Traitors Scheme Using Multi-level Hierarchical Structure for Tardos Fingerprint Based Audio Watermarking*. In SIGMAP 2014 - Proceedings of the 11th International Conference on Signal Processing and Multimedia Applications, Vienna, Austria, 28-30 August, 2014, pages 289–296, 2014. (Cited on pages 8, 66, 69 and 70.)

[Chaabane *et al.* 2015a] Faten Chaabane, Maha Charfeddine and Chokri Ben Amar. *Clustering impact on group-based traitor tracing schemes*. In 15th International Conference on Intelligent Systems Design and Applications (ISDA), Marrakesh, Morocco, December 14-16, 2015, pages 440–445, 2015. (Cited on page 9.)

[Chaabane *et al.* 2015b] Faten Chaabane, Maha Charfeddine, William Puech and Chokri Ben Amar. *A QR-code based audio watermarking technique for tracing traitors*. In 23rd European Signal Processing Conference, EUSIPCO 2015, Nice, France, August 31 - September 4, 2015, pages 51–55, 2015. (Cited on page 8.)

[Chaabane *et al.* 2015c] Faten Chaabane, Maha Charfeddine, William Puech and Chokri Ben Amar. *Towards a Blind MAP-Based Traitor Tracing Scheme for Hierarchical Fingerprints*. In Neural Information Processing - 22nd International Conference, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015, Proceedings, Part IV, pages 505–512, 2015. (Cited on page 9.)

[Chaabane *et al.* 2016a] Faten Chaabane, Maha Charfeddine and Chokri Ben Amar. *Novel two-level tracing scheme using clustering algorithm.* Journal of Information Assurance & Security . 2016, Vol. 11 Issue 4, p179-189. 11p., 2016. (Cited on page 9.)

[Chaabane *et al.* 2016b] Faten Chaabane, Maha Charfeddine, William Puech and Chokri Ben Amar. *An EM-based estimation for a two-level traitor tracing scheme*. In The 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2016) , Budapest, October 2016, 2016. (Cited on page 9.)

[Chaabane *et al.* 2016c] Faten Chaabane, Maha Charfeddine, William Puech and Chokri Ben Amar. *A two-stage traitor tracing strategy for hierarchical fingerprints*. Multimedia Tools Appl., 2016. (Cited on pages 9, 99 and 100.)

[Charfeddine *et al.* 2010] Maha Charfeddine, Maher Elarbi, Mohamed Koubaa and Chokri Ben Amar. *DCT based Blind Audio Watermarking Scheme*. In SIGMAP, pages 139–144, 2010. (Cited on pages 18, 60, 70, 71, 83, 92, 94, 99 and 143.)

[Charfeddine *et al.* 2012] Maha Charfeddine, Maher Elarbi and Chokri Ben Amar. *A new DCT audio watermarking scheme based on preliminary MP3 study*. Multimedia Tools Appl., vol. 70, no. 3, pages 1521–1557, 2012. (Cited on pages 18, 64 and 80.)

[Charpentier *et al.* 2009] Ana Charpentier, Fuchun Xie, Caroline Fontaine and Teddy Furon. *Expectation maximization decoding of Tardos probabilistic fingerprinting code*. In Media Forensics and Security, page 72540, 2009. (Cited on pages 106, 108, 109 and 112.)

[Charpentier *et al.* 2011] Ana Charpentier, Caroline Fontaine, Teddy Furon and Ingemar J. Cox. *An Asymmetric Fingerprinting Scheme Based on Tardos Codes*. In Information Hiding, pages 43–58, 2011. (Cited on page 3.)

[Choi *et al.* 2012] Joonho Choi, Abu S. Reaz and Biswanath Mukherjee. *A Survey of User Behavior in VoD Service and Bandwidth-Saving Multicast Streaming Schemes*. IEEE Communications Surveys and Tutorials, vol. 14, no. 1, pages 156–169, 2012. (Cited on pages 6, 53 and 70.)

[Chor *et al.* 1994] Benny Chor, Amos Fiat and Moni Naor. *Tracing Traitors*. In Proceedings of the 14th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '94, pages 257–270, London, UK, UK, 1994. Springer-Verlag. (Cited on pages 38 and 110.)

[Cormen 2013] Thomas H. Cormen. Foundations of cryptography, pages 138–157. MIT Press, 2013. (Cited on page 2.)

