A High Capacity and Robust Image-Based Watermarking Technique for Relational Databases

Maikel Lázaro Pérez Gort, Claudia Feregrino Uribe,
Jyrki Nummenmaa

Technical Report No. CCC-16-010
November, 2016

Luis Enrique Erro 1
Sta. Ma. Tonantzintla,
72840, Puebla, México.

A High Capacity and Robust Image-Based Watermarking Technique for Relational Databases

Maikel Lázaro Pérez Gort[1], Claudia Feregrino Uribe[1], Jyrki Nummenmaa[2]

1. Coordinación de Ciencias Computacionales, INAOE
Luis Enrique Erro #1, Sta. Ma. Tonantzintla, Puebla, 72840, México
2. School of Information Sciences, University of Tampere
Kalevantie 4, 33100 Tampere, Finland
E-mails: mlazaro2002es@inaoep.mx, cferegrino@inaoep.mx, jyrki.nummenmaa@cs.uta.fi

Abstract. Information hiding techniques have been useful for passing secret messages unnoticed since old times, but nowadays it also has been purposeful to prove ownership of digital assets. The increment of the internet services has provoked easy accessing to illegal or unauthorized copies of datasets, so the piracy is at its best. With watermarking emerging as a tool for ownership proof, traitor tracing, etc., there have been several techniques for multimedia data but no so over relational data. Due to the differences of these data types, another angle is necessary to the conceptions of its watermarking schemes also to deal with new problems that have emerged. With our research, we seek to develop a robust technique based on meaningful signals oriented to watermarking relational data. The watermark must be resilient against common updates but also, it must be resilient against bit level attacks that tries to destroy the watermark.

Keywords. Data usability, embedding capacity, image-based watermarking, ownership proof, relational data, robustness, signal restoration

1. Introduction

The piracy of digital assets has been a concern for those who use them as business products over the web. Truly is that deploying an online enterprise permits the use of the benefits of the internet, allowing the organization to go beyond the geographical obstacles and political borders. The problem is that, due to the easy access and distribution of digital data, piracy and illegal copies cause millions in losses to these businesses.

Computing security is composed of a set of techniques [56] that try to avoid the degradation of the digital assets integrity and protects them from piracy. In the first place is the *Cryptography* which consists of "the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication" [56]. Also, we may find the Steganography, which "is the art of concealing the very existence of information by inserting it in an apparently innocuous object" [7]. Finally, we can find the digital *Watermarking*, a promising set of techniques that allows owners to check the authenticity of the digital asset in case of an accusation of illegal copy [17, 18, 43, 46].

The idea of *Watermarking* a digital asset consists in introducing small changes called marks into the object to be watermarked. All the marks together constitute the Watermark (WM) and each one must

have an insignificant impact on the data usability. Also, every mark should be embedded in such a way that a malicious user cannot destroy the WM without compromising the data quality. So, the watermarking idea is not the prevention of copying or distribution of the digital assets, but a tool to be used in case of legal demands for unauthorized data distribution or false ownership claims [2].

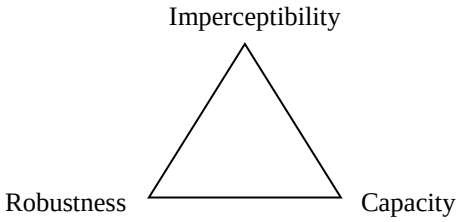


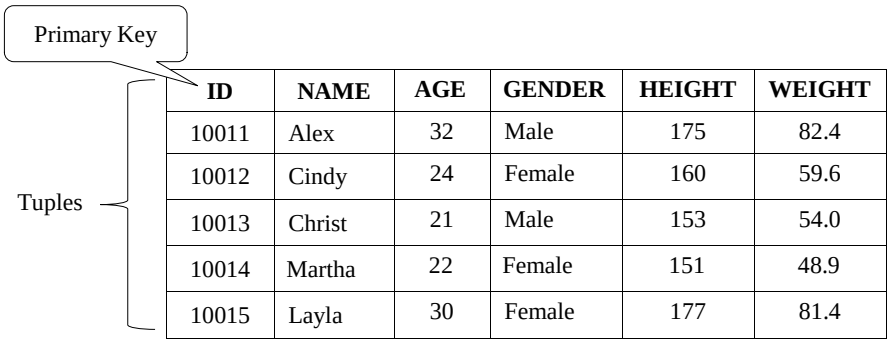
Figure 1. The Watermarking trade-off [8].

There is a Watermarking trade-off that involves the requirements of robustness, imperceptibility, and capacity of the WM (see Figure 1). For example, creating a robust WM may require to embed more marks (increasing the WM capacity) in the object being watermarked, but that would make the WM more perceptible. On the other hand, for making the WM more imperceptible, we could decrease the capacity, but that would compromise the robustness.

"In contrast to *Steganography*, *Watermarking* has the additional requirement of being robust against attackers that try to remove the hidden data or make its presence undetectable" [45]. Also, the use of *Cryptography* makes data unreadable and put in evidence the existence of the secrecy in the digital asset [7].

Several techniques have been proposed for allowing the wide distribution of multimedia data (MMD) with embedded information that permits the verification of the copyright [9, 72, 73]. Although there have been also watermarking schemes for relational databases (RDB) [2, 34, 39, 66] there are still some unsolved problems in this area. These issues are related to the nature of relational data (RD) and the lower maturity of these techniques compared to those created for videos, audios, and images.

Since we are focused in the RDB field, it is important to highlight that a DB scheme may be represented according to the relational model as a collection of tables called relations, linked between each other through a spatial attribute (sometimes is possible a combination of attributes) called Primary Key (PK). The table columns are the attributes (or fields) of the data to store, and the rows (or tuples) correspond to each object registered in the *relation* (see Figure 2). The PK must be unique for each tuple, so that it can be identified [19].



ID	NAME	AGE	GENDER	HEIGHT	WEIGHT
10011	Alex	32	Male	175	82.4
10012	Cindy	24	Female	160	59.6
10013	Christ	21	Male	153	54.0
10014	Martha	22	Female	151	48.9
10015	Layla	30	Female	177	81.4

Figure 2. Example of a relation called "Student" of a generic DB.

The watermarking methods proposed for MMD cannot be used for RD due to the existing differences between these data types, which are [2, 33]:

- Data Redundancy: The multimedia objects are composed of a large sequence of bits providing a convenient cover to hide the marks of the WM, whereas a relation of a DB is a collection of tuples with very low redundancy, an aspect that makes it difficult the embedding of the marks.
- Order of the Data: The relative positions of different parts of multimedia objects do not change, whereas there is no fixed ordering among the tuples and attributes that compose the database relations.
- Frequency of updates: Any portion of multimedia objects is not modified or normally erased, whereas tuples may be inserted, deleted, or updated as part of common and daily database transactions.
- Data appreciation: MMD are oriented to be interpreted by human biological systems (e.g. visual and auditory) and that feature is highly used by watermarking schemes. On the other hand, there is no biological system for raw RD interpretation.

Most of watermarking techniques for MMD use the features of these data types for the generation, embedding and extraction of the WM. For example, the value of pixels of an image usually presents a high correlation with the value of their neighbors, aspect that some techniques use [61]. The attribute values of a database relation should not present this feature at all, according to the right RDB design (low redundancy: exception, the primary key of the relations). Also, MMD can change of domain using some transforms (e.g. Discrete Wavelet Transform (DWT)) and the WM can be embedded in the transformed information, guaranteeing the scattering of the marks widely all over the data [61]. If a similar operation is performed over the RD, the usability may be severely compromised due to the spread of the distortion [2].

Watermark a RDB does not necessary implies marking every one of its relations. Eventually, just DB fragments are sold or distributed, these fragments can consist of subsets of a few linked relations, or just a single relation or even segments from it.

Watermarking RD has proved to be a useful tool in various ways, the more common application is copyright checking in ownership conflicts [42, 48, 64]. Also, embedding different WMs in distinct copies of the data, called fingerprinting, is used to deter illegal copies and to trace traitor users [26, 30, 53]. These two variations of the use of watermarking are classified as robust due to the severity of attacks aimed at their destruction. On the other hand, a WM can be used to control the integrity of the data and protect them against tampering and fraud [25, 47, 51, 76]. This last watermarking type is classified as fragile considering the data owner knows about the WM presence and will be benefited from using it.

This document is organized as follows: In Section 2 we present the theoretical elements of the research area. Here the desired properties of the watermarking schemes for RD and the main challenges that must be faced are explained. Also, the state of the art and other related area techniques considered in this investigation are presented. In Section 3, the proposed scheme is explained. Finally, Section 4 presents the results of the conducted experiments and their analysis. The conclusions constitute the Section 5.

2. Theoretical Background

2.1 Bases of Watermarking and Relational Databases

Watermarking RD consists of two general processes: the embedding and the extraction of the WM. The embedding process is formed by two main sub-processes, the generation of the WM and the embedding of the marks into the relation of the database. The simplest embedding process has at least one parameter, the Secret Key (SK), known only for the data owner (see Figure 3).

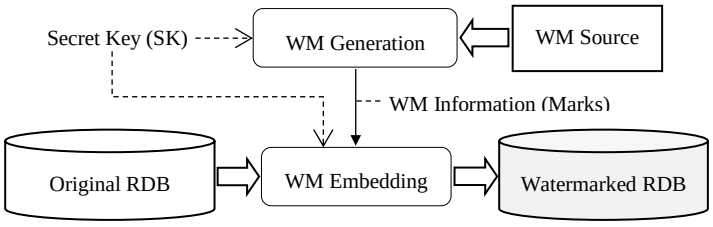


Figure 3. Structure of the WM embedding process.

The extraction process is formed by four sub-processes: detection of the marks, their extraction, the WM reconstruction, and finally (optional) the enhancement of the WM. The value of each private parameter must be the same as the used for the embedding process (see Figure 4).

