



Article

# Blind Digital Watermarking Algorithm against Projection Transformation for Vector Geographic Data

Qifei Zhou <sup>1,2,3</sup> , Na Ren <sup>1,2,3,\*</sup>, Changqing Zhu <sup>1,2,3</sup> and A-Xing Zhu <sup>4</sup> 

<sup>1</sup> Key Laboratory of Virtual Geographic Environment (Nanjing Normal University), Ministry of Education, Nanjing 210023, China; 181301014@njnu.edu.cn (Q.Z.); 09322@njnu.edu.cn (C.Z.)

<sup>2</sup> State Key Laboratory Cultivation Base of Geographical Environment Evolution, Nanjing 210023, Jiangsu, China

<sup>3</sup> Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing 210023, China

<sup>4</sup> Department of Geography, University of Wisconsin-Madison, Madison, WI 53706, USA; azhu@wisc.edu

\* Correspondence: 09359@njnu.edu.cn

Received: 14 October 2020; Accepted: 18 November 2020; Published: 19 November 2020



**Abstract:** Projection transformation is an important part of geographic analysis in geographic information systems, which are particularly common for vector geographic data. However, achieving resistance to projection transformation attacks on watermarking for vector geographic data is still a challenging task. We proposed a digital watermarking against projection transformation based on feature invariants for vector geographic data in this paper. Firstly, the features of projection transformation are analyzed, and the number of vertices, the storage order, and the storage direction of two adjacent objects are designed and used as the feature invariant to projection transformation. Then, the watermark index is calculated by the number of vertices of two adjacent objects, and the embedding rule is determined by the storage direction of two adjacent objects. Finally, the proposed scheme performs blind detection through the storage direction of adjacent features. Experimental results demonstrate that the method can effectively resist arbitrary projection transformation, which indicates the superior performance of the proposed method in comparison to the previous methods.

**Keywords:** digital watermarking; projection transformation; feature invariants; storage order; vector geographic data

## 1. Introduction

Vector geographic data are amongst the most important production materials in geographic information systems (GIS) [1,2]. They are widely used in map applications and services with the advantages of high accuracy and low redundancy. At the same time, many security issues of vector geographic data have emerged, such as piracy, leakage, and malicious tampering [3–5]. The security of vector geographic data is directly related to the sound development of the geographic information industry. Digital watermarking technology is used to embed additional information into digital content and has played an irreplaceable role in the security protection of copyright confirmation for vector geographic data [6–8]. Robustness is a crucial element for measuring the success of a digital watermarking algorithm against intentional or unintentional attacks [9,10].

Many research efforts have been devoted to improving the robustness of watermarking algorithms [11–14]. These efforts have led watermarking algorithms less vulnerable to compression, filtering, point addition and deletion [15–20]. However, as a common processing method of vector geographic data, projection transformation will cause the coordinate values and geometric shapes of vector geographic data to change [21]. Thus, the development of watermarking techniques which can

resist projection transformation has become a crucial research issue in watermarking technology for vector geographic data.

The existing watermarking methods against projection transformation attacks can be categorized into two categories. The first category of studies is non-blind watermarking schemes, in which the original data are required to perform the extraction of the watermark [22–26]. For example, Tong et al. [22] proposed a method based on the same feature points fitting which first extracts feature points and attributes information from vector geographic data, and saves the extracted information separately. When watermark detection is needed, the same algorithm will first extract the feature points and attribute information from the vector geographic data, and then compare it with the stored data to examine if they are the same. This method effectively solved the problem of projection transformation attacks. However, one needs to save the original vector geographic data or additional information which must be saved and stored separately. Thus, this method is not conducive to practical application.

The second category of studies is the blind watermarking scheme, in which the original data or related information are not required to perform the extraction of the watermark. Watermark embedding and extraction are usually achieved by finding the features which are invariant (feature invariants hereafter) to the projection transformation [27–31]. For example, Yang and Hou [28] presented a method based on a kind of polyline transformation, which converts the polyline into a representation of distance and angle. By embedding watermark information in the converted angle coefficient, this method can effectively resist equal-angle projection transformation attacks. Li and Zhu [29] proposed a method based on spatial autocorrelation indices in which the Moran's I coefficient (MC) is regarded as the feature invariant. The MC is calculated by the graphical complexity index and the spatial topological information of geographic objects. Thus, this method has a better ability to resist projection transformation attacks under the condition that the projection transformation does not distort the data significantly. However, these methods cannot resist attacks from projection transformations with significant distortion, which are quite common in geographic analysis.

It can be seen from the above analysis that the non-blind watermarking algorithm requires the participation of original data or other related information to extract the watermark, which leads to limitations in practical application, while the existing blind watermarking algorithms realize blind detection by finding the feature invariants which resist projection transformation attacks. However, a major drawback of blind watermarking algorithms is that the feature invariants are conditional and only stable over projection transformations which do not distort the data substantially. Therefore, how to find robust feature invariants which can resist all projection transformation attacks has become a major challenge in watermarking for vector geographic data.

Aiming to resolve the above problem, this paper presents a blind watermarking algorithm against attacks from projection transformations which would distort data substantially. In vector geographic data, objects are stored in disorder, and rearranging the objects by storage order does not affect the visualization and use value of the data. Therefore, the storage order of the object is meaningless for vector geographic data. We believe that the storage order and the number of vertices of vector geographic data will not change before and after any projection transformation. Then, the storage direction is introduced based on the storage order and the number of vertices, which also remain unchanged in projection transformation. This makes the storage order of the object change from meaningless to meaningful, and can be used to store watermark information. Thus, a watermark synchronization mechanism based on these invariant features can be built and the watermark can be embedded with them. This would avoid the need to store the additional data that are required for watermarking extraction. In this way, the proposed method achieves blind watermarking which can resist arbitrary projection transformation attacks and have good robustness to conventional geometric attacks with good practicability.

The remainder of this paper is organized as follows. In Section 2, we present the feature invariants of projection transformation, which forms the basis of the method. Section 3 describes the detailed

procedure of the proposed method. Then, experiments and results are provided in Section 4, and Section 5 gives the corresponding discussions. Finally, Section 6 draws the conclusions.

## 2. Preliminaries

A vector geographic data set is represented by points, polylines and polygons. Clearly, the representations of polylines and polygons are much more complicated than the representation of points. The complexity in representing polylines and polygons provides the opportunity for extracting feature invariants which are resistant to projection transformation. Thus, we use polylines and polygons as watermarking targets in this paper.

Before introducing the scheme proposed in this paper, there are three issues needing to be addressed: (1) Which features are invariant to arbitrary projection transformation and how to effectively express them; (2) How to use these feature invariants to synchronize watermark information; (3) How to use these feature invariants to embed watermark information. The solution to these three problems is the key to realizing the algorithm in this paper.

### 2.1. Feature Invariants to Projection Transformation

Projection transformation is a process of projecting spatial data from one coordinate system to another, whose basis is map projections. All map projections make distortions on data. According to the properties that are preserved, the map projection can be divided into four types: the equal area projection, the conformal projection, the equidistant projection and the compromise projection. The area is preserved in the equal area projection. The conformal projection can keep shape locally. The distances from one or two specific points to others are preserved in the equidistant projection, and the conformal projection finds a balance in distortions. Thus, projection transformations may alter the relative positions and geometric shapes of vector geographic data. Besides, the projection transformation, including rotation, scaling and translation (RST), is a comprehensive attack. If the selected feature is invariant to the projection transformation, it is also invariant to the RST.

We find that no matter which projection transformation method is used, the number of vertices for each polyline or each polygon remains the same. In other words, the number of vertices for each polyline or each polygon of the vector geographic data does not change after the projection transformation. Thus, the number of vertices can serve as a feature invariant (referred to as number of vertices invariant hereafter).

The storage order refers to the physical order in which the objects are stored in the vector geographic data file, such as the Environmental Systems Research Institute's (ESRI) shapefile format. It is an implicit feature, and most data processing on vector geographic data will ignore the feature and not change it. Therefore, it is very stable to a certain extent. Furthermore, we find that no matter what projection transformation method is used, the data storage order of objects always remains the same. In other words, projection transformation does not change the storage order of any object. Thus, the storage order of objects can serve as a feature invariant (referred to as storage invariant hereafter).

