

Security Issues in Watermarking Applications - A Deeper Look

(An Extended Abstract)

Qiming Li
Computer and Information
Science Department
Polytechnic University
Brooklyn, NY 11201
qiming@isis.poly.edu

Nasir Memon
Computer and Information
Science Department
Polytechnic University
Brooklyn, NY 11201
memon@poly.edu

Husrev T. Sencar
Computer and Information
Science Department
Polytechnic University
Brooklyn, NY 11201
taha@isis.poly.edu

ABSTRACT

Although it is clear that security is an important issue in digital watermarking applications, the main concerns addressed by the current literature are robustness, capacity and imperceptibility. The inadequacy of the prevailing design paradigm in tackling security issues is mainly due to an incomplete assessment of the threat model. The goal of this paper is to take a detailed and rigorous look at the threat model for a variety of watermarking applications. In this extended abstract, we outline the security requirements for a few common watermarking applications and explore in more detail the threat model for a specific application that involves establishing ownership of multimedia content.

Categories and Subject Descriptors

H.5.1 [Information Systems Applications]: Information Interfaces and Presentation—*multimedia information systems*

General Terms

Data Hiding

1. INTRODUCTION

Data hiding is a form of communication where information is conveyed by embedding it in a *cover object*, (e.g., image, video, text, analog/digital waveform, software program, hardware design, etc). Essentially, this is achieved by designing embedding and detection functions that exploit the inherent redundancies of the cover, assuming the presence of a noisy channel. The formulation, design and analysis of data hiding problem has been mostly based on three main criteria: *robustness*, *imperceptibility*, and *capacity*. Robustness characterizes the ability to reliably extract the embedded information from an unintentionally or maliciously modified version of the embedded cover. Imper-

ceptibility refers to maintaining salient (often perceptual) properties and functionality of the cover following embedding. Capacity is the amount of information that can be embedded in and extracted from the cover. These goals are conflicting in nature, and a proper trade-off among them has to be made depending on the requirements of the specific application.

Although data hiding techniques are applicable to vast array of applications, they have been primarily used in multimedia applications (e.g., ownership and copyright protection, authentication, fingerprinting, closed captioning, steganography, etc) wherein the embedder and detector has to cope with an intelligent attacker. In this context data hiding is also commonly referred to as digital watermarking. In the rest of this paper we restrict ourselves to digital watermarking techniques and applications.

Although it is clear that security is an important issue in digital watermarking applications, the main concern addressed by the current literature again has been the robustness of the embedded watermark against possible malicious attacks. In other words, the main determinant of the viability of a watermarking system has been the achievable degree of robustness against a very capable attacker who aims at removing the embedded watermark. However, due to lack of security considerations in the design process, most proposed watermarking systems fall short of achieving their intended goals even if the robustness issue were completely resolved.

For example, in ownership and copyright protection applications, the proposed watermarking based systems have been primarily defeated by a much less intrusive class of attacks that aim at creating confusion at the detector rather than removing the watermark. Similarly, in authentication and fingerprinting applications, possibility of collisions has been the main obstacle facing watermarking techniques. The inadequacy of the prevailing design paradigm in tackling security issues is mainly due to an incomplete assessment of the threat model, which requires a thorough evaluation of how the attacker can impede the goals and purpose of watermarking. Therefore, in addition to fulfilling robustness, imperceptibility and payload requirements, the design of watermarking techniques needs to also address the inherent security issues as defined by the underlying application.

Many of the problems studied in the context of information security and cryptography are similar to the ones encountered in watermarking applications, and it is imperative for watermarking techniques to merge signal processing methods with the vast body of knowledge in these fields. While rigorous threat modeling is a standard practice in cryptography, it can be very challenging in the context of watermarking, where it is common (and in most cases necessary) to tolerate certain noise in the data. In many scenarios, the security of a given scheme depends not only on the cryptographic primitives employed in the protocols, but also on the basic characteristics of the *underlying* signal processing techniques, such as the robustness against certain types of noise, false positives, and so on. It becomes even more subtle when cryptographic primitives are built into signal processing procedures. For example, instead of having arbitrary real watermark sequences, we may require the watermarks to be computed from some discrete cryptographic functions, which essentially imposes additional constraints on the choice of watermarks. In this case, the robustness, imperceptibility and false positives of the scheme can be inevitably affected, which, in turn, will affect the overall security of the system. These effects should be taken into consideration when designing secure systems, and sometimes it can be very difficult to analyze the resulting schemes.