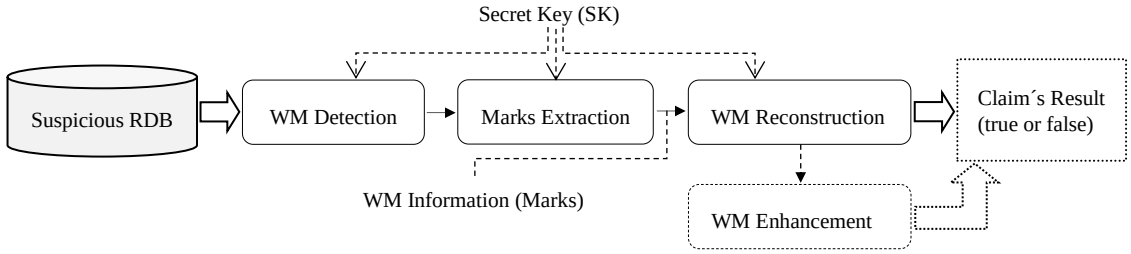


Figure 4. Structure of the WM extraction process.

Also, WM schemes for RDB databases must accomplish the requirements presented below [2, 33]:

- Capacity: It determines the optimum amount of data to be embedded without compromise the database usability. Also, tries about the optimum way to embed and extract this information without raising suspicion the potential attacker.
- Usability: The data usability cannot be degraded by the changes introduced during the WM embedding process. The tolerance to changes may change according to the database purpose.
- Robustness: The embedded WM should be robust enough against malicious attacks or common updates as long as the intent to remove the marks do not compromise the data usability. There is the exception of the fragile WM, oriented to protect the data integrity.
- Blindness: The extraction of the WM should require neither the knowledge of the original database (the unwatermarked copy) nor the watermark information. In case contrary when a scheme does not fulfill this requirement, the WM will be classified as non-blind.
- Security: The security of the watermarking process relies only on some private parameters (e.g. secret key) which should be kept completely secret by the users involved in the operation.
- Public System: The watermarking system should assume that the method used for inserting a watermark is public. The defense must lie only in the choice of the private parameters (e.g. secret key) [6].
- Incremental Watermarking: After the database has been watermarked, the watermarking algorithm should compute the watermark values only for the tuples that will be added modified.
- Non-interference: If multiple WMs are inserted into the same relation of the RDB, then their marks should not interfere with each other.
- False Positives and False Negatives: The false hit is the probability of a valid watermark being detected from unwatermarked data, whereas false miss is the probability of not detecting a valid watermark from watermarked data that has been modified in typical attacks. The false hit and the

false miss should not interfere or affect in the detection of the marks and should not have influence in the WM construction.

Watermarking techniques for RD are classified according to the following criteria [2, 33]:

- Watermark Information: According to the source used to generate the WM (e.g. audio, image, text, other database values, etc.).
- Distortion: The watermarking schemes oriented to embed changes into the database will be classified as distortion-based, otherwise will be classified as distortion-free. The distortion-free methods are commonly used to implement fragile WMs.
- Reversible: If the marked data are returned to its unwatermarked version just after the WM extraction the scheme will be reversible. This classification can be understood as a sub-classification for the distortion-based techniques.
- Intent of Marking: Different watermarking schemes are designed to serve different purposes, mainly, integrity and tamper detection, localization, ownership proof, traitor detection etc. According to that will depend if the WM will be robust or fragile.
- Cover Type: Classification given to the watermarking scheme according to the type of the cover (e.g. type of attributes) into which marks will be embedded.
- Granularity Level: The watermarking can be performed by modifying or adding information at bit level or higher level (e.g. whole a value, attribute or tuple level).
- Verifiability/Detectability: The detection – verification process may be deterministic or probabilistic. Also, it can be performed blindly or non-blindly, and it can be performed publicly (by anyone) or privately (by the owner only).

According to their intent, the robust WM schemes are required by the fingerprinting techniques and the ownership proof. This class of techniques must be resilient against the most complex attacks as well as common database updates:

- Common updates: Daily routine transactions for which databases are implemented. They represent the basic business management operation in any business (data selection, data modification, data insertion and data elimination). If the WM are no conceived to resist this will be easy to destroy.
- Malicious attacks: There are several kinds of attacks oriented to compromise the robust WM techniques, they are as follows [2]:

- *Value modification attacks:* Consist in change the value of the attributes hoping to erase the marks that could be stored on them. These modifications can be at bit level or with a rounding or transforming the numeric attribute values. Also, for multi-word character attributes, can be replacing a specific character like spaces.
- *Set attacks:* Malicious operations that try to simulate the common transactions of databases. These can be performed horizontally (tuple level), vertically (attribute level) or mixed (tuples and attributes), as well as at single level (one element) or massive level (multiple elements). They are sub-classifieds as *subset attacks* (deleting or updating of tuples and/or attributes) or *superset attacks* (insertion of new tuples and/or attributes).
- *Subset Reverse Order Attack:* This attack is performed by exchanging the order or the positions of the tuples and/or attributes trying to make impossible the WM detection. If the WM embedding is sequential somehow this attack will provoke severe damage to it.
- *Brute Force Attack:* In this case, the attacker tries to guess the value of the private parameters (e.g. secret key). This attack can be thwarted by using several private parameters or by parameter's values big enough in size.
- *Collusion Attacks*: Based on the combination of disjoint tuples taken from different copies of the same relation (with the exact scheme). Assuming each copy has different WM, this combination could compromise its extraction. There is a sub-type for this attack called *Majority Attack* which consists in computing each bit of the data to be added in the new relation using a majority function. That variation will exclude from the copy those bits with unusual values, and so, the probability of pass bit portion of the WM that may be identified, will be avoided.
- *False Claim of Ownership:* This attack consists in to provide the traitor with evidence that may raise doubts about the true owner of the data. This attack may be performed by adding a new WM into the watermarked data (*Additive Attack*) or discovering a fictitious WM from the watermarked data based on the random occurrences of the data values (*Reversibility Attack*).

2.2 Notation Used and Useful Functions

As Agrawal and Kiernan [3] were the first ones to propose a WM technique for RDB (also called AHK algorithm), their notation has been used by most subsequent works (see Table 1).

Table 1. Notation used by AHK watermarking RDB techniques.

Notation	Description
η	Number of tuples in the relation of the RDB to be marked.
ν	Number of attributes available for marking from the relation.

ξ	Number of less significant bits (lsb) available for marking in the binary representation of each attribute value.
$1/\gamma$	Defined as Tuple Fraction (TF), is the fraction of tuples to be marked $\gamma \in (1, \eta)$. If the value of γ decreases more tuples will be considered for marking. Ignoring the usability constraints, if $\gamma = 1$ all tuples of the relation will be marked.
ω	Number of marked tuples from the η presented in the relation. Ignoring the usability constraints $\omega \approx \eta / \gamma$.

According to Table 1, data usability will be controlled not only by the DB constraints but the trade-off between the values of γ and ξ as well. These parameters will be responsible for an important part of the watermarking capacity considering the distortion limits on the data to be marked.

Also, the relation to be marked will be identified as *R* with scheme $R(PK, A_0, \ldots, A_{\nu-1})$ where A_i: $i \in (0, \nu-1)$ are all the attributes available to mark. The tuples of the relation *R* are identified as r_j where $r \in R$ and $j \in (0, \eta-1)$. So, the notation $r_j.A_i$ it refers to the attribute *i* of the tuple r_j in the relation R where primary key will be $r_j.PK$.

An important type of methods frequently used by the Watermarking RDB techniques are the one-way hash functions which always guarantee the same output for the same input parameters. The function notation is *H* and for a given message *M* as input, the result will be *h* (operation represented as $h = H(M)$). As one-way function, will be hard to obtain M given *h* such that $M = H'(h)$. Also, given *M* will be hard to find another message M' that guarantees the same result *h* such that $H(M) = H(M')$ [3]. Examples of this type of functions are MD5, SHA-0, SHA-1, etc.

Hash functions are very useful to obtain a unique and secret value that identifies a tuple given unique information about it (e.g. $r_j.PK$ or a unique combination of index values for seeking). Also, combining $r_j.PK$ with the *SK* will improve the secrecy level of the operation.

2.3 Previous Work

Since the first WM technique for RDB proposed in 2002 by Agrawal and Kiernan [3], it has been remarkable the increment of proposals of different nature (see Figure 5). Growing diversity, several works of different types of WM have been created, acting at different level of the RDB and trying to persist despite the RD nature.

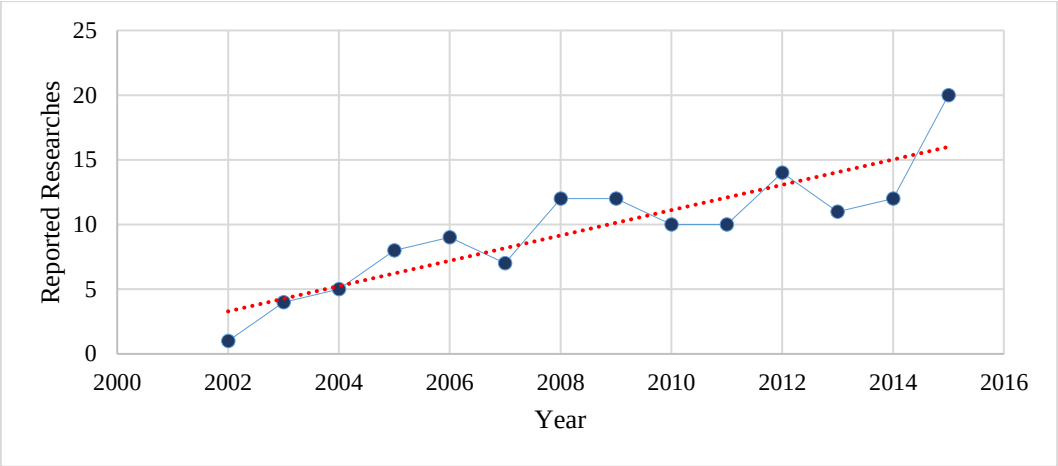


Figure 5. Remarkable increment of WM techniques for RDB.

An important amount of options has been published seeking to accomplish the different intents but most of them have been focused in the ownership proof (see Figure 6). Considering the robustness required for this goal, daily RDB operations tend to destroy the WM as time passes. That is why more researchers still seeking better results in this area. The main challenge is creating a robust WM without compromising data usability in the embedding and remains the imperceptibility so the attacker cannot find evidence at all about which are the marks and where they could be.

