

RESEARCH ARTICLE

Study of the Watermark Source's Topology Role on Relational Data Watermarking Robustness

MAIKEL LÁZARO PÉREZ GORT¹, MARTINA OLLIARO¹, AND AGOSTINO CORTESI¹

Università Ca' Foscari di Venezia Campus Scientifico, 30172 Venice, Italy

Corresponding author: Maikel Lázaro Pérez Gort (maikel.perezgort@unive.it)

Work partially supported by "SERICS" (PE00000014) and "PADS4Health" (PRIN PNRR 2022 P2022MSMAW - CUP N. H53D23010880001) projects under the PNRR MUR program funded by the EU.

ABSTRACT Accessing relational databases through web services increases risks of piracy and tampering. Watermarking guarantees data protection without constraining their access and management. Researchers traditionally compare the extracted watermark with the embedded one after simulating attacks to analyze its robustness. However, the literature has not addressed how much watermark signal is retained after querying multi-word textual attributes and how topological features of the watermark affect robustness. In this work, we fill this gap by evaluating a semantics-preserving approach. We execute different classes of queries in three data sets, including an actual database of a public administration. The results show that 99.99% of the marks can be detected when considering a watermark source with a topological factor close to or equal to 1. As valuable contributions, we formally defined the watermark source's topological factor and the corrosion degree of the watermark detection. This work offers hints to data owners to adjust detection parameters, increasing the chances of spotting the watermark despite having, as evidence, data highly damaged due to malicious operations.

INDEX TERMS Ownership proof, plagiarism detection, relational data, robust watermarking, watermark topology.

I. INTRODUCTION

Relational databases offer the benefits of easy remote access by thirds through the internet after their deployment. Nevertheless, this relatively open entry point is dangerous in the presence of malicious users,¹ increasing the risks of preserving data authenticity. Usually, institutions use ad hoc security methods to protect their digital assets. However, after deployment, they need additional tools to prove ownership after thirds perform data copies and queries. This issue is particularly relevant for news data sets or public administration data, which present high volumes of textual content, and their tampering result in serious misinterpretations [1]. For example, when publishing the results of a political election, authenticity should be preserved even when extracting data relating only to female elected candidates.

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek².

¹Malicious users are those tampering with the data or interested in falsely claiming their ownership.

Watermarking techniques are a powerful tool for protecting data copyright and identifying violations of its integrity and authenticity [2]. They involve inserting a signal, the watermark (a stream of bits, each consisting of a mark), into a digital asset without compromising usability [3]. When required, alleged data owners can try to extract the watermark from a copy of the digital asset suspected to contain it. If both watermarks (the embedded and the extracted one) are similar or equal, it is possible to prove data ownership or unauthorized distribution. Both processes, embedding and extracting the watermark, define the watermark synchronization.² When the watermark is generated from content with independent meaning, such as image, audio, or video files, the watermarking technique is defined as meaningful, and the content used for the watermark generation is identified as the watermark source. On the other hand, techniques not using an independent meaningful external source are classified as meaningless watermarking.

²Synchronization is the process of aligning two signals in time or space [4].

Each of the bits selected from the watermark source is used to generate the marks composing the watermark. Sometimes, besides the bits from the source, other bits from the same data being watermarked can be considered so that the watermark generation can be linked to the content protected [5], [6].

Fig. 1 presents the general architecture of a meaningful watermarking technique. The embedding process takes the watermark source as input to generate the watermark and embeds it into a database, producing a watermarked version of the data as output. The process requires at least a secret key known only by the data owner. After the embedding, the watermarked database enters into its operational phase. Therefore, benign updates resulting from daily transactions will start modifying the data. Also, unauthorized operations, defined as malicious, can be performed to tamper with the data or remove the watermark for thirds to claim the data ownership.

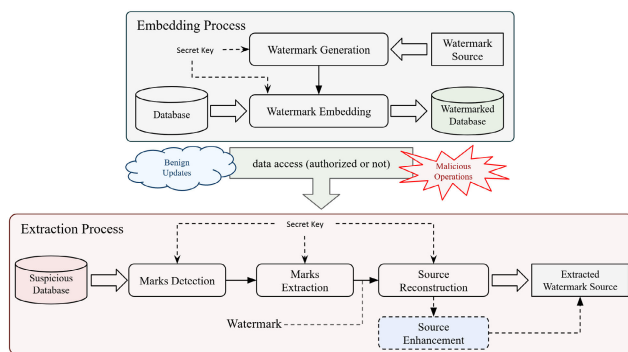


FIGURE 1. Architecture of meaningful watermarking techniques.

In case of suspicions of false ownership claims or other unauthorized operations being performed over the database, the extraction process is carried out, trying to detect and check the watermark in the version of the data object of study. For this, the same private key is used to detect and extract the marks. Once all marks are extracted, the version of the obtained watermark is used to reconstruct the watermark source. If possible, enhancement algorithms can be used to boost the signal build from the watermark. Finally, the version of the extracted source is constructed and compared with the original one, confirming the ownership if both sources are similar enough or concluding about the data quality in the case of other watermark applications [7].

As an approach, watermarking contrasts with other protection mechanisms, such as cryptography, that alter data perception [8]. In the case of watermarking, the presence of the watermark in the protected data must be unnoticed [7], [9].

Researchers first proposed watermarking techniques to secure multimedia data. Nevertheless, over time, they have been successfully used to protect other digital assets, such as compiled code [10], relational databases [11], neural networks [12], etc. Regarding relational data, considering third parties can readily access them through the Internet, watermarks secure them without restricting access or portability.

The literature classifies relational watermarking approaches into two groups: distortion-free and distortion-based [7], [13]. The former maintains data intact, while the latter performs watermark embedding by replacing some bits of the protected data with the marks. Some distortion-free techniques propose interesting approaches, such as protecting datasets with different formats beyond relational data [14]. Nevertheless, given that our work focuses on analyzing the preservation of distortion-based robust watermarks, such approaches are out of this research's scope. Moreover, a group of distortion-based techniques is classified as reversible. They aim to restore the quality of the data once the watermark is extracted. Some of the recent work belonging to this group are the proposals by Liu et al. [15], Wu [16], and Li et al. [17]. Although this is very useful from the point of view of data quality, these techniques are not meant to protect data deployed for long terms, which subdue them to modification due to querying. Considering this, they are out of our scope of work.

Distortion-based techniques replace bits only if the values selected to store the marks (also defined as marks' carriers) tolerate minor errors without affecting the usability and semantics of the data. The threshold to preserve data quality while embedding the watermark varies, considering some data might allow higher distortion than others depending on their purpose in the organization. Higher toleration of changes contributes to higher watermark capacity and, therefore, higher robustness. Suppose the technique can modify more values without compromising data quality. In that case, it is possible to insert more marks, making the watermark more robust and not easily compromised by malicious operations or attacks [4]. Therefore, attackers aiming for its removal must modify a higher volume of data, increasing the risk of making it useless. Nevertheless, attacks might succeed at some point, leaving the path clear for the attacker to claim (with no rights) the data ownership.

A critical goal of relational databases is allowing daily transactional operations (also defined as benign updates) based on Structured Query Language (SQL) queries. Therefore, the volume of data they store increases, involving more carriers of marks each time. Hence, the number of compromised marks also increases after the watermark embedding, denigrating its quality at a point that might compromise its recognition. For a query involving a higher number of tuples and attributes and, therefore, a higher number of carriers, the watermark detection can be compromised even if the query is not so often executed. Consequently, detecting the watermark signal is more difficult for queried data, even when attackers do not tamper with the database. Different watermarking schemes have been proposed to prevent leakage of data stored in third servers [18]. Nevertheless, the origin of the operations that can compromise data queried from protected databases has not been adequately addressed.

In this work, we introduce a systematic approach to the problem of evaluating how queries affect the quality of an embedded watermark in multi-word textual attributes of a relational database. Indeed, when data are public and

subject to processing through queries, it is highly crucial to guarantee ownership detection even after querying it. The related literature addressed this problem for numerical data in [19]. Nevertheless, there needs to be research on semantic-preserving approaches to watermark textual attributes. This type of cover offers interesting opportunities for mark embedding. For instance, when the watermarking technique aims not to compromise semantic consistency, watermark capacity can be increased by inserting multiple marks in a value without affecting the usability of the protected data. However, textual cover type also has robustness and detection challenges that directly influence watermark detection after querying the data, either due to benign updates or malicious operations [20].

Our work is highly relevant to the practice of forensics in relational data. It makes it possible to prove the authenticity of data after suffering modifications resulting from benign or malicious operations.

Our research includes experiments with a public administration database,³ considering how relevant ownership and data quality preservation are for this kind of organization, especially considering the risks of potential misuse of their digital assets. Some examples of malicious operations that can be performed on unprotected data of this kind are:

- data tampering for creating false competencies before applying to a job or setting the successful conclusion of a course for a student to pass to the next semester,
- performing false ownership claims of stolen data by a competing institution featured by predatory behavior.

Our research offers tools for supporting ownership claims so database owners can preserve their intellectual property. We provide experimental evidence from real cases that are meaningful regarding these threats.

A. RESEARCH OBJECTIVES

The main objective of our research is to evaluate the role played by the topological features of the watermark source in meaningful watermarking techniques. We focus mainly on the contribution the source might have to the robustness of the approach. The evaluation to validate our proposal targets watermarked data subjected to a high frequency of updates. Considering that queries are the core of benign updates and malicious operations that seek to compromise watermark detection, we consider different types of queries to increase the proposal's value.

We also aim to formalize the corrosion suffered by the watermark between the embedding and extraction processes when protecting multi-word textual content in a database relation. We study the relevance of multi-word textual content for watermark preservation and compare it with the numerical cover type. By doing this, we highlight the differences the extracted watermark signal experiences due to the corrosion suffered by the detection by querying the data.

We aim to provide data owners with experimental evidence to evaluate options for increasing the likelihood of detecting the watermark after spotting features in the extracted signal that can serve as hints of knowing the type of query applied to the protected data. Given the relevance of watermark detection, we focus our analysis on robust techniques designed for identifying data ownership.

B. PAPER CONTRIBUTION

Compared to marking numerical data, embedding the marks into textual cover types with a semantic protecting approach leads to truthful query results and more chances of preserving the embedded signal through queries [20]. The benefits of selecting textual values as carriers are the possibility of embedding more marks into each attribute (which increases watermark capacity) and the lower transformation that textual data usually suffer when queried (which makes possible carriers' values preservation).

On the other hand, when marking numerical data, small changes in the values result in retrieving misleading information when querying it. However, watermarking approaches can avoid this downside by marking multi-word textual attributes instead while preserving the semantics of the data. When performing watermark embedding on numerical cover types, the algorithm embeds just one mark per value once the carriers are selected. Therefore, regarding watermark persistence through queries, marks can be recovered when carrier attributes are selected. Still, the chances of detecting the original signal from them are low.

In this work, we empirically evaluate the impact of queries on textual databases when watermarked with the semantic preserving technique defined in [20]. We apply several operations over the protected data, following the query classification given in [19], involving different aggression degrees (a.k.a the number of values affected by the query) for each case.⁴ The study performed in this work contributes to preventing data plagiarism. Furthermore, watermarks can identify data owners and their intellectual property. Thus, it is essential to guarantee that the data, when queried, are still protected.

We offer results that allow data owners to adjust detection parameters to increase the chances of spotting the watermark despite highly damaged data. Studied basic operations can be associated with malicious operations in extreme cases, making it possible to develop reverse engineering techniques to build new approaches. Finally, we compare the preservation ability of a semantic watermark to a non-semantic one, highlighting their benefits and downsides.

The main contributions of this paper are as follows:

- a formal definition of the watermark source's topological factor and the corrosion degree of the watermark detection and their role in robust watermarking;

⁴The higher the number of values involved in the query, the higher its aggression degree.

³Courses database from the Ca' Foscari University of Venice.

- an empirical evaluation of the resilience, through query results, of a semantic preserving watermark. We considered queries with different aggression degrees, providing the first study of this kind;
- a set of evidence proving how multi-attribute embedding combined with multi-word textual cover type benefits the watermark capacity while preserving data quality thanks to the synonym substitution approach;⁵
- a comparison of the results with those obtained in [19], where numerical attributes were used for the embedding, highlighting details regarding the watermark propagation through queries when considering different cover types.

C. PAPER STRUCTURE

The rest of this paper is organized as follows. Section II presents the related work, focusing on techniques analyzing queries' effect on data, watermarking approaches using non-numerical attributes as cover type, and works involving semantic components oriented to data quality preservation. Section III introduces the details of our approach. Section IV presents the experiments carried out and the results obtained. Moreover, it formalizes the critical points to consider when analyzing watermarks and data quality preservation. Section V addresses the impact of our research and analyses the results. Section VI concludes.

