Robust Watermarking of Relational Databases with Ontology-Guided Distortion Control

Javier Franco-Contreras, *Member, IEEE,* Gouenou Coatrieux, *Senior Member, IEEE,*

***Abstract*—In this paper, we present a new robust database watermarking scheme the originality of which stands on a semantic control of the data distortion and on the extension of Quantization Index Modulation (QIM) to circular histograms of numerical attributes. The semantic distortion control of the embedding process we propose relies on the identification of existing semantic links in between values of attributes in a tuple by means of an ontology. By doing so, we avoid incoherent or very rare record occurrences which may bias data interpretation or betray the presence of the watermark. In a second time, we adapt QIM to database watermarking. Watermark embedding is conducted by modulating the relative angular position of the circular histogram center of mass of one numerical attribute. We theoretically demonstrate the robustness performance of our scheme against most common attacks (i.e., tuple insertion and deletion). This makes it suitable for copyright protection, owner identification or traitor tracing purposes. We further verify experimentally these theoretical limits within the framework of a medical database of more than one half million of inpatient hospital stay records. Under the assumption imposed by the central limit theorem, experimental results fit the theory. We also compare our approach with two efficient schemes so as to prove its benefits.**

***Index Terms*—Watermarking, Relational Database, Information security, Ontology.**

I. Introduction

The last few years have seen a remarkable increase in the construction, transfer and sharing of databases. This is mainly due to the reinforcement of their economical value and decisional interest, the latter being related in part to the progress of data mining and analysis tools. However, these new access capabilities induce at the same time security risks, as data records may be redistributed or modified without permission. Several examples of information leaks appear each year, even in sensitive areas like defense [1] or health care [2].

Secure access and confidentiality of data are usually achieved by means of cryptographic mechanisms. Nevertheless, once these mechanisms bypassed or more simply when the access is granted, data are no longer protected. Here comes the interest for watermarking, an *a posteriori* protection, that leaves access to data while maintaining them protected in terms of integrity or traceability as example. Watermarking lies in the insertion of a message (some security attributes) or a watermark into a host document (e.g., image or database) by slightly perturbing host data. More precisely, the insertion process is based on the principle of controlled distortion of host data. Watermarking has been successfully applied in multimedia protection [3]–[5], but database watermarking was only introduced in 2002 by Agrawal *et al*. [6]. Since then, several methods have been proposed [7]–[9].

Depending on the embedding modulation, we can distinguish "attribute-distortion-free" methods, that do not modify attributes values from "attribute-distortion-based" methods. The former are usually based on the modulation of the order of tuples within a relation [10]. If one may consider that no data perturbation has been introduced, such a technique makes the watermark dependent on the way the database is stored, inducing constrains on the database management system. As a consequence, the application range this family of methods can be used for is limited. Moreover, these methods are fragile as any reordering of tuples will eliminate the watermark.

For the second class of methods, it is generally assumed that the normal interpretation of data will not be perturbed if some alteration (e.g., modification of attributes' values [11]) is carried out in the database for message insertion. Nevertheless, in order to take into account watermark imperceptibility, most recent "distortion based" schemes consider distortion constraints. For instance, in [7] the embedding process does not modify numerical attributes if some "data usability conditions", measured in terms of the mean squared error, are not respected. Shehab *et al.* consider additional attribute statistics constraints (e.g., mean, standard deviation) on attribute values and adapt the watermark amplitude by means of optimization techniques [9]. In a recent work [12], Kamran and Farooq go one step further. Their watermarking scheme preserves classification results of a prior data-mining process. To do so, attributes are first grouped according to their importance in the mining process. Some local (i.e., for a set of attributes) and global constraints are then defined in the statistical relations between attributes (e.g., mutual information, information gain, etc.). The allowed perturbation of tuples for a set of attributes is obtained by means of optimization techniques. In [13], the same authors introduce the concept of "once for all" usability constraints considering the application framework where a database is sent to several recipients for different purposes. In their approach, the constraints are established in terms of numerical attributes' mean and standard deviation variations defined by the data owner and recipients. The more restrictive set of variations constitute the "once for all" constraints. They then optimize their detection based on these constraints. If a recipient has lower distortion constraints, he will receive a more distorted database leading to a more robust watermark. In [14], Lafaye *et al.* consider a query result approach and

J. Franco-Contreras and G. Coatrieux* are with Institut Mines-TELECOM, TELECOM Bretagne, Inserm U1101 LaTIM, Brest, 29238 FRANCE and Université européenne de Bretagne, FRANCE e-mail: {javier.francocontreras,gouenou.coatrieux}@telecom-bretagne.eu

Manuscript received May, 2015.

look at preserving the response to *a priori* known queries of aggregation, and modulate pairs of tuples in consequence.

As exposed, the above methods focus on preserving the database statistics (of attributes [9], [13] or in-between attributes [12]) and do not take into account the full database semantics that should also be preserved. Semantics refer to the meaning of a piece of information. For instance, let us consider a medical database having two attributes "gender" and "diagnosis". There exist a strong semantic relation between the "gender" value "female" and the value "pregnancy" of "diagnosis". It would be incoherent to have "gender"="male". Although statistics may provide hints about the existence of such semantic links, as they evaluate the dependencies or the co-occurrences of values in the database, they do not allow directly identifying such a situation. In general, watermarked tuples must remain semantically coherent in order to: i) ensure the correct interpretation of the information without introducing impossible or unlikely records; ii) keep the introduced perturbations invisible to attackers. Indeed, an "impossible" tuple can be statistically insignificant but highly semantically detectable [15].

To do so, we propose a new semantic distortion control method which takes advantage of an ontology over the database scheme. As exposed by Gomez-Perez and Benjamins [16], ontologies provide a common vocabulary of an area and define, with different levels of formality, the meaning of the terms and the relations between them. Ontologies have been successfully applied in several domains from data extraction [17] to image annotation and retrieval [18]. To our knowledge, they have not been yet applied to control watermarking distortion. As we will show, one ontology provides semantic knowledge or description of the database that can help us to identify the allowable attribute distortion in a tuple.

Up to now, different modulations have been considered in order to embed a message into a numerical attribute in a database. We can cite as examples the modification of the least significant bits (LSB) [6], the histogram shifting for categorical attributes [19], the modulation of the relative position of circular histograms in groups of tuples [20] or the insertion of fake records [21]. In this work, our distortion control method is applied in conjunction with an adaptation of Quantization Index Modulation (QIM) [22]; robust modulation which in our knowledge has never been considered in database watermarking. This modulation is used so as to modulate the relative angle of the center of mass of circular histograms associated to groups of values of one numerical attribute of the relation. Moreover, we theoretically prove that the use of QIM leads to a scheme that is robust to the most common attacks in the state of the art: tuple insertion and suppression.

The rest of this paper is organized as follows. In Section II we present the main steps of a common chain of database watermarking before explaining how ontologies can be used in order to control the database distortion in Section III. We introduce our scheme and the modulation it is based on in Section IV. In Section V, we theoretically evaluate the performance of our scheme. We then empirically verify these theoretical results in Section VI, where our scheme is evaluated in terms of distortion and complexity in the case of one real medical database of more than one half million patient stay records. We also compare it with the methods of Sion *et al.* [7] and Shehab *et al.* [9], two efficient schemes from the literature in Section VII. Section IX concludes this paper.

II. A COMMON DATABASE WATERMARKING CHAIN

A database DB is commonly defined as a finite set of relations $\{R_i\}_{i=1,\dots,N_R}$. In this work, for sake of simplicity, we consider a DB with one single relation constituted of N unordered tuples $\{t_u\}_{u=1,\dots,N}$, each of M attributes $\{A_1, A_2, \dots, A_M\}$. An attribute A_n takes its values within an attribute domain and $t_u.A_n$ refers to the value of the n^{th} attribute of the u^{th} tuple. Each tuple is uniquely identified by either one attribute or a set of attributes, we call its primary key $t_u.PK$.

Two fundamental stages are considered in most of database watermarking schemes: message embedding and message detection/extraction. As depicted in Fig. 1, the embedding stage includes a preprocessing process, the objective of which is to make the watermark insertion/reading independent of the way database is stored. It usually consists in the construction of groups of tuples, creating a set of N_g non-intersecting groups of tuples $\{G^i\}_{i=1,\dots,N_g}$.

Typically, the group number for one tuple n_u is obtained from the result of a cryptographic hash function applied to its primary key $t_u.PK$, concatenated with a secret watermarking key K_S such as [9]:

$$n_u = H(K_S|H(K_S|t_u.PK)) \mod N_g \tag{1}$$

where '|' represents the concatenation operator and N_g is the number of groups to build. The use of a cryptographic hash function, e.g., Secure Hash Algorithm (SHA), ensures the secure and equal distribution of tuples into groups.

Thus, if N is the total number of tuples in the database, each group will approximately contain $\frac{N}{N_g}$ tuples. By next, one bit or one symbol s_i of the message is embedded per group. To do so, the values of one or several attributes are modified accordingly to the watermarking modulation retained by the user. Thus, one may expect to embed a message corresponding to a sequence of N_g symbols $S = \{s^i\}_{i=1,\dots,N_g}$.

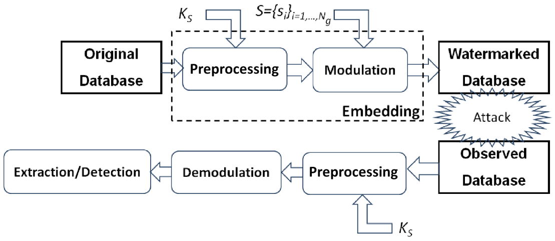


Fig. 1. A common database watermarking chain.

Watermark extraction works in a similar way. First, tuples are distributed into the N_g groups. Depending on the watermarking modulation, one symbol is extracted/detected from each of these groups. If tuple primary keys are not modified, the knowledge of the watermarking key ensures the synchronization between embedding and reading stages.

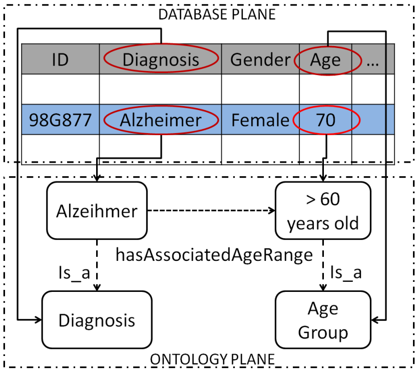


Fig. 2. Existing connection between a relational database and an ontology. Dotted and dashed arrows represent ontological relations between concepts in the ontology. Solid arrows represent connections between attributes or attributes values and ontological concepts.

