Received November 24, 2020, accepted December 3, 2020, date of publication December 9, 2020,
date of current version December 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043450

Commutative Encryption and Watermarking Algorithm Based on Feature Invariants for Secure Vector Map

NA REN[1,2,3], **CHANGQING ZHU**[1,2,3], **DEYU TONG**[4], **(Member, IEEE), WEITONG CHEN**[1,2,3], **AND QIFEI ZHOU**[1,2,3]

[1]Key Laboratory of Virtual Geographic Environment, Nanjing Normal University, Ministry of Education, Nanjing 210023, China
[2]State Key Laboratory of Cultivation Base of Geographical Environment Evolution, Nanjing 210023, China
[3]Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing 210023, China
[4]College of Information Engineering, Nanjing University of Finance and Economics, Nanjing 210023, China

Corresponding author: Changqing Zhu (zcq88@263.net)

This work was supported in part by the National Natural Science Foundation of China under Grant 41971338 and Grant 42071362, and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20191373.

ABSTRACT The existing commutative encryption and watermarking (CEW) methods based on feature invariants can achieve both the robustness of the watermarking algorithm and the security of the encryption algorithm. However, they are only applicable to the raster data such as images, videos, etc. In particular, the organization structure and storage structure of vector map have not been considered in these methods. Therefore, they cannot be used for vector map. This paper derives two feature invariants to solve this problem, which are the sum of inner angles and the storage direction of two adjacent objects according to the inherent characteristics of vector map. Based on these two feature invariants, a new CEW method is proposed in this paper, which includes the feature invariants based watermarking algorithm and the perceptual stream cipher based encryption algorithm on coordinates. Since the coordinate values used in encryption and the feature invariants used in watermarking are independent of each other, the commutativity is achieved for the proposed CEW method. The experiments are given to verify that the proposed CEW method can achieve the commutativity between encryption and watermarking without deteriorating accuracy of data. Besides, it has been verified that the proposed method is more robust to rotate, scaling, translation, and projection transformation compared with the existing CEW methods and has high security. The proposed algorithm has good scalability of encryption, and arbitrary encryption methods based on encrypting the coordinate values can be applied without affecting the extracted feature invariants.

INDEX TERMS Commutative encryption and watermarking, feature invariant, vector map, perceptual stream cipher, lossless.

I. INTRODUCTION

With the rapid development of cloud computing technology and geographical information system (GIS), more and more vector maps are stored, distributed and processed in the cloud because it is convenient and low cost. However, since the vector maps stored in the cloud is out of the user's control, it leads to concerns of data security and privacy leakage [1], [2]. The encryption and watermarking are served as two major techniques for protecting the vector map security [3]–[7]. The encryption technique converts original and meaningful data content into hard-to-understand data content under the control of a key, which protects the data's confidentiality [8]–[10]. Only the authorized customer who has the right key can recover the data correctly, but it cannot achieve the copyright trace for data in plaintext or ciphertext. Compared with the encryption technique, the watermarking technique embeds the invisible information into the data, which can be extracted and used to authenticate data's ownership or identification, but it cannot ensure the security in data transmission [11]–[18]. Therefore, it is possible to overcome the limitation of a single technique by combining both the techniques of encryption and watermarking together to protect both confidentiality and ownership/ identification [19]–[21].

The associate editor coordinating the review of this manuscript and approving it for publication was SK Hafizul Islam.

There are two types of methods that simply combines encryption and watermarking together. The first type is encryption first then watermarking. This method will modify the ciphertext and cause the failure of decryption. The other one is watermarking first then encryption. This method also has the drawback that the secret key of cryptography must be gained by the copyright identification party, resulting in the leakage of key. The above methods are not practical and flexible because watermarking and encryption are not separated and commutative. Commutative encryption and watermarking (CEW) is such a technology that integrates encryption and watermarking and ensures the operation sequence is exchangeable [22], [23]. In this method, the operation of encryption and watermarking do not mutually interfere with each other which is often called commutative property. That means the watermark can be embedded into the encrypted data directly, or it can be extracted correctly from the decrypted data or the encrypted data. The CEW is not a simple combination of encryption and watermarking. However, the current encryption and watermarking techniques have not been involved the commute with the other, which in turn result in the absence of the commute of the CEW derived by using the existing encryption and watermarking techniques directly [24]. Therefore, how to ensure the commute between encryption and watermarking is a key issue in the CEW research.