II. RELATED WORK

Agrawal and Kiernan [3] proposed the first watermarking technique for relational data. In their work, the authors used numerical attributes as carriers. They introduced a way to select tuples, attributes, and bits to proceed with the watermark embedding, later called the AHK algorithm [21]. Since the publication of the AHK algorithm, relational data watermarking techniques have increased in number and diversity. For example, some techniques present database protection as a middleware that generates format-independent watermarks [11], some focus on reducing the distortion caused by the watermark embedding [22], and others allow data owners to define constraints to preserve data usability automatically [23].

Despite the many techniques, most works target numerical attributes for mark embedding while modifying their values and trying to control the distortion using statistical metrics as a reference. A notable downside of those approaches is that any change caused to numerical values results in variations of query outputs. Some researchers propose to use categorical and textual attributes to embed the watermark as an alternative to distortion-based numerical cover-type approaches. In this work, we focused on presenting the techniques depending on their cover type, considering the significant differences that the query produces on the hidden watermark, depending on the type of attribute selected for embedding the marks. Although some techniques consider more than one cover type (e.g., proposals by Zhang et al. [24] and Li et al. [25]), we analyze

the more representative approaches, focusing only on one cover type at a time, given the differences between the benefits and downsides.

A. NUMERICAL WATERMARKING APPROACHES

Managing numerical values using their binary notation is simple and makes it possible to shuffle the bits to reduce the distortion. Nevertheless, the bigger problem of using the numerical cover type has been controlling the distortion without compromising the watermark robustness and the security expected from the watermarking technique.

Among the last numerical cover type contributions, in [26], authors proposed a robust and reversible watermarking technique with distortion control. Their work used a genetic algorithm to select the best secret key and histogram shifting of prediction error to minimize the distortion and improve robustness. Also, Wang and Li proposed a reversible technique that balances the distortion caused during the embedding with robustness using a fitness function [27]. Their technique implements genetic algorithms to obtain the watermark synchronization parameters.

Another example is the proposal by Li et al. [28], where the authors combined the majority voting with a Hamming code to guarantee the correct watermark extraction. They compared the mean and the variance of the original unwatermarked data with the ones obtained from the watermarked data to measure the distortion caused by the embedding. Nevertheless, these metrics might lead to incorrect conclusions, considering that they can indicate that two numerical distributions are similar when, in fact, they are different. Other metrics, such as the Wasserstein distance or the Kullback-Leibler divergence, are more convenient for this purpose [29].

B. CATEGORICAL WATERMARKING APPROACHES

Using carriers storing discrete data types (e.g., categorical values) presents as a challenge the possibility of modifying the entire categorical value with the slightest distortion introduced by data hiding techniques. Furthermore, the watermark embedding process could result in data out of the set of allowed discrete values. The following categorical cover-type approaches address these issues.

In [30] and [31], the authors introduced and analyzed the rights protection issue for categorical relational content through watermarking. They exploited the encoding bandwidth in the semantic association between the categorical attributes and the primary key to embed the watermark. In [32], the authors proposed a technique to satisfy the anonymization constraints and protect the ownership of outsourced medical data. In [33], the authors proposed a lossless technique claiming to achieve higher detection than other proposals. However, their technique's cover type constrains their proposal's applicability, and the analysis of queries on the protected data is out of their scope.

The previous approaches do not consider more complex attributes (e.g., multi-word textual attributes). Moreover, the

⁵Section IV-A presents details about the synonym substitution approach.

changes caused by the embedding compromise the consistency of the inter-attribute semantics [20], resulting in the variation of the results of queries run on the watermarked data.

C. TEXTUAL WATERMARKING APPROACHES

Multi-word textual cover-type approaches represent an alternative to avoid the downsides of numerical and categorical cover-type techniques. As examples, the schemes proposed in [34], [35], [36], [37], and [24] exploit the limitations of the human visual system to embed the watermark into non-numeric multi-word attributes without changing the meaning of the selected textual values or their appearance. However, they are vulnerable to malicious operations aiming at watermark detection, considering attackers can spot the marks using computational methods. Also, they do not address the consequences of semantic distortion caused by the watermark embedding.

In [38], the authors propose another interesting textual cover type technique. They use an image to generate the watermark, classifying their technique as an Image-Based Watermarking (IBW) [7].⁶ The scheme is based on performing a minimum modification to reduce the distortion but never analyzes the semantic perturbation caused by the embedding. In [39], the authors presented a reversible technique applying the histogram shifting model for embedding the watermark into non-numeric attributes. This work does not give details of the queries used to simulate the attacks when analyzing the technique's robustness. The experiments to validate the proposal were performed in datasets storing relatively short textual content.

D. KNOWLEDGE PRESERVATION STRATEGIES

In [40], [41], and [42], the authors proposed other approaches using tuple partitions to synchronize the watermark. In particular, [40] consists of a technique to preserve the database knowledge while balancing data owners' and recipients' usability constraints. None of these proposals guarantees watermark synchronization when the number of partition delimiters compromised due to benign updates or malicious operations is high or when data modifications cause variations of the components used to generate the partitions (e.g., the relation's primary key). Their problem relies on generating partitions for the watermark extraction that are different from the ones used to embed the watermark, thus compromising the detection of the watermark.

In [43] and [44], the authors integrated an ontology-guided distortion control method to identify semantics links between attribute values in a tuple while marking numerical cover types. Nevertheless, these approaches make limited use of the ontologies for controlling semantics and increasing the watermark capacity, considering they select only one numerical attribute for mark embedding. On the other hand, in [20], the authors formalized a multi-word textual

⁶Image-Based Watermarking (IBW) are techniques that use images as watermark sources.

TABLE 1. SQL structure of each *select* and *action* query category [19].

Select Query (Q _S)		Action Query (Q _A)	
Type	Structure	Type	Structure
Q _{S1}	SELECT [attr_list] FROM data_source WHERE condition	Q _{A1}	INSERT INTO data_source ([attr_list]) VALUES ([val_list])
Q _{S2}	SELECT [attr_list], [tran_list] FROM data_source WHERE condition	Q _{A2}	UPDATE data_source SET [val_list] WHERE condition
Q _{S3}	SELECT [attr_list], [aggr_list] FROM data_source WHERE condition GROUP BY [attr_list] HAVING condition	Q _{A3}	DELETE FROM data_source WHERE condition

watermarking approach to preserve intra- and inter-attribute semantics consistency by substituting semantically similar words in a specific context. The fluency, grammaticality, structure, and meaning of a textual value are maintained using a word disambiguation module, which allows the selection of the proper synonyms. This technique guarantees the data's semantic preservation and the watermark's total imperceptibility.

In [25], the authors proposed a semantic-based robust approach for numerical and non-numerical attributes. When working with the numerical cover type, the authors proposed a reversible watermark embedded at the semantic level, trying to preserve the statistical distribution of each attribute. On the other hand, when working with non-numerical attributes, they considered natural language processing to embed the watermark, applying the segmentation and embedding of words.

In this work, we focus on knowledge preservation watermarking techniques that are not reversible, considering the watermark must remain in the data permanently to study the long-term effect of queries on them. We empirically evaluate the queries' impact in textual databases watermarked according to the approach introduced in [20].

The work published in [19] studied the watermark signal's corrosion in numerical databases. In the current work, we refer to the exact structure of the queries for a different aim: to detect the impact of watermark source topology concerning the robustness of the overall watermarking action.

III. WATERMARK SOURCE'S TOPOLOGY

In this work, we adopt the classification criteria of SQL queries formalized in [19], which defines *select* and *action* types, denoted by Q_S and Q_A, respectively (see Table 1). In the table, we highlight the queries' optional clauses in gray color.

Select queries extract content from the database. They are not meant to perform any data modification. Among the types of *select* queries, there are (i) queries operating only on the relation's attributes (denoted by Q_{S1}), (ii) queries that use transformation operations (denoted by Q_{S2}), and (iii) queries that use aggregate operations (denoted by Q_{S3}). In the table, [attr_list] denotes the list of selected attributes, [tran_list] and [aggr_list] are the columns resulting from transformation and aggregate operations, respectively, data_source is the data entry point for the query, and condition stores the filtering criteria defined for the WHERE and HAVING SQL clauses.

On the other hand, *action* queries impact the relation's content and produce the same relation with some amount of modified data as output. There are three categories of them: (i) queries used to insert tuples (denoted by Q_{A1}), (ii) queries used to update values already stored in the relation (denoted by Q_{A2}), and (iii) queries used to delete tuples (denoted by Q_{A3}). In Table 1, $[val_list]$ denotes the list of values considered when inserting a tuple or updating attributes.

We denote the original database by DB and the watermarked one by DB' . We identify the data source obtained from DB' as \mathcal{D}' , comprising a fragment of the watermarked database. Furthermore, we denote by T' the result-set obtained with a *select* query and by \mathcal{D}' the result of an *action* query. The sets of tuples containing the marks in \mathcal{D}' and T' are given by $\eta_{\mathcal{D}'}$ and $\eta_{T'}$, respectively. Similarly, we denote the sets of attributes containing the marks in \mathcal{D}' and T' as $\nu_{\mathcal{D}'}$ and $\nu_{T'}$, respectively. Finally, we define the tuple and attribute complexity weights by $\eta_W = |\eta_{\mathcal{D}'} - \eta_{T'}|$ and $\nu_W = |\nu_{\mathcal{D}'} - \nu_{T'}|$. We used them to quantify the damage caused by the query to the watermark signal, focusing on the number of elements that contain marks and are affected by the operation. Note that the higher the values of η_W and ν_W , the higher the aggression degree of the query.







We obtain the query complexity $\mathcal{X}(q)$ according to (1), where Θ is the function computing the contribution of ν_W and η_W to the degradation of the watermark in the query result [19].

$$\mathcal{X}(q) = \Theta(\nu_W, \eta_W) \quad (1)$$

Knowing the number of elements containing marks affected by the query helps to predict the watermark's quality. Nevertheless, there are other factors to consider. Features of the watermarking technique, such as its cover type, play an important role in synchronizing the watermark in queried data. Furthermore, the number of right-detected marks depends on the query type and the particularities of the extraction process. The number of false positives in *select* query results must be low, especially for cases performing deterministic detection in numerical attributes. On the contrary, when using multi-word textual cover type and contextual-based watermark synchronization, the extracted watermark often presents more false positives.

In this work, we propose (2) to obtain the watermark quality according to the abovementioned factors. In the equation, $\mathcal{X}(E)$ denotes the corrosion degree of the detection process. The corrosion degree describes how the extraction contributes to (or denigrates) the watermark quality depending on the process's false positive detection rate. We identify the embedded watermark signal as wm , the extracted one as wm' , and the quality of the extracted signal as $Q(wm')$. The topological factor $\mathcal{O}(S)$, where S denotes the watermark source, features the relevance of the watermark source when using meaningful approaches (i.e., techniques that generate the watermark using meaningful external sources such as images or audio, among others) [11]. When using a meaningless watermark, the value of $\mathcal{O}(S)$ remains constant for all

TABLE 2. Samples of topologically different binary images.

Case	A	B	C	D	E	F
S						
$\mathcal{O}(S)$	0.7708	0.9583	0.6666	0.9333	1.0000	0.9375

experiments to highlight the role played by $\mathcal{X}(E)$ and $\mathcal{X}(q)$.

$$Q(wm') = \frac{\mathcal{O}(S)}{\mathcal{X}(E) + \mathcal{X}(q)} \quad (2)$$

According to (2), the higher the number of false positives generated by the extraction process, the lower the quality of the detected watermark. The exact relationship occurs between the query's complexity and the detected watermark's quality. For cases when $\mathcal{X}(E)$ and $\mathcal{X}(q)$ present low relevance, the quality presents higher dependency on the topological factor of the watermark source. The following sections formally introduce the watermark source's topological factor and the detection process's corrosion degree.

A. TOPOLOGICAL FACTOR OF THE WATERMARK SOURCE

The use of meaningful watermarks contributes to perceiving the quality of the detected signal by human systems, such as the auditory or the visual. Also, it allows the application of metrics to evaluate the quality of the extracted watermark using the embedded one as a reference. Utilizing meaningful watermarks makes it possible to apply algorithms to enhance the quality of the detected signal. They allow the watermark detection even when the embedding process does not consider all the marks, or benign updates and attacks compromise some marks' values. For example, when using a bilateral symmetric image as a watermark source, one half can contribute to its reconstruction if the other half is damaged. Moreover, neighboring pixels can help to recover the ones missed near them.