[Cox *et al.* 1996] Ingemar J. Cox, Joe Kilian, Tom Leighton and Talal Shamoon. A secure, robust watermark for multimedia, pages 185–206. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996. (Cited on pages 17 and 43.)

[Cox *et al.* 1999] I. J. Cox, M. L. Miller and A. L. McKellips. *Watermarking as communications with side information*. Proceedings of the IEEE, vol. 87, no. 7, pages 1127–1141, Jul 1999. (Cited on page 17.)

[Craver *et al.* 2006] S. Craver, N. Memon, B. L. Yeo and M. M. Yeung. *Resolving Rightful Ownerships with Invisible Watermarking Techniques: Limitations, Attacks, and Implications*. IEEE J.Sel. A. Commun., vol. 16, no. 4, pages 573–586, September 2006. (Cited on page 18.)

[Cvejic *et al.* 2007] Nedeljko Cvejic, Nedeljko Cvejic and Tapio Seppanen. Digital audio watermarking techniques and technologies: Applications and benchmarks. IGI Global, Hershey, PA, USA, 2007. (Cited on page 133.)

[de Rosnay 2002] M. D. de Rosnay. *Digital rights management systems and European law: between copyright protection and access control*. In Web Delivering of Music, 2002. WEDELMUSIC 2002. Proceedings. Second International Conference on, pages 117–124, 2002. (Cited on page 2.)

[Desoubeaux *et al.* 2011] Mathiew Desoubeaux, Gaetan Le Guelvouit and William Puech. *Probabilistic Fingerprinting Codes Used To Detect Traitor Zero-bit Watermark*. In SPIE Proceedings Vol. 7880: Media Watermarking, Security, and Forensics III, 2011. (Cited on page 44.)

[Desoubeaux *et al.* 2012] Mathiew Desoubeaux, Gaetan Le Guelvouit and William Puech. *Fast detection of Tardos Codes with Boneh-Shaw Types*. In Proc. SPIE 8303, Media Watermarking, Security, and Forensics, 2012. (Cited on pages 7, 42, 64, 66, 67, 68, 73, 78, 88, 100, 101, 124 and 143.)

[Desoubeaux *et al.* 2013] Mathieu Desoubeaux, Cédric Herzet, William Puech and Gaëtan Le Guelvouit. *Enhanced blind decoding of tardos codes with new MAP-based functions*. In 15th IEEE International Workshop on Multimedia Signal Processing, MMSP 2013, Pula, Sardinia, Italy, September 30 - Oct. 2, 2013, pages 283–288, 2013. (Cited on pages 107, 110, 111, 116 and 119.)

[Desoubeaux 2013] Mathieu Desoubeaux. *Codes de traçage de traitres pour la protection de contenus numériques*. PhD thesis, 2013. ThÃÍse de doctorat dirigÃl'e par Puech, William Informatique Montpellier 2 2013. (Cited on pages 30, 39 and 43.)

[development] Lerch A (2002) Zplane development. *EAQUAL-Evaluate Audio QUALity, version: 0.1.3alpha*. http://www.mp3-tech.org/programmer/misc.html. (Cited on page 133.)

[Encheva & Cohen 2001] S. Encheva and G. Cohen. *Some new p-ary two-secure frameproof codes*. Applied Mathematics Letters, vol. 14, no. 2, pages 177 – 182, 2001. (Not cited.)

[Encheva & Cohen 2002] Sylvia Encheva and Gérard Cohen. *Frameproof Codes Against Limited Coalitions of Pirates*. Theor. Comput. Sci., vol. 273, no. 1-2, pages 295–304, February 2002. (Cited on page 32.)

[Fahad *et al.* 2014] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou and A. Bouras. *A Survey of Clustering Algorithms for Big Data: Taxonomy and Empirical Analysis*. IEEE Transactions on Emerging Topics in Computing, vol. 2, no. 3, pages 267–279, Sept 2014. (Cited on page 143.)

[Fiat & Tassa 1999] Amos Fiat and Tamir Tassa. *Dynamic Traitor Tracing*. In Michael Wiener, editor, Advances in Cryptology CRYPTO -99, volume 1666 of *Lecture Notes in Computer Science*, pages 354–371. Springer Berlin Heidelberg, 1999. (Cited on page 29.)

[Fontaine 2011] Caroline Fontaine. *How to protect multimedia pieces of content, from their creation to their distribution*. PhD thesis, Université de Bretagne Occidentale, école doctorale SICMA, 2011. (Cited on pages 3, 4, 25, 27 and 43.)