With this perspective, in the Section 2, we discuss various security issues that have to be incorporated in the design and development of watermarking systems. To validate our point of view, in Section 3, we evaluate different approaches to the ownership protection application as an illustrative case study. In the full version of this paper, we will extend these concepts to other applications of watermarking in a more formal framework.

2. SECURITY ISSUES IN MULTIMEDIA WATERMARKING

Due to the nature of multimedia objects, security issues in many application scenarios are often interleaved with signal processing issues. As a result, in many previous works, security issues in multimedia watermarking are not treated in a rigorous manner. In particular, security requirements are often stated in an imprecise way using natural language, and attacker models are often too simplistic. This makes it very difficult to assess the security of the schemes when, in real life, the attackers are very creative intelligent.

In this section we study security issues in multimedia watermarking, and we illustrate how rigorous definitions of security can be formulated for certain applications, and the attackers can be modeled to reflect their complex nature. It should be noted that we try to derive these formulations in a general way so that they can be applied to many scenarios, and as a result they may not capture enough details of the actual applications. Hence, to apply our formulations to real problems, it would be necessary to make some adjustments. Nevertheless, our main focus here is to illustrate the process of translating intuitive but vague statements in natural languages to precise and formal mathematical definitions.

2.1 A General Watermarking Model

We define a *work* to be a vector $I = (x_1, x_2, \dots, x_n)$ where each $x_i \in \mathcal{U}$ for some universe \mathcal{U} that is determined by the representation of the signal. We assume that there is a function $\text{Dist}(\cdot, \cdot)$ that measures the perceptual distance between two works. A *watermark* W is a sequence in \mathcal{W}^n , where \mathcal{W} is a domain determined by both the signal and the watermark generation process. A *key* K is a sequence of m binary bits.

In our general watermarking model, there are three algorithms, a watermark generator \mathcal{G} , a watermark embedder \mathcal{E} , and a watermark detector \mathcal{D} . The watermark generator $G : \{0, 1\}^m \times \{0, 1\}^* \rightarrow \mathcal{W}^n$ is a (randomized) polynomial algorithm that outputs a watermark given a key. We say that a watermark W is *valid* if and only if it is generated from some key K by \mathcal{G} (i.e., $W = \mathcal{G}(K)$ for some K). The embedder \mathcal{E} takes an original work I and a watermark W and outputs a watermarked work \tilde{I} (denoted as $\tilde{I} = \mathcal{E}(I, W)$). Given a work \tilde{I} and a watermark W , the detector \mathcal{D} declares whether W is embedded in \tilde{I} (i.e., $\mathcal{D}(\tilde{I}, W) = 1$), or not ($\mathcal{D}(\tilde{I}, W) = 0$).

Note that there can be many variants to the above watermarking model. For example, the detection may require the original I and may not require a watermark W , and its output may be a string instead of just one bit. The embedding process could employ another embedding key and other information. Nevertheless, the above simple model suits our needs to illustrate rigorous treatment of the security. Other variations can be adapted with minor modifications to our definitions and analysis.

2.2 Threat Model

To understand the security issues in multimedia watermarking and to design secure schemes, the most important task is to define what *security* is in the application scenarios. In a particular application, our security concerns usually consist of two parts: (1) The security requirements (i.e., the goals we want to achieve), and (2) the attacker model (i.e., the type of attackers we are dealing with). In essence, when we say that *a system is secure* we should clearly mean that the system is able to meet some *security requirements* in the presence of certain type of *attackers*. We will refer to these two parts as the *threat model* of the application under consideration.

2.2.1 Security Requirements

Based on security requirements, we can roughly put watermarking applications into the following broad categories, and we illustrate how to define the security of schemes with respect to a relatively simple attacker \mathcal{A} , which is always a probabilistic polynomial algorithm (we will discuss more complex attacker models in Section 2.2.2).

Ownership Proof. In these applications, the data hidden in the cover is used to assert the ownership of the cover work, or to resolve a dispute over the ownership. Typically we require that an attacker cannot (with high probability) hinder some predefined ownership proving process without rendering the cover work useless. For example, an attacker should not be able to remove the watermark easily.