Using binary images as the watermark source is particularly relevant in our work. All benefits previously mentioned are added to the watermarking technique, whereas only one mark is generated for each pixel, considering they can only store 0 or 1 as a value. On the contrary, when selecting a colored image, transforming a pixel value between 0 and 255 produces more marks per pixel (even higher considering each pixel results from combining three values corresponding to the channels of the blue, green, and red colors). Then, using this kind of source requires embedding more marks than when utilizing binary images, which results in higher distortion to the database.

If the image size is small, the number of marks required to be embedded to guarantee the consideration of the entire source during the embedding is also smaller than sources of larger sizes. By considering sources with these characteristics, watermarking techniques become more functional. Furthermore, watermarks generated from binary images with a unique pattern can be easier to detect, even if the queries compromise too many marks. Hence, the size of the image must be as small as possible as long as it allows the representation of unique topological features. Table 2 depicts

samples of images featured with patterns obtained with a small number of pixels.

In this work, we introduce $\mathcal{O}(S)$ to objectively measure the trade-off between a watermark source's size and the presence of unique topological features. Note that obtaining $\mathcal{O}(S)$ depends on the data type of the watermark source (e.g., audio, image, and video files). Since we use a binary image as S , besides the image size, we focus on detecting similar patterns of pixel spreading.

We define a small feature matrix F of size $n \times n$ to analyze the different fragments obtained from the image. We identify them by increasing the column in one unit and the row for each column. That way, the following matrix will overlap the previous one in all columns or rows except one, guaranteeing the inclusion of all possible features in their comparison. A graphical view of this process is depicted in Fig. 2, using a feature matrix F of degree $n = 3$, which ensures the analysis of all possible pixel combinations of Case A shown in Table 2.⁷

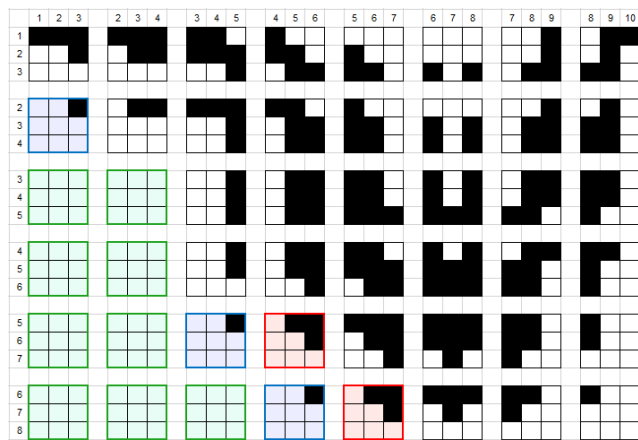


FIGURE 2. Example of image dissection for the topological factor computation (Case A of Table 2).

The process to obtain $\mathcal{O}(S)$ consists of detecting all unique combinations of pixels presented in F and finding their proportion concerning the total number of combinations given the size of the watermark source. For Case A (of size $w = 10$ and $h = 8$, being h and w the height and width of the image, respectively), for the feature matrix of size $n = 3$, the number of combinations is 8×6 . Also, nine feature matrices are composed of white pixels; three have a single black pixel in the top-right position, and two have three black pixels in the top-right corner (see green, blue, and red matrices of Fig. 2, respectively). The rest of the feature matrices contain a unique distribution of pixels. The values computed to obtain $\mathcal{O}(S)$ for Case A are as follows $\mathcal{O}(S) = \frac{(6 \times 8) - (9 + 3 + 2) + 3}{6 \times 8} = \frac{37}{48} = 0.7708$.

⁷To set the size of the feature matrix F , it is important to consider the dimensions of the source S . If n is too big, then F is unfit for spotting unique topological features, given that fewer dissections of S can be performed to analyze the image. This results in identifying more patterns as unique. On the contrary, if n is too small, it is not possible to register all unique patterns in the image, given that the matrix F can be used to register only $2^{(n \times n)}$ cases as unique since each pixel of a binary image can store only one of two values, 0 or 1.

Algorithm 1 breaks down the procedure to obtain $\mathcal{O}(S)$. In the algorithm, F_S is the container of all detected feature matrices. The algorithm's inputs are the watermark source S , its width and height (denoted by w and h , respectively), and the size of the feature matrix (given by n). Lines 2 and 3 consider all possible combinations to generate the feature matrices, and lines 5 and 6 store the matrices with unique patterns in F_S . Finally, line 7 computes the topological factor's final value.

Algorithm 1 Computation of $\mathcal{O}(S)$

```

Input:  $S, w, h, n$ 
Output:  $\mathcal{O}$ 
1  $F_S[] = 0$ 
2 for  $i = 0$  to  $h - (n + 1)$  do
3   for  $j = 0$  to  $w - (n + 1)$  do
4      $F \leftarrow \text{getFeatureMatrix}(S, i, j, n)$ 
5     if  $F$  not in  $F_S$  then
6        $F_S.add(F)$ 
7  $\mathcal{O} = F_S.size() / ((w - n + 1) \times (h - n + 1))$ 
    
```

The last row of Table 2 shows the value of $\mathcal{O}(S)$ for each case. Notice how minor graphical differences between Case B and A result in significant quantitative differences for the topological factor for images of the exact sizes. Also, reducing redundant values from Case C to Case D provokes an increment of $\mathcal{O}(S)$. Notice that the simplicity of the image for Case C causes the smaller $\mathcal{O}(S)$ value. The comparison with Case B vs. Case F and Case C vs. Case D evidences the role played by the size of the image. Finally, the irregular patterns of Case E guarantee the absence of similarity among the feature matrices, which results in the highest possible value of the topological factor.

B. CORROSION DEGREE OF THE DETECTION PROCESS

Similar to the query's complexity, the number of tuples and attributes involved in the watermark detection plays a crucial role in determining the process's complexity. When analyzing the corrosion degree of the detection process for different techniques, the watermark extraction must be performed under the same conditions to keep an objective evaluation. Thus, it is essential to perform watermark synchronization on the same volume of data.

We identify two approaches to obtain the corrosion degree of the extraction process: (i) focusing on the features of the process itself or (ii) focusing on the results obtained with the extraction. The process description considers the technique's verifiability/detectability [7]. It analyses features such as the type of detection (*probabilistic* vs. *deterministic*) and the technique's *blindness*. On the other hand, focusing on the results makes it possible to design and implement the watermarking technique using reverse engineering principles based on the evidence given by the persistence of the watermark in the data selected to perform the extraction.

To obtain the corrosion degree by evaluating the extraction process results, we define the rate of rightly-detected marks according to $R(E) = \omega_{\mathcal{R}}/\omega$, where $R(E)$ denotes the rate, ω is the number of embedded marks, and $\omega_{\mathcal{R}}$ is the number of rightly-detected marks. On the other hand, we define $M(E)$ as the rate of marks missed or not detected by the process, obtained according to $M(E) = (\omega - \omega_{\mathcal{R}})/\omega$.

Regardless of the process features, the corrosion degree decreases when the detection accuracy increases. It experiences an inverse proportion with the rate of rightly detected marks (i.e., $\mathcal{X}(E) \propto 1/R(E)$). On the contrary, the process complexity experiences a direct proportion to the rate of missed marks.

With the definition of the detection process' corrosion degree, we cover all the terms of (2).

IV. EMPIRICAL EVALUATION

In this section, we present the results of the experiments performed to evaluate the resilience of the watermark signal in queried data. We identify as R^1 the version of R containing the watermark. The study measures the quality of the watermark signal detected from the data obtained with each query. In some cases, variations of the extraction process are applied to provide data owners with strategies that might help to identify a stronger watermark signal under suspicion of specific attacks.

The source code of the watermarking techniques used in the experiments, the images used as watermark source, the scheme and content of the databases, and the SQL queries are available in the online GitHub repository at <https://github.com/Gort82/MWTWEval>.

A. EXPERIMENTAL SETUP





As Sections I and II mentioned, we empirically evaluate the watermark preservation in content obtained from querying watermarked data. We used the semantic-driven IBW approach defined in [20] to synchronize the watermark. This approach classifies as a multi-word textual cover type one. It selects the tuples and attributes to embed the marks according to the AHK algorithm. It performs mark embedding by replacing the words with one of their synonyms according to the following procedure. First, the pseudo-random selection of a word from a sentence stored in the relation occurs. Then, the technique identifies the selected word's meaning according to the context (i.e., the sentence containing the word) and uses the set of synonyms linked to the meaning for the replacement. We denote the set of synonyms as \mathcal{Z} . This technique uses a binary image as a watermark source, so each mark consists of a pixel from the image. If the value of the selected mark is 1, the process replaces the word with the first synonym in \mathcal{Z} . On the other hand, if the mark's value is 0, the ϑ -th synonym is selected instead. Depending on its position in \mathcal{Z} , sometimes the synonym chosen for the mark embedding can be the same word. In those cases, the embedding of the mark does not cause any distortion.

A critical part of the watermarking technique in [20] is the word disambiguation (WSD) engine, which chooses the most convenient synonyms for each selected word based on the identified context. It plays a critical role, considering word disambiguation is not 100% accurate. If the sets of synonyms chosen for the extraction differ from those used during the embedding, the watermark synchronization fails. For future reference in this work, we identify the technique presented in [20] as SD-MW (a.k.a Semantic Driven Watermarking).

We also use the multi-attribute numerical-cover type technique introduced in [5] to compare the consequences when querying different watermarked data types. The latter is another IBW technique based on the AHK algorithm that embeds the mark in a pseudo-randomly selected bit out of ξ least significant bits (*lsb*) of the numerical value. For future reference, we identify this technique as MA-NM (a.k.a Multi-Attribute Numerical Watermarking).

We select four images for the watermark generation to evaluate the relevance of the source's topology. Table 3 depicts each one with its corresponding topological factor. Furthermore, we highlight with red color the image pixels that are not considered in the embedding or not detected from the watermarked relation during the extraction. Formally, we define them as missed pixels (See Sections IV-B, IV-D, and IV-C for examples of images with missing pixels).

TABLE 3. Watermark sources used in the experiments.

S	UTM	WWF	Đào	Puzzle
Image				
length (pixels)	82 × 80	40 × 45	20 × 21	6 × 6
$\mathcal{O}(S)$	0.0628	0.0759	0.2924	1.0000

During watermark embedding, the selected techniques can choose the same pixel multiple times to overcome possible watermark degradation due to minor updates. This results in the recurrent embedding of the same mark in different places of the relation. Then, the extraction process carries out a majority voting to decide, from the multiple candidates, the final value of the pixel. That way, the technique corrects minor inconsistencies in extracted values. For this reason, the number of pixels with no values depends on the watermark size. For the same number of tuples and attributes, the bigger the watermark source's length, the higher the number of red pixels in the image generated from the extracted watermark. The experiments of Section IV involving the different watermark sources of Table 3 confirm this statement.

We performed the watermark synchronization with SD-MW on two different datasets storing multi-word textual content. The first is the *Amazon Fine Food Reviews* dataset (denoted as \mathcal{D}_A),⁸ which is composed of 500,000 tuples and 9 attributes. In this case, we mainly select the attribute 'Text' as the carrier,

⁸The *Amazon Fine Food Reviews* dataset is available online in the kaggle repository at <https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews> [45]

TABLE 4. Lengths of multi-word textual attributes from \mathcal{D}_A and \mathcal{D}_B (attributes selected as carriers are highlighted in blue color).

Data	Attribute Name	Attribute Description	Length		
			Max	Mean	
\mathcal{D}_A	ProductId	Id. of the product	1	1	
	UserId	Id. of the user	1	1	
	SUMMARY	Review summary	12.80×10^1	23.42×10^0	
	TEXT	Text of the review	10.33×10^3	43.32×10^1	
	CONTENUTI_ENG	Content of the course (english version)	38.22×10^2	70.69×10^1	
	TESTI_RIF_ENG	Bibliographic references of the course (english version)	39.34×10^2	66.46×10^1	
	OBIETT_FORM_ENG	Objectives of the course (english version)	36.12×10^2	65.00×10^1	
	PREREQ_ENG	Requirements to get enrolled in the course (english version)	20.08×10^2	16.66×10^1	
	\mathcal{D}_B	MOD_VER_APPR_ENG	Exam modality in english (written/oral)	12.10×10^1	8.47×10^0
		ALTRO_ENG	Other notes in english about the course program	32.59×10^2	44.64×10^1
METODI_DID_ENG		Didactic methods applied in the course, in english	31.07×10^2	22.95×10^1	
MOD_VER_DETT_ENG		Details about the course exam and minimum score to pass in english	37.86×10^2	52.25×10^1	
RIS_APPR_ENG		Learning outcomes expected from students in english	37.72×10^2	95.40×10^1	

TABLE 5. Information of the dataset \mathcal{D}_C (numerical cover-type).