[Furon & Pérez-Freire 2009a] Teddy Furon and Luis Pérez-Freire. *EM Decoding of Tardos Traitor Tracing Codes*. In Proceedings of the 11th ACM Workshop on Multimedia and Security, MM&Sec '09, pages 99–106, New York, NY, USA, 2009. ACM. (Cited on pages 106 and 107.)

[Furon & Pérez-Freire 2009b] Teddy Furon and Luis Pérez-Freire. *Worst case attacks against binary probabilistic traitor tracing codes*. CoRR, vol. abs/0903.3480, 2009. (Cited on pages 23, 24, 25, 26, 41, 69, 78 and 146.)

[Furon *et al.* 2008] Teddy Furon, Arnaud Guyader and Frédéric Cérou. Information hiding: 10th international workshop, ih 2008, santa barbara, ca, usa, may 19-21, 2008, revised selected papers, chapter On the Design and Optimization of Tardos Probabilistic Fingerprinting Codes, pages 341–356. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. (Cited on pages 39 and 109.)

[Furon *et al.* 2009] Teddy Furon, Luis Pérez-Freire, Arnaud Guyader and Frédéric Cérou. *Estimating the Minimal Length of Tardos Code*. In Information Hiding, 11th International Workshop, IH 2009, Darmstadt, Germany, June 8-10, 2009, Revised Selected Papers, pages 176–190, 2009. (Cited on pages 39, 44, 85 and 116.)

[Guth & Pfitzmann 2000] Hans-Jürgen Guth and Birgit Pfitzmann. Error- and collusion-secure fingerprinting for digital data, pages 134–145. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000. (Cited on page 24.)

[Hamida *et al.* 2011] Amal Ben Hamida, Mohamed Koubàa and Henri Nicolas. *Hierarchical Traceability Of Multimedia Documents*. In Computational Intelligence in Cyber Security, pages 108–113, 2011. (Cited on pages 41, 53, 60, 61 and 146.)

[Hayashi *et al.* 2007] Naoki Hayashi, Minoru Kuribayashi and Masakatu Morii. *Collusion-Resistant Fingerprinting Scheme Based on the CDMA-Technique*. In IWSEC, pages 28–43, 2007. (Cited on pages 5, 23, 44, 45 and 53.)

[He & Wu 2005] Shan He and Min Wu. Performance study on multimedia fingerprinting employing traceability codes, pages 84–96. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. (Cited on pages 34 and 43.)

[He & Wu 2007] Shan He and Min Wu. *Collusion-Resistant Video Fingerprinting for Large User Group*. IEEE Transactions on Information Forensics and Security, vol. 2, no. 4, pages 697–709, 2007. (Cited on pages 5, 31, 44, 146 and 148.)

[Johnson 1967] StephenC. Johnson. *Hierarchical clustering schemes*. Psychometrika, vol. 32, no. 3, pages 241–254, 1967. (Cited on page 144.)

[Keiler 2006] Florian Keiler. *Real-Time Subband-ADPCM Low-Delay Audio Coding Approach*. In Audio Engineering Society Convention 120, May 2006. (Cited on page 133.)

[Kiayias & Yung 2001] Aggelos Kiayias and Moti Yung. *On Crafty Pirates and Foxy Tracers*. In CCS 2001, Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, Pennsylvania, USA, November 6-8, 2001., pages 22–39, 2001. (Cited on page 34.)

[Kilian *et al.* 1998] J. Kilian, F. T. Leighton, L. R. Matheson, T. G. Shamoon, R. E. Tarjan and F. Zane. *Resistance of digital watermarks to collusive attacks*. In Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on, pages 271–, Aug 1998. (Cited on page 43.)

[Kumar *et al.* 2011] N.M. Kumar, T. Manikandan and V. Sapthagirivasan. *Non blind image watermarking based on similarity in contourlet domain*. In Recent Trends in Information Technology (ICRTIT), 2011 International Conference on, pages 1277–1282, June 2011. (Cited on page 18.)

[Kuribayashi *et al.* 2008] M. Kuribayashi, N. Akashi and M. Morii. *On the systematic generation of Tardos 's fingerprinting codes*. In Multimedia Signal Processing, 2008 IEEE 10th Workshop on, pages 748–753, Oct 2008. (Cited on pages 7, 42, 43, 45, 66, 73, 78, 99 and 100.)