Figure 6. Distribution of studied techniques per WM intent.

Mainly oriented to protect data from illegal distribution and false ownership claim, the robust WM (mostly distortion-based) must persist despite the benign updates and the malicious attacks. Even so, not all studied techniques are tested against all types of attacks and just a few display the distortion degree provoked, allowing the analysis of the usability compromise (see Figure 7).

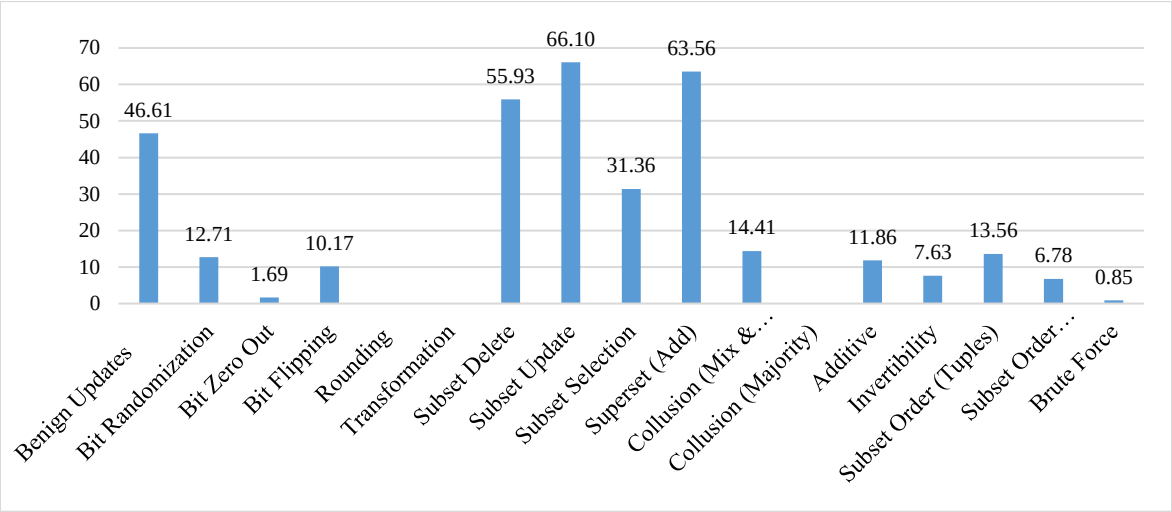


Figure 7. Attention given by the read literature to the attacks and the benign updates.

The main groups for RDB WM are defined according to their intent. Often, the ones oriented to detect tampering and controlling usability violations are distortion-free, while the rest use to be distortion-based. Figure 8 and 9 display a classification of the studied Watermarking techniques proposed by the authors since 2002.

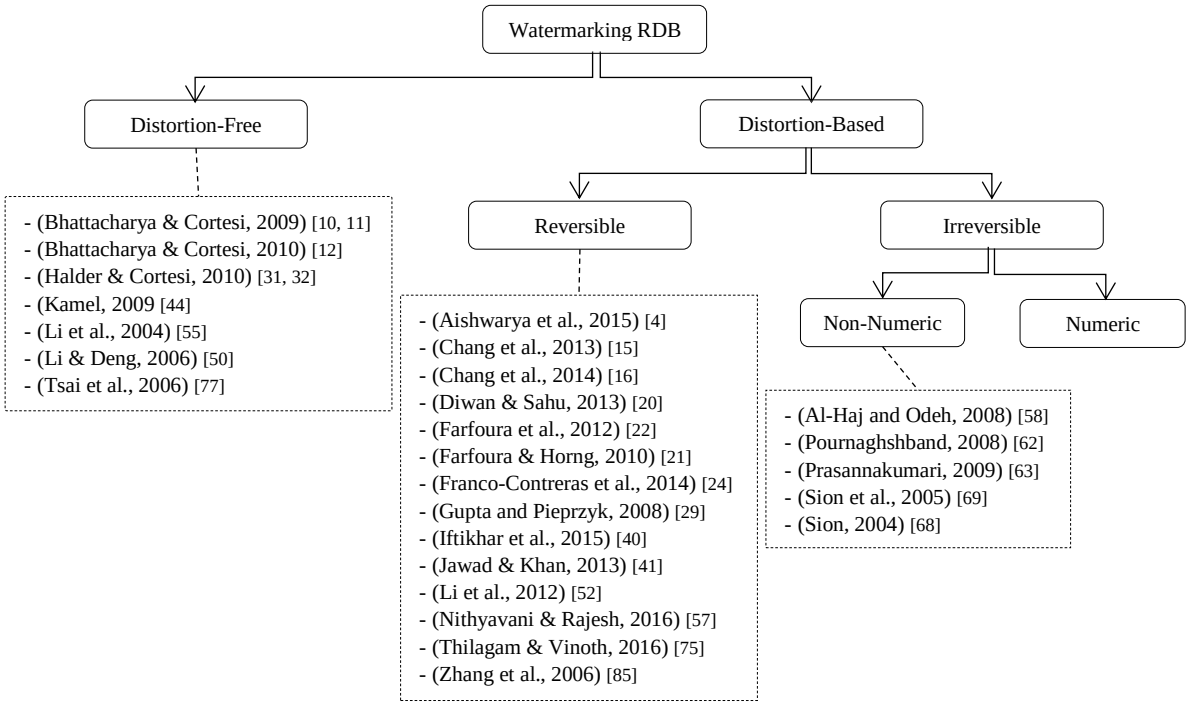


Figure 8. Classification of the studied watermarking techniques for RDB.

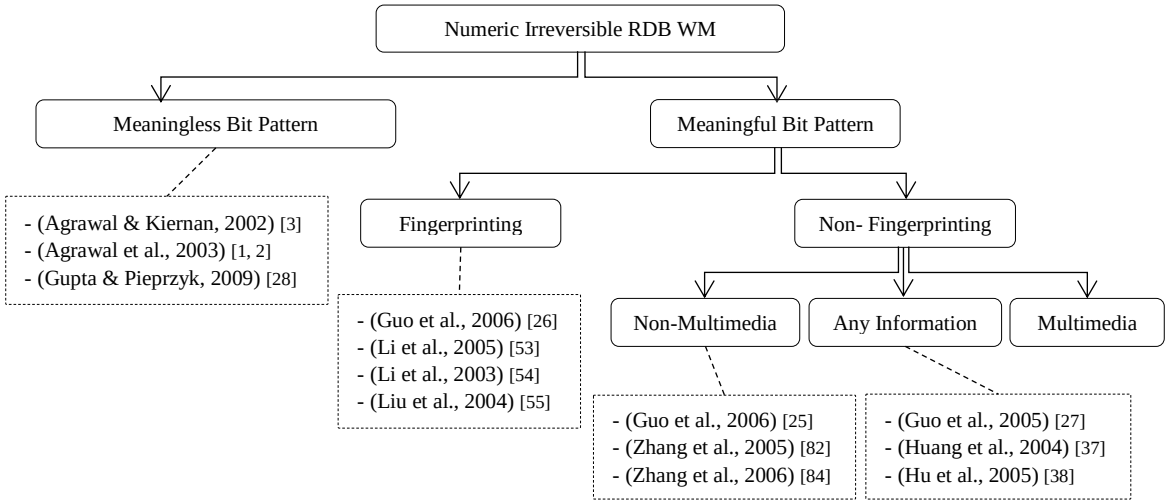


Figure 9. Classification of the studied numeric irreversible watermarking techniques for RDB.

A peculiar type of *Watermarking* techniques for RDB is that who use MD as the source for the WM generation due to the existence of techniques to enhance this kind of signal. Figure 10 makes a special distinction between the Image-Based Watermarking (IBW) (those who use images for the generation) and the others who use other MD data type. This is because the image may be composed of a low number of bits (even lover if is a 1bit-BitMap (BMP) format) and still could be identified by human visibility.

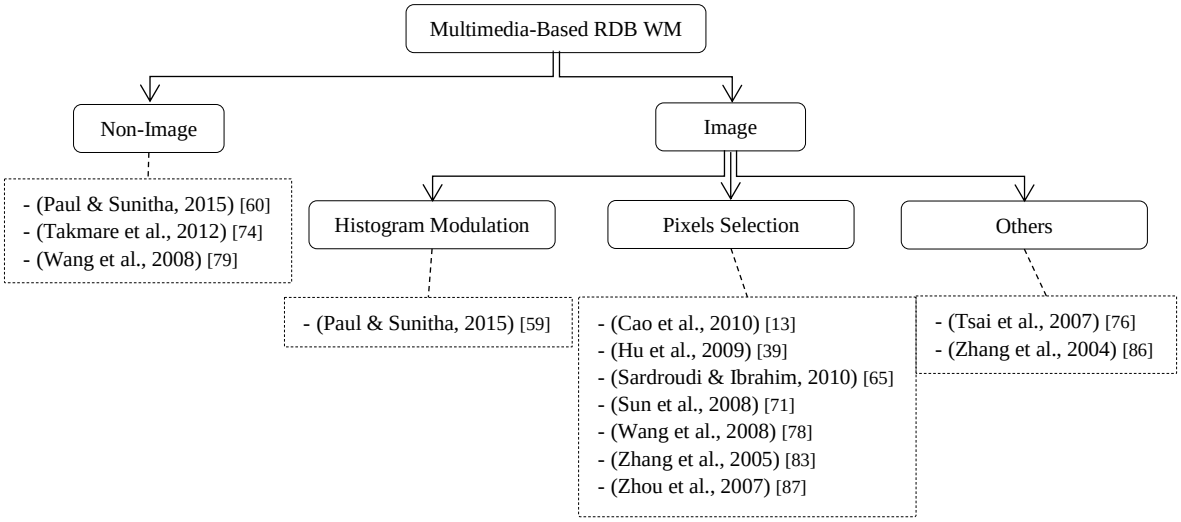


Figure 10. Revised multimedia-based watermarking techniques for RDB.