There are three basic types of methods for achieving the commute of CEW. The first one is based on different data fields, in which the data are separated into two different parts, where one part is encrypted and the other is watermarked. Since the encryption part is independent of the watermarking part, they are naturally commutative [25]–[29]. For example, in [28], the data are partitioned into two parts after wavelet transformation, the high-level coefficients are fully encrypted, while the low-level coefficients are watermarked. In [29], the data are partitioned into important part (such as motion vector), and robust part (such as Discrete Cosine Transform coefficient values), where the former is encrypted and the latter is watermarked. However, only a part of data has been encrypted and the other part of data with watermarking operation is still in plaintext with this method, which will greatly lead to the issue of poor security of data.

The second type of methods is based on homomorphic encryption [30]. Homomorphic encryption is the operation on the encrypted data which can offer the same results after calculations as the operation straight on the original data. Therefore, if the encryption and the watermarking consist of the same homomorphic operation, they will not interfere with each other. For example, in [31], the quasi-commutative encryption and watermarking scheme is proposed based on the homogenous operations of additive modulation. The whole data can be encrypted in this method. Though some schemes based on homomorphic encryption show higher security than the methods based on the different data fields, their robustness of watermarking is relatively lower.

The third type of methods is based on invariant features that are extracted from the data. The invariant features are used for watermarking. The encryption will not have any impact on these features at the same time. Thus, the commute for CEW is achieved [32]–[34]. For example, the feature of global histogram statistics is invariable after scrambling pixel position. This invariant feature has been utilized to watermark and achieve the encryption by scrambling pixel position simultaneously. The security of encryption and robustness of watermarking is thus improved effectively [34]. However, some feature invariants proposed by this kind of methods (e.g., histogram statistics) cannot be applicable for vector map due to the difference in manifestations between vector map and raster data.

From the aforementioned introduction of current studies for CEW, it can be summarized as follows. It is easy to implement the methods based on different data fields, where the operations of encryption and watermarking are separated distinctly. However, the data with watermarking operation are still in plaintext, and suffer from the risks of data leak. The methods based on homomorphic encryption can achieve encryption on all data and improve the security of data. However, the current watermarking methods homomorphic with encryption are still less, which are limited to the simple homomorphic operation such as addition and multiplication. Thus, the robustness of the watermarking algorithm is weak. The methods based on feature invariants achieve encryption for full text by fully utilizing the data features and have good watermarking robust. It provides the basic techniques and approaches for widespread application in vector map. However, there are some obvious differences between vector map and traditional raster data, especially in aspects of data representation, organization methods, storage structure and application environment. Therefore, how to select the representative feature invariants of vector map that satisfy the commute of CEW has become a difficult and urgent problem in the current CEW research of vector map.

This paper aims to realize the commute of CEW for vector map based on feature invariants. Thus, the focus of this method is to construct feature invariants of CEW for vector map. Therefore, the method in this paper consists of three key steps: (1) construct the feature invariants of CEW based on the essential features of vector map; (2) present a feature invariants-based watermarking algorithm; (3) propose an encryption algorithm without changing the feature invariants, and thus ensure the commutativity of CEW for vector map. The remainder of the paper is organized as follows. In Section II, the feature invariants of CEW for vector map are constructed. In Section III, a new CEW algorithm based on feature invariants is presented, which includes two parts, i.e., watermark part and encryption part. Experiments and results are given in Section IV. Discussions are illustrated in Section V. Section VI draws the conclusions.