[Kuribayashi 2012] M. Kuribayashi. *Adaptive iterative detection method for spread spectrum fingerprinting scheme*. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1777–1780, March 2012. (Cited on page 44.)

[Kuribayashi 2014] M. Kuribayashi. *Simplified MAP Detector for Binary Fingerprinting Code Embedded by Spread Spectrum Watermarking Scheme*. IEEE Transactions on Information Forensics and Security, vol. 9, no. 4, pages 610–623, April 2014. (Cited on pages 6 and 107.)

[Laarhoven & de Weger 2011] Thijs Laarhoven and Benne de Weger. *Optimal symmetric Tardos traitor tracing schemes*. CoRR, vol. abs/1107.3441, 2011. (Cited on pages 37, 40, 41 and 55.)

[Laarhoven *et al.* 2013] Thijs Laarhoven, Jeroen Doumen, Peter Roelse, Boris Skoric and Benne de Weger. *Dynamic Tardos Traitor Tracing Schemes*. IEEE Trans. Information Theory, vol. 59, no. 7, pages 4230–4242, 2013. (Cited on page 29.)

[Laftsidis *et al.* 2003] C. Laftsidis, A. Tefas, N. Nikolaidis and I. Pitas. *Robust multibit audio watermarking in the temporal domain*. In Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on, volume 2, pages II–944–II–947 vol.2, May 2003. (Cited on page 18.)

[Lanxun *et al.* 2007] W. Lanxun, Y. Chao and P. Jiao. *An Audio Watermark Embedding Algorithm Based on Mean-Quantization in Wavelet Domain*. In Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on, pages 2–423–2–425, Aug 2007. (Cited on pages 127, 129 and 130.)

[Lie & Chang 2006] Wen-Nung Lie and Li-Chun Chang. *Robust and high-quality time-domain audio watermarking based on low-frequency amplitude modification*. Multimedia, IEEE Transactions on, vol. 8, no. 1, pages 46–59, Feb 2006. (Cited on page 18.)

[Lin *et al.* 2009] W. Sabrina Lin, Shan He and Jeffrey Bloom. *Performance Study and Improvement on ECC-based Binary Anti-collusion Forensic Code for Multimedia*. In Proceedings of the 11th ACM Workshop on Multimedia and Security, MM&Sec '09, pages 93–98, New York, NY, USA, 2009. ACM. (Cited on page 38.)

[Liu *et al.* 2014] Ning Liu, Huajie Cui, S.-H. Gary Chan, Zhipeng Chen and Yirong Zhuang. *Dissecting User Behaviors for a Simultaneous Live and VoD IPTV System*. TOMCCAP, vol. 10, no. 3, page 23, 2014. (Cited on pages 52 and 53.)

[Liu 2005] K.J.R. Liu. Multimedia fingerprinting forensics for traitor tracing. EURASIP book series on signal processing and communications. Hindawi Publishing Corporation, 2005. (Cited on pages 5 and 50.)

[Ma & Ding 2006] Y. Ma and Y. Ding. *Reed-Solomon Codes as Traceability Codes with an Efficient Tracing Algorithm*. In 2006 8th international Conference on Signal Processing, volume 4, 2006. (Cited on page 38.)

[Martinez-Noriega *et al.* 2010] R. Martinez-Noriega, M. Nakano and K. Yamaguchi. *Self-Synchronous Time-Domain Audio Watermarking Based on Coded-Watermarks*. In Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on, pages 135–138, Oct 2010. (Cited on page 18.)

[Mathon *et al.* 2013] B. Mathon, P. Bas, F. Cayre and B. Macq. *Impacts of Watermarking Security on Tardos-Based Fingerprinting*. Information Forensics and Security, IEEE Transactions on, vol. 8, no. 6, pages 1038–1050, June 2013. (Cited on pages 23, 25, 83 and 84.)

[Meerwald & Furon 2012] Peter Meerwald and Teddy Furon. *Toward Practical Joint Decoding of Binary Tardos Fingerprinting Codes*. IEEE Transactions on Information Forensics and Security, vol. 7, no. 4, pages 1168–1180, 2012. (Cited on pages 40, 69 and 77.)

[Nematollahi *et al.* 2012] M.A. Nematollahi, S.A.R. Al-Haddad, S. Doraisamy and M.I.B. Saripan. *Digital Audio and Speech Watermarking Based on the Multiple Discrete Wavelets Transform and Singular Value Decomposition*. In Modelling Symposium (AMS), 2012 Sixth Asia, pages 109–114, May 2012. (Cited on page 18.)