III. Ontology-guided distortion control

A. Relational databases and ontologies

A relational database aims at providing efficient storage and rapid access to large amounts of data. As exposed in Sect. II, it consists of a finite set of relations $\{R_i\}_{i=1,...,N_R}$ where one relation R_i contains a set of N unordered tuples $\{t_u\}_{u=1,...,N}$, each of which having M attributes $\{A_1, A_2, ..., A_M\}$. However, this data structure lacks of semantic information about the meaning and links between different attributes' values in a tuple. An ontology can be herein useful by offering additional semantic pieces of information about the database content. For a specific area of knowledge, an ontology provides a common vocabulary and defines, with different levels of formality, the meaning of the terms and the relations between them [16]. It is composed of concepts, which represent objects or sets of objects within a domain. Concepts in an ontology are linked by means of relations that specify hierarchical or associative interactions between them. Notice that an ontology can be derived from the database by means of data mining operations. However, extracted relations represent only a part of the knowledge one can have about the database content. Notice also that the ontology itself may contain some *a priori* knowledge about concepts and relations statistics, indicating for example if a concept is rare or frequent.

From this standpoint, each domain value, subset or range of values of an attribute A_t can be associated to one ontology concept. We depict in Fig. 2 such a mapping considering the following example. Let us consider a tuple with attributes "diagnosis", "age", ... The value "Alzheimer" in the domain of the attribute "diagnosis" can be associated to a concept "Alzheimer" in a medical ontology. This concept is related to another concept "$\geq$ 60 years old", which can be mapped into a range of possible values for the attribute "age". From a watermarking point of view, this semantic relations make us aware that one attribute age value should not be turned into a value smaller than 60 in a tuple where the "diagnosis" attribute value is "Alzheimer". As exemplified, the value of the attribute A_t in the u^{th} tuple, i.e., $t_u.A_t$, semantically depends on the set $S_{t_u.A_t}$ of values of the other attributes of t_u, i.e., $t_u.\{A_1, ..., A_{t-1}, A_{t+1}, ..., A_M\}$, or a subset of them.

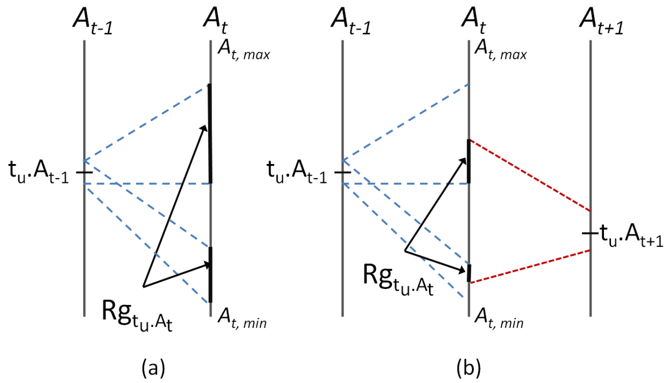


Fig. 3. Identification of allowable range $Rg_{t_u.A_t}$ for an attribute value $t_u.A_t$ from its links with: a) a value in $S_{t_u.A_t}$; b) two values in $S_{t_u.A_t}$. In the first case, $Rg_{t_u.A_t}$ corresponds to the union of different intervals. In the second case, the additional constraints imposed by the second value are represented as the intersection of the allowable ranges imposed by each value in $S_{t_u.A_t}$.

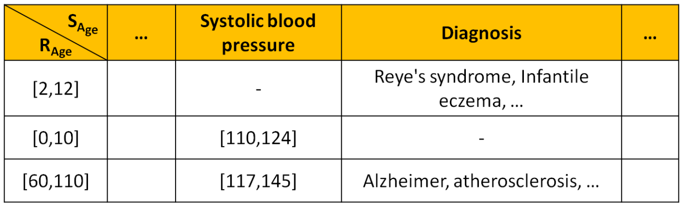
B. Identification of allowable values of a numerical attribute

As a consequence of the assertions exposed in the previous section, we propose to use the concepts and relations of an ontology associated to the database in order to identify the maximum tolerated distortion of its attributes' values. For a numerical attribute A_t, this distortion limit in the tuple t_u is defined by the range of allowable values $t_u.A_t$ can take $Rg_{t_u.A_t}$, under the semantic constraints of $S_{t_u.A_t}$. If we come back to the previous example, where A_t ="age" is an integer, the value $t_u.age$ belongs to an integer range $Rg_{t_u.age}$ imposed by the set $S_{t_u.age}$ ="Alzheimer". In a more general way, if the attribute domain of A_t corresponds to the integer range $[A_{t,min}, A_{t,max}]$, the range $Rg_{t_u.A_t}$ can be defined as the union of N_{rg} different intervals such as: $Rg_{t_u.A_t} = [A_{tmin_1}, A_{tmax_1}] \bigcup ... \bigcup [A_{tmin_{N_{rg}}}, A_{tmax_{N_{rg}}}]$ and $Rg_{t_u.A_t} \subseteq [A_{t,min}, A_{t,max}]$; set of intervals identified from the ontology by querying it considering the other attributes' values in t_u, i.e., $S_{t_u.A_t}$ (see Fig. 3). The knowledge of $Rg_{t_u.A_t}$ will be used as reference to guide the watermark embedding process. It is possible that such a semantic distortion control indicates that an attribute is not appropiate for watermarking. In that situation, an "attribute-distortion-free" watermarking scheme will be more appropriate (see Sect. I).

It is important to notice that the semantic distortion control we propose is complementary to any other statistical distortion control method. For instance, additionally to the ontology constraints one may aim at preserving the correlation or the mutual information between attributes. In a more advanced construction, a global solution associating semantic distortion control and statistics distortion control, such as the technique suggested by Kamran *et al.* [13] can be constructed.

C. Minimization of the number of queries

In practice, the above process requires querying the ontology for each tuple in the database. In order to reduce such a complexity, we propose an ontology preprocessing stage which takes advantage of the fact that in general two numerical attributes have relationships in terms of range of values as



S_Age / R_Age	…	Systolic blood pressure	Diagnosis	…
[2,12]		-	Reye's syndrome, Infantile eczema, …	
[0,10]		[110,124]	-	
[60,110]		[117,145]	Alzheimer, atherosclerosis, …	

Fig. 4. Example of a correspondence table mapping possible ranges of the attribute "age" to associated ranges or sets of values for the other attributes in the relation, in particular the attributes "Systolic blood pressure" and "Diagnosis".

illustrated in the previous paragraph, where $Rg_{t_u.A_t}$ is the range of possible values of A_t in t_u under the constraint $S_{t_u.A_t}$.

Let us generalize and look at this process from the point of view of A_t. From the above, it appears that one of its ranges of values is associated to a range or set of values of the numerical attributes and categorical attributes, respectively, in S_{A_t} (as illustrated in Fig. 3b). Returning to our example with A_t ="age" and as illustrated in Fig. 4, the range of ages $[60, 110]$ can be associated to a range of values $[117, 145]$ of the attribute A_{t+1} ="Systolic blood pressure" and to a set of values {Alzheimer, atherosclerosis,...} of the categorical attribute A_{t+2} ="Diagnosis".

In this context, the preprocessing stage we then propose to perform before the database watermarking process consists in the construction of a correspondence table or mapping between ranges of A_t and ranges of attributes' values in S_{A_t}. Notice that a range of A_t is not necessarily associated to all the attributes in S_{A_t}. The construction of this table is based in the execution of inverse queries going from each value $\{Val_l\}_{l=1,...,L}$ of the domain of A_t to those of S_{A_t}. This results in a set of ranges Val_l may belong to under the constraints of the attributes' values in S_{A_t}.

Figure 4 illustrates such a correspondence table or mapping for the attribute "age" (to be watermarked) in regard with the attributes "Systolic blood pressure" and "Diagnosis". Once the table constructed, it is used during the watermarking process and for one tuple t_u one just has to look for the values of $S_{t_u.A_t}$ in the columns so as to get all the possible ranges of values for the watermarked version of $t_u.A_t$. For instance, as seen in Fig. 4, for the values $t_u.Systolic\,Blood\,Pressure = 113$ and $t_u.Diagnosis$ ="Reye's syndrome", we have $Rg_{t_u.age} = [2, 10]$.

IV. Proposed Watermarking Scheme

The proposed scheme is based on QIM introduced by Chen and Wornell in [22]. In the sequel, we give first QIM principles in the case of discrete signal watermarking. Then, we expose how it is adapted to modulate the phase angle of the vector associated to the center of mass of the circular histogram of one numerical attribute in one group of tuples so as to embed one symbol of message.

A. QIM Modulation and Signals

QIM is based on the quantization of the elements (samples, group of samples or transform coefficients) of a host signal according to a set of quantizers based on codebooks in order to embed the symbols of a message. More clearly, to each symbol s^i issued from a finite set $S = \{s^i_u\}_{u=0,...,U}$ the QIM associates a codebook $\{C_{s^i_u}\}_{u=0,...,U}$ such that:

$$C_{s^i_u} \bigcap C_{s^i_v} = \emptyset \text{ if } u \neq v \quad (2)$$

In order to embed the symbol s^i_u into one element X of the signal, this one is replaced by X_W which corresponds to the nearest element of X in the codebook $C_{s^i_u}$. This process can be seen as:

$$X_W = Q(X, s^i_u) \quad (3)$$

where the function Q returns the nearest element to X in $C_{s^i_u}$. Notice that the watermarking distortion corresponds to the distance between X and X_W. To exemplify this process, let us consider one pixel X of an image, which may take its values from a one-dimensional space $[0, 255]$. We divide this scalar space into non overlapping cells or intervals of equal size. Each cell is then related to only one codebook $\{C_{s^i_u}\}_{u=0,...,U}$ so as to satisfy (2). Consequently, a symbol s^i_u has several representations in $[0, 255]$ and Q corresponds to a scalar quantizer. In the insertion process, if X belongs to a cell that encodes the desired symbol s^i_u, its watermarked version X_W corresponds to the centroid of this cell. Otherwise, X is replaced by the centroid of the nearest cell encoding s^i_u. In the extraction, the knowledge of the cell to which X_W belongs is enough to identify the embedded symbol. This process is illustrated in Fig. 5 in the case of a binary message, i.e., $s^i_u \in 0, 1$ and two codebooks C_0 and C_1 for which the cells are defined according to a uniform scalar quantization of quantization step ρ. In this example, X will be quantized to the nearest square or circle in order to encode s^i_u.