Since the first watermarking technique for RDB was the proposal of Agrawal & Kiernan [3], there are several numbers of extensions to this scheme. The AHK algorithm was created for marking numeric attributes at bit level and is classified as blind. It mainly embeds a meaningless bit sequence into the relation to be marked by selecting the tuples to mark, the attributes and the bit of the attributes

according to a value generated using one-way hash function that takes as inputs SK and $r_j.PK$ (see Section 2.2). They define a Virtual Primary Key (VPK) as $F(r_j.PK) = H(SK \circ H(SK \circ r_j.PK))$ where the operator $\circ$ represents concatenation. So, the *desiderata* for watermarking relational data with random nature are defined by the *mod* operator. Tuple selection is performed if $F(r_j.PK) \bmod \gamma = 0$. In the same way, the attribute index for marking in the selected tuple will be obtained according to $F(r_j.PK) \bmod \nu$, and the bit position $F(r_j.PK) \bmod \xi$.

Agrawal & Kiernan [3] method's main limitations are given by a weak resilience against subset attacks and malicious updates such as data transformations. Also, the meaningless of the watermarking information may compromise the identification despite that the WM could survive the attacks. Finally, they only rely on the usability control in the parameters ξ and γ, and ignore the constraints that may be implemented over the database.

In 2004, Sion *et al* [70] presented a different approach for watermarking RDBs, also a blind technique that marks numeric attributes at bit level. They focused in do not compromising data usability, for that, they calculate data statistics and mark the selected tuple according to the database constraints and the range of allowed error for the data (using the Mean Squared Error (MSE) as metric). The limitation of this proposal is that a tuple sorting is required for defining subsets identifying some tuples as their bounds. These bounds will be called markers and are identified similarly to the tuple selection in AHK Algorithms (if $H(SK, r_j.PK) \bmod e = 0$, where $e = attributes\_set.length/subset\_size$).

Creating subsets are with the aim of taking the watermarking problem into multiple simplified situations and embed the marks with high redundancy, allowing error correction in the extraction process using majority voting. The problem is that some set attacks (e.g. adding or removing tuples) may add or remove some markers, which may compromise the detection of the same subset used in the embedding process and due to this, increment the probability of the synchronization errors. Sion *et al* [70] introduce the idea of using maps for improving the marks localization in the detection process, but this decreases the performance of the watermarking embedding and extraction processes, and violates the blind principle of watermarking, not mentioning that this information may suffer losses in data actualization attacks, so the technique will have to add extra considerations for the maps robustness.

Other proposed technique classified as IBW was the presented in 2010 by Sardroudi & Ibrahim [65]. This scheme uses an image for generating the WM and also checks the usability before modifying the data due to the mark insertion. The main idea is based on AHK algorithm but there are some additional considerations to avoid data degradation. The limitation of this proposal is given by the watermark capacity and the random nature of the algorithm. The first IBW proposed [86] considers a

sequential nature for the marks in the embedding process. The problem is when a *subset-reverse order attack* (over tuples or/and attributes) is performed there may never be a way to identify and recover the WM. So, the solution of previous IBW techniques was a random selection of the image pixels, but this compromises embedding the entire WM (some pixels are considered multiple times while others are ignored). The consequences of this are similar to perform tuple deletion attack to the relation. Sardroudi & Ibrahim's technique presents better results that similar previous schemes but the watermark still is not entirely embedded into the data even when all tuples of the relation are marked ($\gamma = 1$) causing a serious usability deterioration and increasing the risk of providing evidence to the attackers about the locations of the marks.

There are other techniques focused in marking more than one attribute per tuple on the relation. At first glance, the main deficiency of this would be the increasing of data distortion compromising the usability. For example, Jawad, K. & Khan [41] use Genetic Algorithm (GA) for detecting a couple of attributes that combined will provoke less distortion over the tuple and the column (comparing the values of the same attribute two tuples up and two down). According to the studied multi-attribute techniques, a common limitation is that often they define a fixed number of attributes for embedding the marks.

Despite there are some multi-attributes WM techniques for RDBs, none of them use an image as WM and tries to reduce the distortion and at the same time increases the considered pixels in the embedding the way we are proposing as far as we know.

There are also others techniques that may be useful to consider working with them. For example, using binary images for WM generation brings the advantages of using a simpler signal, compared to others, and would allow the chance of not embedding the entire signal (to minimize distortion over RD) and even so, extract a fragment and enhance it using techniques like the proposed by Chan et al. [14], Shen et al. [67], and Zhang and Ye [81]. On the other hand, there are works that use statistics fundaments for variance minimization. That may be useful for selecting the attributes for marking with higher frequency than others, seeking the preservation of data usability. Here we can mention the work of He, X. et al. [35], Kleijnen, J.P. et al. [49]; and Warmuth and Kuzmin [80].

3. Proposed Scheme

Our proposal for watermarking relational data aims to improve the watermark capacity using the idea of spatial image watermarking techniques. In conventional RDB watermarking techniques, the marks are embedded one per tuple so that the operation can be understood as a vertical process. But, if the

marks were embedded in two directions (vertical-horizontal) then the capacity of the watermarking scheme increases considerably.

It seems to be logical that subset attacks have a higher probability of compromising the WM detection if embedding is in one direction; if it is in two, this will improve WM robustness due to the increase of the marks redundancy. This statement is proved by the results of the experiments carried out. Then, despite it may be better to embed more than one mark per tuple, simulating the image techniques that act over the spatial domain, this may compromise the usability. Of course, the higher the number of marked values per tuple, the higher the distortion caused into the data. That is why a data-quality module is added to our scheme.

Our first scheme is based on AHK algorithm, acting over numeric attributes at bit level and does not require the original unwatermarked data neither the WM source for the extraction process. The WM is generated from an image and the marks are scattered more than one per tuple according to a new parameter we introduced. The idea is to extend the scheme Sardroudi & Ibrahim [65] but adding a multi-attribute embedding factor called Attribute Fraction (*AF:* defined as $1/\chi \in (1, \nu)$) controlling the data distortion to be restrictive by considering the maximum error allowed. As with the tuple fraction, if $\chi = 1$ all attributes of the tuple are marked, if χ increases, fewer attributes are marked. Other secret parameters used in our proposal are described in Table 2. We also use the parameters defined in Table 1 as well as the notations and concepts previously described.

Table 2. Parameters required by the proposed scheme so far.

Notation	Description
AL	List of attributes for marking (*AL. size* = ν)
MSB	Range of most significant bits available for the random selection of the value identified as *msb*
LSB	Range of less significant bits available for the random selection of the value identified as *lsb*
IMG	Matrix that represents the array of pixels of the selected image for building the watermark
H	Height of the image used for generating the WM. The randomly selected height position for the pixel extraction is denoted as *h*.
W	Width of the image used for generating the WM. The randomly selected width position for the pixel extraction is denoted as *w*.

The embedding steps of our proposal are presented in Figure 11. We create a VPK for the analyzed tuple (VPK_j) using the function *Create_VPK* and taking as parameter the PK of the tuple and the SK. Then, the tuple is considered for embedding a mark if $VPK_j \bmod \gamma = 0$, otherwise, we pass to the next tuple.

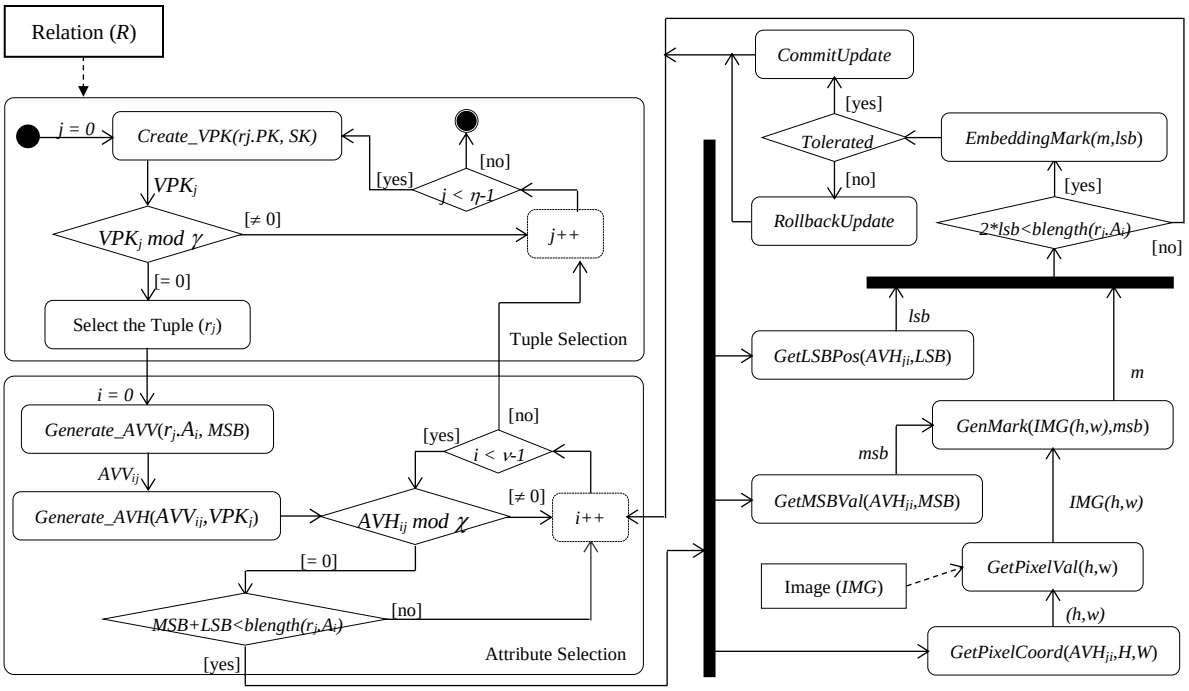


Figure 11. First iteration of the embedding step proposed.

For the considered tuple, we analyze each attribute only if it is included in the *AL* (see Attribute Selection block in Figure 11). The Attribute Virtual Value (AVV) is generated by using the *Generate_AVV* function. If the attribute is selected to be marked, after the mark embedding we could get a new value that may exclude it from the WM extraction process, that is why we have analyzed only the value generated using the attribute MSB. Using the AVV and the VKP_j we generate the Attribute Virtual Hash (AVH) (function *Generate_AVH*) and we proceed to generate the values for the mark embedding on this attribute if $AVH_{ij} \ mod \ \chi = 0$ and the parameters MSB and LSB allow it.