Furthermore, we employ the concept of the storage direction of two adjacent objects. Taking polyline feature as an example, we define that if the number of vertices of the previous polyline in storage is larger than the number of vertices of the subsequent one, the storage direction of two adjacent objects is 1, otherwise, the storage direction is 0. This can be calculated by the Formula (1):

$$Dir(l_i, l_{i+1}) = \begin{cases} 0, & \text{if } N_{l_i} \leq N_{l_{i+1}} \\ 1, & \text{if } N_{l_i} > N_{l_{i+1}} \end{cases} \quad (1)$$

where  $N_{l_i}$  and  $N_{l_{i+1}}$  mean the number of vertices of two adjacent polylines ( $l_i$  and  $l_{i+1}$ ), respectively.

It can be observed that the storage direction of two adjacent objects is related to the relationship between the number of vertices of two adjacent objects. Since the number of vertices of two adjacent

objects and the storage order of two adjacent objects will not be changed by projection transformation, the storage direction of two adjacent objects can serve as a feature invariant.

The number of vertices, the storage order and the storage direction of two adjacent objects reveal that they are global invariants that will not be affected by projection transformation, and thus they are robust.

## 2.2. Watermark Synchronization Mechanism Based on the Number of Vertices Invariant

The watermark used in this paper is a binary sequence containing only 0 and 1, which is used to represent the copyright information to be embedded. In a watermark with the length of  $L$ , the watermark index indicates where a watermark bit is, which ranges from 0 to  $L - 1$ . Figure 1 shows a watermark “1101111010100010” with the length of 16. For example, the watermark bit with a watermark index of 2 is 0, and the watermark bit with a watermark index of 10 is 1.

Watermark index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Watermark bit	1	1	0	1	1	1	1	0	1	0	1	0	0	0	1	0

Figure 1. Demonstration of a watermark.

To ensure watermark synchronization, an operation on the number of vertices of two adjacent objects is needed to compute the location for watermark embedding and extraction. There are a variety of methods to compute the watermark embedding position using the number of vertices.

In this paper, to increase the randomness of the watermark embedding position, we use a multiplication operation to expand the numerical range of the number of vertices and then hash the multiplied feature value to obtain the watermark index of the set of two adjacent objects. It can be expressed as the following equation:

$$Index_{(i+1)/2} = \text{Hash}(N_{l_i} * N_{l_{i+1}}) \quad (2)$$

where  $Index_{(i+1)/2} \in [0, L_w - 1]$ , and  $L_w$  means the length of the watermark information  $w$ .

As Equation (2) shows, Hash is a function that depends only on the number of vertices of two adjacent objects. The important advantage of using Hash is that the number of vertices still exists even when the data are attacked by projection transformation. Therefore, the watermark will be located at the same position as it was before the attack of projection transformation, which leads to correct watermark extraction.

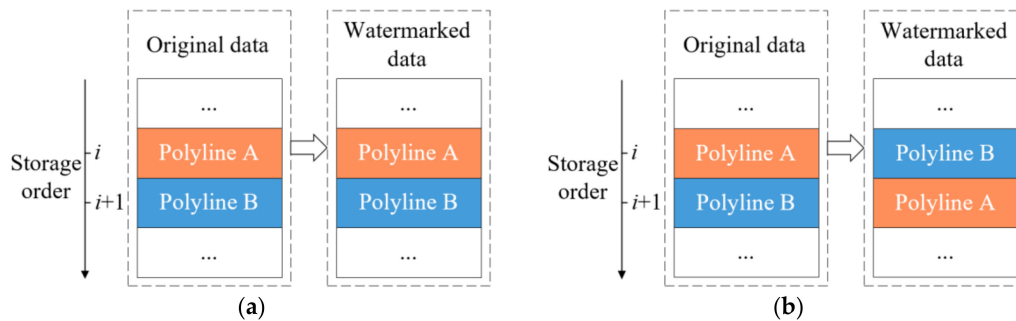
Moreover, the object and the embedding position of watermarking will become a many-to-one mapping relationship. That is to say, multiple polylines or polygons can be embedded in one watermark index at the same time. In the case of object addition and deletion, this feature can not only ensure the synchronization relationship of watermark information, but also improve the robustness of the algorithm effectively.

## 2.3. Watermark Embedding Rule Based on the Storage Direction

Given by the definition of the storage direction of two adjacent objects in Section 2.1, the quantified values of the storage direction of two adjacent objects are 0 and 1. Generally, watermark information consists of a binary sequence of 0 and 1. Thus, the storage direction of two adjacent objects is considered to represent the watermark.

The watermark embedding rule based on the storage direction of two adjacent objects is described as that the quantified value of the storage direction of two adjacent objects needs to be consistent with the embedded watermark information. That is, when the storage direction of two adjacent objects is not the same as the embedded watermark information, the storage order of two adjacent objects needs to be reversed. Since the objects in vector geographic data are originally stored out of order, rearranging the objects by storage order does not affect the accuracy and visualization of the data. Then, the storage direction of two adjacent objects 0 means that the embedded watermark information is 0,

and the storage direction of two adjacent objects 1 means that the embedded watermark information is 1. Figure 2 demonstrates the process of watermark embedding, where polyline A and polyline B are two adjacent objects stored in polyline data. Polyline A is at  $i$ -th storage position and has 30 vertices. It is followed by polyline B with 60 vertices. So, the storage direction of them is 0, which can be calculated by the Formula (1). Every two adjacent objects can be embedded with a watermark bit. For polyline A and polyline B, when the watermark bit is 0, which is equal to the storage direction of them, do nothing to the two polylines, as shown in Figure 2a. However, when the watermark bit is 1, reverse them by storage order to implement watermark embedding, as shown in Figure 2b.



**Figure 2.** Demonstration of watermark embedding: (a) When the watermark bit is 0; (b) When the watermark bit is 1.

The storage direction is invariant to projection transformation due to the fact that both the number of vertices invariant and the storage invariant of the vector geographic data keep unchanged. Therefore, the watermark can be extracted successfully after the projection transformation. Additionally, a good tradeoff between good robustness and data fidelity can be achieved because there is no change in coordinates caused by embedding watermark.

### 3. Methods

#### 3.1. Basic Idea

The aim of this paper is to solve the challenge that the watermark must be extracted correctly after projection transformation. To solve the three problems in Section 2, the following aspects are mentioned for the design of the proposed watermarking method: (1) The number of vertices, the storage order and the storage direction of two adjacent objects can serve as the feature invariants to projection transformation. Additionally, the storage direction of two adjacent objects is quantized to be 0 or 1 by comparing the size of the number of vertices between two adjacent objects. (2) In order to solve the watermark synchronization, the number of vertices of two adjacent objects is used to obtain the embedding watermark location. This implies that the extracted watermark is located in the same position as it was embedded. (3) The storage direction of two adjacent objects is used to embed the watermark and represent the binary watermark 0 and 1. The proposed watermark embedding rule effectively utilizes the invariance of the storage order. It not only does not change the coordinate values of the data, but it also effectively resists projection transformation attacks.

In this section, we explain the proposed watermarking scheme based on the feature invariants in detail. There are three procedures in the proposed scheme: watermark information generation, watermark embedding and watermark extraction. The various procedures are described in detail in the following section.

#### 3.2. Watermark Information Generation

In order to increase the secrecy of the watermarking, the watermark information is preprocessed by scrambling transformation before construction.

Suppose the key is  $k$ ,  $w_0$  is the original watermark and  $S$  is a function to generate the new watermark information  $w$ . Then the function defines:

$$w = S(w_0, k) \quad (3)$$

where the Arnold transform [32] is selected as  $S$  function to permute the original watermark. This is a commonly used scrambling method for watermarking [33–35]. Arnold transform is a cyclic transformation with a period  $T$  for the two-dimensional  $n \times n$  signals, and  $k$  in Equation (3) ranges from 1 to  $T - 1$ . The calculation of the Arnold transform can be mathematically expressed as:

$$\begin{bmatrix} r' \\ c' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} r \\ c \end{bmatrix} (\text{mod } n) \quad (4)$$

where mod is the modulo operation to get the remainder after division.  $r$  and  $c$  are the row number and column number of the element in the signal, respectively.  $r'$  and  $c'$  are scrambled ones. Since the watermark in this paper is a one-dimensional signal, the reshape operation is introduced. In Figure 3, the scrambling process of a watermark is demonstrated. Here, we set  $k$  to 2, so for a watermark “1101111010100010” with the length of 16, the encrypted result is “1110010001101101”.