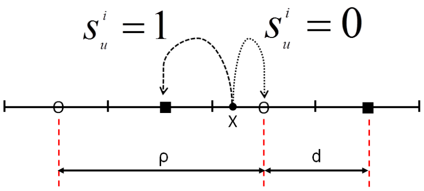


Fig. 5. Example of QIM in the case where X is a scalar value for the embedding of a binary symbol. Codebooks are based on an uniform quantization of quantization step ρ. Cells centered on squares represent $C_1(s^i_u = 1)$ while cells centered on circles represent $C_0(s^i_u = 0)$. Notice that $d = \rho/2$ establishes the measure of robustness to signal perturbations.

An extension of this approach, whose purpose is to reduce the distortion, is the Compensated QIM [22] where a fraction of the quantization error is added back to the quantized value so as to better manage the watermark robustness/imperceptibility tradeoff.

B. Adapted QIM for attribute circular histogram watermarking

In this work, we modulate the angle of the vector associated to the center of mass of the circular histogram of an attribute in a group of tuples. In this section, the steps for the computation of this center of mass are first evoked before explaining QIM codebook construction and presenting our complete scheme.

Calculation of the center of mass: Let us consider the numerical attribute A_t selected for embedding, which takes its values in the integer range $[0, L-1]$. Then, for one group of tuples G^i secretly built as depicted in section II, the histogram of the attribute A_t is calculated and mapped onto a circle. The histogram center of mass C^i of the group G^i and its associated vector V^i are computed, as illustrated in Fig. 6(a). The module and phase of V^i can be calculated from its Cartesian coordinates given by:

$$\begin{array}{c} X_i = \frac{1}{Mass_i}\sum_{l=0}^{L-1} n_l \cos(\frac{2\pi l}{L}) \\ Y_i = \frac{1}{Mass_i}\sum_{l=0}^{L-1} n_l \sin(\frac{2\pi l}{L}) \\ Mass_i = \sum_{l=0}^{L-1} n_l \end{array} \tag{4}$$

where n_l is the cardinality of the circular histogram class l of G^i (i.e., when A_t takes the integer value l). As a consequence, the module of V^i equals $R = \sqrt{X^2 + Y^2}$ and its phase, we also call mean direction μ_i, is given by:

$$\mu_i = \begin{cases} \arctan(Y/X) \text{ if } X > 0 \\ \frac{\pi}{2} \text{ if } X = 0, Y > 0 \\ -\frac{\pi}{2} \text{ if } X = 0, Y < 0 \\ \pi + \arctan(Y/X) \text{ else} \end{cases} \tag{5}$$

In the sequel, in order to embed a symbol s^i into a group G^i, we modulate the value of μ_i. Our choice in working with the circular histogram center of mass stands on the fact that it provides a more robust embedding space against tuple deletion and insertion and attributes' values modification attacks than conducting embedding directly at the attribute value level. More clearly, this feature is less sensitive to such attacks. For instance, the removal of some tuples will not make vary too much the angular position of the center of mass.

Construction of the codebooks: For sake of simplicity, the considered message is a sequence of bits $S = \{0, 1\}$. Thus, two codebooks C_0 and C_1 are necessary. Another simplification we make in this work is that only one cell is associated to each codebook as illustrated in Fig. 6(b). Two questions need then to be answered: the determination of the cell's boundaries and the position of their centroids.

Let us define μ as the mean direction of A_t calculated over all the tuples of the database. Based on the fact attribute circular histograms of tuple groups are all positioned around this mean direction, we decided to define the cells' frontiers as the intersection between μ and the unit circle as illustrated in Fig. 6(b). So in order to encode 0 or 1 the histogram will be rotated to the left or to the right of this frontier.

Unlike the previously presented QIM based on uniform scalar quantization, the centroids C_{q0} and C_{q1} of our cells C_0 and C_1 respectively do not correspond to the cell's center. This allows us to better refine the imperceptibility/robustness trade-off. C_{q0} and C_{q1} are defined as:

$$C_{q0} = \mu - \Delta,\ C_{q1} = \mu + \Delta \tag{6}$$

where Δ corresponds to the rotation angle shift, a user defined parameter that allows controlling the compromise robustness/distortion. As defined the maximum robustness is achieved when $\Delta = \frac{\pi}{2}$ while the maximum distortion is achieved when $\Delta = \pi$. The main difference of our modulation with QIM and compensated QIM exposed above stands in the cell centroid position which is no longer at the cell's center and also on the fact the quantization error is not added back (see Section IV-A).

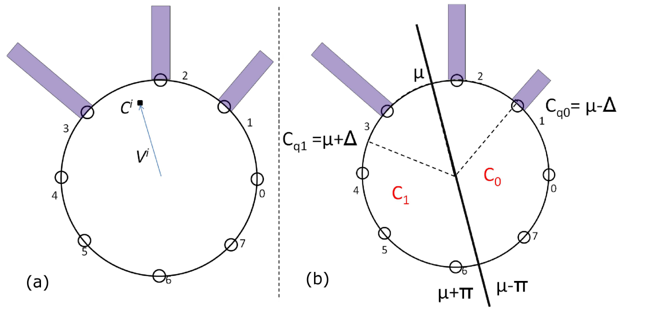


Fig. 6. a) Histogram mapping of one group G^i onto a circle. The angle of the vector pointing its center of mass is modulated in order to embed one message symbol $s^i \in 0, 1$. b) Codebook cells C_0 and C_1 with their corresponding centroids C_{q0} and C_{q1} respectively.

To sum up, each codebook is then associated to a one cell (see Fig. 6(b)) defined as:

$$C_0 = (\mu - \pi, \mu),\ C_1 = (\mu, \mu + \pi) \tag{7}$$

Message embedding and extraction - Complete scheme: Let us now consider the embedding of the binary symbol $\{s^i\}_{i=0,...,N_g-1} = \{0/1\}$ into the group of tuples $\{G^i\}_{i=0,...,N_g-1}$. As stated above, the mean direction value μ_i is replaced by the centroid of the cell coding the value of s^i, resulting in μ_i^w. The embedding of a symbol $s^i = \{0/1\}$ into μ_i can be synthesized as:

$$\mu_i^w = \mu + (2s^i - 1)\Delta \tag{8}$$

where μ_i^w is the watermarked mean direction, Δ is the rotation angle shift that allows the rotation of V^i so as to align it onto the cell centroid. This rotation is performed in the linear domain, i.e., on the attribute histogram, by modifying the attribute's values for certain tuples under distortion constraints (see Sect. III). We come back on attribute modification with more detail in Section IV-C. Regarding the message extraction stage, groups of tuples are reconstructed and angles μ_i^{det} are calculated in each group. It is important to remark that the value of μ should be known by the detector in order to make it possible to reconstruct the dictionaries C_0 and C_1 and extract the message. μ can be sent to the reader as part of the watermarking key or recomputed from a set of reference tuples our algorithm would not have watermarked. More clearly, in this latter case, the embedder does not insert any symbols of the message into the first groups of tuples, as example, allowing the reader to retrieve from them the value of μ.

At the reading stage, the value of μ_i^{det} can differ from μ_i^w in case the watermarked database has been attacked. The cell to which μ_i^{det} belongs allows us to extract the value of s^i in in the i^{th} watermarked group of tuples:

$$s^i = 0 \text{ if } \mu_i^{det} \in C_0,\ s^i = 1 \text{ if } \mu_i^{det} \in C_1 \tag{9}$$

C. Linear histogram modification

As exposed, a rotation of the center of mass vector V_i can be performed by changing the values of the attribute A_t in a certain number of tuples of the group G^i. For instance, if we call α the elementary angle between two consecutive bins of the circular histogram of A_t, $\alpha = \frac{2\pi}{L}$, modifying μ_i of α in the clockwise direction results in adding -1 to the attribute's value for every tuple in the group. Notice that $+1$ (resp. -1) is the minimal perturbation of an integer value.

In practice, we execute an iterative process so as to modify the attribute's values in G^i to rotate V^i onto C_{q0} or C_{q1}. In an iteration, the attribute values in the group are increased of $+1$ (resp. -1) under the distortion constraints extracted from the ontology as exposed in section III, so as to rotate positively (resp. negatively) μ_i^w to make it converge to C_{q1} (resp. C_{q0}). More clearly, one value is not modified if its modification results in a violation of the semantic distortion constraints (see Sect.III). One can compute the number n_{mods} of tuples to be modified of the minimal perturbation $+1$ (resp. -1) in an iteration. n_{mods} depends on the elementary angle α and the number of tuples N_{G^i} in the group:

$$n_{mods} = \text{round}(\frac{|\mu_i^w - C_{q0}|}{\alpha} N_{G^i}) \tag{10}$$

As exposed, after each iteration the distance between μ^w and C_{q1} (resp. C_{q0}) decreases. However because A_t is an integer attribute, being at least modifiable of ± 1, μ_i^w may not reach the codebook cell centroid after an infinite number of iterations. This is why we introduced a user defined parameter ε, such as our algorithm stops when $|\mu_i^w - C_{q0}| < \epsilon$. Notice that the lowest value ϵ can take depends on the attribute. Indeed, due to the fact A_t is an integer, its circular histogram center of mass can be rotated of a minimal angle $\frac{2\pi}{L}\frac{N_g}{N} = \alpha\frac{N_g}{N}$ (i.e., a modification of ± 1 of one individual attribute's value). This results in a minimum value of ϵ for this attribute of a half of this rotation, $\epsilon_{min} = \frac{\alpha}{2}\frac{N_g}{N}$.

V. Theoretical Performance

In this section, we theoretically evaluate the performance of our scheme in terms of robustness against two of the most common database attacks: tuple deletion and tuple insertion. As we will show, robustness depends on the number of groups N_g, the rotation angle shift Δ (on which depends the codebook cell centroids), the statistical distribution of the watermarked mean directions μ_i^w as well as on the strength of the database modifications, i.e percentage of deleted/inserted tuples. Before entering into details, we first introduce some useful results of circular statistics.