[Neubauer & Herre 1998] C. Neubauer and J. Herre. *Digital Watermarking and Its Influence on Audio Quality*. In 105th AES Convention, San Francisco, 1998. Preprint 4823. (Cited on page 80.)

[Nuida *et al.* 2007] Koji Nuida, Satoshi Fujitsu, Manabu Hagiwara, Takashi Kitagawa, Hajime Watanabe, Kazuto Ogawa and Hideki Imai. *An Improvement of Tardos's Collusion-secure Fingerprinting Codes with Very Short Lengths*. In Proceedings of the 17th International Conference on Applied Algebra, Algebraic Algorithms and Error-correcting Codes, AAECC'07, pages 80–89, 2007. (Cited on pages 40 and 44.)

[Panda *et al.* 2012] Sandeep Panda, Sanat Sahu, Pradeep Jena and Subhagata Chattopadhyay. *Comparing Fuzzy-C Means and K-Means Clustering Techniques: A Comprehensive Study*. In David C. Wyld, Jan Zizka and Dhinaharan Nagamalai, editors, Advances in Computer Science, Engineering & Applications, volume 166 of *Advances in Intelligent*

and Soft Computing, pages 451–460. Springer Berlin Heidelberg, 2012. (Cited on page 144.)

[Peikert *et al.* 2003] Chris Peikert, Abhi shelat and Adam Smith. *Lower Bounds for Collusion-secure Fingerprinting*. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 472–479, 2003. (Cited on pages 5, 36 and 42.)

[Pérez-Freire & Furon 2009] Luis Pérez-Freire and Teddy Furon. *BLIND DECODER FOR BINARY PROBABILISTIC TRAITOR TRACING CODES*. In IEEE International Workshop on Information Forensics and Security, London, United Kingdom, December 2009. (Cited on page 107.)

[Petitcolas *et al.* 1998] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn. *Attacks on Copyright Marking Systems*. In Proceedings of the Second International Workshop on Information Hiding, pages 218–238, London, UK, UK, 1998. Springer-Verlag. (Cited on page 85.)

[Pfitzmann & Schunter 1996] Birgit Pfitzmann and Matthias Schunter. Advances in cryptology — eurocrypt '96: International conference on the theory and application of cryptographic techniques saragossa, spain, may 12–16, 1996 proceedings, chapter Asymmetric Fingerprinting, pages 84–95. Springer Berlin Heidelberg, 1996. (Cited on page 29.)

[Pfitzmann & Waidner 1997] Birgit Pfitzmann and Michael Waidner. Advances in cryptology — eurocrypt '97: International conference on the theory and application of cryptographic techniques konstanz, germany, may 11–15, 1997 proceedings, chapter Anonymous Fingerprinting, pages 88–102. Springer Berlin Heidelberg, 1997. (Cited on page 30.)

[Pradhan *et al.* 2012] Chittaranjan Pradhan, Shibani Rath and Ajay Kumar Bisoi. *Non Blind Digital Watermarking Technique Using {DWT} and Cross Chaos*. Procedia Technology, vol. 6, no. 0, pages 897 – 904, 2012. 2nd International Conference on Communication, Computing & Security [ICCCS-2012]. (Cited on page 18.)

[Qureshi *et al.* 2015] Amna Qureshi, David MegÃ■as and Helena RifÃă-Pous. *Framework for preserving security and privacy in peer-to-peer content distribution systems*. Expert Systems with Applications, vol. 42, no. 3, pages 1391 – 1408, 2015. (Cited on page 44.)

[Saha *et al.* 2014] Bidyut Jyoti Saha, Arun, Kunal Kumar Kabi and Chittaranjan Pradhan. *Non blind watermarking technique using enhanced one time pad in DWT domain*. In Computing, Communication and Networking Technologies (ICCCNT), 2014 International Conference on, pages 1–6, July 2014. (Cited on page 18.)

[Schaathun 2008] Hans Georg Schaathun. *On error-correcting fingerprinting codes for use with watermarking*. Multimedia Systems, vol. 13, no. 5, pages 331–344, 2008. (Cited on page 38.)