As we mentioned, the WM is generated using an image where each randomly selected pixel (functions *GetPixelCoord and GetPixelVal*) will be *xored* with the *msb* (function *GenMark*). Then the created mark is embedded in the *lsb* position selected from the range given as parameter (function *EmbeddingMark*). The mark embedding is done only if the condition of the *lsb* position respect to the binary length (function *blength*) of the attribute value is met, and the changes will be committed if the data constraints and the tolerated error allow it.

The extraction process follows the same steps of the embedding process and uses the same parameter values as well as the same functions. One step can be modeled as the reverse of the other. The extraction process is deterministic and a majority voting is performed at the end of the extraction to use the redundancy of the embedded marks aiming to avoid the impact of the attacks and benign updates.

3.1 Avoiding the Degradation of the Data

To avoid compromising the data usability, we take into consideration not only the random values selected for marking the attribute (e.g. the *lsb*) but also the attribute value itself. So, if the attribute finally is marked or not, it gives to our technique a higher level of randomness, an aspect that improves the technique, making it unpredictable for the attackers.

As the technique of Sardroudi & Ibrahim, we only mark an attribute if the length of its value in binary notation is at least twice of the position of the LSB selected for marking. For example, if we have a binary number 111 (7 in decimal) and the second *lsb* is selected, we are provoking a change in the value of only 2 units, but respect the original value, this represents an alteration of 28.75%. With this restriction, we are making available the marking only if the *lsb* is 1 (distortion in one unit ≈ 14.29%, also the minimum available degradation) or forcing the selection of the second *lsb* only if the length of the binary value is 5 or higher. For example, if the number is 10111 (23 in decimal), changing the second *lsb* will provoke just an alteration of the 8.7% of its value.

The second consideration is that the selected positions for the *lsb* and the *msb* cannot overlap. For example, we can have in a relation of the database, numeric attribute values that require 10 digits for their binary representation and the available ranges for the random selection of 3 for the *lsb* and 4 for the *msb*. In this case the random positions selected for the *lsb* and the *msb* will never overlap, but if we have a number represented by only 6 digits, there is the probability that the same selected value for the *msb* in the generation of the mark may be selected as the *lsb* to be modified (e.g. *msb* = 4, *lsb* = 3). Due to this, we could obtain a wrong value for the mark in the WM extraction process. Also, the number formed by the MSB range is used for deciding if the attribute will be marked according to the AF. If we change some value of the MSB, we may not consider this attribute in the WM extraction process.

One solution to this problem may be zero padding to the left of the binary representation and make all values of the same length, but this would provoke the detection of a lot of *msb* with same values and it can make the technique predictable. The solution we are proposing is that the sum of the ranges indicated for the selection of the LSB and the MSB must be equal or higher to the length of the binary representation of the numeric value selected to mark.

Another consideration at bit level is the change of the values at the right of the selected *lsb* to reduce the difference of the new attribute value compared against the original. Other *lsb*'s will change their value to the opposite of the value inserted. For example, if we have the binary value 1001101 (77 in decimal) and the *lsb* selected to be marked is the 4th, then the new value would be 1000101 (69 in decimal), but if the other *lsb*'s are changed according to this rule, the new value will be 1000111 (71

in decimal); so, the distortion will be less. The number of *lsb* places to change will be indicated by parameter and this will add again a higher random nature to our technique, due to the fact that not all *lsb* have the same value compared with the marked one. Sardroudi & Ibrahim [65] apply the same concept but they act only over a single *lsb* (right next to the marked).

On the other hand, to avoid violating the database constraints we work with transactions using the classic blocks of *commit-rollback* for the case the operation requires to be reversed due to some violations of the constraints. In that way, we are also keeping the semantic of the data despite the modifications in the marking process.

The last consideration is to allow users (if they desire) to indicate the maximum amount of allowed error for each attribute considered in the list of attributes available for marking. This value will be a percentage of the attribute value before to be marked. In case of exceeding the maximum allowed error, the data owner can try watermarking the relation using different parameters. Also, a general amount of maximum allowed error is indicated to control the relation distortion as a whole.

4. Experimental Results

The conducted experiments were performed using the same dataset that Sun et al. [71] and Sardroudi & Ibrahim [65] used in their experiments since we are comparing our results against theirs. We used the *Forest Cover Type* relational dataset [23] available for public download. This dataset has 581,012 tuples with 54 attributes, we are using a subset of the original data with 30,000 tuples and 10 numeric attributes. We are also using the Universiti Teknologi Malaysia (UTM) logo with size of 82*80 pixels as image with one bit per pixel (see Figure 12 a)).

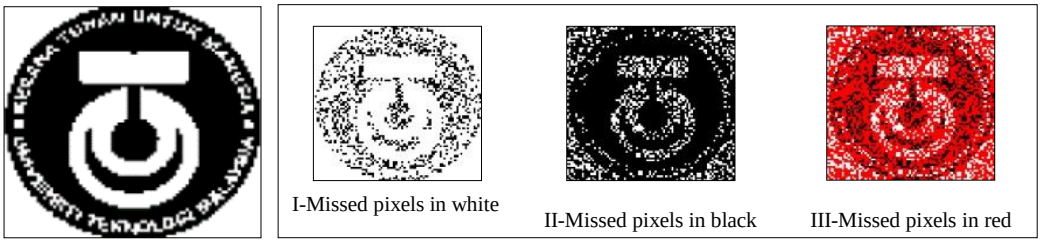


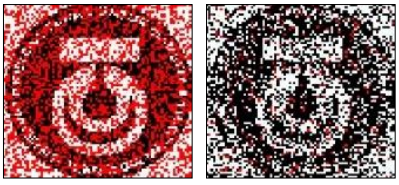
Figure 12. Samples of the images from the experiments.

The implementation of our proposal consists of a software with client-server architecture. The client application was implemented using Java 1.8 as the programming language and Eclipse Mars.1 as Integrated Development Environment (IDE). The server technology was Oracle Database 12c with Oracle SQL Developer as Database Management Interface. The metric used for evaluating and comparing the results so far is the Correction Factor (CF), which compares each pixel values of the

embedded image against the extracted one using the exclusive or operator and the negation of extracted pixels. The maximum value of this metric is 100 and it means an exact similarity of the two images. It excludes from the operation the missed pixels [65]. (See Formula 1)

$$CF = \frac{\sum_{i=1}^{x} \sum_{j=1}^{y} (Img_{org}(i,j) \oplus \overline{Img_{ext}(i,j)})}{x*y} * 100\% \quad (1)$$

To avoid visual confusion and appreciating better the CF obtained values, we assume 'Virtual Value' to the missed pixels. Since our image is binary (in black and white), we do not assign any of these values but the red color (see Figure 12 b)), in that way it is clearer the distortion caused in the extracted image. Also, assigning the 'Virtual Value' may be useful for recognizing better images. For example, Figure 13 a) shows an image with missed pixels due to *tuple deletion attacks*, its CF is lower than the image of Figure 13 b) (noisy image due to an aggressive *tuple addition attack*) even so, for human appreciation is more similar to the original embedded image.



a) CF = 58 b) CF = 69

Figure 13. False color role in the image appreciation.

4.1 Watermark Capacity

The first feature of the proposed WM technique that is analyzed is the capacity. It is worth to remember that in DB watermarking schemes, it is common not to include the whole images as a WM but a part of it. So, we consider a higher number of pixels of the image than any other proposed technique for the WM generation, using a random method (see Figure 14).

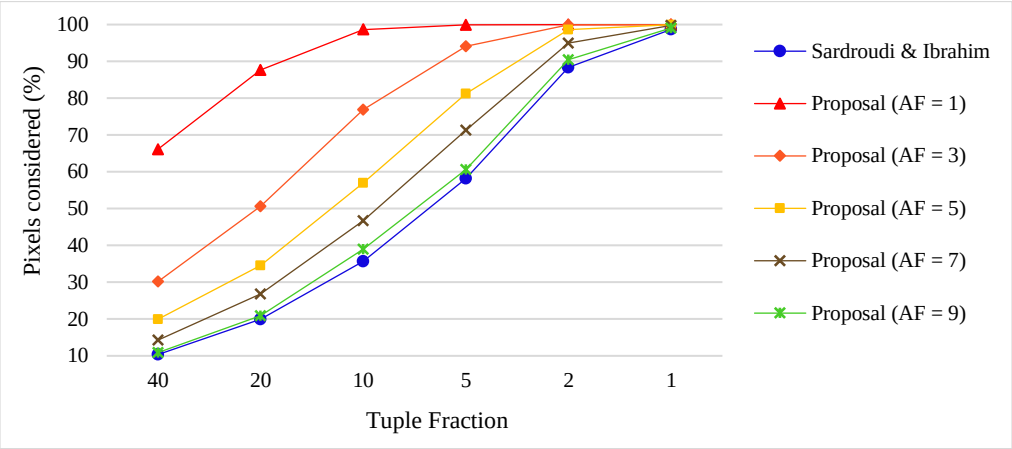


Figure 14. Percentage of image pixels considered for the WM generation.

A higher value of AF (AF = 9) guarantees almost (a little higher percentage) the same results of the considered pixels for the WM generation than the proposal of Sardroudi & Ibrahim [65]. When the AF value decreases, the number of embedded pixels increases, and when TF = 1, almost all pixels are considered. To avoid data quality degradation, we consider the parameters combination that allows the WM identification even when not all pixels are embedded. Table 3 shows the extracted image and its corresponding CF according to different parameters used for the WM embedding.

Table 3. Embedded image for different parameter values.