A. Preliminary Results

Let us consider the circular data distribution of one attribute θ (i.e., its histogram mapped onto a circle). This can be seen as the p.d.f $f(\theta)$ of a discrete random variable θ which takes L values around the circle in the finite set $\{\frac{2\pi l}{L}\}_{l=0,...,L-1}$. The mean direction μ of θ (or equivalently the phase of the vector associated to the center of mass of θ circular histogram) can be estimated based on a finite number of θ samples. Based on the Law of large numbers, it was shown by Fisher and Lewis [23] that for any circular data distribution $f(\theta)$ the difference between the real mean direction and its estimated value tends to zero as the number of samples used in the estimation tends to ∞. With the help of the central limit theorem, they also proved that the distribution of the mean direction estimator approaches a normal distribution centered on the real mean direction of the circular data distribution.

Considering one numerical attribute, a database of N tuples and N_g groups, we can obtain the variance $\sigma^2_{\mu_i}$ as in [24]

$$\sigma^2_{\mu_i} = \frac{\sigma_s^2}{\frac{N}{N_g} R^2} \tag{11}$$

where: R corresponds to the module of the center of mass vector (i.e., V^i, see Section IV) and σ_s^2 is defined as [24]:

$$\sigma_s^2 = \sum_{l=0}^{L-1} \sin^2(\frac{2\pi l}{L}) f(\frac{2\pi l}{L}) \tag{12}$$

The values $\{\frac{2\pi l}{L}\}_{l=0,...,L-1}$ are the bins of the circular histogram attached to the attribute A_t and $f(\frac{2\pi l}{L})$ their corresponding probabilities. Notice that the above normal distribution assumption of μ_i is verified in the cases when the central limit theorem is empirically verified, i.e., $\frac{N}{N_g} \geq 30$ (see [25] for further details).

B. Robustness Performance

Let us consider the watermarking of one numerical attribute A_t in a database by means of the scheme presented in Sect. IV, where two unique cell codebooks C_0 and C_1 with centroids $C_{q0} = \mu - \Delta$ and $C_{q1} = \mu + \Delta$ respectively are used so as to embed a sequence S of symbols $s^i \in \{0, 1\}$. The result of such an insertion process on the normal distribution of the original mean direction μ^i (see Fig. 7 and section IV) is illustrated in Fig. 8 which gives the p.d.f of the watermarked angles μ_i^w. One can easily identify the centroids of the codebook cells as well as the frontier between the two cells (or codebooks) established by μ.

As exposed in Sect. IV, the modulation of μ_i is performed by introducing a controlled distortion into the integer values of A_t. This modification is carried out by means of an iterative process that stops when $|\mu_i^w - C_{q0}| < \epsilon$ (resp. $|\mu_i^w - C_{q1}| < \epsilon$) with the error ϵ fixed by the user. Thus, contrarily to the QIM, the p.d.f distribution of μ_i^w does not present only two peaks in the cell centroids, but two Gaussians centered in C_{q0} and C_{q1} with a variance that depends on the error ϵ, as seen in Fig. 8 in the case s^i is uniformly distributed (i.e., $\mathbb{P}_0 = \mathbb{P}_1 = \frac{1}{2}$).

Performance in terms of robustness of our scheme depends on the probability that a group of tuples changes of embedded symbol after an attack. We propose to compute these probabilities considering two common database attacks or modifications: tuple addition or tuple removal. To do so, we need to express their impact on the p.d.f of the watermarked angles, i.e., of the random variable μ_i^w given in Fig. 8.

Notice that for the sake of simplicity, we consider in the sequel that the error ϵ, with $|\mu_i^w - C_{q0}| < \epsilon$ (resp. $|\mu_i^w - C_{q1}| < \epsilon$), equals zero. We thus make the hypothesis that the p.d.f $f_{\mu_i^w}(\mu_i^w)$ of μ_i^w is such as:

$$f_{\mu_i^w}(\mu_i^w) = \begin{cases} \mathbb{P}_0 \;\; if \;\; \mu_i^w = \mu - \Delta \\ \mathbb{P}_1 \;\; if \;\; \mu_i^w = \mu + \Delta \\ \quad 0 \;\; otherwise \end{cases} \tag{13}$$

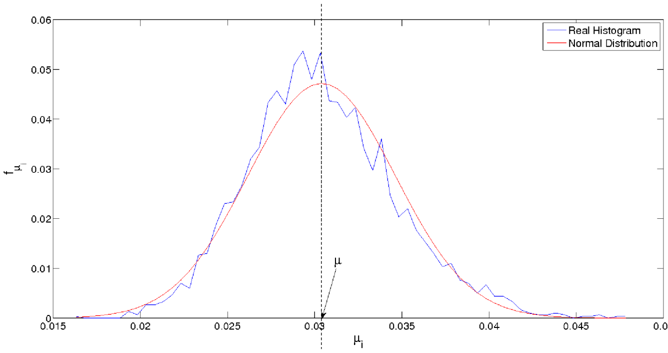


Fig. 7. Distribution of the mean direction μ_i for an exponentially distributed numerical attribute taking its values in $[0, 707]$ ($L = 708$) with a number of groups $N_g = 500$. As shown, the real distribution obtained by means of the normalized histogram perfectly fits a normal distribution with the theoretically calculated statistical moments.

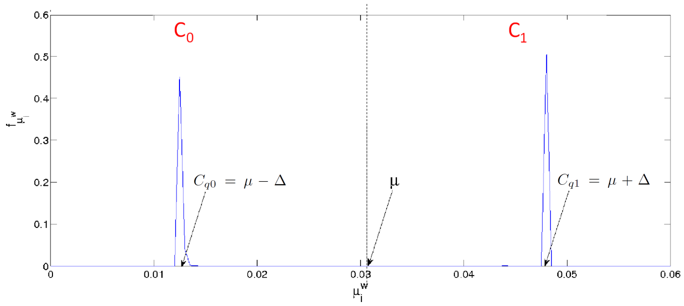


Fig. 8. μ_i^w distribution after the embedding process for an exponentially distributed numerical attribute taking its values in $[0, 707]$ ($L = 708$) with $N_g = 500$ and $\Delta = 2\frac{2\pi}{L} = 0.0177$, in the case s^i is uniformly distributed (i.e., $\mathbb{P}_0 = \mathbb{P}_1 = \frac{1}{2}$)

where $\mathbb{P}_0$ and $\mathbb{P}_1$ are the symbol probabilities (i.e., $\mathbb{P}_0 = Pr(s^i = 0)$, $\mathbb{P}_1 = Pr(s^i = 1)$, $\mathbb{P}_0 + \mathbb{P}_1 = 1$). Under this hypothesis, the probability of symbol error $\mathbb{P}_e$ is reduced increasing as a consequence the theoretical robustness of our scheme. As we will see in the experimental section, ϵ can be made small enough to reach such performance at the price however of time computation increase which depends on the number of iterations of our algorithm (see section IV).

The p.d.f $f_{\mu_i^w}(\mu_i^w)$ of the watermarked angles can be expressed in terms of the conditional p.d.f of μ_i^w given the quantization cell C_0 or C_1 they belong to. Then, we have:

$$f_{\mu_i^w}(\mu_i^w) = \mathbb{P}_0\, f_{\mu_i^w}(\mu_i^w|\mu_i^w \in C_0) + \mathbb{P}_1\, f_{\mu_i^w}(\mu_i^w|\mu_i^w \in C_1) \tag{14}$$

The conditional p.d.f of μ_i^w given cell correspond to:

$$f_{\mu_i^w}(\mu_i^w|\mu_i^w \in C_0) = \begin{cases} 1 \ if \ \mu_i^w = C_{q0} \\ 0 \ otherwise \end{cases} \tag{15}$$

$$f_{\mu_i^w}(\mu_i^w|\mu_i^w \in C_1) = \begin{cases} 1 \ if \ \mu_i^w = C_{q1} \\ 0 \ otherwise \end{cases} \tag{16}$$

Notice that under the assumption of $\epsilon = 0$, each conditional p.d.f can be seen as a normal distribution centered in the cell centroid with a variance equal to zero.

1) Deletion Attack: In this attack, N_d tuples are randomly eliminated. Based on the fact tuples are uniformly distributed into N_g groups (see Sect. II), we can assume that each group G^i loses in average $\frac{N_d}{N_g}$ tuples. This reduction perturbs the accuracy of μ_i^w which by definition is an estimator of the attribute mean direction in the group G^i (see Sect. IV-B). If we consider the previously exposed conditional distributions of μ_i^w given the cell they belong to, this attack only modifies their variances, leaving their means unchanged. We can model this variance modification as the addition of a centered normally distributed random variable ψ to the value of μ_i^w, such as $\mu_i^{del} = \mu_i^w + \psi$.

Due to the fact the tuples in a group have not been watermarked with the same amplitude or distortion, the impact of the deletion attack variably depends on each tuple. This makes impossible to theoretically calculate the value of the variance σ_ψ^2 of ψ and consequently the probability of symbol error $\mathbb{P}_e$. However, we can obtain an upper bound of P_e considering all the tuples in a group are modified with the same maximum distortion. The variance of ψ is then obtained from (17), with σ_s^2 (see Sect. V-A) calculated over all the tuples in the database.

$$\sigma_\psi^2 = \frac{\sigma_s^2}{\frac{N-N_d}{N_g}R^2} \tag{17}$$

The resulting conditional p.d.f $f_{\mu_i^{del}}(\mu_i^{del}|\mu_i^w \in C_0)$ (resp. C_1), i.e., the p.d.f of μ_i after watermarking and deletion attack, is a normal density function given by:

$$\begin{aligned} f_{\mu_i^{del}}(\mu_i^{del}|\mu_i^w \in C_0) \sim \mathcal{N}(C_{q0}, \sigma_\psi^2) \\ f_{\mu_i^{del}}(\mu_i^{del}|\mu_i^w \in C_1) \sim \mathcal{N}(C_{q1}, \sigma_\psi^2) \end{aligned} \tag{18}$$

2) Insertion Attack: In this situation the attacker inserts N_i tuples. Herein, we assume that added attribute values follow the same distribution as the original un-watermarked attribute A_t. As previously, the fact a cryptographic hash function is used to construct groups of tuples (see Sect. II) allows us to consider a uniform distribution of new tuples among these groups $\{G^i\}_{i=1,\dots,N_g}$. Such an attack can thus be modeled by a mixture of two populations: the watermarked tuples and the added un-watermarked tuples with mixture proportion parameters p_1 and p_2 such as $p_2 = 1 - p_1$ with $p_1 = \frac{N}{N+Ni}$, where N is the number of tuples in the original database.