Attribute Name	Attribute Description	BL(Avg)	Max	Min	Mean	StdDev
ELEVATION	Elevation in meters	11.99	3849	1863	2780.08	322.30
ASPECT	Aspect in degrees azimuth	7.11	360	0	144.60	108.17
SLOPE	Slope in degrees	4.13	61	0	14.19	8.13
HOR_DIST_TO_HYDROLOGY	Horz Dist to nearest surface water features	7.13	1343	0	207.04	183.73
VERT_DIST_TO_HYDROLOGY	Vert Dist to nearest surface water features	4.33	554	-146	38.63	50.41
HOR_DIST_TO_ROADWAYS	Horz Dist to nearest roadway	11.40	7117	0	2643.32	1895.24
HILLSHADE_9AM	Hillshade index at 9am, summer solstice	7.99	254	0	215.53	27.04
HILLSHADE_NOON	Hillshade index at noon, summer solstice	8.00	254	99	221.77	19.61
HILLSHADE_3PM	Hillshade index at 3pm, summer solstice	7.57	248	0	136.57	38.05
HOR_DIST_TO_FIRE_POINTS	Horz Dist to nearest wildfire ignition points	11.67	7173	0	3210.02	2157.09

considering it stores texts with higher lengths. The second dataset consists of the database of courses from the Ca' Foscari University of Venice (denoted as \mathcal{D}_B).⁹ In the latter, the number of attributes storing long textual content is higher than in the *Amazon Fine Food Reviews* dataset, and the length of the textual content, on average, is also higher. Table 4 presents the description of the attributes of both datasets (\mathcal{D}_A and \mathcal{D}_B) that are more relevant to our research. The table shows each attribute's maximum and average length (columns **Max** and **Mean**, respectively), using the number of characters of their values as references.

On the other hand, we used the dataset *Forest Cover Type* (denoted as \mathcal{D}_C)¹⁰ to evaluate the watermark preservation in numerical data. In this case, we perform the watermark synchronization with MA-NM. We considered the same number of tuples for each dataset (i.e., $\eta = 3,628$) for a fair comparison of the results. For the case of \mathcal{D}_C , we use only the first 10 attributes. Notice that *Forest Cover Type* dataset comprises 581,012 tuples and 54 numerical attributes.

⁹The content of the courses database from the Ca' Foscari University of Venice is available online at <https://www.unive.it/pag/10478/>

¹⁰The *Forest Cover Type* dataset is available online in the University of California Irvine (UCI) machine learning repository at <http://kdd.ics.uci.edu/databases/coverttype/coverttype.html> [46]

Table 5 presents the attributes in \mathcal{D}_C that we use in our research. The table shows the average of each attribute's binary length (column **BL(Avg)**),¹¹ their maximum and minimum values (columns **Max** and **Min**, respectively), and the mean and standard deviation (columns **Mean** and **StdDev**, respectively) of the numerical distribution of their values.

Since this work targets the quality of the watermark and the data resulting from the query, we use metrics oriented to evaluate those aspects. Considering that the technique uses an image to generate the watermark and generates an image from the extracted watermark, we compute the Correction Factor (CF $\in [0, 100]$) and the Structural Similarity Index (for our purposes, SSIM $\in [0, 1]$) to evaluate the quality of the watermark.¹²

The Correction Factor, computed with (3), compares the pixels of the image built from the embedded watermark (given by I_{emb}) against the ones of the image generated from the

¹¹The binary length **BL** denotes the number of bits used for the value binary representation. The bigger the binary length, the higher the cover for mark embedding.

¹²For the sake of simplicity, we use binary images as the watermark source. Even though the SSIM is more often used to evaluate gray-scale image similarity, considering them as sources increases the capacity of the watermark, compromising the data quality due to the distortion required during the embedding.

extracted watermark (given by I_{ext}). In (3), h and w represent the height and width of the images, respectively. If both watermarks are identical, then $CF = 1$. On the contrary, the watermarks are different if $CF = 0$.

$$CF = \frac{\sum_{i=1}^h \sum_{j=1}^w (I_{emb}(i, j) \oplus \overline{I_{ext}(i, j)})}{h \times w} \times 100 \quad (3)$$

Considering that the Correction Factor is a correlation-based metric, high CF values do not always fit to describe the similarity between images, especially for cases when tuples are added to R (see Section IV-D). We use SSIM for a more objective evaluation of the watermark quality. The SSIM represents image similarities, considering a perception closer to how the Human Visual System works. We compute this metric according to (4), using windows of exact size defined by x and y . In (4), N denotes the windows' sizes, μ_x and μ_y represent the average of x and y respectively, σ_x^2 and σ_y^2 their variance, and σ_{xy} their covariance. Also, the symbols C_1 and C_2 constitute stabilization constants aiming to characterize the saturation effects of the visual system at low luminance and contrast regions and that assure numerical stability when the denominators are close to zero [47]. If $SSIM = 1$, the metric describes a perfect structural similarity between the two images. On the other hand, if $SSIM = 0$, both images lack similarity.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (4)$$

We implemented the selected techniques based on a client-server architecture. We used Java 1.8 and Eclipse Integrated Development Environment (IDE) 4.21 for the client layer and Oracle Database 19C database server with Oracle SQL Developer 21.4 IDE for the server side. The run-time environment was a 4.20 GHz Intel i7-7700K PC with 32.0 GHz of RAM and Windows 10 Pro OS.

B. QUALITY OF THE SYNCHRONIZED WATERMARK

The first results describe the quality of the embedded watermark. We use them to set the comparison point before performing the queries on R' . According to the AHK algorithm [48], the tuple fraction (denoted by $\gamma \in [1, \eta]$) allows controlling the number of watermarked tuples. The algorithm sets the trade-off according to $\gamma \approx \eta/\omega$, where η is the number of tuples in R , and ω is the number of watermarked tuples (also used to identify the number of embedded marks if the technique embeds just one mark per tuple, see Section III-B). The tuple fraction states the marking of one of each γ tuples, so when $\gamma = 1$, the process watermarks all tuples, resulting in a more substantial presence of the watermark signal in R' at the cost of higher distortion.

Table 6 depicts the quality of the embedded watermark using each one of the sources previously introduced in Table 3 and marking different numbers of tuples by considering different tuple fraction values. In the table, SD-MW¹ identifies the case of SD-MW embedding the watermark into \mathcal{D}_A , and SD-MW² into \mathcal{D}_B (see the carriers for each dataset in Table 4).

TABLE 6. Quality of the embedded watermark using each technique and marking different tuple numbers.

Tech.	$\gamma = 1$				$\gamma = 5$			
	UTM	WWF	Đào	Puzzle	UTM	WWF	Đào	Puzzle
MA-NM								
	0.37	0.83	1.00	1.00	0.11	0.33	0.83	1.00
SD-MW ¹								
	42.91	87.44	100.00	100.00	10.06	31.444	79.52	100.00
SD-MW ²								
	0.63	0.98	1.00	1.00	0.24	0.64	0.99	1.00
	79.43	99.38	100.00	100.00	26.67	67.44	98.80	100.00
	0.85	1.00	1.00	1.00	0.49	0.95	1.00	1.00
	98.52	100.00	100.00	100.00	58.44	96.05	100.00	100.00

TABLE 7. Quality of the detected watermark.

Tech.	$\gamma = 1$				$\gamma = 5$			
	UTM	WWF	Đào	Puzzle	UTM	WWF	Đào	Puzzle
MA-NM								
	0.37	0.83	1.00	1.00	0.11	0.33	0.83	1.00
SD-MW ¹								
	42.91	87.44	100.00	100.00	10.06	31.44	79.52	100.00
SD-MW ²								
	0.40	0.80	1.00	1.00	0.14	0.40	0.86	1.00
	64.61	94.55	100.00	100.00	20.25	55.05	91.42	100.00
	0.69	1.00	1.00	1.00	0.37	0.76	1.00	1.00
	94.40	100.00	100.00	100.00	50.36	90.22	100.00	100.00

On the other hand, watermark embedding in \mathcal{D}_B is performed with MA-NM, considering approximately one attribute per tuple, according to $\delta = 9$, where $\delta \in [1, \nu]$ denotes the attribute fraction and ν the number of attributes in R . Same as the tuple fraction, if $\delta = 1$, all tuple attributes are used as carriers, causing higher distortion during the embedding [5].

For each case, we show the values of SSIM and CF to describe the embedded watermark quality, along with the image built from the extracted signal. The table shows that when the number of tuples and attributes remain constant if the watermark length is smaller, the process excludes fewer pixels, embedding a stronger signal and guaranteeing higher robustness. Nevertheless, as previously mentioned, using a small-length source can deprive the watermark of structural particularities that might increase the chances of its identification. This is why using a source that is too small is not recommended.

Increasing capacity contributes to robustness, but techniques will only succeed if they do not compromise the quality of the database when embedding the watermark. For this reason, the embedding process must produce as little distortion as possible. The watermark must be unnoticed in R' . The use of synonym replacement when watermarking multi-word textual attributes is a great strategy to avoid the impact of distortion. Nevertheless, high word disambiguation accuracy is essential to guarantee watermark recognition, given that techniques based on the word disambiguation process present

TABLE 8. Quality of the detected watermark (Q_{S1} + horizontal filtering).

Tech.	$\gamma = 1$				$\gamma = 5$			
	UTM	WWF	Đào	Puzzle	UTM	WWF	Đào	Puzzle
MA-NM								
	0.21	0.61	0.97	1.00	0.06	0.18	0.56	1.00
SD-MW ²								
	23.49	62.11	97.61	100.00	4.87	16.83	52.38	100.00
MA-NM								
	0.55	0.91	1.00	1.00	0.22	0.59	0.94	1.00
SD-MW ²								
	79.58	98.66	100.00	100.00	29.31	72.66	97.38	100.00

a higher probability of adding false positives than deterministic numerical cover-type techniques.

Table 7 shows the quality of the image built from the extracted watermark for the experiments in Table 6. When using multi-word textual cover type techniques, if the detection process selects the wrong set of synonyms, it can assign incorrect values to the marks. If the number of marks with wrong values is too high, the majority voting also assigns incorrect values to the pixels of the reconstructed image. Therefore, the final image will contain salt and pepper noise.

C. EFFECT OF SELECT QUERIES

Analyzing the quality of the watermark detected after performing queries on R^1 can help consider alternatives to increase the technique's resilience. In particular, when using meaningful watermarking approaches, such as IBW, there is the alternative of changing the watermark source by another one, including features allowing its reconstruction despite the partial signal extraction. Furthermore, the quality of the extracted watermark can offer evidence (or cause suspicion) of the performance of a particular malicious operation over the data. Therefore, variations in the watermark extraction process can boost the quality of the detected signal, increasing the chances of its recognition.

We perform empirical evaluation from the simplest to the most complex case (i.e., from considering queries that do not affect watermark traces in the resulting dataset to queries that cause the most damage). We exclude the Puzzle watermark source from the results shown in Figs. 3 to 10, given that we always obtain the maximum quality when using it (i.e., $CF = 100\%$ and $SSIM = 1$). The first experiments correspond to *select* queries that recover all marks embedded in R^1 . This is Q_{S1} where $\mathcal{X}(q) = 0$ due to satisfying the conditions $\nu_A = 0$ and $\eta_A = 0$. These conditions must guarantee the detection of each one of the embedded marks. Nevertheless, the watermark detection complexity $\mathcal{X}(E)$ is higher in SD-MW, mainly because of the WSD engine, which causes a reduction in the detected watermark quality. In this sense, SD-MW does not offer the same guarantees as MA-NM, which keeps a lower rate of false positives (see Table 7).

The number of missed marks increases for *select* queries that consider all carriers but apply horizontal filtering with the SQL WHERE clause. Fewer marks directly impact the quality of the image built from the extracted watermark signal. We highlight

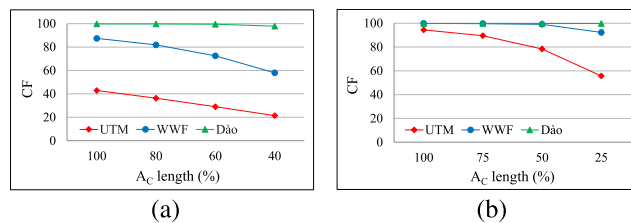


FIGURE 3. CF of detected watermarks after losing carriers in T^1 (a) Watermark signal detected in T^1 (technique: MA-NM, data source D_C). (b) Watermark signal detected in T^1 (technique: SD-MW, data source D_B).