DEFINITION 1 *A watermarking scheme is t -resistant to re-*

removal attacks if for any attacker \mathcal{A} and given any cover work \tilde{I} watermarked by W , it is computationally infeasible for \mathcal{A} to compute any work I' such that $\text{Dist}(\tilde{I}, I') < t$ and $\mathcal{D}(I', W) = 0$.

The phrase *computationally infeasible* here follows the standard definition in cryptography. That is, it is computationally infeasible for \mathcal{A} to compute I' given any \tilde{I} if and only if $\Pr[\mathcal{A}(\tilde{I}) = I']$ is negligible w.r.t. n . Note that a quantity X is negligible w.r.t. n if and only if for all sufficiently large n and any fixed polynomial $q(\cdot)$, we have $X < 1/q(n)$.

A stronger requirement is that an attacker should not be able to prevent the legitimate owner from proving his/her ownership of the work (by creating an ambiguity about the ownership).

DEFINITION 2 A watermarking scheme is resistant to ambiguity attacks if for any attacker \mathcal{A} and any cover work \tilde{I} , it is computationally infeasible for \mathcal{A} to compute a valid watermark W such that $\mathcal{D}(\tilde{I}, W) = 1$.

Although watermarking schemes that are resistant to removal and ambiguity attacks may be sufficient to resolve disputes over ownership of works, it is insufficient for *ownership assertion* where the original creator may be absent. In this case we would need a trusted third party. We will give more details in this scenario in Section 3.

In some scenarios, the owner of the work may want to prevent attackers from declaring arbitrary work as created by the owner by copying the watermark contained in an authentic cover work to the target. This is often referred to as the *copy attack*.

DEFINITION 3 A watermarking scheme is t -resistant to copy attacks if for any attacker \mathcal{A} and any cover work $\tilde{I} = \mathcal{E}(I, W)$ for some original I and watermark W , it is computationally infeasible for \mathcal{A} to compute a work I' such that $\text{Dist}(I, I') > t$, yet $\mathcal{D}(I', W) = 1$.

Fingerprinting. Hidden marks have been used to trace illegal redistribution of copyrighted material for centuries. The main concern here is that the legitimate copyright owner should be able to *identify* the source of illegal copies with a tracing algorithm \mathcal{T} , even when some of the attackers collude. Formally, let $S = \{I_1, \dots, I_k\}$ be a set of k works fingerprinted by the scheme, and let coalition C be a set of any c works from S .

DEFINITION 4 A fingerprinting scheme is (c, t) -collusion resistant if for any coalition C and for any work $J = \mathcal{A}(C)$ such that $\text{Dist}(J - I) < t$ for some work $I \in S$ and threshold t , there exists an efficient algorithm \mathcal{T} such that $\mathcal{T}(J) \in C$.

In some cases, we only require that the tracing algorithm to succeed with high probability.

DEFINITION 5 A fingerprinting scheme is (c, t, ϵ) -collusion resistant if for any coalition C and for any work $J = \mathcal{A}(C)$

such that $\text{Dist}(J - I) < t$ for some work $I \in S$ and threshold t , there exists an efficient algorithm \mathcal{T} such that $\Pr[\mathcal{T}(J) \in C] > 1 - \epsilon$.

Sometimes a weaker notion of security may be sufficient, where we only require that the colluded attackers cannot frame other users.

DEFINITION 6 A fingerprinting scheme is (c, t) -frameproof if given any coalition C , it is computationally infeasible for any attacker \mathcal{A} to compute a work J such that $\text{Dist}(J - I) < t$ for some work $I \in S \setminus C$ and threshold t .

There can be other security requirements. For instance, it may be desirable for the copyright owner to *prove* the source of such illegal redistribution to a third party (say, a judge). In some scenarios, the legitimate distribution process involves multiple entities, and the copyright owner may want to be sure that these entities are honest.

Authentication. Applications in this category usually concerns about the *integrity* and the *originality* of the work. That is, a receiver of the cover work should be convinced that it is created by a certain entity, and that it has not been tampered with. At the same time, it may be desirable to tolerate certain amount of noise that could appear in the communication channels. We assume that there is an efficient verification algorithm \mathcal{V} that takes a received work J and some auxiliary information K (e.g., a key) as the input and outputs either a **yes** when J is considered authentic or a **no** when J is not authentic.