[Schäfer *et al.* 2010] Marcel Schäfer, Waldemar Berchtold, Sascha Zmudzinski and Martin Steinebach. *Zero False Positive 2-secure Fingerprinting Watermarking Based on Combining Hamming Distance Conditions and Parent Pair Search*. In Proceedings of the 12th ACM Workshop on Multimedia and Security, MM&Sec '10, pages 169–174. ACM, 2010. (Cited on pages 29 and 38.)

[Schäfer 2016] Marcel Schäfer. *Collusion Secure Fingerprint Watermarking*. PhD thesis, Technische Universität Darmstadt, Darmstadt, 2016. (Cited on pages 28, 30, 34, 35, 39 and 124.)

[Shahid *et al.* 2013] Zafar Shahid, Marc Chaumont and William Puech. *H.264/AVC video watermarking for active fingerprinting based on Tardos code*. Signal, Image and Video Processing, vol. 7, no. 4, pages 679–694, 2013. (Cited on page 6.)

[Simone & Skoric 2011] Antonino Simone and Boris Skoric. Asymptotically false-positive-maximizing attack on non-binary tardos codes, pages 14–27. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. (Cited on page 40.)

[Simone & Škorić 2012] Antonino Simone and Boris Škorić. *Accusation probabilities in Tardos codes: beyond the Gaussian approximation*. Designs, Codes and Cryptography, vol. 63, no. 3, pages 379–412, 2012. (Cited on page 35.)

[Skoric *et al.* 2007] Boris Skoric, Stefan Katzenbeisser and Mehmet Utku Celik. *Symmetric Tardos fingerprinting codes for arbitrary alphabet sizes*. IACR Cryptology ePrint Archive, vol. 2007, page 41, 2007. (Cited on pages 35, 37, 39, 40, 106, 108, 110, 112 and 119.)

[Staddon *et al.* 2001] J. N. Staddon, D. R. Stinson and Ruizhong Wei. *Combinatorial properties of frameproof and traceability codes*. IEEE Transactions on Information Theory, vol. 47, no. 3, pages 1042–1049, Mar 2001. (Cited on pages 32 and 34.)

[Tardos 2003] Gábor Tardos. *Optimal probabilistic fingerprint codes*. In STOC, pages 116–125, 2003. (Cited on pages 5, 34, 36, 39, 42, 73, 99, 100, 106 and 113.)

[Tassa 2005] Tamir Tassa. *Low Bandwidth Dynamic Traitor Tracing Schemes*. Journal of Cryptology, vol. 18, no. 2, pages 167–183, 2005. (Cited on pages 41 and 42.)

[Thiede *et al.* 2000] Thilo Thiede, William C. Treurniet, Roland Bitto, Christian Schmidmer, Thomas Sporer, John G. Beerends and Catherine Colomes. *PEAQ - The ITU Standard for Objective Measurement of Perceived Audio Quality*. J. Audio Eng. Soc, vol. 48, no. 1/2, pages 3–29, 2000. (Cited on page 133.)

[Thilagavathi *et al.* 2015] N. Thilagavathi, D. Saravanan, S. Kumarakrishnan, Sakthivel Punniakodi, J. Amudhavel and U. Prabu. *A Survey of Reversible Watermarking Techniques, Application and Attacks*. In Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering ; Technology (ICARCSET 2015), ICARCSET '15, pages 37:1–37:7, 2015. (Cited on page 3.)

[Tomas-Buliart *et al.* 2008] Joan Tomas-Buliart, Marcel Fernandez and Miguel Soriano. New considerations about the correct design of turbo fingerprinting codes, pages 501–516. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. (Cited on page 38.)

[Tomas-Buliart *et al.* 2009] Joan Tomas-Buliart, Marcel Fernandez and Miguel Soriano. Improvement of collusion secure convolutional fingerprinting information codes, pages 76–88. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. (Cited on page 38.)

[Trappe *et al.* 2003] Wade Trappe, Min Wu, Z. Jane Wang and K. J. Ray Liu. *Anti-collusion fingerprinting for multimedia*. IEEE Transactions on Signal Processing, vol. 51, no. 4, pages 1069–1087, 2003. (Cited on pages 5, 25, 31 and 145.)

[Wagner 1983] Neal R. Wagner. *Fingerprinting*. In Proceedings of the 1983 IEEE Symposium on Security and Privacy, pages 18–, Washington, DC, USA, 1983. (Cited on pages 4, 5, 22, 27, 28 and 29.)

[Walia & Suneja 2013] E. Walia and A. Suneja. *Fragile and blind watermarking technique based on Weber's law for medical image authentication*. Computer Vision, IET, vol. 7, no. 1, pages 9–19, February 2013. (Cited on page 18.)

