Power: A Metric for Evaluating Watermarking Algorithms *

Radu Sion, Mikhail Atallah, Sunil Prabhakar
Computer Sciences, Purdue University
West Lafayette, IN, 47907, USA
Email: [sion, mja, sunil]@cs.purdue.edu

Abstract

An important parameter in evaluating data hiding methods is hiding capacity [3] [11] [12], [16] i.e. the amount of data that a certain algorithm can "hide" until reaching allowable distortion limits.

One fundamental difference between watermarking [4] [5] [7] [8] [9] [13] [14] [17] [18] [19] and generic data hiding resides exactly in the main applicability and descriptions of the two domains. Data hiding aims at enabling Alice and Bob to exchange messages [1] [6] [15] in a manner as resilient and stealthy as possible, through a medium controlled by the evil Mallory. On the other hand, digital watermarking is deployed by Alice to prove ownership over a piece of data, to Jared the Judge, usually in the case when Tim the Thief 1 benefits from using/selling that very same piece of data or maliciously modified versions of it.

In the digital framework, watermarking algorithms that make use of information hiding techniques have been developed and hiding capacity was naturally used as a metric in evaluating their power to hide information.

Whereas the maximal amount of information that a certain algorithm can "hide" (while keeping the data within allowable distortion bounds) is certainly related to the ability to assert ownership in court, it is not directly measuring its "power of persuasion" 2, in part also because it doesn’t consider directly the existence and power of watermarking attacks.

In this paper we show why, due to its particularities, watermarking requires a different metric, more closely related to its ultimate purpose, claiming ownership in a court of law. We define one suitable metric (watermarking power) and show how it relates to derivates of hiding capacity. We prove that there are cases where considering hiding capacity is sub-optimal as a metric in evaluating watermarking methods whereas the metric of watermarking power delivers good results.

1 Introduction

Defining a unified watermarking evaluation metric is a non-trivial task. Most domain specific metrics are derived from the concept of watermarking capacity and do not directly relate to the main purpose of watermarking per se, i.e. claiming ownership in court. Theoretical approaches [3] [10] [11] [12] [16] explore the broader area of steganography and information hiding in a generic manner.

Proof of ownership is usually achievable by demonstrating that the particular piece of data exhibits a certain rare property (read “hidden message” or “watermark”), usually known only to Alice (with the aid of a “secret” - read “watermarking key”), the property being so rare that if one considers any other random piece of data similar (in terms of usability, see below) to the one in question, this property is “very improbable” to apply.

There is a threshold determining Jared’s convincibility related to the “very improbable” assessment. Nevertheless this defines a main difference from steganography: Jared doesn’t care what the property is, as long as Alice can prove it is she who embedded/induced it to the original (non-watermarked) data object.

It is to be stressed here that another particularity of watermarking is the emphasis on ‘detection’ rather than ‘extraction’. Extraction of a watermark (or bits of it) is usually a part of the detection process but just complements the process up to the extent of increasing the ability to convince in court. If recovering the watermark data in itself becomes more important than detecting the actual existence of it (aka. ‘yes/no answer’) then this is a drift towards covert communication and pure steganography.

The paper is structured as follows. Section 2 formalizes some of the main concepts such as watermark, usability, usability domains, watermarking algorithm and defines our metric, watermark power.

---

* Portions of this work are supported by Grant EIA-99035-45 from the National Science Foundation, and by sponsors of the Center for Education and Research in Information Assurance and Security.

1 Tim’s middle name is Mallory.

2 i.e., convinced will Jared the Judge and the Jury be when presenting the algorithm and applying it on an object in dispute.
Section 3 presents a simple scenario where metrics derived from hiding capacity perform poorly in evaluating a certain watermarking method whereas the concept of power delivers good results. Section 4 defines future envisioned research issues.

2 Model and Definitions.

Let $\mathbb{D}$ be the domain of all possible data objects to be considered for watermarking.