The conditional p.d.f $f_{\mu_i^{ins}}(\mu_i^{ins}|\mu_i^w \in C_0)$ (resp. C_1), i.e., the p.d.f of μ_i after watermarking and tuple insertion, is a normal density function. Its mean $\bar{\mu_i}^{ins,0}$ (resp. $\bar{\mu_i}^{ins,1}$), which corresponds to the conditional mean given that ${\mu_i}^w \in C_0$ (resp. C_1), and its variance $\sigma_{\mu_i^{ins}}$ can be calculated as:

$$\begin{aligned} \bar{\mu_i}^{ins,0} = \bar{E}[{\mu_i}^{ins}|\mu_i^w \in C_0] = p_1 C_{q0} + p_2 \mu \\ \bar{\mu_i}^{ins,1} = \bar{E}[{\mu_i}^{ins}|\mu_i^w \in C_1] = p_1 C_{q1} + p_2 \mu \end{aligned} \tag{19}$$

$$\sigma^2_{\mu_i^{ins}} = p_2^2 \frac{\sigma_s^2}{\frac{N_i}{N_g}R^2} \tag{20}$$

The cell conditional p.d.f are obtained as:

$$\begin{aligned} f(\mu_i^{ins}|\mu_i^w \in C_0) \sim \mathcal{N}(\bar{\mu_i}^{ins,0}, \sigma^2_{\mu_i^{ins}}) \\ f(\mu_i^{ins}|\mu_i^w \in C_1) \sim \mathcal{N}(\bar{\mu_i}^{ins,1}, \sigma^2_{\mu_i^{ins}}) \end{aligned} \tag{21}$$

3) Probabilities of error: The robustness of our scheme is characterized by the symbol error probability $\mathbb{P}_e$, that is to say the probability the symbol of a group changes after an attack. $\mathbb{P}_e$ can be determined through hypothesis testing problem with the following set of hypothesis:

- H_0 corresponds to the case $s_i = 0$, i.e., $\mu_i^w \in C_0$.
- H_1 corresponds to the case $s_i = 1$, i.e., $\mu_i^w \in C_1$.

The probability the watermark reader returns the wrong symbol value, i.e., $\mathbb{P}_e$, results from the acceptance of H_0 (resp. H_1) when the correct hypothesis is H_1 (resp. H_0). Thus, $\mathbb{P}_e$ is calculated as:

$$\mathbb{P}_e = \mathbb{P}_0 Pr(H_1|H_0) + \mathbb{P}_1 Pr(H_0|H_1) \tag{22}$$

$\mathbb{P}_e$ can be refined depending on the database attack::

$$\begin{aligned}\mathbb{P}_{e,del} &= \mathbb{P}_0 \int_{C_1} f(\mu_i^{del}|H_0) d\mu_i^{del} + \mathbb{P}_1 \int_{C_0} f(\mu_i^{del}|H_1) d\mu_i^{del} \\ \mathbb{P}_{e,ins} &= \mathbb{P}_0 \int_{C_1} f(\mu_i^{ins}|H_0) d\mu_i^{ins} + \mathbb{P}_1 \int_{C_0} f(\mu_i^{ins}|H_1) d\mu_i^{ins}\end{aligned} \tag{23}$$

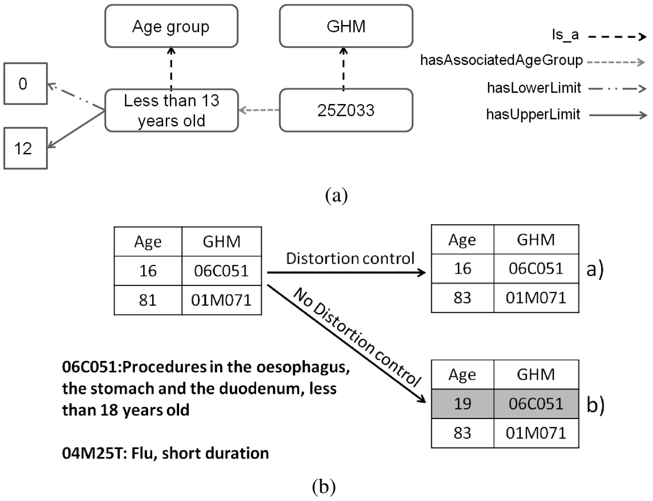
where $\mathbb{P}_{e,del}$ and $\mathbb{P}_{e,ins}$ correspond to the probability of symbol error under a deletion and an insertion attack respectively

VI. Experimental Results

A. Experimental dataset and Ontology

The following experiments have been conducted on a test database constituted of one relation of 508000 tuples issued from one real medical database containing pieces of information related to inpatient stays in French hospitals. In this table, each tuple associates fifteen attributes like the hospital identifier (*id_hospital*), the patient stay identifier (*id_stay*), the patient age (*age*), the stay duration (*dur_stay*), the attribute GHM (patient homogeneous group), the attribute ICD10 principal diagnosis and several other data useful for statistical analysis of hospital activities. If *age* and *dur_stay* are numerical attributes, GHM and ICD10 are categorical attributes. GHM is the French equivalent of the the Diagnosis-Related Groups (DRG) of the Medicare system in the USA. Its attribute domain consists in a list of codes intended for treatment classification and reimbursement. A GHM code results from a function that takes as input the patient age, the ICD10 principal and associated diagnostics, the stay duration, and several others element we can not detail herein due to space limitation. In this experiment, for sake of simplicity, we summed up the domain ontology to the relations between the attribute GHM, age and stay duration. More clearly, our ontology represents a subset of the rules associated to the calculation of the GHM codes. For instance, as depicted in Fig. 9(a), the code "$25Z033$: VIH related disease, age lower than 13 years old, level 3" is related to the group of ages "Less than 13 years old" which corresponds to a numerical range $(0, 12)$. Notice that in our implementation, the domain ontology was implemented in Protégé [26] and queried by means of the SPARQL query language.

In order to constitute the groups of tuples (see Sect. II), the attributes *id_stay* and *id_hospital* were considered as the primary key. Two numerical attributes were considered for message embedding and watermarked independently: patient age (*age*) and stay duration (*dur_stay*). These attributes were chosen because of the specificity of their distributions. The attribute *age* is slightly uniformly distributed, while the attribute *dur_stay* presents an exponential distribution concentrated over the lower values of its domain.



(a)

Age	GHM
16	06C051
81	01M071

Age	GHM
16	06C051
83	01M071

Age	GHM
19	06C051
83	01M071

06C051:Procedures in the oesophagus, the stomach and the duodenum, less than 18 years old

04M25T: Flu, short duration

(b)

Fig. 9. a) Example of concepts and relations in the domain ontology used in the identification of semantic distortion limits. b) Example of modification of two tuples taking and not into account semantic distortion limits. Semantically incorrect tuples are highlighted.

Illustrative example of the ontology interest: An example presenting the advantage of controlling semantically the database distortion by means of an ontology is given in Fig. 9(a). This latter shows an extract of the original database with only two tuples and the corresponding watermarked database extracts with and without semantic distortion constraints, i.e., tables a) and b) respectively. As it can be seen from table a) and b), taking into account the ontology avoids the apparition of incoherent tuples. Indeed, the GHM code $06C051$ corresponds to patients younger than 18 years old, if this constraint is satisfied in table a), this is not the case in table b) where the watermarked age value is 19 (see the shaded tuple). Such an incoherent value makes the tuple suspect to an attacker and can perturb the normal interpretation of data in a subsequent data-mining process.

B. Performance criteria

The performance of our scheme is evaluated in terms of statistical distortion, robustness against tuple suppression and insertion attacks and complexity. In order to get a global vision of the variation of the attribute's distribution, we quantify its statistical distortion through the variations of the attribute's mean and standard deviation, the Kullback-Leibler divergence (D_{KL}) (see (24)) and the mean absolute error (MAE) (see (25)) between histograms of the attribute before and after watermark embedding. If we call h_{A_t} and $h_{A_t^{wat}}$ the histograms of the original attribute A_t and of its watermarked version A_t^{wat} respectively, we have:

$$D_{\mathrm{KL}}(h_{A_t} \| h_{A_t}^{wat}) = \sum_{l=0}^{L-1} \ln\left(\frac{h_{A_t}(l)}{h_{A_t^{wat}}(l)}\right) h_{A_t}(l) \tag{24}$$

$$\text{MAE} = \frac{1}{L \cdot N} \sum_{l=0}^{L-1} \left| h_{A_t}(l) - h_{A_t^{wat}}(l) \right| \tag{25}$$

We recall that the attribute's domain of A_t corresponds to the integer range $[0, L-1]$ and N is the total number of tuples in the database. Robustness is evaluated by means of the bit error rate (BER), i.e., the probability the value of an extracted symbol is incorrect after attacks, we compute as:

$$\text{BER} = \frac{\sum_{i=1}^{N_g} (s^i \oplus s^{i,det})}{N_g}; \tag{26}$$

Complexity is established as the computation time or more clearly, the amount of time taken by the execution of the insertion and the extraction processes. It is important to notice that all of the following results are given in average after 30 random simulations with the same parameterization but different group configuration.

TABLE I
INTRODUCED STATISTICAL DISTORTION IN TERMS OF MEAN, STANDARD DEVIATION, KULLBACK-LEIBLER DIVERGENCE (D_{KL}) AND HISTOGRAMS MAE FOR THE ATTRIBUTE AGE, CONSIDERING A TEST DATABASE OF N= 508000 TUPLES FOR DIFFERENT NUMBER OF GROUPS AND VARIOUS ROTATION ANGLE SHIFTS Δ. α IS THE ELEMENTARY ANGLE (SEE SECTION IV-B). MOMENTS' VARIATIONS ARE INDICATED IN PARENTHESIS.