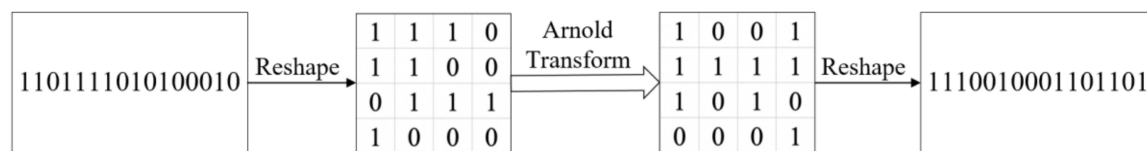


Figure 3. Demonstration of watermark scrambling.

### 3.3. Watermark Embedding

We exploit the number of vertices invariant of two adjacent objects to keep the watermark synchronization of the proposed watermarking scheme. The feature invariant will prevent the destroying of the watermark information when the watermarked vector geographic data are attacked by arbitrary projection transformations. The proposed embedding process is illustrated in Figure 4.

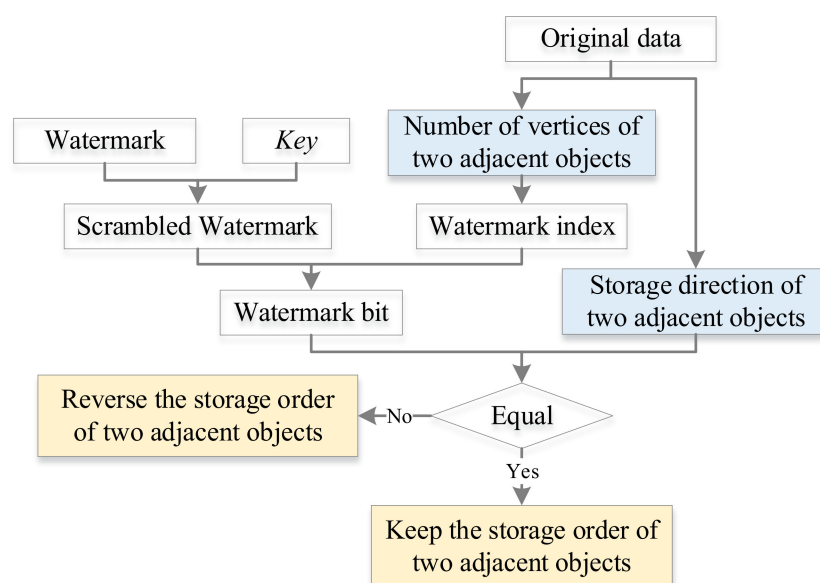


Figure 4. Watermark embedding procedure.



The following are the specific steps of the embedding procedure.

Step 1: Read the vector geographic data  $Vector$ , expressed by  $Vector = \{Vector_1, Vector_2, \dots, Vector_m\}$ , with the size of  $m$ .

Step 2: Read the binary watermark information  $w = \{w_1, w_2, \dots, w_{L_w}\}$ , generated by the Equation (3), with the length of  $L_w$ . In addition, save the watermark  $w$  and the key  $k$ .

Step 3: Calculate the watermark index by the number of vertices of two adjacent objects, and the position of embedding watermark  $Index_{(i+1)/2}$  is obtained by Equation (2). Then obtain the embedded watermark bit  $w_{Index_{(i+1)/2}}$ .

Step 4: Calculate the storage direction  $Dir(Vector_i, Vector_{i+1})$  of two adjacent objects using Equation (1).

Step 5: Embed the watermark using the rule as follows: The embedding form is judged by whether the embedded watermark bit  $w_{Index_{(i+1)/2}}$  is consistent with the storage direction  $Dir(Vector_i, Vector_{i+1})$  of two adjacent objects. If they are the same, the storage order of two adjacent objects remains unchanged; otherwise, reverse the storage order of two adjacent objects.

Step 6: Repeat these operations for all objects. Then, the watermark can be embedded in every object to obtain the final watermarked vector geographic data. Save the watermarked data  $Vector'$ .

As described above, the watermark is embedded by changing the storage order of two adjacent objects, and the coordinates of the original data are not modified. Therefore, it does not affect the fidelity of the data.

### 3.4. Watermark Extraction

The overall flowchart of the watermark extraction processes is shown in Figure 5. The process of watermark extraction is the reverse of the embedding process. In this paper, the watermarking extraction algorithm is blind because it does not require the original data at the extraction stage.

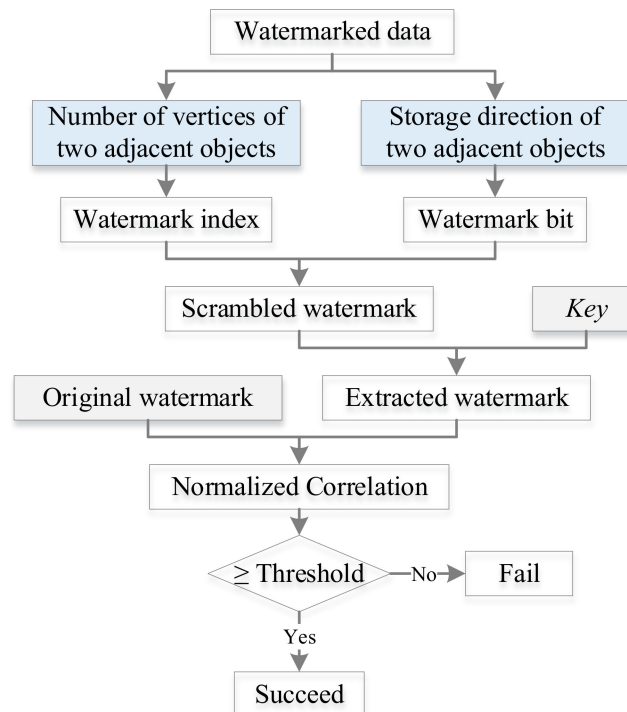


Figure 5. Watermark extraction procedure.

Actually, the embedding watermark position  $Index'_{(i+1)/2}$  and the storage direction  $Dir'(Vector'_i, Vector'_{i+1})$  of two adjacent objects are calculated in the same way as the watermark embedding.

The current watermark bit can be obtained by judging the storage direction of two adjacent objects. If the storage direction of two adjacent objects is 1, the watermark bit is 1. If the storage direction of two adjacent objects is 0, the watermark bit is 0. The watermark bit is estimated as follows:

$$w'_{Index'_{(i+1)/2}} = \begin{cases} 0, & \text{if } Dir'(Vector'_i, Vector'_{i+1}) = 0 \\ 1, & \text{if } Dir'(Vector'_i, Vector'_{i+1}) = 1 \end{cases} \quad (5)$$

It should be noticed that each watermark bit may be extracted multiple times, and therefore the majority voting mechanism is employed to determine the watermark bit. The majority voting mechanism determines the possible value which has the maximum frequency. That is, if more than half of these values are 0, then  $w'_{Index'_{(i+1)/2}}$  is 0. Otherwise,  $w'_{Index'_{(i+1)/2}}$  is 1. Then, the Arnold inverse transform is implemented using the same key  $k$  to obtain the decrypted watermark  $w''$ .

Correlation between the original watermark and the extracted watermark is computed to check how much watermark is extracted. As we know, Normalized Correlation (NC) refers to the normalized correlation, which is one of the metrics used to find the quality of the extracted watermark concerning the original watermark. It is found by using the following formula:

$$NC = \frac{\sum_i w_i * w''_i}{\sqrt{\sum_i w_i^2} \sqrt{\sum_i w''_i^2}} \quad (6)$$

where  $w_i$  is the original watermark value and  $w''_i$  is the extracted watermark value, respectively. NC ranges from 0 to 1, where high NC denotes strong similarity. If the NC is higher than a preset threshold, it indicates that the data contain watermark information.