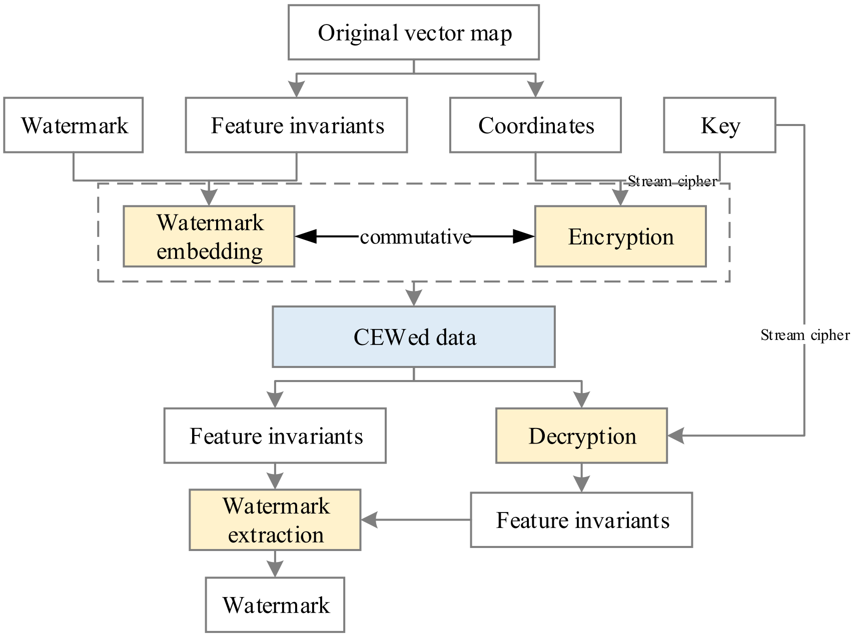


FIGURE 1. The proposed algorithm.

II. FEATURE INVARIANTS OF CEW FOR VECTOR MAP

A. THE FEATURE INVARIANT OF SUM OF INNER ANGLES

The sum of interior angle of vector map refers to the sum value of the inner angles formed by all vertex inside polylines (the beginning vertex and the end vertex are allowed to be virtually connected) or polygons. Suppose one vector map includes N objects, one of which is represented by $P_i, 0 \leq i < N$, NP_{P_i} represents the vertex number of the current feature, the sum of inner angles of one object can be represented as:

$$SumA_i = \left(N_{P_i} - 2\right) \times 180^\circ \tag{1}$$

where $SumA_i$ is the sum of inner angles of i-th object of polyline/polygon. Subtracting 2 from N_{P_i} is because a polygon with n vertices can be considered to be made up of (n − 2) triangles.

From Eq. (1), it can be observed that the sum of inner angles is only related to the number of vertices in the current object. It is noted that the number of vertices does not change with the operation, such as rotation, scaling, translation and projection transformation, etc. That is to say, the sum of inner angles has invariability under the operation of rotation, scaling, translation, and projection transformation.

B. THE FEATURE INVARIANT OF STORAGE DIRECTION OF TWO ADJACENT OBJECTS

Suppose two adjacent objects are P_i and P_{i+1}, and the number of vertices in these two adjacent objects are N_{P_i} and $N_{P_{i+1}}$, respectively. The quantized value of the storage direction of two adjacent objects, namely $PDir_{(i+1)/2}$ is expressed as:

$$PDir_{(i+1)/2} = \begin{cases} 1, & N_{P_i} > N_{P_{i+1}} \\ 0, & N_{P_i} \leq N_{P_{i+1}} \end{cases} \tag{2}$$

It can be observed from Eq. (2) that the storage direction of two adjacent objects are only related to the number of vertices in two adjacent objects. Therefore, the storage direction also has invariability under operation of rotation, scaling, translation and projection transformation.

III. PROPOSED METHOD

A. BASIC IDEA

The basic idea of this paper is to establish the feature invariants of CEW for vector map, propose a watermarking algorithm based on the constructed feature invariants, and present an encryption algorithm based on coordinate values that keeps the constructed feature invariants unchanged. In this method, the derived feature invariants are the sum of inner angles and the storage direction of two adjacent objects, which are independent of the coordinates of vector map. Therefore, the proposed watermarking algorithm based on feature invariants and the encryption algorithm based on coordinates do not interfere with each other.