Considering any reasonable security assumptions and attacks, it becomes clear that a correct watermarking algorithm has to assure that the domain of all possible watermarked data objects (i.e. results from watermarking objects in $\mathbb{D}$) should be a subset of $\mathbb{D}$. For simplicity we assume that any considered algorithm produces watermarked objects only in $\mathbb{D}$ or that $\mathbb{D}$ is simply the union of all the closures over $\mathbb{D}$ of all resulting watermarked objects from considered algorithms.

For example in case of digital media objects we can simply assume that $\mathbb{D}$ is the set of all variable sized bit strings over $B = \{0, 1\}$.

Objects $d \in \mathbb{D}$ have associated values induced by the object creator. Watermarking tries to protect this association between the value carrying object and its creator. Complex objects can exhibit different value levels when put to different uses. We need a way to express the different associated values of objects, in different usability domains.

Usability Domain: A usability domain is defined as a set of functionals, $e = \{f | f : \mathbb{D} \rightarrow [0, 1]\}$, quantifying value in terms of usability. In a real world algorithm the considered usability domain is constructed by mapping real world properties to actual parametrized functionals in $e$. Also most likely $|e| = 1$, that is, each domain contains only one significant function of usability. Notation: Let the set of all usability domains be $\mathcal{U}$. 

Usability: The usability of an object $d \in \mathbb{D}$ corresponding to a certain domain $e \in \mathcal{U}$ ($e = \{f_1, f_2, ..., f_\delta\}$) is defined as $u(d, e)$ where $u : \mathbb{D} \times U \rightarrow [0, 1]$. $u(d, e)$ is a combination of all the elements of $e$. For simplicity we will assume that $|e| = 1$. In this case we define $e = \{u\}$.

The concept of usability enables the definition of a certain threshold below which the object is not “usable” anymore in the given domain. In other words, it “lost its value” to an unacceptable degree. The notion of usability is related to distortion. A highly distorted object (e.g. as result of watermark embedding or attacks) will likely suffer a drop in its distortion domain usability.

For simplicity, in the following we consider a single usability domain $e \in \mathcal{U}$, unless otherwise specified.

Change in Usability: The difference in usability is defined as $\Delta u : \mathbb{D} \times \mathbb{D} \times U \rightarrow [-1, 1]$, where $\Delta u(d_1, d_2, e) = u(d_1, e) - u(d_2, e)$. This quantity is easier to derive from a real world mapping and has a higher impact on the actual embedding decisions made.

Usability Vicinity: Let $V \subseteq U$ be a set of usability domains and a maximum allowed difference in usability $\Delta u_{\text{max}}$. Then we say that element $x \in \mathbb{D}$ is in the radius $\Delta u_{\text{max}}$ usability vicinity of $d \in \mathbb{D}$ with respect to $V$ \footnote{And vice-versa. It is commutative.} if and only if $\forall v \in V, \Delta u(d, x, v) < \Delta u_{\text{max}}$.

Note that the usability vicinity of a certain object $d \in \mathbb{D}$ with respect to a considered set of usability domains $V \subseteq U$ defines actually the set of possible watermarked versions of $d$ with respect to $V$ and $\Delta u_{\text{max}}$.

Watermark: Given an un-marked object $d \in \mathbb{D}$ and the considered watermarked version of it, $d' \in \mathbb{D}$, where $d'$ is within radius $\Delta u_{\text{max}}$ usability vicinity of $d$, a key $k \in K$ \footnote{Where $K$ is the set of secrets (i.e. keys) involved in the watermarking process. In many cases we assume that $K$ is also a subset of all variable sized binary bit strings or can be easily mapped to one. Sometimes $K \subseteq \mathbb{D}$.}, a set of usability domains $V \subseteq U$ and a maximum allowed difference in usability $\Delta u_{\text{max}}$, a watermark can be asserted by a special property functional $w : \mathbb{D} \times \mathbb{D} \rightarrow B$, defined by the following: $w(d, k) = 0$, $w(d', k) = 1$ and there exists $\epsilon \in (0, 1)$ such that the probability $P(w(x, k) = 1 | w(d', k) = 1) < \epsilon$, for any $x \in \mathbb{D}, x \neq d'$, inside the radius $\Delta u_{\text{max}}$ usability vicinity of $d (1)$. \footnote{The definition becomes more complex in case of specifying different maximum allowed changes in usability for each given domain. It is simple to derive that case from this one.} Notation: Let $\mathbb{W}_D$ be the set of all $w$ over a given $\mathbb{D}$.