## 4. Experiments and Results

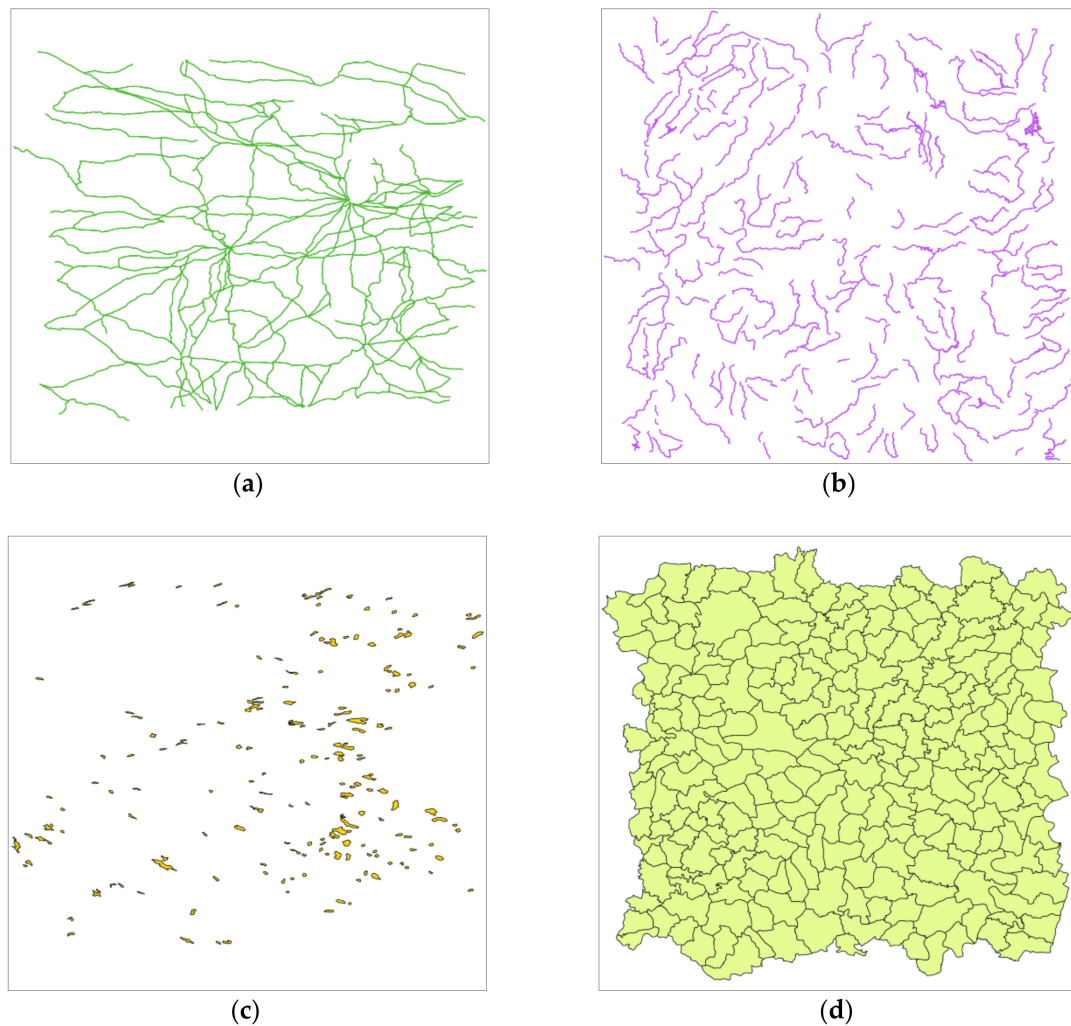
### 4.1. Experimental Data and Parameters

The proposed scheme is applied to polylines and polygons in vector geographic data. Four vector maps at the same scale of 1:4,000,000 are used. The data format of the experimental data is the ESRI shapefile format, which is a popular geospatial vector data format for GIS software. They contain different types of data from different regions, namely, railroads, rivers, lakes, and administrative divisions, which are denoted by Railroad, River, Lake, and Division in turn. They have no map projection and their geographic coordinate systems are the same, namely, the World Geodetic System (WGS) 1984, whose angular unit is the degree, prime meridian is Greenwich and datum is WGS 1984. Figure 6 shows the host vector maps in the plate carrée projection, one of the most common and simple map projections. Table 1 shows the detailed information of four vector maps.

**Table 1.** Detailed information of the vector maps.

Name	Feature Type	Objects	Vertices
Railroad	Polyline	488	25,677
River	Polyline	354	33,767
Lake	Polygon	174	4086
Division	Polygon	236	34,321





**Figure 6.** The host vector maps: (a) Railroad; (b) River; (c) Lake; (d) Division.

In the step of watermark generation, the meaningless binary watermark [11]  $w_0$  presents the copyright of “OWNERSHIP IS GEOMARK”. The length of the watermark is 130 bits. Before embedding, the binary watermark information  $w$  is obtained by scrambling  $w_0$  with Arnold transform, and the key  $k$  used to scramble the original watermark is 6.

In the step of evaluation,  $NC$  gives us an intuition of similarity between the extracted watermark information and the original watermark information. The threshold of  $NC$  is an empirical value. A good threshold should present the tradeoff between the false positive and false negative rates [9]. In this paper, the threshold of  $NC$  is set as 0.75. In other words, the watermark will be extracted successfully when the value of  $NC$  calculated is higher than the threshold. When the value of  $NC$  is 1, it is indicated that the extracted watermark is identical to the original one.

#### 4.2. Watermark Extraction Results

After embedding watermark information in the experimental data, watermark extraction can check if the watermark is embedded successfully. When the watermarked data have not suffered any attack, the watermark should be detected completely, which is the basic requirement of all the watermarking methods.

Table 2 shows the extracted watermark information from the watermarked data with the proposed method. It can be observed that all the values of  $NC$  are 1.00, and therefore the copyright “OWNERSHIP

IS GEOMARK" is embedded and detected in the experimental data successfully. This indicates that the proposed method meets the basic requirement of watermarking.

**Table 2.** The results of watermark extraction.

Extraction	Railroad	River	Lake	Division
NC	1.00	1.00	1.00	1.00

#### 4.3. Invisibility

The previous section proved that the watermark is indeed embedded in the data, and then we need to verify the invisibility of the watermark. The watermark invisibility can be seen as the perceptual similarity between the watermarked vector geographic data and the host data, which means that after embedding the watermark, no visible distortion should appear in the vector geographic data [16]. It is one of the most critical factors, and a good watermarking method should satisfy certain requirements of invisibility [36].

In this subsection, we evaluate how much the watermark embedding visually affects the watermarked data when compared to the original ones. The change rates of coordinate values of four test data are calculated by Equation (7) in the plate carrée projection and the experimental results are listed in Table 3:

$$rc = \frac{\text{num}\left(\sqrt{(x'_i - x_i)^2 + (y'_i - y_i)^2} \neq 0\right)}{N_{\text{vector}}} \quad (7)$$

where  $N_{\text{vector}}$  means the total number of vertices of the vector geographic data, and  $\text{num}()$  denotes the number of changed coordinate values.  $(x'_i, y'_i)$  means the watermarked coordinate value, and  $(x_i, y_i)$  means the original coordinate value.  $rc$  ranges from 0 to 1, and the lower value of  $rc$  indicates less data changed. When the vector geographic data are not changed, the value of  $rc$  becomes 0.

**Table 3.** The change rates of coordinate values.

Distortion	Railroad	River	Lake	Division
Vertices	25,677	33,767	4086	34,321
Changed vertices	0	0	0	0
$rc$	0	0	0	0

We can see from Table 3 that there is no coordinate changed after embedding a watermark. The number of changed vertices of the four experimental data is 0, so the corresponding change rates of coordinate values, i.e., the  $rc$  values, are 0. This means that the data accuracy of the original data and the watermarked data is exactly the same. Therefore, the experimental results show that the proposed method has good watermark invisibility. It is also important to note that the proposed scheme is a lossless watermarking that does not affect the data accuracy of the watermarked data at all.

#### 4.4. The Robustness of Projection Transformation

The robustness of projection transformation refers to the ability of the watermarking algorithm to extract the watermark information correctly even after the attacks of projection transformations on watermarked data. A good watermarking algorithm resisting projection transformation should resist arbitrary projection transformation. As the evaluation index of the experimental results, the higher the  $NC$ , the stronger the robustness. If the  $NC$  between the extracted watermark and the original one is always one after different map projections, it indicates that the algorithm can resist arbitrary projection transformation.