The flowchart of this method is shown in Figure 1. From the original data to the CEWed data, we can see that the proposed CEW method consists of two parts, namely the watermark part and the encryption part. First, feature invariants and coordinates are extracted from the original data. Then the watermark is embedded in the feature invariants, and the encryption is applied to the coordinates with a key. The watermark and the encryption are commutative. Afterwards, the watermark extraction is the inverse process of the watermark embedding, and the decryption is also the inverse process of the encryption. Furthermore, the watermark extraction and the decryption can be exchangeable. As aforementioned, the sum of inner angles of two adjacent objects is utilized to determine the bit of watermarking. In this

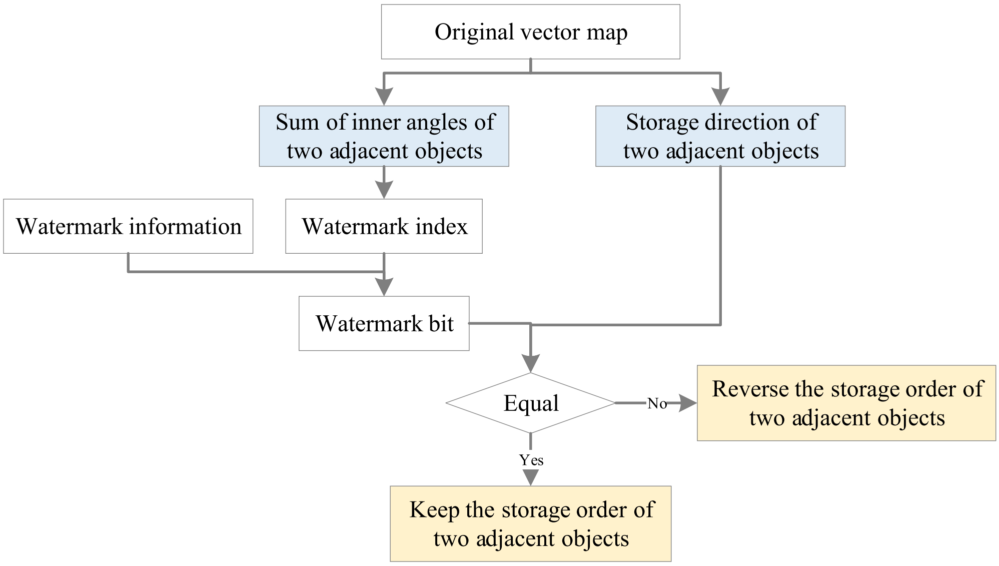

Original vector map
Sum of inner angles of two adjacent objects
Storage direction of two adjacent objects
Watermark information
Watermark index
Watermark bit
Equal
No
Reverse the storage order of two adjacent objects
Yes
Keep the storage order of two adjacent objects

FIGURE 2. The diagram of embedding watermark.

way, the synchronization of watermarks can be achieved before and after their embedding processes. In addition, the storage directions of two adjacent objects are used to embed the watermark information. In this way, not only the coordinates of data are unchanged before and after the watermark embedding, but also the robustness of the watermarking algorithm can be enhanced. The encryption algorithm is proposed for the coordinates based on perceptual stream cipher to achieve the commutativity between the encryption and the watermarking. This encryption algorithm does not affect the feature invariants, and therefore does not interfere with the watermarking part.

B. WATERMARK PART

1) WATERMARK EMBEDDING

The detailed diagram of watermark embedding method is shown in Figure 2. First, sum of inner angles and storage direction of two adjacent objects are calculated from the original vector map. Then, the watermark index is determined by the sum of inner angles, and the watermark bit at the watermark index is obtained. Finally, the watermark bit is embedded by making the storage direction of the two adjacent objects equal to the watermark bit.

The basic steps of watermark embedding are listed as follows.