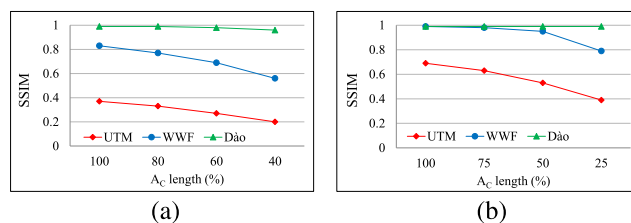


FIGURE 4. SSIM of detected watermarks after losing carriers in T^1 (a) Watermark signal detected in T^1 (technique: MA-NM, data source D_C). (b) Watermark signal detected in T^1 (technique: SD-MW, data source D_B).

the positions with undetected marks with red pixels. Table 8 shows the watermark quality for these cases. For each one, the number of tuples resulting from the horizontal filtering is 1758, which describes an increment of $\mathcal{X}(q)$.

For cases based on Q_{S1} queries excluding attributes from the selection, the quality of the watermark signal will depend on the number of carriers in the resulting table T^1 . Given the significant content stored in multi-word textual attributes and the possibility of storing more than one mark on each value, the quality of the watermark signal remains higher for SD-MW than for MA-NM, independently of the number of carriers lost from the list A_C (see Figs. 3 and 4). For this experiment, we used SD-MW with the dataset D_B , considering that D_A has only one attribute with content large enough to achieve good watermark synchronization. On the other hand, if the *select* query excludes the only carrier considered for D_A , the watermark synchronization fails. From the figures, the relevance of the watermark source is evident. Notice that the source of the highest topological factor makes it possible to obtain the highest robustness.

Moreover, when applying the same query as the one used for the results shown in Figs. 3 and 4, but selecting only 1758 tuples due to adding horizontal filtering to the query (as in Table 8), the quality of the detected watermark decreases due to the loss of marks. However, the quality proportion remains in each case, considering the watermark source, technique, and the number of remaining carriers (see Figs. 5 and 6).

A general view of the consequences of applying horizontal filtering to Q_{S1} queries is depicted in Figs. 7 and 8. In the figure, the same colors and markers are used for the same watermark sources, depicting the signals extracted after applying queries with filters in a lighter version of the color. Also, the legend includes the suffix '-F' at the end of the name of each watermark detected from filtered data. The general

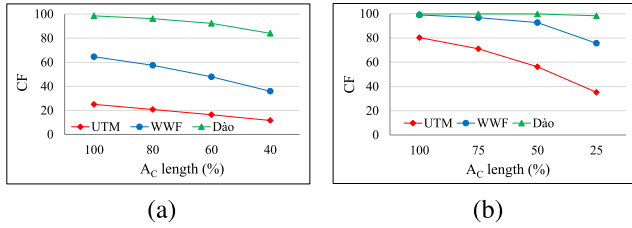


FIGURE 5. CF of detected watermarks from T' losing carriers and applying horizontal filtering (a) Watermark signal detected in T' (technique: MA-NM, data source \mathcal{D}_C). (b) Watermark signal detected in T' (technique: SD-MW, data source \mathcal{D}_B).

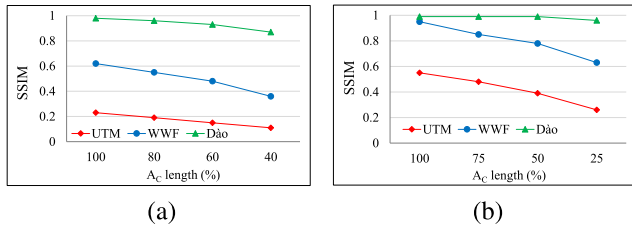


FIGURE 6. SSIM of detected watermarks from T' losing carriers and applying horizontal filtering (a) Watermark signal detected in T' (technique: MA-NM, data source \mathcal{D}_C). (b) Watermark signal detected in T' (technique: SD-MW, data source \mathcal{D}_B).

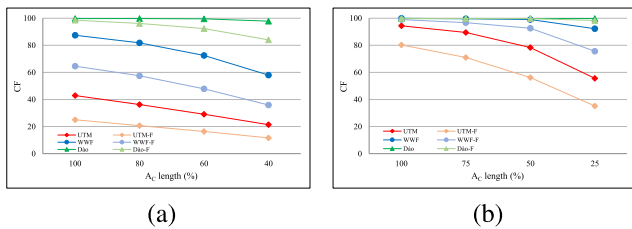


FIGURE 7. CF of detected watermarks from T' losing carriers. The results are obtained after applying Q_{S1} queries used to obtain results from Figs. 3 and 5 (same queries excluding and applying horizontal filtering) (a) Watermark signal detected in T' (technique: MA-NM, data source \mathcal{D}_C). (b) Watermark signal detected in T' (technique: SD-MW, data source \mathcal{D}_B).

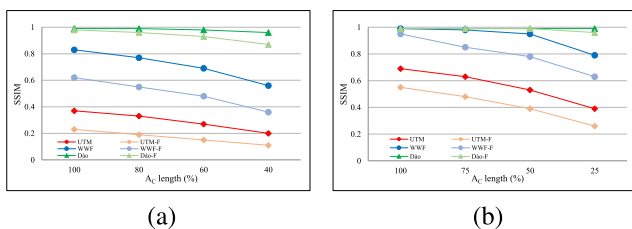


FIGURE 8. SSIM of detected watermarks from T' losing carriers. The results are obtained after applying Q_{S1} queries used to obtain results from Figs. 4 and 6 (same queries excluding and applying horizontal filtering) (a) Watermark signal detected in T' (technique: MA-NM, data source \mathcal{D}_C). (b) Watermark signal detected in T' (technique: SD-MW, data source \mathcal{D}_B).

view given by these figures clearly shows the denigration of the watermark signal's quality due to the exclusion of tuples containing carriers after applying filters. Notice how a light line of the same color is drawn below each dark line.

The following results depict the resilience of the watermark after applying a query of type Q_{S2} to the watermarked dataset. The query merges two carriers to produce a new column. For this case, we analyze the watermark's preservation when using SD-MW on \mathcal{D}_B , considering it contains more

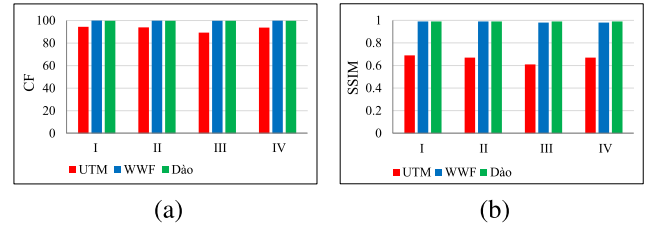


FIGURE 9. Watermarks detected from the resulting T' of Q_{S2} excluding horizontal filtering (a) CF of detected watermark signal. (b) SSIM of detected watermark signal.

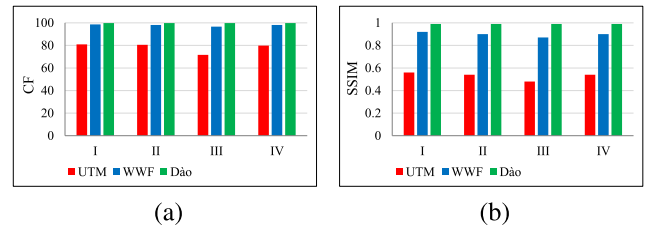


FIGURE 10. Watermarks detected from the resulting T' of Q_{S2} with horizontal filtering (a) CF of detected watermark signal. (b) SSIM of detected watermark signal.

than one multi-word textual attribute. We present four alternatives to form A_C for detecting the watermark: (I) considering the original carriers, (II) using the original carriers plus a new column generated by concatenating the attributes `MOD_VER_DETT_ENG` and `RIS_APPR_ENG`, (III) considering all carriers except the last one, which is replaced by the new column, and (IV) taking into account all carriers except the ones used to generate the new column, which replaces them in A_C (see Figs. 9 and 10).

When the query used to obtain the results shown in Fig. 9 is applied with horizontal filters, leaving only 1758 tuples in the result as before, all detection strategies offer similar results with just a slight denigration of the watermark signal quality for cases using the sources with the minimal topological factor (see Fig. 10). There is no significant damage to the watermark signal when using this query type with multi-word textual attributes.

The last type of *select* query is Q_{S3} . From all *select* queries, this is the one causing more damage to the watermark detection. The main reasons for losing carriers when applying this type of query (given that they are used to decide if the tuple is marked, which attribute within the tuple stores the mark, and to perform the generation of the mark itself) when performing `GROUP BY` operations, and the loss of carriers when using aggregate functions. To generate T' with Q_{S3} , we consider two different grouping criteria to check the quality of the watermark extracted from \mathcal{D}_B . We use the attributes highlighted in blue color in Table 9 for grouping. Furthermore, we obtained each group's maximum `CONTENUTI_ENG` value as an aggregate operation. Those alternatives depict the consequences of losing carriers and primary keys, considering the number of groups is very different for each case. When selecting `AF_GEN_DES`, the number of tuples obtained from the query is closer to the ones contained in the dataset. On the other hand, when using `SETT_COD`, the number of tuples is much lower.

TABLE 9. Number of groups according to \mathcal{D}_B attributes' values.

Attribute	Groups	Attribute	Groups
TIPO_CORSO_COD	10	AF_GEN_COD	2665
CDS_COD	83	AF_GEN_DES	2563
NOME_CDS	78	SETT_COD	187
REGDID_COD	145	MATRICOLA	1120
PDS_COD	63	COGNOME	1061
PDS_DES	82	NOME	617

Table 10 depicts the signal detected after applying Q_{S3} for each case. As expected, the watermark is entirely compromised. Nevertheless, this type of query also compromises the data quality. Therefore, if the reasons for their execution are outside regular operations of the organization (e.g., digital reports generation and data summarising), the data owner must consider if claiming the resulting dataset's ownership is worthy, given its quality.

TABLE 10. Signal detected after applying Q_{S3} with no horizontal filtering.

GROUP BY							
AF_GEN_DES				SETT_COD			
UTM	WWF	Đào	Puzzle	UTM	WWF	Đào	Puzzle
0	0.03	0.06	0	0.01	0	0	0
47.71	47.16	49.76	50.00	13.29	32.44	48.57	52.77

We do not perform experiments applying the same Q_{S3} queries with horizontal filtering, considering that the results obtained in Table 10 already show how compromised synchronization can be. For these cases, the outcome would result in losing carriers after executing the filtering conditions. Therefore, the extracted signal produces a noisy image as before. Their only difference will be the higher number of red pixels in the images for this case due to tuple filtering.

D. EFFECT OF ACTION QUERIES

Regarding *action* queries, we proceed with the watermark detection by performing experiments considering Q_{A1} , Q_{A2} , and Q_{A3} , respectively. These queries are the core of benign operations and might be used to implement malicious operations or attacks seeking the removal of the watermark. Therefore, the number of marks contrasting with the correct values is expected to be proportional to the number of modified values. Nevertheless, incremental watermarking implementation contributes to correcting the noise added to the watermark signal due to *action* queries [7], [11].

Table 11 depicts the quality of the watermarks detected with SD-MW after applying Q_{A1} queries. The number of inserted tuples increases for each experiment by 10% of the number of tuples originally watermarked in \mathcal{D}_B . As previously mentioned, in SD-MW, the performance of the WDS engine is responsible for adding false positives. Nevertheless, in these experiments, the false positives also increased due to the number of inserted tuples. The higher the number of false positives, the higher the presence of salt and pepper noise

in the images reconstructed from the extracted watermark. In particular, for the case of the UTM watermark, higher salt and pepper noise is added, given the low topological factor of the watermark source. On the contrary, for the cases of Đào and Puzzle watermarks, the majority voting in combination with a higher topological factor contributes to avoiding all noise from the false positives detected in \mathcal{D}^I .

TABLE 11. Detected watermarks after inserting different tuple numbers.

Q_{A1}	0	10	30	50	70	90
UTM						
	0.69	0.66	0.60	0.57	0.53	0.52
	94.40	93.53	91.63	90.03	88.46	87.19
WWF						
	0.99	0.98	0.98	0.97	0.93	0.89
	99.94	99.72	99.77	99.38	98.33	97.88
Đào						
	1.00	1.00	1.00	1.00	1.00	1.00
	100.00	100.00	100.00	100.00	100.00	100.00
Puzzle						
	1.00	1.00	1.00	1.00	1.00	1.00
	100.00	100.00	100.00	100.00	100.00	100.00

Table 12 presents the results regarding the watermark detected after performing attribute updates with Q_{A2} . For this case, we obtained similar results to those shown for Q_{A1} . The watermarks in the table correspond to the detection after one of the carriers (pseudo-randomly selected) takes the value of the same attribute stored in a different tuple. Selecting the value assigned in the update from R^I makes it possible to consider values within the domain of the corresponding attribute. This time, we also considered a different number of tuples for each case (increasing by 10% the initial number of tuples in R). This way, we illustrate the degree of damage caused to the watermark by modifying different volumes of data in the protected relation. As depicted, the watermark's resilience remains proportional to the topological factor of its source.