Number of groups		$\Delta = \alpha$	$\Delta = 2\alpha$	$\Delta = 3\alpha$
Mean	100	50.113 (0.06%)	50.159 (0.1%)	50.153 (0.15%)
	500	50.138 (0.11%)	50.164 (0.17%)	50.204 (0.25%)
	1000	50.158 (0.16%)	50.194 (0.23%)	50.227 (0.29%)
Std. dev.	100	25.24 (0.01%)	25.266(0.11%)	25.306 (0.27%)
	500	25.233 (0.01%)	25.258 (0.08%)	25.304 (0.26%)
	1000	25.222 (0.05%)	25.25 (0.05%)	25.295 (0.23%)
D_{KL}	100	0.001	0.002	0.004
	500	0.001	0.002	0.004
	1000	0.001	0.002	0.004
MAE	100	$2.36\ 10^{-4}$	$2.73\ 10^{-4}$	$4.54\ 10^{-4}$
	500	$2.54\ 10^{-4}$	$3.55\ 10^{-4}$	$4.82\ 10^{-4}$
	1000	$3\ 10^{-4}$	$3.73\ 10^{-4}$	$4.82\ 10^{-4}$

TABLE II
INTRODUCED STATISTICAL DISTORTION IN TERMS OF MEAN, STANDARD DEVIATION, KULLBACK-LEIBLER DIVERGENCE (D_{KL}) AND DISTANCE BETWEEN HISTOGRAMS WITH NO SEMANTIC CONSTRAINTS

Nb.groups		$\Delta = \alpha$	$\Delta = 2\alpha$	$\Delta = 3\alpha$
Mean	100	50.2009 (0.24%)	50.3265 (0.49%)	50.4003 (0.64%)
	500	50.2387 (0.32%)	50.3247 (0.49%)	50.5198 (0.88%)
	1000	50.3071 (0.45%)	50.3948 (0.63%)	50.5141 (0.87%)
Std. dev.	100	25.2 (0.14%)	25.1865(0.19%)	25.1878 (0.19%)
	500	25.1797 (0.22%)	25.1751 (0.24%)	25.1895 (0.18%)
	1000	25.1562 (0.31%)	25.1599 (0.3%)	25.1753 (0.24%)
D_{KL}	100	0.0188	0.0393	0.0728
	500	0.0195	0.0347	0.0632
	1000	0.0242	0.0349	0.0557
MAE	100	$6.11\ 10^{-4}$	$9.07\ 10^{-4}$	0.0011
	500	$6.41\ 10^{-4}$	$8.62\ 10^{-4}$	0.0011
	1000	$7.27\ 10^{-4}$	$8.75\ 10^{-4}$	0.0011

C. Statistical Distortion Results

As stated above, we evaluate the statistical database distortion through the variations of the the mean, the standard deviation, the D_{KL} and the histogram MAE of the attribute. These variations mainly depend on the rotation angle shift Δ of the center of mass and of the number of tuples per group. Table I provides the results we obtained for the attribute *Age* for a different number of groups $N_g = 100, 500, 1000$ and values of Δ which are multiples of the elementary angle α (see Section IV-B). We recall that our test database contains $N = 508000$ tuples. The original mean and standard deviation of *Age* are 50.078 and 25.236 respectively. As it can be seen they tend to differ from their original value with the value of Δ but the variation remains below 1%. It is the same for the D_{KL} and the histograms MAE which quantify the distortion of the attribute's distribution. These measures increase with the number of groups but the augmentation is not significant. Thus, if our scheme minimizes semantic distortion, it also induces low statistical distortions and may not bias most data-mining operations.

In order to evaluate the gain of performance in terms of distortion when considering our semantic distortion control, the same experiments were conducted applying our scheme with the same parameterization but without the ontology. Obtained results are given in Table II. As we can see, without semantic constraints the distortion is at least 3 times greater whatever the criteria.

D. Robustness Results

Robustness or the BER of our scheme against tuple deletion and insertion attacks mainly stands on the rotation angle shift Δ and on the number of tuples per group; this latter established by the number of groups N_g as the number of tuples in the database N is fixed. In this experiment, the attribute *Age* of our test database was watermarked with an uniformly distributed binary message S considering different values of Δ (as multiples of the angle α) and N_g. These watermarked databases were then attacked by tuple insertion or deletion. The degree of the attack is measured in percentage from 20% to 99%, i.e., the percentage of tuples added to or deleted from the protected database. A fixed value of $\epsilon = 0.0001$ was also considered. Herein, we confront experimental to theoretical performance we established in section VI-D.

In Fig. 10, we show the BER we achieved in the case of a deletion attack for two values of N_g and the lowest rotation angle shift value, i.e., $\Delta = \alpha$. As it can be seen, experimental curves are lower than the theoretical BER upper limit we defined in Section V-B1. However, they tend to this limit along with the increase of the degree of the deletion attack. Fig. 12 provides more tuple deletion attack robustness results making varying Δ and N_g. Obviously, the BER increases along with the degree of the attack but also with the number of groups. This is due to the limited size of the database, the more the number of group increases, the more the number of tuples per group decreases. In general, decreasing the number of tuples per group by mean of a deletion attack or a high number of groups directly impacts the mean direction estimation and consequently the robustness of the scheme. At the same time, the higher the value of Δ, the further the codebook cell centers are (see section IV-B) and the greater the robustness is.

Similar experiments were conducted regarding the tuple addition attack. As in section V-B2, where we theoretically established the BER, new added tuple attributes' values follow the original distribution of the attribute. Results are provided

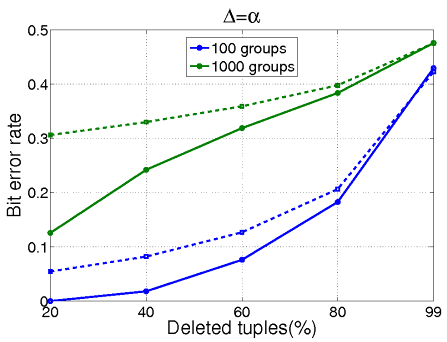


Fig. 10. Tuple deletion attack - BER obtained for the attribute *Age* considering $\Delta = \alpha$ and $N_g = 100$ and 1000 groups. Theoretical and experimental results are depicted indicated with a dashed and solid lines, respectively.

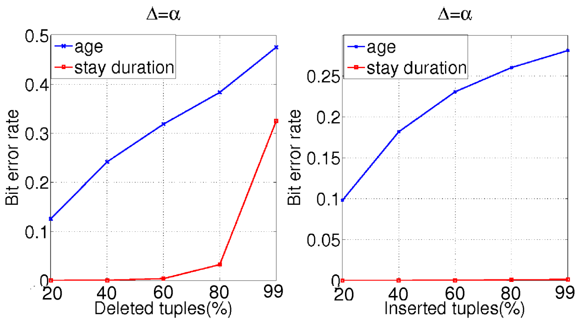


Fig. 11. BER for the attributes *age* and *dur_stay* with $N_g = 1000$ and $\Delta = \alpha$ considering the tuple deletion attack (left) and the tuple insertion attack (right).

in Fig. 13. First, it can be seen that experimental results fit theoretical ones. A little error can be seen, which is related to the hypothesis we made in section VI-D with ϵ equals 0. Beyond, as in the previous attack, the BER decreases with the increase of Δ, and increases along with the number of groups. Notice also, that our scheme better resists to the addition attack than the deletion attack. This is due to the fact that the addition attack only increases the variance of the p.d.f. of μ_i^w while the deletion attack impacts also its mean.

A third experiment was conducted so as to evaluate the influence of the attribute distribution itself. To do so attributes *age* and *dur_stay* were watermarked using the same number of groups $N_g = 1000$ and rotation angle shift $\Delta = \alpha$. As depicted in Fig. 11, the BER obtained for the attribute *dur_stay* is the lowest one due to its low dispersion around its mean values, which makes its mean direction more stable faced to the addition or suppression of tuples.

Finally, we have also evaluated the robustness of our scheme against two common attribute value modification attacks: i) Gaussian noise addition attack ; ii) uniform noise addition attack. Considering the watermarking of the attribute age, the first attack was parameterized with a standard deviation $\sigma = 2$ and the latter with an amplitude in [-4; 4]. Notice that, with such a parameterization, both attacks can be considered as "strong" in our experimental framework. For a capacity or number of groups $N_g = 1000$ and $\Delta = \alpha$, we obtained a BER$\approx$ 0.04 in the first case and BER$\approx$ 0.09 in the second. These BERs are small and will decrease for smaller and higher values of N_g and Δ, respectively. Thus, we can conclude that our scheme is highly robust against attribute value modifications.

E. Computation Time

The computation time of our scheme depends on the construction of groups of tuples, on the identification of the allowable distortion and the watermark embedding/extraction processes. The complexity increases along with the number of tuples in the relation, i.e., N. However, the complexity of the allowable distortion identification task is in addition dependent on the number of attributes considered for watermarking and of the number of attributes semantically connected with them (complexity of the ontology queries).

In this experiment, the attribute *Age* was considered for embedding. It is important to notice that all the following computation times are those of an implementation of our scheme made with Matlab® running on a Intel® Xeon® E5504 running at 2Ghz with 3GB of physical memory and four cores. Table III evaluates the elapsed time for the first two tasks for several values of N. It can be seen that the time increases linearly with N and with the number of attributes semantically connected with the one to be watermarked.

The complexity of the insertion process depends on the number of groups N_g, on the value of Δ and the value of the error ϵ manually fixed by the user. The smaller ϵ is, the more iterations our scheme will have to make to reach this value. On its side, the extraction stage complexity is essentially related with the number of groups. Indeed, once the groups reconstituted one just has to interpret their center of mass values to decode the message. Results for both stages processes are given in Table IV. It can be observed that the extraction complexity increases with the number of groups only. The insertion computation time increases also with Δ. Indeed, our scheme iteratively modified the tuples of one group so as to reach the codebook cell centroid which encodes the desired bit under an epsilon value constraint. In order to evaluate the influence of the value of ϵ on the computation time, we insert a watermark into the attribute *age* considering $N_g = 500$ and $\Delta = \alpha$ and several values of ϵ. Results are depicted in Fig. 14. As expected, the computation time inversely grows with ϵ until reaching an asymptote in $\epsilon = 2.79 \cdot 10^{-5}$. This later value is directly related to the definition of the attribute *age*, an integer variable (see Sect.IV-C).

TABLE III
COMPUTATION TIME FOR THE IDENTIFICATION OF SEMANTIC DISTORTION LIMITS AND THE CONSTRUCTION OF GROUPS USING MATLAB®.