In plain words, a watermark can be defined as a special induced (through watermarking) property ($w$) of a certain watermarked object $d' \in \mathbb{D}$, so rare, that if we consider any other object $x \in \mathbb{D}$, with a “close-enough” usability level with the original object $d$, the probability that $x$ exhibits the same property can be upper-bounded.

Note: Intuitively, one main challenge of watermarking is to find/derive $d'$ such that, given only $d'$ it will be very hard for an attacker to determine an $x \neq d'$ inside the usability vicinity of $d$. 

Algorithm: A watermarking algorithm can be described as a functional $\mathcal{A} : \mathbb{D} \times K \rightarrow \mathbb{W} \times \mathbb{D}$, which, given as input an object $d \in \mathbb{D}$ provides a watermarked version of the object, $d'$, and an associated property functional $w$ that enables watermark detection. Notation: Let $\mathcal{A}_D$ be the set of all $\mathcal{A}$ over a given $\mathbb{D}$.

Attack: Given a watermarking algorithm $\mathcal{A} \in \mathcal{A}_D$, an object $d \in \mathbb{D}$ and its watermarked version $d' \in \mathbb{D}$, $\Delta u_{\text{max}}, \epsilon_{\text{max}}, \forall k \in K$ and noting $\alpha(d, k) = (d', w)$, a watermarking attack is defined by $z : \mathbb{D} \rightarrow \mathbb{D}$ such that $\Delta u(z(d'), d) < \Delta u_{\text{max}}$ and $w(z(d'), k) = 0$. In
other words, an attack tries to maintain the attacked
watermarked object within the usability vicinity of
the original non-watermarked one, while making it
impossible to recover the watermark. Notation: Let
\( Z_w \) be the set of all attacks \( z \) for a given \( w \).

Watermark Power: In designing a new metric for the power of a certain watermark we have to take
into consideration two main aspects, namely (i) how
"rare" is the watermark and (ii) how easy it is to
find and apply a successful attack for it. The "rarity"
of the watermark is modeled by \( \varepsilon \) as defined above.
Estimating real-life raw attack-ability of algorithms is
basically intractable, thus we have to assume that a
successful attack is always available.

A powerful marking method should result in a
watermarked object \( d' \), at the "outskirts" of the
allowable usability vicinity of the original \( d \), making
it hard/impossible for an attacker to directly derive \( d \)
or the considered vicinity set.

For a given watermarked object \( d' \) and associated
property \( w \), we define the power of the watermark as:
\[
power(w, d') = (1 - \varepsilon) \times \left( \frac{1}{|d|} \right)
\]
\((2)\)

Note: \( V_{d', d} \) is the intersection of the usability
vicinities of \( d \) and \( d' \). It defines the target space for
any successful attack, effectively modeling the ease
of finding a non-watermarked, usable version of \( d' \).

Note: \( P(3 \varepsilon > 0) \) defines the probability that a
successful attack can be found for a given algorithm.
In the following we will consider it \( 1 \) (highly likely) \(^8\).
If we consider \( P(3 \varepsilon > 0) = 1 \) we have a formula
easier to sample and compute:
\[
power(w, d') = (1 - \varepsilon) \times \left( \frac{1}{|d|} \right)
\]
\((2-2)\)

The power of a certain watermark is directly related
to its convince-ability towards Jared the Judge. The
weaker the watermark (higher the false hit probability
upper-bound) the less convincing it will be \(^9\).