DEFINITION 7 An authentication scheme is $(t_1, \epsilon_1, t_2, \epsilon_2)$ -secure if for any authentic work I , the following conditions hold:

1. For any random I' such that $\text{Dist}(I - I') \leq t_1$,

$$\Pr[\mathcal{V}(I', K) = \text{yes}] \geq 1 - \epsilon_1.$$

2. For any attacker \mathcal{A} and $J = \mathcal{A}(I)$ such that $\text{Dist}(I - I') > t_2$,

$$\Pr[\mathcal{V}(J, K) = \text{yes}] < \epsilon_2.$$

When $t_1 = t_2 = 0$, it becomes similar to traditional authentication without any tolerance of noise, and any modification will be considered as tampering. Also, the distance function Dist could depend on features extracted from the cover work. In that case it is sometimes referred to as *content-based authentication*. Furthermore, although we put ϵ_1 and ϵ_2 as constants here, in some applications we may require them to be negligible quantities.

Besides integrity and originality, sometimes we may have additional requirements on the *deniability*. In some cases, we need to make sure that the sender of the work cannot deny sending it to the receiver (sometimes referred to as *non-repudiation*). In other cases, it may be desirable for the sender to be able to deny it (*plausible deniability*).

2.2.2 Attacker Model

While it is relatively easier to find the right security requirements, it is more subtle to define what kind of attackers we want to deal with. The failure to model attackers properly usually results in schemes that are later found to be insecure. There are many such examples in the literature. For example, in an ownership proof scenario, one might easily assume that to create an ambiguity about the ownership, an attacker *has to* find a fake watermark first and compute the fake original later. In that case, the resulting system can be secure if the attacker follows this assumption, but nothing can be claimed otherwise. Although this seems to be a natural assumption at the first glance, it is proved to be too strong and lead to insecure schemes, as we will see in Section 3.

In general, we will not be able to predict what kind of algorithms the attackers will use in the future. Therefore, we need *rigorous treatment* to these attackers. That is, instead of making assumptions about the algorithms the attackers will use, the resulting system would be far more convincing if we only make assumptions about the computational capabilities of the attackers. For example, we should describe the attackers as whether they have limited or unlimited computing power, whether they have the access to just one cover work or multiple cover works, whether they have the access to a watermark embedding oracle and/or a watermark detection oracle, and if they do have the access to some oracle, whether they can only make one batch of queries, or they can make queries adaptively based on the answers returned by the oracle, and so on and so forth.

The definitions given in Section 2.2.1 assumes that the attacker has limited computing power (i.e., polynomial time algorithm), and has access to only one work (e.g., in ownership proof), or a fixed number of works (e.g., in fingerprinting). If this assumption does not hold, the definitions need to be adapted accordingly. For example, when we consider the security against ambiguity attacks where the attacker has the access to multiple watermarked works, we would have the following definition of security.

DEFINITION 8 *A watermarking scheme is resistant to ambiguity attacks for multiple works if for any attacker \mathcal{A} , any fixed polynomial $q(\cdot)$ and any $q(n)$ cover works $(\tilde{I}_1, \dots, \tilde{I}_{q(n)})$, it is computationally infeasible for \mathcal{A} to compute a valid watermark W and a j ($1 \leq j \leq q(n)$) such that $\mathcal{D}(\tilde{I}_j, W) = 1$.*

From signal processing point of view, attackers can also be categorized by the type of signal processing operations that they can perform. For example, a naive attacker can only add white noise to the signal, others may compress the signal, perform geometric distortions, or pass it through some filter. This categorization is particularly important in the formulation of security against removal attacks, since it is extremely difficult to design schemes resistant against all possible signal processing operations, and the formulation would not be very meaningful if no scheme can satisfy it in practice. Hence, instead of using Definition 1, we may want to use the following definition, where \mathcal{A}_S is an attacker who can only perform the types of signal processing operations described by S .