Method	Tuple Fraction					
	40	**20**	**10**	**5**	**2**	**1**
Sardroudi & Ibrahim						
	10	19	35	58	88	98
Proposal (AF = 9)						
	10	20	38	60	90	99
Proposal (AF = 6)						
	16	30	50	75	97	99
Proposal (AF = 3)						
	30	50	76	94	99	99
Proposal (AF = 1)						
	66	87	98	99	99	99

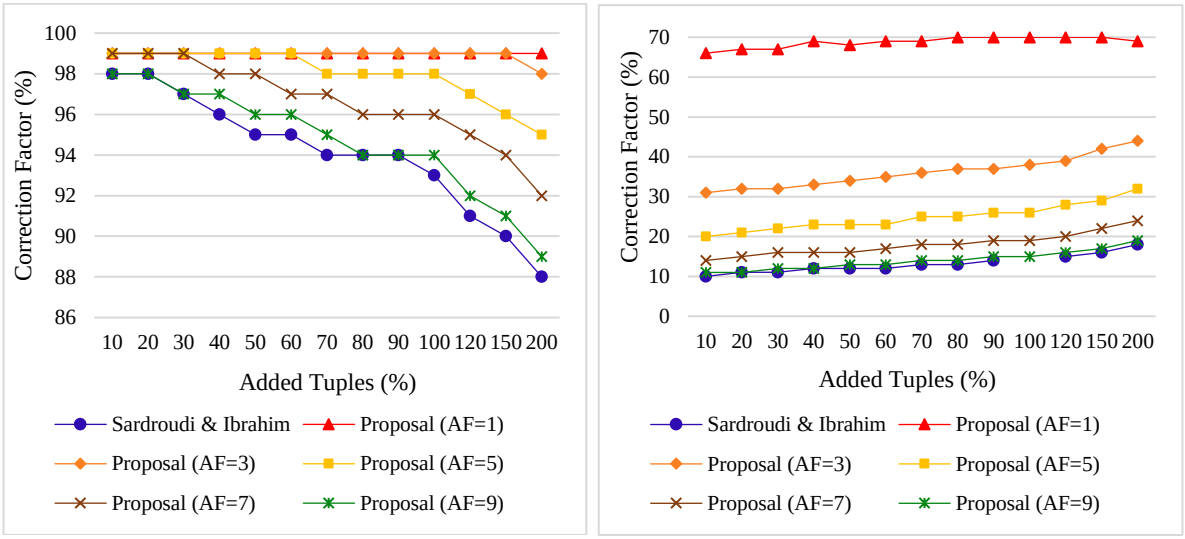
Figure 14 and Table 3 give a clear idea of the minimum number of pixels required for embedding, avoiding to mark a high number of attributes for not compromising the usability of the data and still so, have a clear detection of the WM. Even so, it is important to understand that embedding a higher number of pixels guarantees the robustness of the WM against malicious attacks and benign updates.

4.2 Robustness Analysis

The experiments carried out for the robustness analysis were focused on the *Subset* and *Superset Attacks*. The results show the resiliency of our proposal against malicious modifications. Figures 15-18 show how even for a high percentage of alterations of the watermarked data, the WM can remain. It is important to understand that in some cases (e.g. *Tuple Addition*) the attacker may not try to modify the data with the same or higher severity that we do in our experiments. That would compromise the data quality and would make it useless for his purposes, even so, we stress the WM detection for some cases to see if it could be extracted and identified anyway.

For the *Tuple Addition Attack* (see Figure 15) the WM is easy to identify despite the addition of even 200% of new tuples with values generated randomly inside the set and domain of the original values. For TF = 1 (highly aggressive of the embedding), using different values of the AF, our technique guarantees a better resilience than Sardroudi & Ibrahim (see Figure 15 a)) work. Even so, the main tendency is the degradation of the WM as the number of added tuples increases.

For TF = 40 (less aggressive embedding), the CF is close to the obtained for Sardroudi & Ibrahim in the case of AF = 9, but for other cases it improves the WM recognition (see Figure 15 b)). Despite the low values of the CF, the main tendency of the WM here is to improve in a discrete way.



a) CF variation (TF = 1). b) CF variation (TF = 40).

Figure 15. CF for tuples addition attack.

Considering that a TF of 40 is not an aggressive embedding, it is interesting to know at which level the embedding can be conducted to get better parameters, avoiding a high data distortion. And even guaranteeing that tuple addition attacks, far for compromising the WM, it contributes to its enhancement thanks to the random nature of the attacks and the majority voting used in the extraction

process. Figure 16 shows the tendencies of the WM using different TF for the case of AF=7 (low embedding degree at attribute level). From this experiment, it is observed that the value of TF that may guarantee a change in the negative impact of the *Tuple Addition Attack* is between 5 and 10.

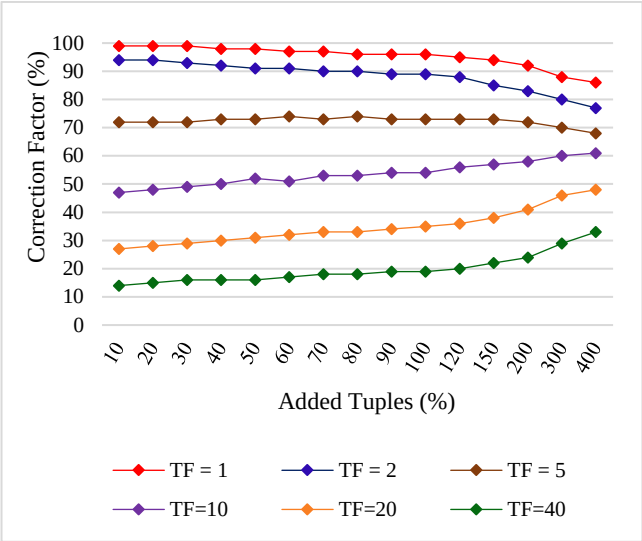


Figure 16. Tuples addition attack (AF = 7, different TF).

For the *Tuple Deletion Attacks* (see Figures 17 and 18), the WM degradation is directly linked to the TF and AF values. For example, for TF=1 (see Figure 17) there is a high resilience of the WM. The different AF always guarantees a high CF. The WM detection problem begins when the dropped tuples are more than the 90%, but in this case, the data usability is completely lost.

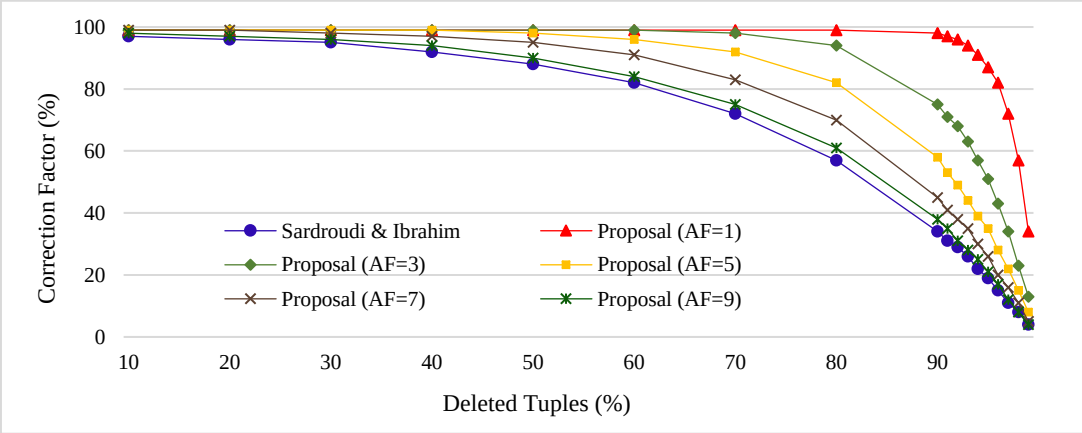
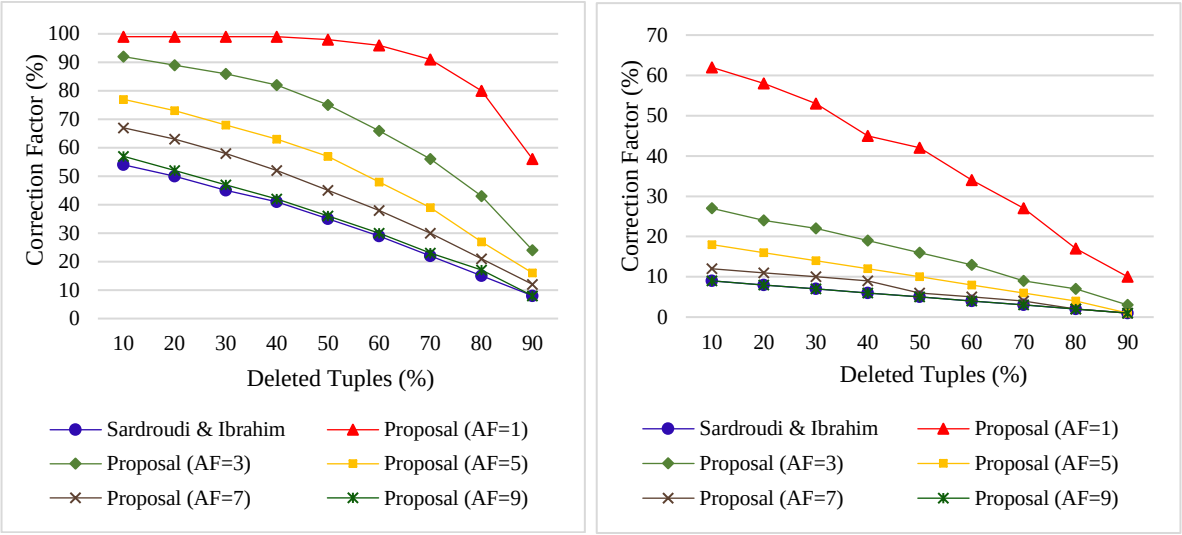


Figure 17. CF for tuples deletion attack (TF = 1).

For the case of TF=5 (see Figure 18 a)), the Tuple Deletion Attack starts to compromise the WM recognition with a lower modification than the used in the case of TF=1, but the WM still can be recognized adding the false value for the missed pixels. According to the Table 3, even for a CF of 35, if the low value is due to missed values, the WM can be identified. So, the key value for

compromising the WM robustness drops between the 60% and 70% of the tuples. That compromises the usability of the watermarked data once more, considering losing more than 50% of the information.



a) CF variation (TF = 5). b) CF variation (TF = 40).