[Wang *et al.* 2003] Z. J. Wang, Min Wu, Hong Zhao, K. J. R. Liu and W. Trappe. *Resistance of orthogonal Gaussian fingerprints to collusion attacks*. In Multimedia and Expo, 2003. ICME '03. Proceedings. 2003 International Conference on, volume 1, pages I–617–20 vol.1, July 2003. (Cited on page 43.)

[Wang *et al.* 2004] Z. Jane Wang, Min Wu, Wade Trappe and K. J. Ray Liu. *Group-Oriented Fingerprinting for Multimedia Forensics*. EURASIP J. Adv. Sig. Proc., vol. 2004, no. 14, pages 2153–2173, 2004. (Cited on pages 5, 6, 40, 41, 44, 53, 60, 61, 143, 146 and 148.)

[Wang *et al.* 2005] Z Jane Wang, Min Wu, Hong Vicky Zhao, Wade Trappe and KJ Ray Liu. *Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation*. Image Processing, IEEE Transactions on, vol. 14, no. 6, pages 804–821, 2005. (Cited on page 43.)

[Wang *et al.* 2011] Jian Wang, Ron Healy and Joe Timoney. *A robust audio watermarking scheme based on reduced singular value decomposition and distortion removal*. Signal Processing, vol. 91, no. 8, pages 1693 – 1708, 2011. (Cited on pages 127, 129 and 130.)

[Wu & Liew 2012] Xin-Wen Wu and AlanWee-Chung Liew. *Near-Optimal Collusion-Secure Fingerprinting Codes for Efficiently Tracing Illegal Re-distribution*. In Cyberspace Safety and Security, volume 7672, pages 352–361. Springer Berlin Heidelberg, 2012. (Cited on page 42.)

[Xie *et al.* 2008] Fuchun Xie, Teddy Furon and Caroline Fontaine. *On-off keying modulation and tardos fingerprinting*. In MM&Sec, pages 101–106, 2008. (Cited on page 18.)

[Xie 2010] Fuchun Xie. *Robust and Secure Watermarking for Multimedia Traitor Tracing*. Theses, Université Rennes 1, September 2010. (Cited on page 34.)

[Xu *et al.* 2007] Xiaojuan Xu, Hong Peng and Chengyuan He. Applications of fuzzy sets theory: 7th international workshop on fuzzy logic and applications, wilf 2007, camogli, italy, july 7-10, 2007. proceedings, chapter DWT-Based Audio Watermarking Using Support Vector Regression and Subsampling, pages 136–144. Springer Berlin Heidelberg, 2007. (Cited on page 129.)

[Yagi *et al.* 2007] Hideki Yagi, Toshiyasu Matsushima and Shigeichi Hirasawa. New traceability codes against a generalized collusion attack for digital fingerprinting, pages 252–266. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. (Cited on page 32.)

[Ye *et al.* 2013] Conghuan Ye, Hefei Ling, Fuhao Zou and Zhengding Lu. *A new fingerprinting scheme using social network analysis for majority attack*. Telecommunication Systems, vol. 54, no. 3, pages 315–331, 2013. (Cited on pages 6, 18, 41, 60, 61 and 146.)

[Yong *et al.* 2014] Zhao Yong, Zhang Aixin and Lu Songnian. *DCT fingerprint classifier based group fingerprint*. In Audio, Language and Image Processing (ICALIP), 2014 International Conference on, pages 292–295, July 2014. (Cited on pages 60, 143 and 148.)

[Yu & Sattar 2003] Dan Yu and Farook Sattar. *A New Blind Watermarking Technique Based on Independent Component Analysis*. In Proceedings of the 1st International Conference on Digital Watermarking, pages 51–63, Berlin, Heidelberg, 2003. (Cited on page 18.)

[Zeng & Qiu 2008] Gaorong Zeng and Z. Qiu. *Audio watermarking in DCT: Embedding strategy and algorithm*. In 2008 9th International Conference on Signal Processing, pages 2193–2196, Oct 2008. (Cited on page 18.)

[Zhu *et al.* 2006] Yan Zhu, Wei Zou and Xinshan Zhu. *Collusion Secure Convolutional Fingerprinting Information Codes*. In Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, ASIACCS '06, pages 266–274, New York, NY, USA, 2006. ACM. (Cited on page 38.)