DEFINITION 9 *A watermarking scheme is (t, S) -resistant to removal attacks if for any attacker \mathcal{A}_S and given any cover work \tilde{I} watermarked by W , it is computationally infeasible for \mathcal{A}_S to compute any work I' such that $\text{Dist}(\tilde{I}, I') < t$ and $\mathcal{D}(I', W) = 0$.*

The above definitions only serve as examples. Since there are many different combinations of the attacker types, it would be impractical to list all of them in this extended abstract. We will give more examples of different attacker models in our full paper.

3. OWNERSHIP APPLICATION

Based on the definitions given in Section 2, in this section, we briefly describe the attack model for ownership applications and identify the requirements of watermarking based ownership system to withstand such attacks. The approaches to establishing rightful ownership over digital works deploy watermarking techniques as a tool to generate a protected version of the original work so that even its variants can be traced to the owner of the work. In this regard, an ownership protection system has 3 essential components, namely, watermark generation, embedding, and detection. The first two components need to be initiated by the owner when an original work is created and their operation might also involve a trusted third party. The third one, on the other hand, involves the ownership claimant(s) and may also require the engagement of a trusted third party in establishing ownership of a work.

In this extended abstract, we will rely on the following generic setting; however, in the full version of this paper, we will consider all possible scenarios for ownership application individually. To protect ownership of an original work I , the owner embeds a watermark W in I and publishes the marked-work $\tilde{I} = \mathcal{E}(I, W)$. Accordingly, in case of an ownership dispute the owner presents I and shows the presence of W in the possibly modified version of \tilde{I} , \hat{I} with $\text{Dist}(\tilde{I}, \hat{I}) < t$ for some properly selected threshold t , as a proof of ownership, i.e., $\mathcal{D}(\hat{I}, W) = 1$. Correspondingly, the goal of an ownership protection system is to ensure that a pirate, who has access to \tilde{I} , cannot claim ownership on \tilde{I} or any of its variant \hat{I} .

Assuming the pirate has the ability to obtain a fake original work I_p and a fake watermark W_p based on the available \tilde{I} or from a variant \hat{I} , a pirate can successfully attack the ownership protection system by achieving any of the following goals:

Ownership deadlock: The pirate is able to provide an ownership proof that is as conclusive as the actual owner's proof. Therefore, ownership cannot be established and a deadlock arises as

$$\begin{array}{l} \text{Proof} \\ \text{Owner : } \quad \text{Dist}(I, \tilde{I}) < t, \quad \mathcal{D}(\tilde{I}, W) = 1, \quad \mathcal{D}(I_p, W) = 1 \\ \text{Pirate : } \quad \text{Dist}(I_p, \tilde{I}) < t, \quad \mathcal{D}(\tilde{I}, W_p) = 1, \quad \mathcal{D}(I, W_p) = 1 \end{array} \quad (1)$$

Counterfeit ownership: The pirate is able to provide an ownership proof that is more convincing than that of the actual owner. Consequently, the pirate may proclaim counter-

feit ownership over the work whose ownership is in question which can be expressed as

$$\begin{array}{l}
 \textit{Proof} \\
 \textit{Owner} : \quad \text{Dist}(I, \tilde{I}) < t, \quad \mathcal{D}(\tilde{I}, W) = 1, \quad \mathcal{D}(I_p, W) = 0 \\
 \textit{Pirate} : \quad \text{Dist}(I_p, \tilde{I}) < t, \quad \mathcal{D}(\tilde{I}, W_p) = 1, \quad \mathcal{D}(I, W_p) = 1
 \end{array}
 \tag{2}$$

Theft of Ownership: The pirate obtains a protected work, pretends it to be his original and generates his protected copy from the already protected work. The pirate claims ownership over the work assuming the actual owner will remain oblivious about the pirate. Therefore, the pirate is able to prove his ownership on the owners marked work as

$$\begin{array}{l}
 \textit{Proof} \\
 \textit{Owner} : \quad \tilde{I} = \mathcal{E}(I, W) \\
 \textit{Pirate} : \quad I_p = \mathcal{E}(\tilde{I}, W_p), \quad \text{Dist}(\tilde{I}, I_p) < t, \quad \mathcal{D}(I_p, W_p) = 1,
 \end{array}
 \tag{3}$$

A watermarking based ownership protection system can be deemed to be secure if the pirate cannot find any vulnerabilities of the system to launch any of these attacks. This essentially requires determining the potential weaknesses that might lead to these attacks. In this regard, the ability to create an *ownership deadlock* requires

1. obtaining a fake watermark W_p .
2. deriving a fake original work I_p and

from the publicly available \tilde{I} that satisfy (1). The former objective can be easily achieved by exploiting the fact that watermarking techniques are statistical methods, and therefore, they are prone to false-positives. On the other hand, the latter capability requires that the embedding operation be invertible so that a watermark can be removed from a given marked work.