Step 1: The copyright information is mapped to a meaningless watermark information, which is a binary sequence including zeros and ones. The watermark information is represented by $W = \left\{w_j | w_j = 0, 1\right\}$, where $0 \le w_j \le N_W - 1$, and N_W represents the length of watermark information.

Step 2: Read the vector map, and combine two adjacent objects into one object pair (P_i, P_{i+1}). Then, the number of vertices is got for each object pair $\left(N_{P_i}, N_{P_{i+1}}\right)$.

Step 3: Calculate the sum of internal angle of each object according to formula (1), then $(SumA_i, SumA_{i+1})$ is obtained.

Step 4: In order to ensure the synchronization relationship of watermark information effectively, the index of the watermark bit is calculated by,

$$Index = ((SumA_i + k_1) \times k_2 + SumA_{i+1}) \mod N_W \quad (3)$$

where k_1 and k_2 are two random primes. Therefore, the embedding watermark bit of the current object pair is w_{Index}.

Step 5: The quantized value of the storage direction of them, namely $PDir_{(i+1)/2}$ is calculated according to Eq. (2).

Step 6: Then, the watermark is embedded in the storage direction for each object pair. The embedding rule is given as:

$$\left(P'_i, P'_{i+1}\right) = \begin{cases} (P_i, P_{i+1}), & \textit{if } PDir_{(i+1)/2} = w_{Index} \\ (P_{i+1}, P_i), & \textit{if } PDir_{(i+1)/2} \neq w_{Index} \end{cases} \quad (4)$$

Through Eq. (4), it is easy to see that if the storage direction of the current object pair is the same as the watermark bit, then the storage order of the current object pair will not be changed. Otherwise, the storage order of the object pair is reversed. Finally, the embedded watermark bit is equal to the storage direction of the current object pair.

Step 7: After embedding the watermark for all objects, the watermarked vector map is obtained.

During the aforementioned watermark embedding process, the coordinates of vector map are not changed. This could provide good precondition for the commutativity of the proposed encryption algorithm in this paper.

2) WATERMARK EXTRACTION

Watermark extraction is the inverse process of watermark embedding, and the watermark bit is determined by the storage direction of each object pair. Therefore, the watermark bit can be determined by,

$$w_{Index'} = \begin{cases} 0, & if\ PDir'_{(i+1)/2} = 0 \\ 1, & if\ PDir'_{(i+1)/2} = 1 \end{cases} \tag{5}$$

where $PDir'_{(i+1)/2}$ is the storage direction of the object, which is acquired as the same procedure of embedding watermark. $Index'$ is the embedding watermark index, which can also be acquired as the same procedure of embedding watermark.

It is noticed that one watermark bit may be extracted for several times, and the majority rule is used to determine the watermark information for each watermark bit. The majority rule is given as:

$$w'_j = \begin{cases} 0, & if\ N_{jw_0} \geq N_{jw_1} \\ 1, & if\ N_{jw_0} < N_{jw_1} \end{cases} \tag{6}$$

where N_{jw_0} represents the number of extracting the watermark information of 0 for the bit of j, and N_{jw_1} represents the number of extracting the watermark information of 1 for the bit of j. The watermark information W' can thus be extracted with the steps mentioned above.

Then, the bit error rate (BER) is employed to estimate the similarity between the extracted watermark $W' = \left\{w'_0, w'_1, \ldots, w'_{N_w-1}\right\}$ and the original watermark $W = \left\{w_0, w_1, \ldots, w_{N_w-1}\right\}$,.

$$BER = \frac{\sum_{i=0}^{N_W-1} \left|w'_i - w_i\right|}{N_W} \tag{7}$$

BER ranges from 0 to 1, where the lower BER means that less watermark changed.

C. ENCRYPTION PART

1) PERCEPTUAL STREAM CIPHER

Stream cipher is a symmetric key encryption system used to encrypt streams of plaintext data. It produces an infinite pseudo-random sequence of n-bit symbols, which is, actually, the keystream. Encryption is then XORed with the plaintext bits, thereby producing a sequence of ciphertext bits. Decryption is done in a similar process, with an identical keystream sequence generated and XORed with ciphertext to produce plaintext [35], [36].