To evaluate the robustness of projection transformation, the projection transformation is implemented by four types of map projections, namely the equal area projection, the conformal projection, the equidistant projection and the compromise projection. The first three can preserve the area, shape and distance of data to some extent, respectively, but not all factors. The last type, the compromise projection, ensures that all the factors remain unchanged in a small view, not in a large view. As shown in Table 4, twelve map projections are chosen, where each type includes three subtypes, which will show as many aspects of distortions as possible. It is also more conducive to verifying the feasibility of the proposed method.

**Table 4.** Map projections.

Projection Type	Projection Name	Short Name
Equal area projection	Equal-Area Cylindrical Projection	Eqacylin
	Gall Orthographic Projection	Gortho
	Lambert Azimuthal Equal-Area Projection	Eqaaazim
Conformal projection	Mercator Projection	Mercator
	Lambert Conformal Conic Projection	Lambert
	Stereographic Projection	Stereo
Equidistant projection	Equidistant Azimuthal Projection	Eqdazim
	Equidistant Cylindrical Projection	Eqdcylin
	Equidistant Conic Projection	Eqdconic
Compromise projection	Robinson Projection	Robinson
	Winkel I Projection	Winkel
	Aitoff Projection	Aitoff

In addition, we compare the proposed method with two existing methods, named Tong [22] and Li [29], which are the representatives of non-blind and blind watermarking, respectively. When testing Tong's method, the 20% number of feature points of the original data is saved, and the watermark is embedded in the third position after the decimal point of the original data. Meanwhile, to facilitate comparison, in Li's algorithm, we calculate *NC* values on the basis of watermark similarity. Tables 5–8 give comparison results of different watermarking schemes in terms of *NC* under four types of map projections, respectively.  $\sqrt{\phantom{x}}$  means the *NC* value is higher than the threshold, and  $\times$  means the *NC* value is lower than the threshold. The corresponding graphs of the four experimental data are shown in Figures 7–10, respectively, in which the dotted line represents the threshold of *NC*.

**Table 5.** The results after equal area projection transformation.

Projection Number	Projection Name	Method	NC			
			Railroad	River	Lake	Division
1	Eqacylin	Proposed	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Tong's	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Li's	0.76 $\sqrt{\phantom{x}}$	0.77 $\sqrt{\phantom{x}}$	0.00 $\times$	0.88 $\sqrt{\phantom{x}}$
2	Gortho	Proposed	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Tong's	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Li's	0.72 $\times$	0.73 $\times$	0.00 $\times$	0.47 $\times$
3	Eqaaazim	Proposed	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Tong's	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$	1.00 $\sqrt{\phantom{x}}$
		Li's	0.76 $\sqrt{\phantom{x}}$	0.77 $\sqrt{\phantom{x}}$	0.00 $\times$	0.88 $\sqrt{\phantom{x}}$

**Table 6.** The results after conformal projection transformation.

Projection Number	Projection Name	Method	NC			
			Railroad	River	Lake	Division
4	Mercator	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.77√	0.00×	0.88√
5	Lambert	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.77√	0.77√	0.00×	0.88√
6	Stereo	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.77√	0.00×	0.88√

**Table 7.** The results after equidistant projection transformation.

Projection Number	Projection Name	Method	NC			
			Railroad	River	Lake	Division
7	Eqdazim	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.77√	0.00×	0.88√
8	Eqdcylin	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.75√	0.00×	0.85√
9	Eqdconic	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.78√	0.00×	0.92√

**Table 8.** The results after compromise projection transformation.

Projection Number	Projection Name	Method	NC			
			Railroad	River	Lake	Division
10	Robinson	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.76√	0.00×	0.85√
11	Winkel	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.75√	0.76√	0.00×	0.82√
12	Aitoff	Proposed	1.00√	1.00√	1.00√	1.00√
		Tong's	1.00√	1.00√	1.00√	1.00√
		Li's	0.76√	0.77√	0.00×	0.88√

From Tables 5–8, overall, it is clear to see that both the proposed method and Tong's method have the NC value of 1 under twelve projection attacks and four experimental data. However, the NC values of Li's scheme vary with different data and map projections—31% of them are below the threshold, and the maximum is only 0.92. In detail, the NC values of Li's method vary significantly with different experimental data; in particular, all of its NC values for Lake are 0. This is because the distribution of polygons in Lake is discrete, and so the MC based on neighboring relationships does not exist. In contrast, it shows that the proposed method can resist projection transformation regardless of the data type of the host data. At the same time, clearly, Figures 7–10 also show that the proposed method maintains the NC value of 1 under the four types (12 subtypes) of projections and four experimental

data. Thus, the proposed method has stronger robustness than Li's, and achieves the same effect as Tong's on resisting projection transformation attacks without the help of the original data.

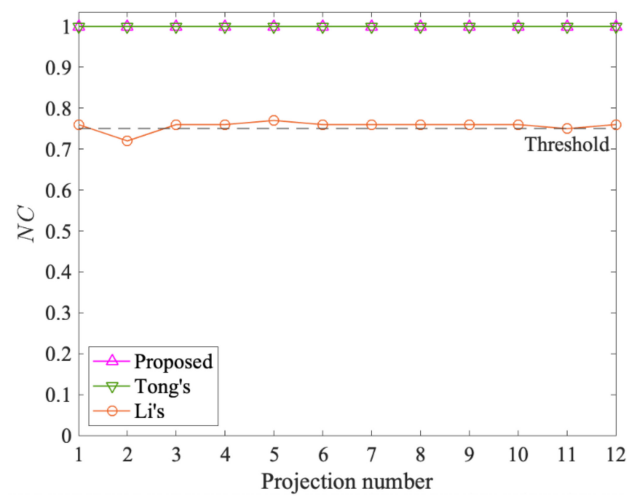


Figure 7. The results of Railroad.

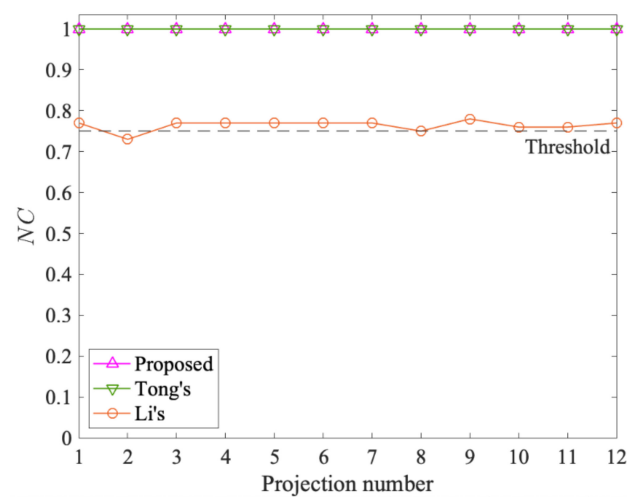


Figure 8. The results of River.

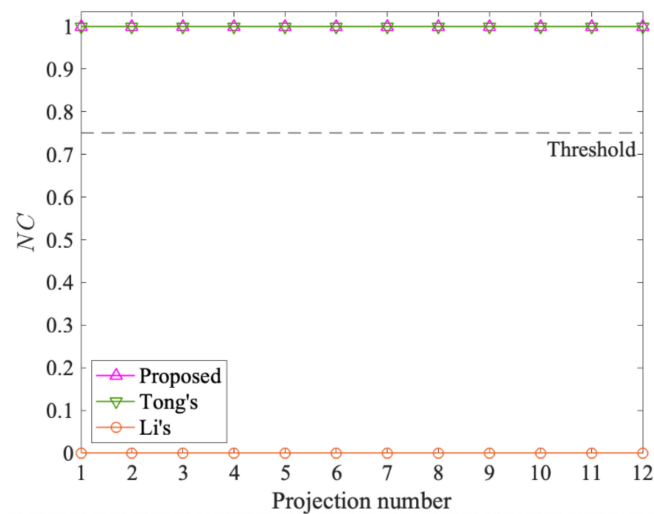


Figure 9. The results of Lake.