The ability to claim *counterfeit ownership* on a protected work requires all the capabilities needed to create an ownership deadlock plus the ability to remove the owner's watermark W from \tilde{I} in obtaining the fake original W_p , see (2). *Theft of ownership*, on the other hand, is based on intractability of pirate's actions involving a marked work by its owner.

Based on the above attack model, the security requirements of a watermarking technique used for establishing unambiguous ownership can be determined as the following:

1. robustness;
2. low probability of false-positives;
3. non-invertibility of embedding; and
4. involvement of a trusted party.

Without satisfying these requirements, a watermarking technique will not be secure with respect to above described three type of attacks.

The most trivial attack on the class of watermarking techniques that intend to resolve rightful ownership is the removal of the watermark from the marked work. Therefore, the first requirement of an embedding/detection scheme is the robustness against all forms of malicious modification. Unfortunately, this is too strong an assumption to make in the presence of an intelligent and resourceful attacker, and the possibility of *unconditional* robustness is still an open question. However, *conditional* robustness can be achieved against an attacker, who is restricted to a limited set of attacks, by careful design of the embedding/detection operations.

The second requirement refers to the ease with which a pirate can extract a fake watermark from a marked work. (i.e., ambiguity attack). This is the basis for the ambiguity attacks which subsequently enable achieving ownership deadlocks and establishing counterfeit ownership [3]. Essentially, this vulnerability is a direct consequence of relatively high false-positive detection rate of the detector, and it can only be avoided by reducing the probability of false-positives [2].

The third requirement is aimed to ensure that watermark embedding cannot be reversed. That is, a watermark cannot be subtracted from a marked work to produce a fake original. Since this is not achievable, one way to cope with this is by making it difficult for the pirate to obtain a valid watermark [5]. This is usually achieved by cryptographic means which essentially generate watermarks through a one-way relation from the original work. It should be noted that if the false-positive of the underlying watermarking scheme is high, ambiguity attacks would always succeed. On the other hand, when the false-positive is very low the invertibility of an embedding scheme does not pose a significant risk (as brute-force searching of the watermark space will not be feasible), and non-invertibility further complicates the task of the pirate. In [4], it is proved that non-invertible watermarking scheme is possible for certain class of embedding techniques

The last requirement concerns with the theft of ownership attack which is primarily due to owner's inability to track how his protected copy is utilized by others. The only way to achieve such a capability is by the inclusion of a third party which registers each work against its owner and gets involved in watermark generation [1]. Therefore, ownership can be claimed only with the approval of the trusted party. In practice, however, this requirement is one of the most difficult to satisfy due to the complexity in the task of registration as will be addressed in the full version of the paper.

4. REFERENCES

- [1] A. Adelsbach and A. R. Sadeghi. Advanced techniques for dispute resolving and authorship proofs on digital works. In *Proc. of SPIE: Security and Watermarking of Multimedia Contents V*, volume 5020, 2003.
- [2] H. T. Sencar and N. Memon. Combatting ambiguity attacks via selective detection of embedded watermarks. *Submitted to IEEE Trans. Information Forensics and Security*, 2006.
- [3] M. Ramkumar and A. N. Akansu. A robust protocol for proving ownership of multimedia content. *IEEE*

Transactions on Multimedia, 6(3):469–478, 2004.

- [4] Q. Li and E. Chang. On the possibility of non-invertible watermark schemes. In *Proc. of IHW'04, Lecture Notes in Computer Science*, volume 3200, pages 13–24. Springer-Verlag, 2004.
- [5] S. Craver, N. Memon, B. Yeo, and M. Yeung. Resolving rightful ownership with invisible watermarking techniques: Limitation, attacks, and implications. *IEEE Journal on Selected Areas in Communications*, 16(4):573–586, 1998.