RC4 is probably the most widely used stream cipher and is used in many applications [37]. It has two parts, namely the Key Scheduling Algorithm (KSA) and the Pseudo-Random Generator Algorithm (PRGA). KSA turns a random key into an initial permutation S of $\{0, 1, \ldots, N_S - 1\}$, where N_S is the size of the RC4 permutation. PRGA uses this pseudorandom permutation to generate the arbitrary number of pseudorandom keystream bytes [38], [39].

According to the discussion above, the coordinates of vector map can be expressed by binary system, and encrypted by RC4 stream cipher. Since the feature invariant based watermarking algorithm and the coordinates are independent of each other, the stream cipher based on coordinates will not affect the feature invariant proposed in this paper. However, straightforward RC4 stream cipher conducted on the coordinates will change geographic range of the vector map totally and possible generate invalid coordinate values, which cause significant degradation on visual effect and severe unevenness on spatial distribution. The traditional RC4 stream cipher is improved and a perceptual stream cipher method is proposed in this paper to solve this problem.

The main idea of the perceptual stream cipher is to keep the prior n bits unchanged and encrypt the other bits. More specifically, suppose an arbitrary value is represented in the binary form as $(a_1a_2a_3\ldots a_nb_1b_2b_3\ldots b_m)_2$, and the encryption will skip the bits $a_1a_2a_3\ldots a_n$ and only encrypt the bits $b_1b_2b_3\ldots b_m$ with keystream $c_1c_2c_3\ldots c_m$. Hence, the parameter n controls the number of bits, which cannot be modified. It implies that the coordinate values are encrypted by a stream cipher with range control.

2) CALCULATION OF THE ENCRYPTION BITS BASED ON THE PERCEPTUAL STREAM CIPHER

Based on the perceptual stream cipher, the n_x and n_y for x and y coordinate as the parameters can be calculated to construct the encryption bits. Firstly, read the coordinate values of vertices $V_i = \{(x_i, y_i)\,|i \in [1, N_V]\}$ from the vector map, where N_V means the number of the vertices. Then, each coordinate is converted into the binary form as $(V_i)_2 = \left\{((x_i)_2\,, (y_i)_2)\,|i \in [1, N_V]\right\}$. Then, the binary length of $(x_i)_2$ and $(y_i)_2$ are obtained as L_x and L_y, respectively. Then, get the coordinate range of the vector map $[x_{\min}, x_{\max}]$ and $[y_{\min}, y_{\max}]$, where $x_{\min}$ and $x_{\max}$ represent the minimum x-coordinate value and the maximum x-coordinate value, respectively. $y_{\min}$ and $y_{\max}$ represent the minimum y-coordinate value and the maximum y-coordinate value, respectively. Finally, calculate the n_x and n_y for x and y coordinate as the parameters of perceptual stream cipher with the following equations:

$$\begin{cases} n_x = \max(\lceil \log_2 x_i \rceil - \lceil \log_2(x_{\max} - x_{\min}) \rceil - 1, 0) \\ n_y = \max(\lceil \log_2 y_i \rceil - \lceil \log_2(y_{\max} - y_{\min}) \rceil - 1, 0) \end{cases} \tag{8}$$

where $\lceil\ \rceil$ means the operator of rounding up to an integer and max function means to choose the maximum value.

3) DATA ENCRYPTION AND DECRYPTION

After n_x and n_y are obtained, we can set the RC4 input key with *key*, and then generate the keystream $c_1c_2c_3\ldots c_{m_x}$ with the length of $m_x = L_x - n_x$, and continually generate the keystream $c_{m_x+1}c_{m_x+2}c_{m_x+3}\ldots c_{m_x+m_y}$ with the length of $m_y = L_y - n_y$. Therefore, skip the first n_x bit of $(x_i)_2$ and encrypt its remaining m_x bit with the keystream of $c_1c_2c_3\ldots c_{m_x}$ by exclusive OR operation. Encrypt $(y_i)_2$ in a similar way with the keystream of $c_{m_x+1}c_{m_x+2}c_{m_x+3}\ldots c_{m_x+m_y}$. The encrypted map can be obtained by converting the encrypted binary number into the decimal form.