Relative Algorithm Power: If given two
algorithms \( a_1, a_2 \in \mathbb{A} \) we say that \( a_1 \) is weaker than
\( a_2 \) with respect to \( d \) if, whenever applied to the same
object \( d \in \mathbb{D} \) and key \( k \in \mathbb{K} \), \( a_1 \) returns an associated
(property function,object) pair \((w_1, d_1)\) that is weaker
than the one returned by \( a_2 \).

Weighted Algorithm Power: Whereas relative
algorithm power indeed compares two applications (i.e.
\(^6\)) the present definition requires more careful attention and
many future refinements are envisioned. It is given only as an
illuminative example of the new proposed approach and is not to
be taken directly to implementation.

\(^6\) The notation \( \power(w) \) will be used if \( d' \) is implied by the
context.

\(^7\) In a previous paper the power of the watermark was
wrongly defined as \( \frac{1}{|d|} \).

\(^8\) It is to be noted that in real life, a certain watermark
embedded into an object can be 'viewed' through different
property functionals, that is there can be multiple \( w \)'s that
reveal the given base watermark with different \( \varepsilon \)s. This basically
-corresponds to different methods of watermark detection. The
concept of power is also distinguishing among them.

3 Scenario

One could argue that, using capacity as a measure of
the power of "persuasion" of a certain algorithm works, if, instead of hiding a known text (e.g. "This
is the property of Alice"), the algorithm hides a
hash of it or some other form of encrypted secret.
This - the argument goes - will increase the actual
"persuasiveness" of the algorithm and will tightly relate
capacity to the convince-ability towards Jared, because,
after all, only Alice could have known the encoded
secret and the probability that anyone else might know
it, is computationally zero and thus secure.

Whereas the above argument makes a good point of
showing that hiding a secret (i.e. that looks like
random "garbage" to the attacker) is much better
than...
hiding a plain-text message, it misses the point in case of existing trivial attacks on the algorithm as a whole.

In the following we present a simple scenario in which a metric based mainly on hiding capacity cannot predict the weakness of a marking algorithm, whereas the **weighted algorithm watermarking power** metric performs well.

For illustration purposes, consider an algorithm in the space of LSB watermarking algorithms. Those algorithms are known to be weak, because of trivial attacks that can be successfully deployed against them [2] [13] [14].

Let $\mathbb{D}$ be the space of images (e.g. JPEG pictures). Given some normalized image distortion metric in a trivial usability domain (e.g. HVS - Human Visual System), $m: \mathbb{D} \times \mathbb{D} \rightarrow [0,1]$, let's consider $d \in \mathbb{D}$ and the usability vicinity of $d$ of radius $\Delta u_{max} = m(d,d')$. $d' \in \mathbb{D}$ is obtained from $d$ by altering the LSB subset of $d$ such that $m(d,d')$ is maximal.

If we assume (like it is generally understood) that altering LSB information (to the extent given above) is usually tolerable, because the given usability vicinity contains $O(2^{[LSB]})$ elements (many), and because $P(u(z, k) = 1|w(d'), k) = 1$ as defined in (1) cannot be upper bound (i.e. it is 1), the power of any LSB algorithm tends to 0 (weaker) rendering the algorithm (rightfully) useless in Court.

On the other hand, if *available encoding capacity* would have been a major factor in providing Court confidence, and knowledge about obvious attacks (e.g. zero all LSB info) would not have been available, then the alleged non-zero confidence may have determined its undeserved use in Court proofs.

Although the example is not the most accurate one (defining a “perfect” case is out of the space requirements of this poster) it certainly is relevant intuitively, linking the idea of measuring watermarking algorithm quality to its final goal, ownership proof and creation rights affirmation in Court.

### 4 Conclusions

We defined main generic watermarking domain issues and presented a new metric aiming at qualitatively measuring watermarking algorithms.

We stressed our metric, *watermarking power*, versus any arbitrary domain-specific metric as being a better formula, essentially linked to the underlying final purpose of any watermarking algorithm. A simple example was given, outlining the main differences.

Further work is required in refining the given metric and determining different data specific applications. Integration with existing domain specific work is another point of future interest.

### References