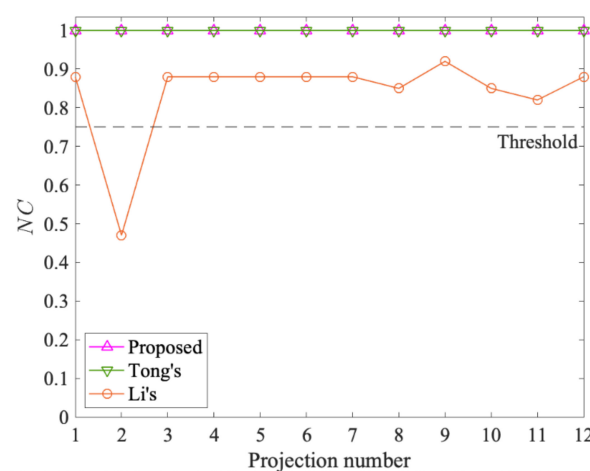


Figure 10. The results of Division.

Moreover, each of these four types of projections can only preserve some features of the data, such as area, angle and distance, at the expense of others, which means they can show the most comprehensive distortion. The experimental results show that the method in this paper can completely resist the twelve subtypes of projection transformation derived from the four types given. This is because after the projection transformation, although the shapes and coordinate values of the vector geographic data are deformed to different degrees, the number of vertices, the storage order, and the storage direction keep unchanged. Further, the watermark synchronization mechanism and the watermark embedding rule based on these feature invariants are maintained. Therefore, the proposed method can resist attacks from arbitrary projection transformation.

#### 4.5. The Robustness of RST

RST belongs to geometric transformation, which refers to rotation, scaling and translation. Generally, the RST attack is for the projected data in the projected coordinate system. Therefore, watermarked data should have the ability to resist RST attacks after undergoing projection transformation attacks.

To verify the robustness of RST, we choose Lambert projection to project the watermarked data and then execute the RST attack with different parameters in turn. The rotation attack uses the data center as a fixed point and then rotates 0 to 360 degrees with an interval of 30 degrees. The scaling attack acts to scale the data from 0.1 to 10 times. The translation attack translates the data with a certain ratio of the maximum width of the data in the X-axis and Y-axis at the same time. Figures 11–13 give the experimental results.

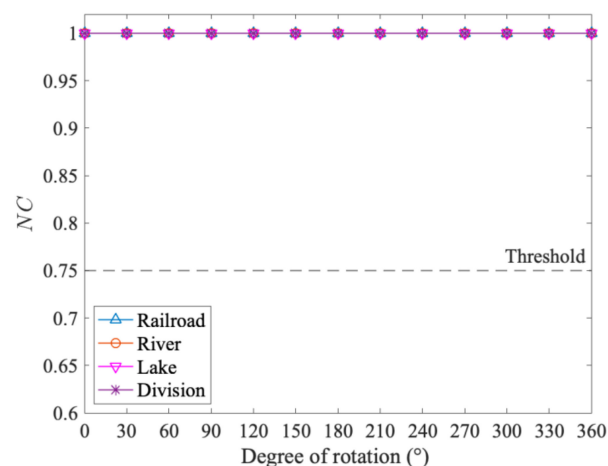


Figure 11. The results of rotation attack.



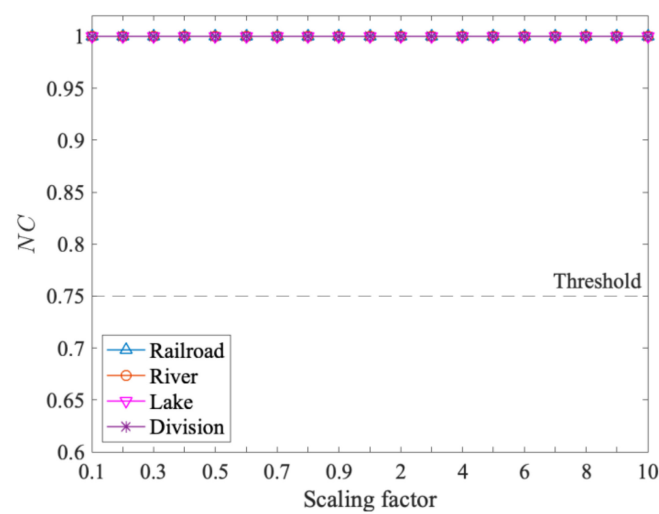


Figure 12. The results of scaling attack.

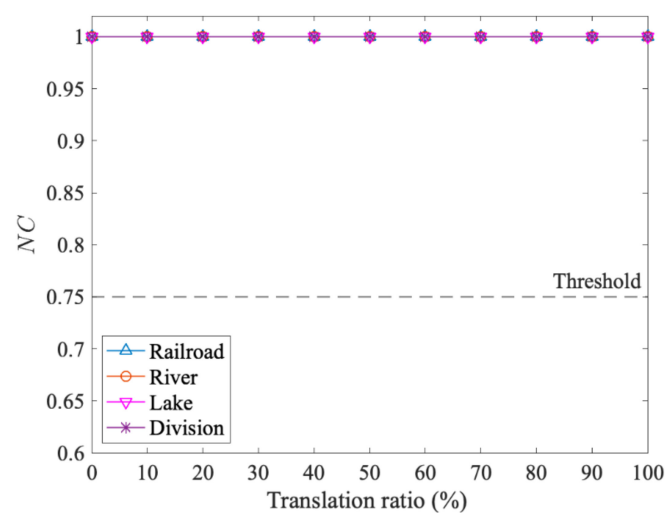


Figure 13. The results of translation attack.

From Figures 11–13, it is easy to see that the NC values of four experimental data are all one under different extents of RST attack. In detail, the proposed method can resist the attack of any angle of rotation. It can handle the extreme situation in scaling and translation; for example, the data are scaled to 0.1 and 10 times the original one, and the translation ratio is 100%. Thus, it shows that the proposed algorithm can completely resist the RST attacks.

#### 4.6. The Robustness of Adding and Deleting Objects

The adding and deleting of objects for vector geographic data are also common operations in geographic analysis. Different from projection transformation and RST attacks, these operations will increase or decrease the number of objects in the vector geographic data. A good watermarking algorithm should also be able to resist the attacks from adding and deleting objects.

In the subsection, an attack of addition is conducted by adding objects to the watermarked data in storage order. This is usually quantified by the addition ratio, which is defined as the percentage of the number of non-watermarked objects added to the number of the original objects. The attack of deletion is just the reverse operation of the addition attack, and the deletion ratio is used to represent the degree of deletion. Figure 14 presents the simulation results concerning the robustness against

adding objects. The adding ratio is changed from 10% to 200%. The corresponding results of deleting objects with the deletion ratio of 10% to 90% are given in Figure 15.

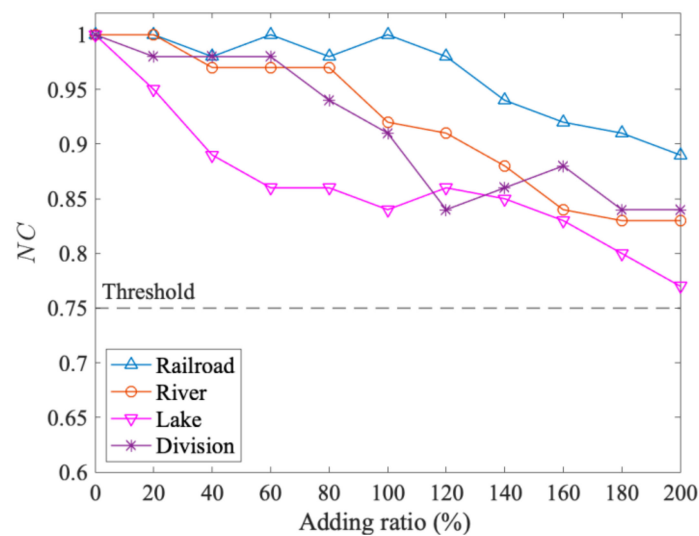


Figure 14. The results after adding objects.

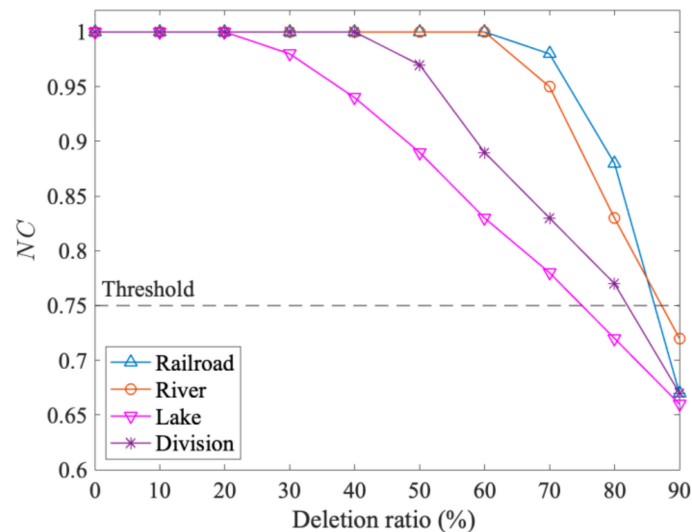


Figure 15. The results after deleting objects.