Figure 18. CF for tuples deletion attack.

Finally, for the case of the highest value of TF according to our experiments (see Figure 18 b)), the WM is seriously compromised. Here, a limit is clearly appreciated for the low change due to embedding. So, if the data owner wants to guarantee the resilience of the WM against the *Tuple Deletion Attack*, he must consider a TF value lower than 40.

5. Conclusions

Designing a distortion-based robust watermarking technique for RD that guarantees the WM persistence despite benign updates and malicious attacks without compromising the data usability is a non-trivial activity. It is required to randomly embed each mark composing the WM several times with no dependence of the data to be marked and controlling the distortion in the process. Defined RDB constrains must not be violated as well as the semantic defined over the DB.

IBW schemes have emerged as a promising alternative to embed small WMs on RD, allowing the redundant embedding and guaranteeing the reconstruction of the signal after their extraction. Binary images are even better considering the simplicity of the signal and the existence of techniques for improving it in case of noise addition. Yet, there are limitations still unsolved, that in our opinion, not all the data capacity for embedding the marks is used. Also, most techniques depending on the relation PK, qualified as robust, have not been tested against important types of attacks.

Preliminary results show that it is possible to increase the WM capacity without necessarily compromising the data usability. On the other hand, using the right parameters allows to embed a low-level distortion WM that far from being compromised it is discretely benefited from addition tuple attacks. For example, marking only 13% of the attributes and under a severe attack of 100% of tuples addition, 96% of the watermark can be extracted. Also, while previous techniques embed up to 61% of the watermark, for the same conditions we guarantee embedding 99.96% of the marks using random selection.

References

[1]. Agrawal, R., Haas, P.J. and Kiernan, J., 2003. A system for watermarking relational databases. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data* (pp. 674-674). ACM.

[2]. Agrawal, R., Haas, P.J. and Kiernan, J., 2003. Watermarking relational data: framework, algorithms and analysis. *The VLDB journal*, 12(2), pp.157-169.

[3]. Agrawal, R. and Kiernan, J., 2002, August. Watermarking relational databases. In *Proceedings of the 28th international conference on Very Large Data Bases* (pp. 155-166). VLDB Endowment.

[4]. Aishwarya, C., Aishwarya, S. and Sathish Saravanan, P., 2015. Reversible Watermarking Technique based on Time Stamping in a Relational Data. International *Journal of Scientific Research in Science, Engineering and Technology (IJSRSET).* (Vol. 1. Issue: 2, pp. 230 – 232, ISSN: 2394-4099).

[5]. Al-Najjar, Y.A. and Soong, D.C., 2012. Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI. *International Journal of Scientific & Engineering Research*, *3*(8), p.1.

[6]. Auguste, K., 1883. La cryptographie militaire. *Journal des sciences Militaires*, 9, p.538.

[7]. Baran, B., Gomez, S. and Bogarin, V., 2001, January. Steganographic watermarking for documents. In *System Sciences, 2001. Proceedings of the 34th Annual Hawaii International Conference on* (pp. 10-pp). IEEE.

[8]. Barni, M. and Bartolini, F. eds., 2004. Watermarking systems engineering: enabling digital assets security and other applications. CRC Press.

[9]. Bassia, P., Pitas, I. and Nikolaidis, N., 2001. Robust audio watermarking in the time domain. *Multimedia, IEEE Transactions on*, 3(2), pp.232-241.

[10]. Bhattacharya, S. and Cortesi, A., 2009, December. A Distortion Free Watermark Framework for Relational Databases. In *Proceedings of the 4th International Conference on Software and Data Technologies (ICSOFT '09)*, pp.229–234, Sofia, Bulgaria. INSTICC Press.

[11]. Bhattacharya, S. and Cortesi, A., 2009, December. A generic distortion free watermarking technique for relational databases. In *International Conference on Information Systems Security* (pp. 252-264). Springer Berlin Heidelberg.

[12]. Bhattacharya, S. and Cortesi, A. (2010). Database authentication by distortion-free watermarking. In *Proceedings of the 5th* International *Conference on Software and Data Technologies (ICSOFT '10)*, pages 219–226, Athens, Greece. INSTICC Press.

[13]. Cao, Z., Sun, J. and Hu, Z., 2010. Image algorithm for watermarking relational databases based on chaos. In *Advances in Wireless Networks and Information Systems* (pp. 411-418). Springer Berlin Heidelberg.

[14]. Chan, T.F., Esedoglu, S. and Nikolova, M., 2005, September. Finding the global minimum for binary image restoration. In *IEEE International Conference on Image Processing 2005* (Vol. 1, pp. I-121). IEEE.

[15]. Chang, C.C., Nguyen, T.S. and Lin, C.C., 2013. A blind reversible robust watermarking scheme for relational databases. *The Scientific World Journal*, *2013*.

[16]. Chang, C.C., Nguyen, T.S. and Lin, C.C., 2014, October. A Blind Robust Reversible Watermark Scheme for Textual Relational Databases with Virtual Primary Key. In *International Workshop on Digital Watermarking* (pp. 75-89). Springer International Publishing.

[17]. Cox, I., Miller, M., Bloom, J., Fridrich, J. and Kalker, T., 2007. *Digital watermarking and steganography*. Morgan Kaufmann.

[18]. Cox, I.J. and Miller, M.L., 1997, June. Review of watermarking and the importance of perceptual modeling. In *Electronic Imaging'97* (pp. 92-99). International Society for Optics and Photonics.

[19]. Darwen, H., 2009. *An Introduction to Relational Database Theory*. Bookboon.

[20]. Diwan, T. D. and Sahu, V., 2013. Robust and Reversible Relational Database Protection Using Watermarking Technique. International *Journal for Advance Research in Engineering and Technology (IJARET).* (Vol. 1. Issue: 7, pp. 7 – 12, ISSN: 2320-6802).

[21]. Farfoura, M.E., Horng, S.J. and Wang, X., 2013. A novel blind reversible method for watermarking relational databases. *Journal of the Chinese Institute of Engineers*, *36*(1), pp.87-97.

[22]. Farfoura, M.E., Horng, S.J., Lai, J.L., Run, R.S., Chen, R.J. and Khan, M.K., 2012. A blind reversible method for watermarking relational databases based on a time-stamping protocol. *Expert Systems with Applications*, *39*(3), pp.3185-3196.

[23]. Forest CoverType, The UCI KDD Archive. Information and Computer Science. University of California, Irvine. Source: http://kdd.ics.uci.edu/databases/covertype/covertype.html.

[24]. Franco-Contreras, J., Coatrieux, G., Cuppens, F., Cuppens-Boulahia, N. and Roux, C., 2014. Robust lossless watermarking of relational databases based on circular histogram modulation. IEEE *transactions on information forensics and security*, *9*(3), pp.397-410.

[25]. Guo, H., Li, Y., Liu, A. and Jajodia, S., 2006. A fragile watermarking scheme for detecting malicious modifications of database relations. *Information Sciences*, 176(10), pp.1350-1378.

[26]. Guo, F., Wang, J. and Li, D., 2006, April. Fingerprinting relational databases. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 487-492). ACM.

[27]. Guo, F., Wang, J., Zhang, Z., Ye, X. and Li, D., 2005, August. An improved algorithm to watermark numeric relational data. In *International Workshop on Information Security Applications* (pp. 138-149). Springer Berlin Heidelberg.

[28]. Gupta, G. and Pieprzyk, J., 2009, December. Database relation watermarking resilient against secondary watermarking attacks. In *International Conference on Information Systems Security* (pp. 222-236). Springer Berlin Heidelberg.

[29]. Gupta, G. and Pieprzyk, J., 2008, January. Reversible and blind database watermarking using difference expansion. In *Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop* (p. 24). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[30]. Gursale, N. and Arti M., 2014. A Robust, Distortion Minimization Fingerprinting Technique for Relational Database. *International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC).* (Vol. 2. Issue: 6, pp. 1737 – 1741, ISSN: 2321-8169).

[31]. Halder, R. and Cortesi, A., 2010, December. A persistent public watermarking of relational databases. In *International Conference on Information Systems Security* (pp. 216-230). Springer Berlin Heidelberg.

[32]. Halder, R. and Cortesi, A., 2010, October. Persistent watermarking of relational databases. In *Proceedings of the IEEE International Conference on Advances in Communication, Network, and Computing (CNC 2010), October* (pp. 4-5).

[33]. Halder, R., Pal, S. and Cortesi, A., 2010. Watermarking Techniques for Relational Databases: Survey, Classification and Comparison. *J. UCS*, 16(21), pp.3164-3190.

[34]. Hanyurwimfura, D., Liu, Y. and Liu, Z., 2010, June. Text format based relational database watermarking for non-numeric data. In *Computer Design and Applications (ICCDA), 2010 International Conference on* (Vol. 4, pp. V4-312). IEEE.

[35]. He, X., Ji, M., Zhang, C. and Bao, H., 2011. A variance minimization criterion to feature selection using laplacian regularization. *IEEE transactions on pattern analysis and machine intelligence*, *33*(10), pp.2013-2025.

[36]. Hore, A. and Ziou, D., 2010, August. Image quality metrics: PSNR vs. SSIM. In *Pattern recognition (icpr), 2010 20th international conference on* (pp. 2366-2369). IEEE.

[37]. Huang, M., Cao, J., Peng, Z. and Fang, Y., 2004, September. A new watermark mechanism for relational data. In Computer *and Information Technology, 2004. CIT'04. The Fourth International Conference on* (pp. 946-950). IEEE.

[38]. Hu, T.L., Chen, G., Chen, K. and Dong, J.X., 2005, October. Garwm: Towards a generalized and adaptive watermark scheme for relational data. In *International Conference on Web-Age Information Management* (pp. 380-391). Springer Berlin Heidelberg.

[39]. Hu, Z., Cao, Z. and Sun, J., 2009, April. An image based algorithm for watermarking relational databases. In *Measuring Technology and Mechatronics Automation, 2009. ICMTMA'09. International Conference on* (Vol. 1, pp. 425-428). IEEE.