TABLE 1. Meta information of experimental maps.

Experimental dataset	Geographical region	Coordinate system	Vertices	Scale	Map accuracy
Yonkers	New York, USA	WGS84	37584	1:1000	$10^{-6\circ}$
LianYunGang	Jiangsu, China	Albers	2779	1:50000	$5m$
DaFeng	Jiangsu, China	Albers	38373	1:50000	$5m$

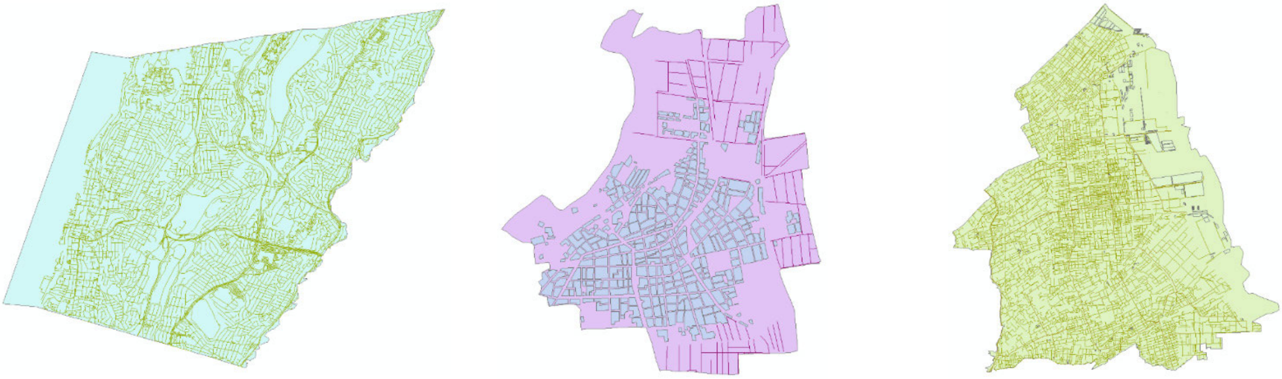

(a) Yonkers
(b) LianYunGang
(c) DaFeng

FIGURE 3. Visual expression of the experimental maps.

By using the same random key, i.e. *key*, the number of bits n_x and n_y, the inverse processes of data encryption are performed to obtain the decrypted vector map.

IV. EXPERIMENTS AND RESULTS

A. EXPERIMENTAL MAPS AND PARAMETER SETTINGS

Three vector maps in the shapefile format are chosen to verify the proposed CEW method. Three vector maps, including polyline and polygon layers, have been used in the experiments. They are named as Yonkers, LianYunGang and DaFeng. Their meta information is listed in Table 1, including the geographical region, the coordinate system, the number of vertices, scale and map accuracy. The map of Yonkers is downloaded from OpenStreetMap (http://download.geofabrik.de), and the other maps are provided by National Earth System Science Data Center, National Science & Technology Infrastructure of China (http://www.geodata.cn). It has to been noted that there are obvious differences in their coordinated system, vertices, scale and map accuracy. The visual expressions of the three vector maps are shown in Figure 3.

The experiments are conducted to verify the commutativity, imperceptibility, watermark robustness and security of the proposed CEW method. The experiments are performed on a platform of MATLAB 2018b with a CPU of i7-8700 and a memory of 16GB. The binary watermark sequence is used as the watermark and its length is 100 bits in the experiments for testing the watermarking method. Two random prime numbers for calculating the *Index* of embedding watermark bit are $k_1 = 199$ and $k_2 = 613$.

In the experiments for testing the encryption method, the secret key for stream cipher is $key_1 = abcdef1234567890$.