It can be observed in Figure 14 that the correlation of the extracted watermark and the original watermark, i.e., NC values, with the proposed method are higher than the threshold, 0.75, under all addition attacks. For example, even the attacking strength is quite strong, e.g., when the adding ratio is 200%, the NC values of the four test data are still higher than 0.77. In addition, we also noticed that with the increase of the addition ratio, the NC values of all data show a downward trend.

Meanwhile, as shown in Figure 15, the NC values are reduced with the deletion ratio, with a downward trend. However, the NC values of all data are still above the threshold even when 70% of objects are deleted. Moreover, the NC value of Lake starts to fall below the threshold when 80% of objects are deleted, and then all the NC values of the test data become lower than the threshold as more objects are deleted. It can be deduced that the watermark will be deleted along with the watermarked data under the attacks of deleting objects. However, the watermark could be extracted in the remaining objects, and the watermark can be extracted under certain limits. Therefore, to sum up, the proposed scheme is resilient to attacks of addition and deletion.

## 5. Discussion

The above experimental results show that the proposed method based on feature invariants has strong robustness, especially in projection transformation attacks. It achieves the same effect as non-blind watermarking and is better than blind watermarking. To better understand the characteristics of the proposed method, more discussions are given from three perspectives in the following.

### 5.1. The Characteristics of Resisting Arbitrary Projection Transformation

The proposed method opens a new direction in resisting projection transformation by using feature invariants, i.e., the number of vertices invariant, the storage order invariant and the storage direction. Different from non-blind watermarking with the help of original data, these feature invariants exist in both the original data and the watermarked data, and do not need to be recorded in advance. Therefore, the detection procedure based on these feature invariants is totally blind. Besides, the method is also different from other blind watermarking methods that are based on coordinate values. These feature invariants perfectly avoid the change of coordinate values caused by projection transformation. For example, the number of vertices of two adjacent objects comes from a kind of statistical characteristic, and the storage order and storage direction are the ones that exist in the storage level of the data.

In addition, the experimental results also prove that these feature invariants used by this method work very well in all of the four types of projections (i.e., equal area projection, conformal projection, equidistant projection and compromise projection), not just in one of them. As mentioned earlier, these four types of map projections cover all possible situations that cause the distortion of projection transformation. Being able to resist these four types of projections at the same time indicates the ability to resist arbitrary projection transformation. Thus, the feature invariants of the proposed method in this paper can completely resist arbitrary projection transformation.

### 5.2. The Characteristics of Lossless Watermarking

The proposed method is a kind of lossless watermarking method, which is a method that can realize copyright protection without causing damage to the vector geographic data [16]. The characteristics of lossless watermarking of the proposed method are achieved effectively, as shown in Table 3. None of the coordinate values in four sets of experimental data are modified; that is, the embedding of the watermark has no impact on the data accuracy. This is because the watermark of the method is embedded in the storage direction of adjacent objects, and the changing of the storage direction does not affect the coordinate values of the adjacent objects. In detail, the storage order between objects is disordered and meaningless in vector geographic data. Changing disordered objects to ordered objects makes the proposed method a lossless watermarking. Besides, compared with other non-lossless watermarking, the method has a wider application, especially in some industries with extremely high data accuracy requirements.

To better show the lossless characteristics of the proposed method, one of the experimental data sets, Railroad, is taken as an example. Compared to Tong's method [22], the comparison results are shown in Table 9. It is shown that the number of changed vertices in the proposed method is 0, and the  $rc$  value is also 0. However, 5053 vertices in Tong's method are modified, and the corresponding  $rc$  value is 19.68%, which means that the accuracy of 19.68% of the vertices cannot be guaranteed. It would be unacceptable in some fields, such as military and medical fields, which require extremely high data precision. On the contrary, the method in this paper does not have this issue, due to the fact that it has no impact on data accuracy. Therefore, the proposed method is lossless and can be applied to various application scenarios.

**Table 9.** Comparison results of lossless and non-lossless watermarking.

Distortion	The Proposed	Tong's
Vertices	25,677	25,677
Changed vertices	0	5053
<i>rc</i>	0	19.68%

### 5.3. The Analysis of the Watermark Synchronization Mechanism

The proposed method employs the watermark synchronization mechanism based on the number of vertices invariant, in which the majority voting mechanism plays a crucial role. The watermark synchronization mechanism allows the watermark to be embedded redundantly. That is, each watermark bit is embedded in the data multiple times, which provides the condition for the majority voting mechanism. When the number of objects changes in map editing, the majority voting mechanism can reduce the wrong result induced by object addition or deletion. Therefore, the probability is greatly improved for extracting the complete watermark information, and the proposed method has strong robustness against addition attacks and deletion, as verified in the experiments of adding and deleting objects.

Besides, the watermark index to be embedded is determined by the number of vertices of adjacent objects in the watermark synchronization mechanism. It is very useful for polyline and polygon data, since the distribution of the number of vertices is relatively discrete in polylines and polygons. However, it is not suitable for point data. This is because a point object generally contains only one vertex, which leads to the result that all of the watermark indices will have the same fixed value in the point data. Therefore, the watermark indices cannot be fully mapped, that is, the watermark information cannot be fully embedded in the point data.

## 6. Conclusions

How to resist projection transformation for vector geographic data in GIS is a challenging issue in robust watermarking research. To solve the issue, a novel robust and blind watermarking method, based on feature invariants, is proposed. In the method, the number of vertices of two adjacent objects is used to calculate the embedding watermark index, which can effectively guarantee the synchronization mechanism of the watermark. At the same time, the storage direction of two adjacent objects is used to embed the watermark, which does not require the proposed algorithm to modify the coordinates, and thus maintains the integrity of the data. Experimental results verify that the method can not only have good robustness in resisting arbitrary projection transformations, but also achieves blindness and losslessness, which makes it easily applicable in practice, especially in GIS industries with high data accuracy requirements. For example, in intelligent transportation and military navigation, both copyright protection and the accuracy of vector geographic data are always the key factors to be considered. It requires the data keep the watermark under various attacks, including projection transformation. Furthermore, the data accuracy should not be affected by watermark embedding because any slightest error on the data will lead to failure in navigation decisions. Therefore, the proposed method provides an effective means for these applications, which strikes a good balance between being robust to attacks and being lossless in as regards data. The proposed method used the number of vertices for the first time to change the meaningless storage order to a meaningful storage direction, compared with the state-of-the-art of watermarking research for vector geographic data. This method of transforming a meaningless feature in vector geographic data into a meaningful one and embedding watermark information in the feature can provide a valuable reference for current watermarking research. However, the main limitation of our scheme lies in that it is not suitable for points. How to convert points into polylines or polygons is a key issue. A feasible approach seems to be to construct the Voronoi diagram from points so that every point has a unique Thiessen polygon. This will be further investigated in our future works.