[40]. Iftikhar, S., Kamran, M. and Anwar, Z., 2015. RRW—A Robust and Reversible Watermarking Technique for Relational Data. *IEEE Transactions on Knowledge and Data Engineering*, *27*(4), pp.1132-1145.

[41]. Jawad, K. and Khan, A., 2013. Genetic algorithm and difference expansion based reversible watermarking for relational databases. *Journal of Systems and Software*, 86(11), pp.2742-2753.

[42]. Jiang, C., Chen, X. and Li, Z., 2009, August. Watermarking relational databases for ownership protection based on DWT. In *Information Assurance and Security, 2009. IAS'09. Fifth International Conference on* (Vol. 1, pp. 305-308). IEEE.

[43]. Johnson, N.F., Duric, Z. and Jajodia, S., 2001. *Information Hiding: Steganography and Watermarking-Attacks and Countermeasures: Steganography and Watermarking: Attacks and Countermeasures* (Vol. 1). Springer Science & Business Media.

[44]. Kamel, I., 2009. A schema for protecting the integrity of databases. *computers & security*, *28*(7), pp.698-709.

[45]. Katzenbeisser, S. and Petitcolas, F., 2016. *Information hiding*. Artech House.

[46]. Katzenbeisser, S. and Petitcolas, F., 2000. *Information hiding techniques for steganography and digital watermarking*. Artech house.

[47]. Khan, A. and Husain, S.A., 2013. A fragile zero watermarking scheme to detect and characterize malicious modifications in database relations. *The Scientific World Journal*, 2013.

[48]. Khanduja, V. and Verma, O.P., 2012. Identification and Proof of Ownership by Watermarking Relational Databases. *International Journal of Information and Electronics Engineering*, 2(2), p.274.

[49]. Kleijnen, J.P., Ridder, A.A. and Rubinstein, R.Y., 2013. *Variance reduction techniques in Monte Carlo methods* (pp. 1598-1610). Springer US.

[50]. Li, Y. and Deng, R.H., 2006, March. Publicly verifiable ownership protection for relational databases. In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security* (pp. 78-89). ACM.

[51]. Li, Y., Guo, H. and Jajodia, S., 2004, October. Tamper detection and localization for categorical data using fragile watermarks. In *Proceedings of the 4th ACM workshop on Digital rights management* (pp. 73-82). ACM.

[52]. Li, Z., Liu, J. and Tao, W., Robust and Reversible Relational Database Watermarking Algorithm Based on Clustering and Polar Angle Expansion.

[53]. Li, Y., Swarup, V. and Jajodia, S., 2005. Fingerprinting relational databases: Schemes and specialties. *Dependable and Secure Computing, IEEE Transactions on*, 2(1), pp.34-45.

[54]. Li, Y., Swarup, V. and Jajodia, S., 2003, October. Constructing a virtual primary key for fingerprinting relational data. In *Proceedings of the 3rd ACM workshop on Digital rights management* (pp. 133-141). ACM.

[55]. Liu, S., Wang, S., Deng, R.H. and Shao, W., 2004, December. A block oriented fingerprinting scheme in relational database. In *International Conference on Information Security and Cryptology* (pp. 455-466). Springer Berlin Heidelberg.

[56]. Menezes, A.J., Van Oorschot, P.C. and Vanstone, S.A., 1996. *Handbook of applied cryptography*. CRC press.

[57]. Nithyavani, G. and Rajesh, P., 2016. Secure Spread, Robustic and Reversible Watermarking Techniques for Relational Databases. *International Journal of Engineering Development and Research (IJEDR).* (Vol. 4. Issue: 1, pp. 150 – 153, ISSN: 2321-9939).

[58]. Odeh, A. and Al-Haj, A., 2008. Robust and blind watermarking of Relational Database System. *Journal of Computer Science*, *4*(12), pp.1024-1029.

[59]. Paul, A. and Sunitha, E.V., 2015, March. Distortion less watermarking of relational databases based on circular histogram modulation. In *Circuit, Power and Computing Technologies (ICCPCT), 2015 International Conference on* (pp. 1-5). IEEE.

[60]. Paul, A. and Sunitha, E.V., 2015. Watermarking of Relational Databases using Video. *International Journal of Science, Engineering and Technology Research (IJSETR)*. (Vol. 4. Issue: 4, pp. 790 – 794, ISSN: 2278 – 7798).

[61]. Potdar, V.M., Han, S. and Chang, E., 2005, August. A survey of digital image watermarking techniques. In *Industrial Informatics, 2005. INDIN'05. 2005 3rd IEEE International Conference on* (pp. 709-716). IEEE.

[62]. Pournaghshband, V., 2008, March. A new watermarking approach for relational data. In *Proceedings of the 46th Annual Southeast Regional Conference on XX* (pp. 127-131). ACM.

[63]. Prasannakumari, V., 2009. A robust tamperproof watermarking for data integrity in relational databases. *Research Journal of Information Technology*, *1*(3), pp.115-121.

[64]. Rao, U.P., Patel, D.R. and Vikani, P.M., 2012. Relational database watermarking for ownership protection. *Procedia Technology*, 6, pp.988-995.

[65]. Sardroudi, H.M. and Ibrahim, S., 2010, November. A new approach for relational database watermarking using image. In *Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on* (pp. 606-610). IEEE.

[66]. Shehab, M., Bertino, E. and Ghafoor, A., 2008. Watermarking relational databases using optimization-based techniques. *Knowledge and Data Engineering, IEEE Transactions on*, 20(1), pp.116-129.

[67]. Shen, Y., Lam, E.Y. and Wong, N., 2008. A signomial programming approach for binary image restoration by penalized least squares. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *55*(1), pp.41-45.

[68]. Sion, R., 2004, March. Proving ownership over categorical data. In *Data Engineering, 2004. Proceedings. 20th International Conference on* (pp. 584-595). IEEE.

[69]. Sion, R., Atallah, M. and Prabhakar, S., 2005. Rights protection for categorical data. *IEEE Transactions on Knowledge and Data Engineering*, *17*(7), pp.912-926.

[70]. Sion, R., Atallah, M. and Prabhakar, S., 2004. Rights protection for relational data. *Knowledge and Data Engineering, IEEE Transactions on*, 16(12), pp.1509-1525.

[71]. Sun, J., Cao, Z. and Hu, Z., 2008, December. Multiple watermarking relational databases using image. In *MultiMedia and Information Technology, 2008. MMIT'08. International Conference on* (pp. 373-376). IEEE.

[72]. Swanson, M.D., Zhu, B. and Tewfik, A.H., 1998. Multiresolution scene-based video watermarking using perceptual models. *Selected Areas in Communications, IEEE Journal on*, 16(4), pp.540-550.

[73]. Swanson, M.D., Zhu, B. and Tewfik, A.H., 1996, September. Transparent robust image watermarking. In *Image Processing, 1996. Proceedings., International Conference on* (Vol. 3, pp. 211-214). IEEE.

[74]. Takmare, S.B., Gupta, R.K. and Chandel, G.S., 2012. Voice Based Watermarking Technique for Relational Databases. *International Journal of Scientific & Technology Research*, *1*(10), pp.65-67.

[75]. Thilagam, T. and Vinoth, R., 2016. RRW - A Resilient Reversible Watermarking Technique for the Preclusion of Information from Cyber Punkers. *International Journal of Engineering Research (IJOER)*. (Vol. 4. Issue: 1, pp. 141 – 146, ISSN: 2321-7758).

[76]. Tsai, M.H., Hsu, F.Y., Chang, J.D. and Wu, H.C., 2007, November. Fragile database watermarking for malicious tamper detection using support vector regression. In *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on* (Vol. 1, pp. 493-496). IEEE.

[77]. Tsai, M.H., Tseng, H.Y. and Lai, C.Y., 2006, October. A Database Watermarking Technique for Temper Detection. In Proceedings *of the 2006 Joint Conference on Information Sciences (JCIS '06)*, Kaohsiung, Taiwan. Atlantis Press.

[78]. Wang, C., Wang, J., Zhou, M., Chen, G. and Li, D., 2008. Atbam: An arnold transform based method on watermarking relational data. In *2008 International Conference on Multimedia and Ubiquitous Engineering (MUE 2008)*.

[79]. Wang, H., Cui, X. and Cao, Z., 2008, May. A speech based algorithm for watermarking relational databases. In *2008 International Symposiums on Information Processing*.

[80]. Warmuth, M.K. and Kuzmin, D., 2012. Online variance minimization. *Machine learning*, *87*(1), pp.1-32.

[81]. Zhang, J. and Ye, W., 2011, October. A fast algorithm for binary image restoration. In *Image and Signal Processing (CISP), 2011 4th International Congress on* (Vol. 2, pp. 590-593). IEEE.

[82]. Zhang, Y., Niu, X. and Zhao, D., 2005. A method of protecting relational databases copyright with cloud watermark. *International Journal of Information and Communication Engineering*, *1*(7), pp.337-341.

[83]. Zhang, Y., Niu, X.M., Wu, D., Zhao, L., Liang, J.C. and Xu, W.J., 2005. A method of verifying relational databases ownership with image watermark. In *The 6th International Symposium on Test and Measurement, Dalian, PR China* (pp. 6316-6319).

[84]. Zhang, Y., Niu, X., Zhao, D., Li, J. and Liu, S., 2006, August. Relational databases watermark technique based on content characteristic. In *First International Conference on Innovative Computing, Information and Control-Volume I (ICICIC'06)* (Vol. 3, pp. 677-680). IEEE.

[85]. Zhang, Y., Yang, B. and Niu, X.M., 2006. Reversible watermarking for relational database authentication. *Journal of Computers*, *17*(2), pp.59-66.

[86]. Zhang, Z.H., Jin, X.M., Wang, J.M. and Li, D.Y., 2004, August. Watermarking relational database using image. In *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on* (Vol. 3, pp. 1739-1744). IEEE.

[87]. Zhou, X., Huang, M. and Peng, Z., 2007, March. An additive-attack-proof watermarking mechanism for databases' copyrights protection using image. In *Proceedings of the 2007 ACM symposium on Applied computing* (pp. 254-258). ACM.