Finally, Table 13 shows the quality of the watermark detected after deleting a different number of tuples from R^I . After applying Q_{A3} queries, the loss of carriers causes a degradation of the watermark signal. As expected, the number of red pixels gets more significant as the number of tuples affected by the queries increases. Nevertheless, we spot an intriguing effect. The loss of carriers compromises the majority voting performance, and some false positives make their way to the image built from the watermark signal, slightly increasing the presence of salt and pepper noise. This fact is more straightforward to spot in the watermarks experiencing higher resilience to Q_{A3} queries, given the higher topological factor of the sources (a.k.a Đào and Puzzle).

TABLE 12. Detected watermarks after updating different tuple numbers.

Q_{A2}	0	10	30	50	70	90
UTM						
	0.69	0.67	0.63	0.61	0.58	0.56
	94.40	93.41	92.24	90.88	89.00	87.98
WWF						
	0.99	0.99	0.99	0.95	0.93	0.95
	99.94	99.88	99.83	99.72	99.16	99.05
Đào						
	1.00	1.00	1.00	1.00	1.00	1.00
	100.00	100.00	100.00	100.00	100.00	100.00
Puzzle						
	1.00	1.00	1.00	1.00	1.00	1.00
	100.00	100.00	100.00	100.00	100.00	100.00

TABLE 13. Detected watermarks after deleting different tuple numbers.

Q_{A3}	0	10	30	50	70	90
UTM						
	0.69	0.67	0.61	0.54	0.45	0.22
	94.40	93.00	87.97	79.40	64.05	31.00
WWF						
	0.99	0.99	0.94	0.92	0.82	0.54
	99.94	99.83	99.55	98.94	94.50	69.44
Đào						
	1.00	1.00	1.00	1.00	1.00	0.93
	100.00	100.00	100.00	100.00	100.00	96.66
Puzzle						
	1.00	1.00	1.00	1.00	1.00	1.00
	100.00	100.00	100.00	100.00	100.00	100.00

Figs. 11, 12, and 13 allow us to make a broader comparison when performing different *action* queries on the watermarked dataset. We compared the synchronized watermark for these cases, considering all watermark sources, techniques, and data sources. Given the high quality obtained by using the Puzzle watermark source, we do not include those results in the figures to avoid unnecessary overloading of the graphics. By using this source, it is possible to obtain the maximum values of SSIM and CF in all cases, except for the experiment performed updating 90% of tuples using SD-MW¹ to synchronize the watermark, having CF = 86.11, and SSIM = 0.68 as results, which is higher than when using SD-MW¹ with Đ o (see Fig. 12).

Fig. 11 shows the quality of the watermark signal detected after applying Q_{A1} queries. In the figure, the Đ o watermark

depicts the higher resilience, thanks to its topological factor. In this case, results do not depend on the technique used to perform the synchronization. On the other hand, when using UTM and WWF as watermark sources, the best results are obtained using the SD-MW technique on \mathcal{D}_B , given the higher cover of the data source for mark embedding. For the case of tuple insertion, it is clear the way SSIM and CF differ due to the noise added to the image reconstructed from the watermark signal. The CF could mislead the actual quality of the signal, given that it is possible to obtain high CF values by sheer chance from noisy images. Thus, when the number of false positives is high, it is recommended to use SSIM instead.

Fig. 12 depicts a similar comparison as Fig. 11 but in terms of Q_{A2} queries. As expected, the watermark synchronized over a single carrier experiences the highest degradation in this case (see SD-MW¹ in the figure). When considering more than one carrier, we randomly select the attribute to be updated. Even if the number of updated attributes is higher for SD-MW² (a.k.a 1/4), the signal quality remains higher than for MA-NM (a.k.a 1/10) due to the higher cover of multi-word textual attributes of \mathcal{D}_B .

The last results show the quality of the detected watermark after performing Q_{A3} queries over the protected dataset. Contrary to when inserting data, SSIM and CF are both good metrics to perceive the watermark quality after deleting tuples from R^1 . As a natural process, the watermark quality will degrade as the number of deleted tuples increases. Nevertheless, it is clear how $\mathcal{O}(S)$ could challenge this outcome to the point of forcing attackers to delete too many tuples to erase traces of the watermark, which could compromise the data quality for both the attacker and the data owner. As stated in (2), the relevance of $\mathcal{O}(S)$ vs. the way false positives increase the corrosion degree of the detection process $\mathcal{X}(E)$ and how the query complexity $\mathcal{X}(q)$ compromises mark carriers are evident.

We remind the reader that key features to overcome these data modifications in the image built from the watermark signal are the recurrent embedding of marks and the majority voting in the extraction process. Also, from the point of view of data preservation, the SD-MW technique guarantees higher capacity while preserving usability. Finally, if the approach considers implementing incremental watermarking, the detection process will benefit from new marks added when inserting or updating the data.

E. TOPOLOGICAL FACTOR'S ROLE GENERALIZATION

To validate the role that the topological factor must play in describing the resilience expected when selecting a binary image as the watermark source, we performed a set of experiments seeking the confirmation or disqualification of previous results on a larger scale. Multiple images of size 40×45 were pseudo-randomly generated and grouped according to their topological factor. We performed the experiments with three categories of images, classified according to the interval of values to which their topological factor belongs (see Table 14).

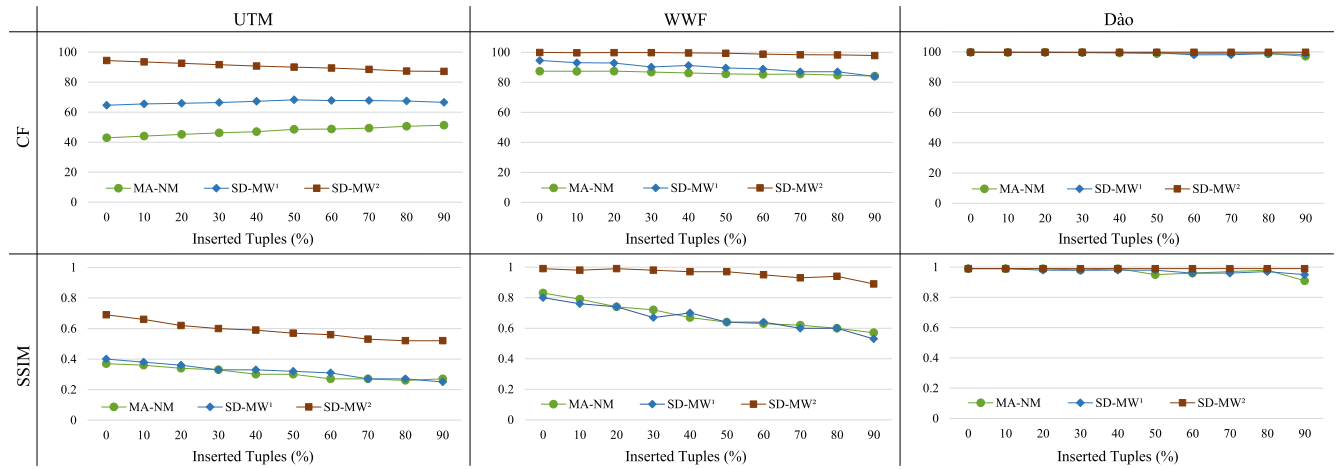


FIGURE 11. Quality of detected watermarks using different cover type techniques after inserting a different number of tuples in R^1 .

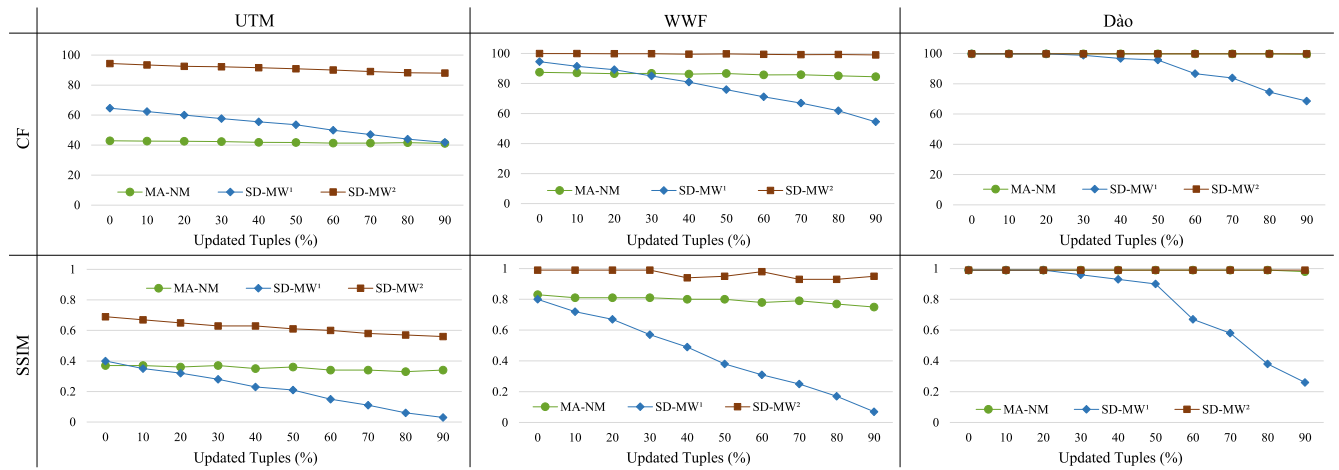


FIGURE 12. Quality of detected watermarks using different cover type techniques after updating a different number of tuples in R^1 .

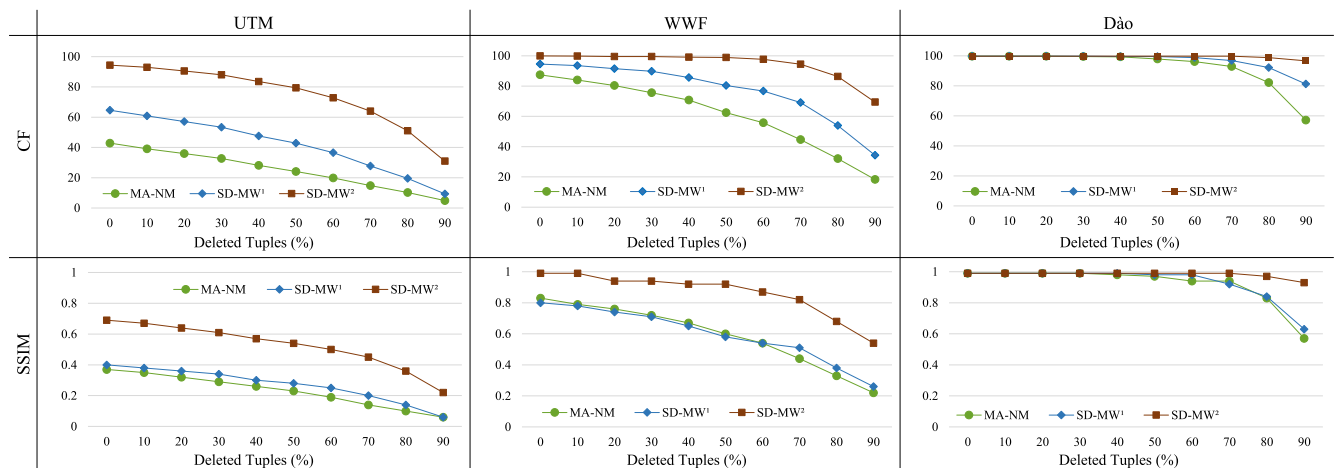


FIGURE 13. Quality of detected watermarks using different cover type techniques after deleting a different number of tuples in R^1 .