Identification	$N = 200000$	$N = 400000$	$N = 500000$
One attribute	3.54s	4.63s	4.98s
Two attributes	6.97s	9.15s	10.72s
Group creation	$N = 200000$	$N = 400000$	$N = 500000$
	67s	134s	170s

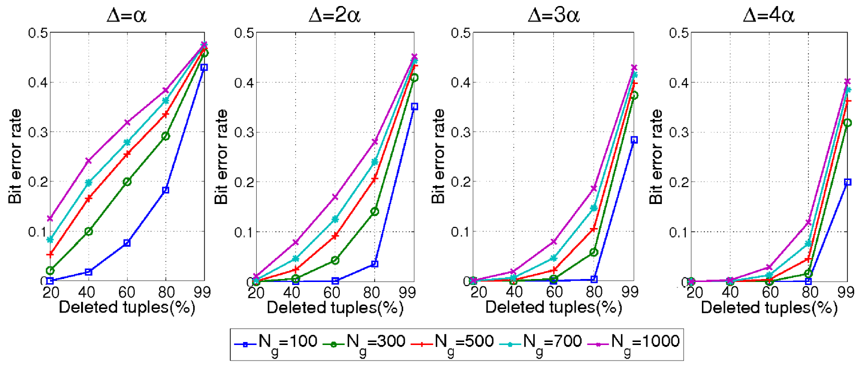


Fig. 12. BER for the attribute *Age* with different rotation angle shifts Δ taking $N_g = 100, 300, 500, 700$ and 1000 groups for a tuple deletion attack.

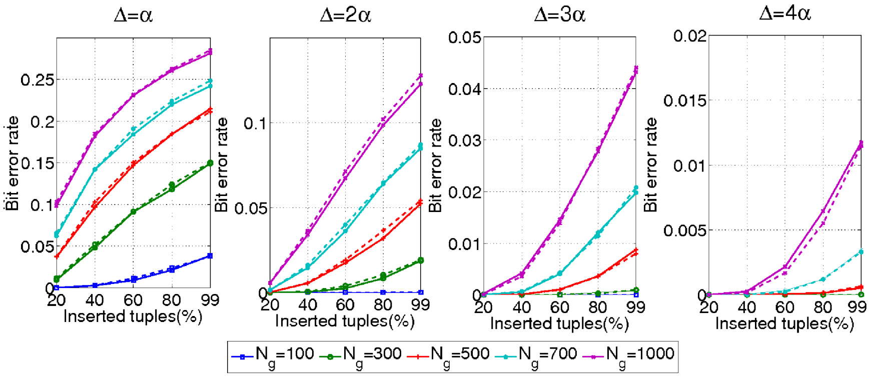


Fig. 13. BER for the attribute *Age* with different rotation angle shifts Δ taking $N_g = 100, 300, 500, 700$ and 1000 groups for a tuple insertion attack. Theoretical and experimental results are depicted indicated with a dashed and solid lines, respectively.

TABLE IV
COMPUTATION TIME FOR THE INSERTION AND THE DETECTION STAGES FOR THE ATTRIBUTE AGE WITH $\epsilon = 0.0001$ USING MATLAB®.

Insertion	$N_g = 100$	$N_g = 500$	$N_g = 700$
$\Delta = \alpha$	2s	8.5s	11.3s
$\Delta = 2\alpha$	2.7s	10.4s	14s
$\Delta = 3\alpha$	3.5s	12.35s	17s
$\Delta = 4\alpha$	4.05s	14.2s	19.6s

Detection	$N_g = 100$	$N_g = 500$	$N_g = 700$
	0.3s	1s	1.3s

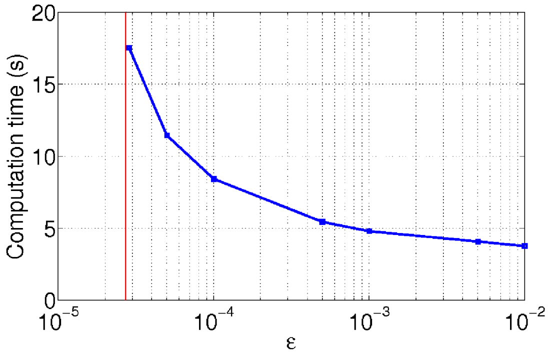


Fig. 14. Computation time for the attribute *Age* with $\Delta = \alpha$ taking $N_g = 500$ and several values of ϵ. The vertical solid line represents the asymptotic value of ϵ for this attribute..

VII. Performance Comparison Results with State of Art Methods

As exposed in Sect. I, distortion control methods one can find in the literature are based on statistical criteria. As example, most recent schemes take the attributes' mean and standard deviation variations [9], [13], the mean squared error [7] as well as attributes values co-occurrences (evaluated in terms of the correlation, the information gain and so on) [12] as constraints. Even though these statistical aspects are linked to the database semantics, they represent only a part of the knowledge attached to the database content. For instance, the modification of the age of a newborn can be statistically insignificant but it will be easily pointed out by means of a semantic analysis. Our semantic distortion control allows avoiding such a situation. In a more general way, it completes the statistical distortion control.

Based on this statement and for fair comparison, we decided to compare our scheme with the ones of Sion *et al.* [7] and Shehab *et al.* [9] (two efficient robust methods) under statistical distortion constraints only. By doing so, we only compare our scheme based on the adapted QIM without considering ontology based semantic distortion control (see Sect. IV) with these two approaches.

The method by Sion *et al.* is based on the modification of the attribute value statistics in a group of tuples G^i so as to embed one bit $s^i = \{0, 1\}$ of the message S (see Sect. II). To do so, a threshold value is first derived from G^i: $Tr = avg + c\sigma$, where avg and σ are the mean and standard deviation values of A_t in G^i and $c \in (0, 1)$ is a user defined parameter. The embedded bit value is encoded depending on the number ν_c of watermarked attributes values over or under this threshold. More clearly, for a group of N_t tuples, a bit value 0 is embedded if $\nu_c < N_t \nu_{false}$ and a bit value is embedded if 1 if $\nu_c > N_t \nu_{true}$ where $\nu_{true}, \nu_{false} \in (0, 1)$ are used defined parameters exploited so as to control watermark robustness. At the reading stage, the message is extracted simply by verifying if ν_c is greater than $N_t \nu_{true}$ or smaller than $N_t \nu_{false}$. At the same time, [7] was slightly modified, without changing intrinsically the strategy they follow, so as to adapt the construction of groups to the one the other schemes use. In the method of Shehab *et al.* [9], watermarking is presented as a constrained optimization problem, where a dissimulation function Θ is maximized or minimized depending on the bit value to embed in a group of tuples. Optimization space is limited by the quality constraints set. In the example given by the authors, Θ represents the number of elements the value of which exceeds a certain threshold defined as in [7] $Tr = avg + c\sigma$. At the detection, the value of Θ is calculated and the detected bit is a 1 (resp. 0) if the obtained value is greater (resp. smaller) than a threshold T. The value of T is calculated so as to minimize the BER.

In the following experiments, 506800 tuples of the attribute *age* of mean 50.078 and standard deviation 25.236 were watermarked and the same statistical constraints, allowing a change in data values within ± 10 percent were considered for all methods. We considered a watermark S of 500 uniformly distributed bits and, consequently, a number of groups $N_g = 500$. All schemes were parameterized so as to ensure a similar distortion in terms of mean and standard deviation, that is to say: $c = 0.85$, $\nu_{false} = 0.05$ and $\nu_{true} = 0.09$ for [7]; $c = 85$ for [9]; and a rotation angle shift $\Delta = 1$ for our scheme. As in the previous section, results are given in average after 30 random simulations.

A. Attribute P.D.F preservation

Although all the methods preserve the attribute's mean and standard deviation, they do not have the same behavior in preserving the attribute's p.d.f, as shown by the D_{KL} and the MAE criteria in Table V. [9] provides the best results but at the price of a very high complexity due to the use of an optimization process (see below).

TABLE V
DISTANCE BETWEEN DISTRIBUTIONS IN TERMS OF THE D_{KL} AND THE MAE FOR OUR SCHEME AND THE METHODS PROPOSED BY SION *et al.* [7] AND SHEHAB *et al.* [9].

Method	D_{KL}	MAE
Sion *et al.* [7]	0.0805	0.00218
Shehab *et al.* [9]	$1.916\ 10^{-4}$	$5.619\ 10^{-5}$
Proposed Method	0.0024	$2.8288\ 10^{-4}$

B. Robustness

With the same parameterization, three attacks were considered so as to evaluate algorithms' robustness: insertion and deletion of tuples and attributes values modification. All of them were performed impacting a percentage of tuples in the range $20\% - 99\%$. The attribute modification consisted in the addition of a centered Gaussian noise of std. dev. $\sigma = 2$.

As depicted in Table VI, our method performs in general better than [7] and [9], being [7] the worst solution. [9] provides a better robustness than our method in the case of a suppression attack only. In the case of an attribute alteration attack, our scheme provides a BER 100 times smaller than [9]. Here is the interest of working with the angular position of the center of mass. It also achieves better performance regarding the tuple insertion attack.

TABLE VI
BIT ERROR RATE FOR OUR SCHEME AND THE METHODS PROPOSED BY SION *et al.* [7] AND SHEHAB *et al.* [9] FOR VARIOUS ATTACKS.

Deletion	**20%**	**40%**	**60%**	**80%**	**99%**
Sion *et al.* [7]	0.2643	0.3183	0.3453	0.3875	0.4387
Shehab *et al.* [9]	0.0652	0.0776	0.0944	0.1548	0.464
Proposed Method	0.0434	0.1208	0.2041	0.3053	0.4624
Insertion	**20%**	**40%**	**60%**	**80%**	**99%**
Sion *et al.* [7]	0.487	0.4932	0.5	0.5	0.5
Shehab *et al.* [9]	0.074	0.0956	0.1268	0.1776	0.218
Proposed Method	0.0263	0.0721	0.1104	0.1449	0.1656
Modification	**20%**	**40%**	**60%**	**80%**	**99%**
Sion *et al.* [7]	0.5	0.5	0.5	0.5	0.5
Shehab *et al.* [9]	0.1192	0.134	0.2392	0.4386	0.4804
Proposed Method	0.0028	0.0031	0.0039	0.0043	0.0037

C. Complexity

Computation time is used so as to evaluate the complexity of these approaches. Regarding the embedding process, it is conducted in about 3s with our method and the one by Sion *et al.*. It takes about 4 hours to [9], due to its optimization process. The detection stage duration is approximately the same in all cases, being of less than 3s.