**Author Contributions:** Conceptualization, Qifei Zhou, Na Ren, and A-Xing Zhu; methodology, Qifei Zhou, Na Ren and Changqing Zhu; validation, Qifei Zhou and Na Ren; formal analysis, Qifei Zhou and Na Ren; writing—original draft preparation, Na Ren; writing—review and editing, Qifei Zhou, A-Xing Zhu and Changqing Zhu; visualization, Qifei Zhou and Na Ren; supervision, Na Ren, Changqing Zhu and A-Xing Zhu; funding acquisition, Na Ren and Qifei Zhou. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Natural Science Foundation of China under Grant 41971338; the Natural Science Foundation of Jiangsu Province under Grant BK20191373; and the China Scholarship Council under Grant 201906860010.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lü, G.; Batty, M.; Strobl, J.; Lin, H.; Zhu, A.X.; Chen, M. Reflections and speculations on the progress in Geographic Information Systems (GIS): A geographic perspective. *Int. J. Geogr. Inf. Sci.* **2019**, *33*, 346–367. [\[CrossRef\]](#)
2. Goodchild, M.F. Reimagining the history of GIS. *Ann. GIS* **2018**, *24*, 1–8. [\[CrossRef\]](#)
3. Peng, F.; Lin, Z.; Zhang, X.; Long, M. Reversible Data Hiding in Encrypted 2D Vector Graphics Based on Reversible Mapping Model for Real Numbers. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2400–2411. [\[CrossRef\]](#)
4. Wang, N.; Men, C. Reversible fragile watermarking for 2-D vector map authentication with localization. *Comput. Des.* **2012**, *44*, 320–330. [\[CrossRef\]](#)
5. Tang, Z.; Zhang, Y.; Huang, J.; He, H.; Ding, Y. A novel infringement detection method for GIS vector data. *ISPRS Int. J. Geo-Inf.* **2019**, *9*, 12. [\[CrossRef\]](#)
6. Zhu, C. Research Progresses in Digital Watermarking and Encryption Control for Geographical Data. *Acta Geod. Cartogr. Sin.* **2017**, *46*, 1609–1619. [\[CrossRef\]](#)
7. López, C. Watermarking of digital geospatial datasets: A review of technical, legal and copyright issues. *Int. J. Geogr. Inf. Sci.* **2002**, *16*, 589–607. [\[CrossRef\]](#)
8. Abubahia, A.; Cocea, M. Advancements in GIS map copyright protection schemes—a critical review. *Multimed. Tools Appl.* **2017**, *76*, 12205–12231. [\[CrossRef\]](#)
9. Ingemar, J.C.; Miller, M.L.; Jeffrey, A.B.; Fridrich, J.; Kalker, T. *Digital Watermarking and Steganography*, 2nd ed.; Morgan Kaufmann Publishers: Burlington, MA, USA, 2008.
10. Shih, F.Y. *Digital Watermarking and Steganography: Fundamentals and Techniques*, 2nd ed.; Taylor & Francis; CRC Press: Boca Raton, FL, USA, 2017.
11. Wang, Y.; Yang, C.; Zhu, C.; Ding, K. An efficient robust multiple watermarking algorithm for vector geographic data. *Information* **2018**, *9*, 296. [\[CrossRef\]](#)
12. Peng, Y.; Lan, H.; Yue, M.; Xue, Y. Multipurpose watermarking for vector map protection and authentication. *Multimed. Tools Appl.* **2018**, *77*, 7239–7259. [\[CrossRef\]](#)
13. Liu, Y.; Yang, F.; Gao, K.; Dong, W.; Song, J. A zero-watermarking scheme with embedding timestamp in vector maps for Big Data computing. *Cluster Comput.* **2017**, *20*, 3667–3675. [\[CrossRef\]](#)
14. Yan, H.; Zhang, L.; Yang, W. A normalization-based watermarking scheme for 2D vector map data. *Earth Sci. Inform.* **2017**, *10*, 471–481. [\[CrossRef\]](#)
15. Peng, Z.; Yue, M.; Wu, X.; Peng, Y. Blind watermarking scheme for polylines in vector geo-spatial data. *Multimed. Tools Appl.* **2015**, *74*, 11721–11739. [\[CrossRef\]](#)
16. Zhou, Q.; Ren, N.; Zhu, C.; Tong, D. Storage Feature-Based Watermarking Algorithm with Coordinate Values Preservation for Vector Line Data. *KSII Trans. Internet Inf. Syst.* **2018**, *12*, 3475–3496. [\[CrossRef\]](#)
17. AL-ardhi, S.; Thayananthan, V.; Basuhail, A. RST invariant watermarking technique for vector map based on LCA-transform. *Telkommika* **2020**, *18*, 500. [\[CrossRef\]](#)
18. Liu, J.; Yang, Y.; Ma, D.; He, W.; Wang, Y. A novel watermarking algorithm for three-dimensional point-cloud models based on vertex curvature. *Int. J. Distrib. Sens. Netw.* **2019**, *15*. [\[CrossRef\]](#)
19. Da, Q.; Sun, J.; Zhang, L.; Kou, L.; Wang, W.; Han, Q.; Zhou, R. A Novel Hybrid Information Security Scheme for 2D Vector Map. *Mob. Netw. Appl.* **2018**, *23*, 734–742. [\[CrossRef\]](#)
20. Yang, C.; Zhu, C.; Wang, Y.; Rui, T.; Zhu, J.; Ding, K. A Robust Watermarking Algorithm for Vector Geographic Data Based on Qim and Matching Detection. *Multimed. Tools Appl.* **2020**. [\[CrossRef\]](#)

21. Yan, J.; Song, X.; Gong, G. Averaged ratio between complementary profiles for evaluating shape distortions of map projections and spherical hierarchical tessellations. *Comput. Geosci.* **2016**, *87*, 41–55. [[CrossRef](#)]
22. Tong, D.; Ren, N.; Zhu, C.; Lin, W. A Watermarking Algorithm Resisting to Projection Transformation for Vector Geographic Data. *J. Geo-Inf. Sci.* **2016**, *18*, 1037–1042. [[CrossRef](#)]
23. Doncel, V.R.; Nikolaidis, N.; Pitas, I. An Optimal Detector Structure for the Fourier Descriptors Domain Watermarking of 2D Vector Graphics. *IEEE Trans. Vis. Comput. Graph.* **2007**, *13*, 851–863. [[CrossRef](#)] [[PubMed](#)]
24. Zhu, C.; Yang, C.; Wang, Q. A watermarking algorithm for vector geo-spatial data based on integer wavelet transform. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2008**, *37*, 15–18.
25. Zope-Chaudhari, S.; Venkatachalam, P. Robust Watermarking for Protection of Geospatial Data. *2012 IACSIT Hong Kong Conf.* **2012**, *29*, 34–38.
26. Huang, X.; Gu, J. A Non-Blind Detection Watermarking Algorithm for 2-Dimensional Engineering Drawings. *J. Eng. Graph.* **2006**, *27*, 158–161.
27. Yang, H.; Min, L.; Hou, X. A digital watermarking algorithm against projection transformation based on modulation scheme. *Sci. Surv. Mapp.* **2015**, *40*, 19–22. [[CrossRef](#)]
28. Yang, H.; Hou, X. A Watermarking Scheme against Equal-Angle Projection for Vector Map Data. *J. Geomat. Sci. Technol.* **2014**, *31*. [[CrossRef](#)]
29. Li, A.B.; Zhu, A.X. Copyright authentication of digital vector maps based on spatial autocorrelation indices. *Earth Sci. Inform.* **2019**, *12*, 629–639. [[CrossRef](#)]
30. Wang, X.; Huang, D.; Zhang, Z. A Robust Zero-Watermarking Algorithm for Vector Digital Maps Based on Statistical Characteristics. *J. Softw.* **2012**, *7*, 2349–2356. [[CrossRef](#)]
31. Xi, X.; Zhang, X.; Liang, W.; Xin, Q.; Zhang, P. Dual Zero-Watermarking Scheme for Two-Dimensional Vector Map Based on Delaunay Triangle Mesh and Singular Value Decomposition. *Appl. Sci.* **2019**, *9*, 642. [[CrossRef](#)]
32. Arnold, V.I.; Avez, A. *Ergodic Problems of Classical Mechanics*; W. A. Benjamin: New York, NY, USA, 1968.
33. Saadi, S.; Merrad, A.; Benziane, A. Novel secured scheme for blind audio/speech norm-space watermarking by Arnold algorithm. *Signal Process.* **2019**, *154*, 74–86. [[CrossRef](#)]
34. Loan, N.A.; Hurrah, N.N.; Parah, S.A.; Lee, J.W.; Sheikh, J.A.; Bhat, G.M. Secure and Robust Digital Image Watermarking Using Coefficient Differencing and Chaotic Encryption. *IEEE Access* **2018**, *6*, 19876–19897. [[CrossRef](#)]
35. Zhang, Z.; Wu, L.; Yan, Y.; Xiao, S.; Sun, H. An improved reversible image watermarking algorithm based on difference expansion. *Int. J. Distrib. Sens. Netw.* **2017**, *13*. [[CrossRef](#)]
36. Ahmed Elmousalami, A.M.T. *Digital Image Watermarking Using Fuzzy Logic and Naturalness Preserving Transform*; Kansas State University: Manhattan, KS, USA, 2004.

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).