Twenty images were generated for each group for a total of 60 sources. For each of them, the generated watermark was embedded and later extracted after executing each *action*

query used in Section IV-D. We selected the *action* queries to generalize the topological factor's validation, considering they are featured by a uniform behavior in terms of the detected

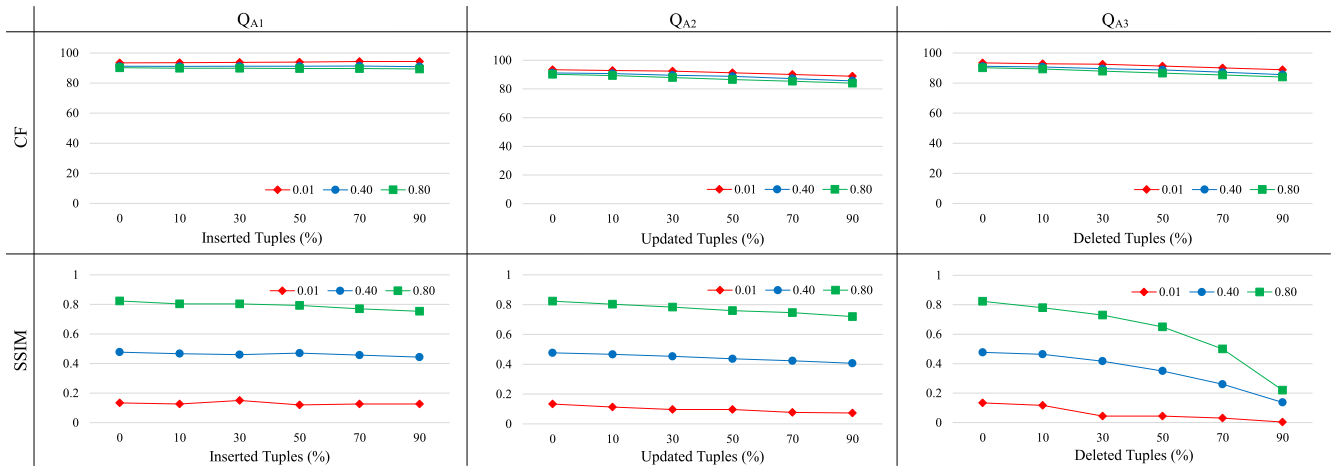


FIGURE 14. Generalization of the topological factor's role in the resilience of the watermark in R' after performing different action queries.

TABLE 14. Groups of images defined according to their topological factor.

Group's Name	Range of Values	
	Min	Max
$O(S) = 0.01$	0.009	0.014
$O(S) = 0.40$	0.350	0.449
$O(S) = 0.80$	0.750	0.849

watermark quality with respect to the volume of modified data. For these experiments, the *action* queries were performed considering degrees of 10%, 30%, 50%, 70%, and 90% of the tuples originally stored in R . The results are depicted in Fig. 14, where the average of SSIM and CF are used as references for each group of images and each degree of tuples affected by the queries. For SSIM, the standard deviation of the registered values was never higher than 0.34, having a mean of $0.0174(\pm 0.0112)$, which describes a good consistency of the values registered for the quality of the synchronized watermark for each topological factor. On the other hand, for CF, the standard deviation of the registered values was $0.6369(\pm 0.7365)$, which describes a slightly larger dispersion compared to SSIM.

Nevertheless, Fig. 14 perfectly illustrates the contrast of CF and SSIM as metrics for registering the similarities between the embedded and the extracted watermarks. For these cases, as a side-effect of using images of the same sizes but different topological factors, SSIM experiences a behavior consistent with the degradation of the watermark, given the number of tuples affected by the queries. Also, SSIM reports values consistent with the topological factor in terms of describing higher resiliency for the watermarks generated with sources having higher topological factors. For all cases in the figure, the signal depicted with the red line, corresponding to the group $O(S) = 0.01$, describes a lower watermark quality compared to the signal in blue and green (groups of topological factors 0.40 and 0.80, respectively). On the contrary, CF fails to describe these differences properly, always reporting a high watermark quality, featured by a way smaller denigration

while the number of tuples affected by the queries increases. Also, by very small differences, there is a shift of roles where a higher resilience is described for the images with lower topological factors.

V. PROPOSAL'S IMPACT

In [19], the authors analyzed the impact of queries on the numerical cover type. Nevertheless, researchers have not examined their impact on the multi-word textual cover type. Furthermore, they still need to address the robustness regarding the topological features of the watermark source for meaningful techniques. In this section, we analyze the results that validate our statements and highlight the main differences in our work concerning previous research in the field.

A. THEORETICAL AND PRACTICAL IMPLICATIONS

Three main aspects feature our research and make it different from previous work: (i) the study of SQL queries as the core of *benign updates* and *malicious operations* that contribute to the degradation of the watermark robustness when using multi-word carriers, (ii) the analysis of the relevance of the cover type against the corrosion degree of the detection process, and (iii) the contribution of the watermark source to the technique's robustness.

Most techniques analyze robustness by increasing the degree of data subjected to the operation without offering alternatives for the extraction or analyzing the fragments of the queries involved. We address these issues to encourage new ways to seek mark preservation and prevent watermark degradation. We are confident that this way of analyzing the watermarking approach can contribute to the design of proactive watermark synchronization, increasing the technique's robustness even when an important number of marks gets compromised due to the attacks.

So far, meaningful techniques have yet to consider the contribution of the watermark source to the technique's robustness. They do not evaluate robustness considering

different sources or try to explain why some sources might be better than others. We offer various examples and provide an objective way to assess the expected success of the techniques, which can prevent the data owner from selecting the wrong source, avoiding time wasting by having to restore the data to start the whole embedding process with a different source.

B. ANALYSIS OF RESULTS

For numerical cover-type techniques, there is a direct link between increasing the capacity to benefit robustness and compromising data quality. As results show, when using meaningful watermarking techniques, the watermark quality increases as the tuple factor γ decreases (see Tables 6 and 7). Nevertheless, the price paid by causing more distortion is not always permissible. This is even more critical when more attributes are marked for each tuple selected as the attribute fraction ν gets closer to 1. Under the requirement of querying the data, the damage caused by the distortion of numerical data is even higher, especially if the queries use criteria based on numerical values to filter the selected tuples. The results will change drastically, compromising both the watermark detected in T^1 or \overline{D}^1 , and the authenticity of the data contained in T^1 . This is due to obtaining a different number of tuples before and after the embedding as a result of excluding values from the filtering due to the distortion and considering others previously ignored. Since more elements need to be marked to increase the watermark quality, η_W and ν_W increase as well, contributing to the growth of $\mathcal{X}(q)$ and resulting in a poor quality of the watermark $Q(wm')$, according to (2).

The direct solution to avoid the downsides of the numerical cover type is selecting textual attributes as mark carriers. In our work, we take particular advantage of multi-word textual attributes, considering they have longer content and, therefore, a more extensive cover for mark embedding. The more significant content also contributes to increasing the accuracy of word disambiguation when performing mark embedding through synonym substitution, as in [20], so no distortion compromising data quality and watermark imperceptibility is produced.

Because word sense disambiguation is not a 100% accurate process, the extracted watermark signal may contain noise [19]. The features of the watermark source can contribute to overcoming the loss of marks as a result of querying the watermarked data. They can help overcome the damage caused by marks extracted that do not match the embedded ones. Therefore, a source with a high topological factor directly benefits the watermark synchronization, mainly when the technique considers recurrent embedding of marks and majority voting in the extraction.

As stated in (2), when $\mathcal{O}(S)$ increases and $\mathcal{X}(q)$ remains the same, the quality of the watermark detected will depend on $\mathcal{X}(E)$. This lets clear the benefits of the multi-word textual cover type compared to the numerical one and highlights the relevance of the techniques' cover type. The term $\mathcal{X}(E)$ expresses the capacity of the technique to experience resilience to the natural degradation of the watermark in

relational data thanks to the types of attributes in the database. The relationship between the topological factor $\mathcal{O}(S)$, the corrosion degree of the watermark detection $\mathcal{X}(E)$, and the query complexity $\mathcal{X}(q)$ rules the expected quality of the synchronized watermark in databases protected using meaningful watermarking techniques already deployed in a public server.

VI. CONCLUSION

In this paper, we evaluated how different types of queries affect the watermark persistence in multi-word textual data stored in relational databases. Since SQL queries are the core of benign updates and malicious attacks, our experiments offer evidence to help data owners choose among a set of extraction alternatives to recover high-quality watermark signals, even if, in some cases, the degree of damage caused to the data and to the embedded watermark is too high. We presented the framework to evaluate how the queries compromise the watermark and formalized the role played by the topology of the watermark source. One significant contribution lies in defining the watermark source's topological factor, which other techniques can utilize to ensure higher robustness. Furthermore, we proved that the features of the extraction process and the technique's cover type determine the quality of the extracted signal.

The conclusions of the empirical evaluation presented in this work are relevant when using meaningful watermarking techniques [11]. The flexibility of computing the topological factor depends on the type of watermark source. On the other hand, the requirements of the recurrent selection of the bits used to generate the marks and the majority voting to enhance the extracted watermark quality are mandatory. They are the only features creating a dependency between the results we obtained and the watermarking techniques. For this reason, as long as the length of the watermark source allows the representation of unique topological features that contribute to the watermark detection, the smaller, the better.

For synonym replacement approaches, the accuracy of the WSD engine plays a crucial role in the number of false positives extracted from the result set. Therefore, if the WSD engine performance is poor, the majority voting function might be compromised, resulting in the extraction of a low-quality watermark signal, which could prevent watermark recognition if the source presents a low topological factor. On the other hand, if the recommendations given in this work are followed, the watermark preservation will experience a behavior similar to the one we have shown. In future work, we pretend to extend the capabilities of multi-word textual cover types regarding semantic consistency and protection of data stored in relational structures and content managed in ontologies.

REFERENCES

- [1] M. T. Ahvanooy, Q. Li, X. Zhu, M. Alazab, and J. Zhang, "ANiTW: A novel intelligent text watermarking technique for forensic identification of spurious information on social media," *Comput. Secur.*, vol. 90, Mar. 2020, Art. no. 101702, doi: 10.1016/j.cose.2019.101702.