To sum up, our approach provides better robustness performance than the scheme of Sion *et al.* [7] and it introduces less statistical distortion. Regarding the scheme by Shehab *et al.* [9], our scheme is more robust against tuple insertion and attribute's values modification attacks. It introduces more distortion in terms of the distance between distributions but is of a much lower complexity.

VIII. DISCUSSION

The security of the proposed scheme stands on the construction of groups of tuples. This task is conducted by means of a cryptographic hash function, e.g., SHA, which takes the secret watermarking key K_S as an input. This procedure ensures that, in the absence of K_S, it is impossible for an attacker to correctly reconstruct the groups of tuples and consequently, to extract the sequence S. Thus, the only way he or she has to erase the watermark is by modifying attributes' values or deleting a set of tuples from the database.

The use of an ontology improves the watermark imperceptibility/masking. It is important to notice that the more visible the watermark, the easier for an attacker to identify the watermarked tuples. The ontology indicates the set of semantically coherent values the embedder can use when watermarking/modifying an attribute in a tuple so as to keep this modification invisible to an attacker. This domain ontology is public, i.e., a priori known of everyone. Indeed, a user that regularly exploits a database will learn with time the relationships that exist between its attributes' values. Beyond, the ontology can also serve to identify incoherent tuples introduced by any watermarking scheme that does not take care of the semantic interpretation of records. Such incoherent records can help the attacker by providing clues about the presence of a watermark or of the scheme that has been used.

Notice that after our ontology based semantic analysis, the number of watermarkable tuples can be reduced, impacting the watermark robustness. Even though in this work tests have been performed considering only one attribute, we recommend watermarking several attributes in a relation. Using several attributes at the same time for message embedding will increase the watermark robustness. One can for example repeat the message in different attributes and use majority vote at the reading stage. This solution will require the user to store the attributes' mean values and to send them to the watermark reader as part of the watermarking key (see Sect. IV-B). In the case several attributes are watermarked at different time, by or for different users, multiple watermarking keys will have to be managed. In order to overcome this issue, an alternative we propose in Sect. IV-B is to make possible that the watermark reader retrieves the attributes' mean values by itself. This can be achieved by not watermarking some secretly selected tuples, i.e., tuples the attributes of which are not modified by the embedding process and that can be retrieved and used by the reader for attributes' mean-values computation. In the extreme case where none of the attributes are semantically watermarkable, one must consider the use of distortion free schemes [10] (see Sect. I).

IX. Conclusion

In this paper, we have proposed a new robust database watermarking scheme the originality of which stands in a novel semantic distortion control method and a QIM adapted to the modulation of attributes' circular histograms. As we have shown, semantic distortion control aims at two main objectives: i) ensure the correct interpretation of the information contained in the database, by preserving the semantic links in between attributes values. ii) make the watermarking invisible to an attacker. Our method is suitable to be combined with existing methods that include statistical distortion control. Message embedding is performed by application of QIM to the center of mass of circular histograms for a numerical attribute. It is robust against most common database attacks: tuple deletion and insertion as well as attributes' values modification. Our scheme is appropriate for copyright protection or traitor tracing, embedding for example a user identifier. In addition, we have theoretically established and verified experimentally the performance of our method in terms of robustness. The proposed results allow the user to correctly select our scheme's parameters under constraints of robustness and capacity.

Acknowledgment

This work was supported by the French armament procurement agency (DGA) - Projet DGA RAPID FRAG&TAG - and by the Brittany Region Council. The authors are very grateful to the Department of Medical Information and Archives, CHU Lille; UDSL EA 2694; Univ Lille Nord de France; F-59000 Lille, France, for the experimental data used in this study.

References

[1] S. Rogers. Wikileaks embassy cables: download the key data and see how it breaks down. In The Gurdian Online. Accessed: 2013-11-21. [Online]. Available: http://www.theguardian.com/news/datablog/2010/nov/29/wikileaks-cables-data

[2] M. McNickle. Top 10 data security breaches in 2012. In Healthcare Finance News. Accessed: 2013-11-21. [Online]. Available: http://www.healthcarefinancenews.com/news/top-10-data-security-breaches-2012

[3] W.-H. Lin, Y.-R. Wang, and S.-J. Horng, "A wavelet-tree-based watermarking method using distance vector of binary cluster," *Expert Systems with Applications*, vol. 36, no. 6, pp. 9869 – 9878, 2009.

[4] D. Rosiyadi, S.-J. Horng, P. Fan, X. Wang, M. K. Khan, and Y. Pan, "Copyright protection for e-government document images," *IEEE Multimedia*, vol. 19, no. 3, pp. 62–73, 2012.

[5] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. Morgan Kaufmann Inc., 2008.

[6] R. Agrawal and J. Kiernan, "Watermarking relational databases," in *Proc. Int. Conf. Very L. DB*, 2002, pp. 155 – 166.

[7] R. Sion, M. Atallah, and S. Prabhakar, "Rights protection for relational data," *IEEE T. Knowl. Data En.*, vol. 16, no. 12, pp. 509 – 1525, 2004.

[8] D. Gross-Amblard, "Query-preserving watermarking of relational databases and xml documents," *ACM Trans. Database Syst.*, vol. 36, pp. 3:1–3:24, 2011.

[9] M. Shehab, E. Bertino, and A. Ghafoor, "Watermarking relational databases using optimization-based techniques," *IEEE T. Knowl. Data En.*, vol. 20, pp. 116–129, 2008.

[10] Y. Li, H. Guo, and S. Jajodia, "Tamper detection and localization for categorical data using fragile watermarks," in *Proc. ACM workshop on DRM*, 2004, pp. 73–82.

[11] F. Guo, J. Wang, and D. Li, "Fingerprinting relational databases," in *Proc. of the ACM Symp. on Applied computing*, 2006, pp. 487–492.

[12] M. Kamran and M. Farooq, "A formal usability constraints model for watermarking of outsourced datasets," *IEEE T. Inf. Foren. Sec.*, vol. 8, no. 6, pp. 1061–1072, 2013.

[13] M. Kamran, S. Suhail, and M. Farooq, "A robust, distortion minimizing technique for watermarking relational databases using once-for-all usability constraints," *IEEE T. Knowl. Data En.*, vol. 25, no. 12, pp. 2694 – 2707, 2013.

[14] J. Lafaye, D. Gross-Amblard, C. Constantin, and M. Guerrouani, "Watermill: An optimized fingerprinting system for databases under constraints," *IEEE T. Knowl. Data En.*, vol. 20, pp. 532–546, 2008.

[15] J. Franco-Contreras, G. Coatrieux, P. Massari, S. Darmoni, N. Cuppens-Boulahia, F. Cuppens, and C. Roux, "Data Quality Evaluation in Medical Database Watermarking," in *MIE 2015*, 2015, accepted paper.

[16] A. Gomez-Perez and V. R. Benjamins, "Applications of ontologies and problem-solving methods," *AI Magazine*, vol. 20, no. 1, pp. 119–123, 1999.

[17] W. Su, J. Wang, and F. H. Lochovsky, "Ode: Ontology-assisted data extraction," *ACM T. DB Syst.*, vol. 34, no. 2, pp. 12:1–12:35, 2009.

[18] L. Hollink, G. Schreiber, J. Wielemaker, and B. Wielinga, "Semantic annotation of image collections," in *Workshop on Knowl. Markup and Semantic Annot. KCAP*, 2003, pp. 0–3.

[19] G. Coatrieux, E. Chazard, R. Beuscart, and C. Roux, "Lossless watermarking of categorical attributes for verifying medical data base integrity," in *Proc. of the IEEE EMBC*. IEEE, 2011, pp. 8195–8198.

[20] J. Franco-Contreras, G. Coatrieux, N. Cuppens-Boulahia, F. Cuppens, and C. Roux, "Robust lossless watermarking of relational databases based on circular histogram modulation," *IEEE T. Inf. Foren. Sec*, vol. 9, no. 3, pp. 397–410, 2014.

[21] V. Pournaghshband, "A new watermarking approach for relational data," ser. ACM-SE 46, 2008, pp. 127–131.

[22] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE T. Inform. Theory*, vol. 47, no. 4, pp. 1423–1443, 1999.

[23] N. I. Fisher and T. Lewis, "Estimating the common mean direction of several circular or spherical distributions with differing dispersions," *Biometrika*, vol. 70, no. 2, pp. 333–341, 1983.

[24] R. G. McKilliam, "Lattice theory, circular statistics and polynomial phase signals," Ph.D. dissertation, Univ. of Queensland, Australia, 2010.

[25] M. Berenson, T. Krehbiel, and D. Levine, *Basic Business Statistics: Concepts and Applications*. Prentice-Hall, 2012.

[26] "Protégé ontology editor," http://protege.stanford.edu/ Accessed: 2013-12-09.

Javier Franco-Contreras (M'11) received the Ph.D. degree in Signal Processing and Telecommunication from Institute Mines-Telecom, TELECOM Bretagne in collaboration with University of Rennes I, France, in 2014. He is currently a Post-Doctoral researcher in the Department of Information and Image Processing, TELECOM Bretagne and Laboratory of Medical Information Processing, INSERM U1101, Brest, France, focusing on the protection of outsourced medical databases by means of watermarking. His main areas of research interest are medical information security and privacy as well as watermarking.

Gouenou Coatrieux (M'06-SM'13) is currently a Full Professor at the Department of Information and Image Processing, Institute Mines-Telecom, Telecom Bretagne; LaTIM-Inserm U1101 (France) and the head of the Joint laboratory on Security and Processing of Externalized Medical Image Data (SePEMeD). He received the Ph.D. degree in Signal Processing and Telecommunication (2002) from the University of Rennes I (France) in collaboration with the Institute Mines-Telecom, Telecom ParisTech (France). His primary research interests concern medical information system security and medical data protection by means of watermarking and encryption. He is an Associate Editor of the IEEE Journal on Biomedical and Health Informatics, Digital Signal Processing, and Innovation and Research in BioMedical Engineering. He is a member of the International Federation for Medical and Biological Engineering "Global Citizen Safety and Security Working Group" and the European Federation for Medical Informatics "Security, Safety, and Ethics Working Group", and has contributed to the Technical Committee of "Information Technology for Health" of the IEEE Engineering in Medicine and Biology Society.