- [2] R. Singh, M. Saraswat, A. Ashok, H. Mittal, A. Tripathi, A. C. Pandey, and R. Pal, "From classical to soft computing based watermarking techniques: A comprehensive review," *Future Gener. Comput. Syst.*, vol. 141, pp. 738–754, Apr. 2023, doi: [10.1016/j.future.2022.12.015](https://doi.org/10.1016/j.future.2022.12.015).
- [3] R. Agrawal and J. Kiernan, "Watermarking relational databases," in *Proc. 28th Int. Conf. Very Large Databases*. Amsterdam, The Netherlands: Elsevier, 2002, pp. 155–166, doi: [10.1016/b978-155860869-6/50022-6](https://doi.org/10.1016/b978-155860869-6/50022-6).
- [4] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. San Mateo, CA, USA: Morgan Kaufmann, 2007, doi: [10.1016/b978-0-12-372585-1.x5001-3](https://doi.org/10.1016/b978-0-12-372585-1.x5001-3).
- [5] M. L. Pérez Gort, C. Feregrino Uribe, and J. Nummenmaa, "A minimum distortion: High capacity watermarking technique for relational data," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.* Philadelphia, PA, USA: ACM, 2017, pp. 111–121, doi: [10.1145/3082031.3083241](https://doi.org/10.1145/3082031.3083241).
- [6] M. L. P. Gort, M. Olliaro, and A. Cortesi, "Reducing multiple occurrences of meta-mark selection in relational data watermarking," *IEEE Access*, vol. 10, pp. 62210–62231, 2022, doi: [10.1109/ACCESS.2022.3182099](https://doi.org/10.1109/ACCESS.2022.3182099).
- [7] R. Halder, S. Pal, and A. Cortesi, "Watermarking techniques for relational databases: Survey, classification and comparison," *J. Universal Comput. Sci.*, vol. 16, no. 21, pp. 3164–3190, 2010, doi: [10.3217/jucs-016-21-3164](https://doi.org/10.3217/jucs-016-21-3164).
- [8] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press, 2018, doi: [10.1201/9781439821916](https://doi.org/10.1201/9781439821916).
- [9] A. Alqassab and M. Alanezi, "Relational database watermarking techniques: A survey," *J. Phys., Conf. Ser.*, vol. 1818, no. 1, Mar. 2021, Art. no. 012185, doi: [10.1088/1742-6596/1818/1/012185](https://doi.org/10.1088/1742-6596/1818/1/012185).
- [10] A. Dey, S. Bhattacharya, and N. Chaki, "Software watermarking: Progress and challenges," *INAE Lett.*, vol. 4, no. 1, pp. 65–75, Mar. 2019, doi: [10.1007/s41403-018-0058-8](https://doi.org/10.1007/s41403-018-0058-8).
- [11] S. Rani and R. Halder, "Comparative analysis of relational database watermarking techniques: An empirical study," *IEEE Access*, vol. 10, pp. 27970–27989, 2022, doi: [10.1109/ACCESS.2022.3157866](https://doi.org/10.1109/ACCESS.2022.3157866).
- [12] Y. Nagai, Y. Uchida, S. Sakazawa, and S. Satoh, "Digital watermarking for deep neural networks," *Int. J. Multimedia Inf. Retr.*, vol. 7, no. 1, pp. 3–16, Mar. 2018, doi: [10.1007/s13735-018-0147-1](https://doi.org/10.1007/s13735-018-0147-1).
- [13] M. Mazhar and R. Dhakad, "Survey on relational database watermarking employing evolutionary methods," *J. Inf. Technol. Sci.*, vol. 9, no. 1, pp. 13–29, Jan. 2023, doi: [10.46610/joits.2023.v09i01.003](https://doi.org/10.46610/joits.2023.v09i01.003).
- [14] S. Rani and R. Halder, "An efficient format-independent watermarking framework for large-scale data sets," *Expert Syst. Appl.*, vol. 208, Dec. 2022, Art. no. 118085, doi: [10.1016/j.eswa.2022.118085](https://doi.org/10.1016/j.eswa.2022.118085).
- [15] Q. Liu, H. Xian, J. Zhang, and K. Liu, "A random reversible watermarking scheme for relational data," in *Proc. Int. Conf. Secur. Privacy Commun. Syst.*, vol. 462. Cham, Switzerland: Springer, 2022, pp. 413–430, doi: [10.1007/978-3-031-25538-0](https://doi.org/10.1007/978-3-031-25538-0).
- [16] S. Wu, "Robust reversible database watermarking scheme for local distortion," in *Proc. 5th Int. Conf. Commun., Inf. Syst. Comput. Eng. (CISCE)*, Guangzhou, China, Apr. 2023, pp. 402–405, doi: [10.1109/CISCE58541.2023.10142826](https://doi.org/10.1109/CISCE58541.2023.10142826).
- [17] C. Li, X. Han, W. Qi, and Z. Guo, "An improved reversible database watermarking method based on histogram shifting," in *Proc. ACM Workshop Inf. Hiding Multimedia Secur.*, Chicago, IL, USA, Jun. 2023, pp. 103–114, doi: [10.1145/3577163.3595091](https://doi.org/10.1145/3577163.3595091).
- [18] H. Kumar, "Review on prevention of data leakage in cloud server by utilizing watermarking and double encryption techniques," in *Proc. 9th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, vol. 1. Coimbatore, India, Mar. 2023, pp. 1619–1625, doi: [10.1109/ICACCS57279.2023.10112767](https://doi.org/10.1109/ICACCS57279.2023.10112767).
- [19] M. Olliaro, M. L. Pérez Gort, and A. Cortesi, "Empirical analysis of the impact of queries on watermarked relational databases," *Expert Syst. Appl.*, vol. 204, Oct. 2022, Art. no. 117491, doi: [10.1016/j.eswa.2022.117491](https://doi.org/10.1016/j.eswa.2022.117491).
- [20] M. L. Pérez Gort, M. Olliaro, A. Cortesi, and C. Feregrino Uribe, "Semantic-driven watermarking of relational textual databases," *Expert Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114013, doi: [10.1016/j.eswa.2020.114013](https://doi.org/10.1016/j.eswa.2020.114013).
- [21] J. Lafaye, "An analysis of database watermarking security," in *Proc. 3rd Int. Symp. Inf. Assurance Secur.*, Aug. 2007, pp. 462–467, doi: [10.1109/IAS.2007.11](https://doi.org/10.1109/IAS.2007.11).
- [22] K. Jawad and A. Khan, "Genetic algorithm and difference expansion based reversible watermarking for relational databases," *J. Syst. Softw.*, vol. 86, no. 11, pp. 2742–2753, Nov. 2013, doi: [10.1016/j.jss.2013.06.023](https://doi.org/10.1016/j.jss.2013.06.023).
- [23] M. Kamran and M. Farooq, "A formal usability constraints model for watermarking of outsourced datasets," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 6, pp. 1061–1072, Jun. 2013, doi: [10.1109/TIFS.2013.2259234](https://doi.org/10.1109/TIFS.2013.2259234).
- [24] Y. Zhang, Z. Wang, Z. Wang, and C. Liu, "A robust and adaptive watermarking technique for relational database," in *Proc. CNCERT*. Beijing, China: Springer, 2021, pp. 3–26, doi: [10.1007/978-981-16-9229-1](https://doi.org/10.1007/978-981-16-9229-1).
- [25] W. Li, N. Li, J. Yan, Z. Zhang, P. Yu, and G. Long, "Secure and high-quality watermarking algorithms for relational database based on semantic," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 7440–7456, Jul. 2022, doi: [10.1109/TKDE.2022.3194191](https://doi.org/10.1109/TKDE.2022.3194191).
- [26] D. Hu, D. Zhao, and S. Zheng, "A new robust approach for reversible database watermarking with distortion control," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 6, pp. 1024–1037, Jun. 2019, doi: [10.1109/TKDE.2018.2851517](https://doi.org/10.1109/TKDE.2018.2851517).
- [27] C. Wang and Y. Li, "A copyright authentication method balancing watermark robustness and data distortion," in *Proc. 26th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, Rio de Janeiro, Brazil, May 2023, pp. 1178–1183, doi: [10.1109/CSCWD57460.2023.10152729](https://doi.org/10.1109/CSCWD57460.2023.10152729).
- [28] D. Li, C. Ma, H. Gao, and X. Jin, "LBP feature and hash function based dual watermarking algorithm for database," *Data Knowl. Eng.*, vol. 148, Nov. 2023, Art. no. 102228, doi: [10.1016/j.datak.2023.102228](https://doi.org/10.1016/j.datak.2023.102228).
- [29] M. L. Pérez Gort, M. Olliaro, and A. Cortesi, "A quantile-based watermarking approach for distortion minimization," in *Proc. Int. Symp. Found. Pract. Secur.* Paris, France: Springer, 2021, pp. 162–176, doi: [10.1007/978-3-031-08147-7](https://doi.org/10.1007/978-3-031-08147-7).
- [30] R. Sion, "Proving ownership over categorical data," in *Proc. 20th Int. Conf. Data Eng.*, 2004, pp. 584–595, doi: [10.1109/ICDE.2004.1320029](https://doi.org/10.1109/ICDE.2004.1320029).
- [31] R. Sion, M. Atallah, and S. Prabhakar, "Rights protection for categorical data," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 7, pp. 912–926, Jul. 2005, doi: [10.1109/TKDE.2005.116](https://doi.org/10.1109/TKDE.2005.116).
- [32] E. Bertino, B. Chin Ooi, Y. Yang, and R. H. Deng, "Privacy and ownership preserving of outsourced medical data," in *Proc. 21st Int. Conf. Data Eng.*, Tokyo, Japan, 2005, pp. 521–532, doi: [10.1109/ICDE.2005.111](https://doi.org/10.1109/ICDE.2005.111).
- [33] C.-C. Lin, T.-S. Nguyen, and C.-C. Chang, "LRW-CRDB: Lossless robust watermarking scheme for categorical relational databases," *Symmetry*, vol. 13, no. 11, p. 2191, Nov. 2021, doi: [10.3390/sym13112191](https://doi.org/10.3390/sym13112191).
- [34] A. Al-Haj and A. Odeh, "Robust and blind watermarking of relational database systems," *J. Comput. Sci.*, vol. 4, no. 12, pp. 1024–1029, Dec. 2008. [Online]. Available: <https://thescipub.com/pdf/jcssp.2008.1024.1029>
- [35] D. Hanyurwimfura, Y. Liu, and Z. Liu, "Text format based relational database watermarking for non-numeric data," in *Proc. Int. Conf. Comput. Design Appl.*, vol. 4. Qinhuangdao, China, Jun. 2010, pp. V4-312–V4-316, doi: [10.1109/ICCDA.2010.5541119](https://doi.org/10.1109/ICCDA.2010.5541119).
- [36] L. Zhang, W. Gao, N. Jiang, L. Zhang, and Y. Zhang, "Relational databases watermarking for textual and numerical data," in *Proc. Int. Conf. Mech. Sci., Electr. Eng. Comput. (MEC)*, Jilin, China, Aug. 2011, pp. 1633–1636, doi: [10.1109/MEC.2011.6025791](https://doi.org/10.1109/MEC.2011.6025791).
- [37] S. Melkundi and C. Chandankhede, "A robust technique for relational database watermarking and verification," in *Proc. Int. Conf. Commun., Inf. Comput. Technol. (ICCICT)*, Mumbai, India, Jan. 2015, pp. 1–7, doi: [10.1109/ICCICT.2015.7045676](https://doi.org/10.1109/ICCICT.2015.7045676).
- [38] W. Li, J. Yan, and Z. Zhang, "Relational database watermarking based on Chinese word segmentation and word embedding," in *Proc. 29th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Honolulu, HI, USA, Aug. 2020, pp. 1–6, doi: [10.1109/ICCCN49398.2020.9209600](https://doi.org/10.1109/ICCCN49398.2020.9209600).
- [39] K. A. E. Drandaly, W. Khedr, I. S. Mohamed, and A. M. Mostafa, "Digital watermarking scheme for securing textual database using histogram shifting model," *Comput., Mater. Continua*, vol. 71, no. 3, pp. 5253–5270, 2022, doi: [10.32604/cmc.2022.023684](https://doi.org/10.32604/cmc.2022.023684).
- [40] M. Kamran, S. Suhail, and M. Farooq, "A robust, distortion minimizing technique for watermarking relational databases using once-for-all usability constraints," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 12, pp. 2694–2707, Dec. 2013, doi: [10.1109/TKDE.2012.227](https://doi.org/10.1109/TKDE.2012.227).
- [41] M. B. Imamoglu, M. Ulutas, and G. Ulutas, "A watermarking technique for relational databases based on partitioning," in *Proc. 23rd Signal Process. Commun. Appl. Conf. (SIU)*, Malatya, Turkey, May 2015, pp. 2094–2097, doi: [10.1109/SIU.2015.7130283](https://doi.org/10.1109/SIU.2015.7130283).
- [42] S. Rani, D. K. Koshley, and R. Halder, "Partitioning-insensitive watermarking approach for distributed relational databases," in *Transactions on Large-Scale Data- and Knowledge-Centered Systems XXXVI*. Berlin, Germany: Springer, 2017, pp. 172–192, doi: [10.1007/978-3-662-56266-6_8](https://doi.org/10.1007/978-3-662-56266-6_8).
- [43] J. Franco-Contreras, G. Coatrieux, N. Cuppens-Boulahia, F. Cuppens, and C. Roux, "Ontology-guided distortion control for robust-lossless database watermarking: Application to inpatient hospital stay records," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Chicago, IL, USA, Aug. 2014, pp. 4491–4494, doi: [10.1109/EMBC.2014.6944621](https://doi.org/10.1109/EMBC.2014.6944621).

- [44] J. Franco-Contreras and G. Coatrieux, "Robust watermarking of relational databases with ontology-guided distortion control," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 1939–1952, Sep. 2015, doi: [10.1109/TIFS.2015.2439962](https://doi.org/10.1109/TIFS.2015.2439962).
- [45] J. J. McAuley and J. Leskovec, "From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews," in *Proc. 22nd Int. Conf. World Wide Web*, May 2013, pp. 897–908, doi: [10.1145/2488388.2488466](https://doi.org/10.1145/2488388.2488466).
- [46] Colorado-State-University. (1999). *Forest CoverType, The UCI KDD Archive*. Univ. California. [Online]. Available: <http://kdd.ics.uci.edu/databases/covertype/covertype.html>
- [47] D. Brunet, E. R. Vrscay, and Z. Wang, "On the mathematical properties of the structural similarity index," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1488–1499, Apr. 2012, doi: [10.1109/TIP.2011.2173206](https://doi.org/10.1109/TIP.2011.2173206).
- [48] J. Kiernan, R. Agrawal, and P. J. Haas, "Watermarking relational data: Framework, algorithms and analysis," *VLDB J. Int. J. Very Large Data Bases*, vol. 12, no. 2, pp. 157–169, Aug. 2003, doi: [10.1007/s00778-003-0097-x](https://doi.org/10.1007/s00778-003-0097-x).



Her research interests include string static analysis through abstract interpretation theory and relational data watermarking.

MARTINA OLLIARO received the Ph.D. degree in computer science from Università Ca' Foscari di Venezia Campus Scientifico, Venice, Italy, and Masaryk University, Brno, Czech Republic, in 2021. She worked as a Research Fellow with Università Ca' Foscari di Venezia Campus Scientifico. She is currently an Information Technology Consultant with Blue Reply and a Subject Expert with the Department of Environmental Sciences, Informatics, and Statistics, Ca' Foscari University.



His research interests include relational databases theory, information security and privacy, and data usability and authenticity.

MAIKEL LÁZARO PÉREZ GORT received the Ph.D. degree in computer sciences from the National Institute of Astrophysics, Optics, and Electronics (INAOE), Puebla, Mexico, in 2020. During his career, he has worked with different institutions regarding database management, usability, and security issues. He is currently a Researcher with Università Ca' Foscari di Venezia Campus Scientifico, and also a Non-Tenured Track Assistant Professor with Ca' Foscari University.



He coordinates the MAE Italy-India Project 2017–2020 "Formal Specification for Secured Software System."

...