B. COMMUTATIVITY

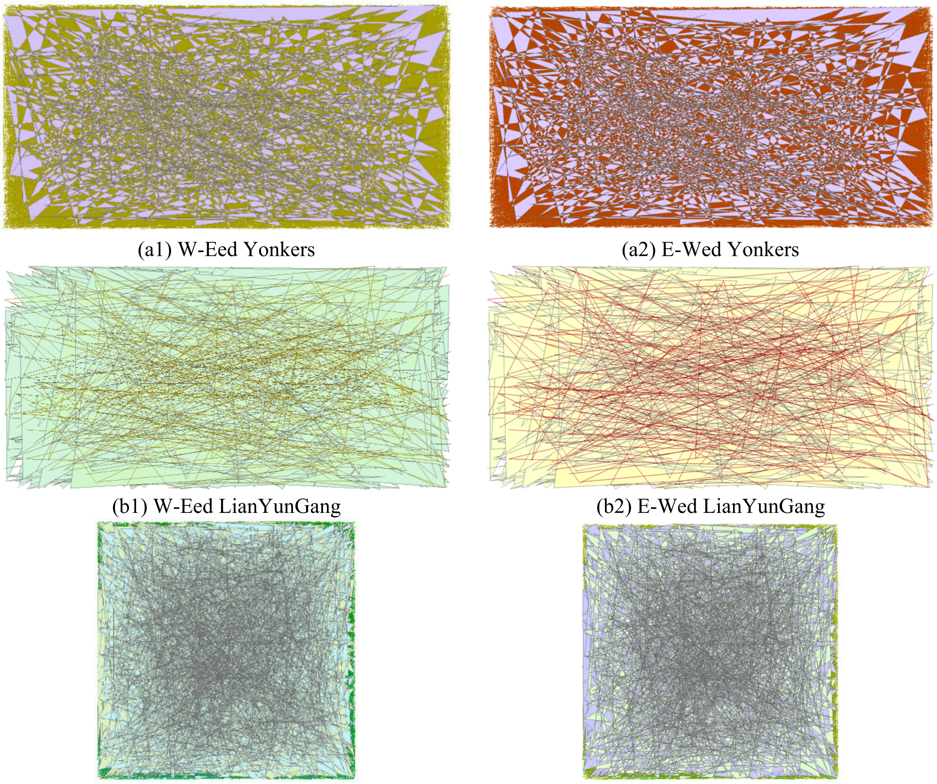
Commutativity is a basic indicator of the CWE algorithm and differs from other algorithms that combining encryption and watermarking. It is used to assess whether encryption and watermark interfere with each other. Commutativity enables the processing order of encryption and watermarking will not affect the final result, and watermark can be extracted from watermarked data no matter it is decrypted or not.

The commutativity between watermark embedding and encryption, and the commutativity between watermark extraction and decryption are investigated in this section. The results are shown in Figure 4. The watermarked-encrypted (W-Eed) maps, which are obtained by watermarking at first and then encrypting, are shown in Figure 4(a1), 4(b1) and 4(c1). The encrypted-watermarked(E-Wed) maps, which are obtained by encrypting at first and then watermarking, are shown in Figure 4(a2), 4(b2) and 4(c2). As shown in Figure 4, there is no difference (except the rendering style) between the W-Eed maps and E-Wed maps.

For evaluating the difference between the W-Eed maps and E-Wed maps objectively, the root-mean-square error (RMSE) is used here to measure the errors between the processed data and the original data. RMSE is calculated according to the mean error of vertices coordinates as:

$$RMSE = \sqrt{\frac{1}{N_v}\sum_{i=1}^{N_v}\left(x_i' - x_i\right)^2 + \left(y_i' - y_i\right)^2} \tag{9}$$

where N_v means the total number of vertices of the vector map. $\left(x_i', y_i'\right)$ means the processed coordinate value, and (x_i, y_i) means the original coordinate value. The higher value of RMSE implies that the error between these two maps are



(a1) W-Eed Yonkers

(a2) E-Wed Yonkers

(b1) W-Eed LianYunGang

(b2) E-Wed LianYunGang

(c1) W-Eed DaFeng

(c2) E-Wed DaFeng

FIGURE 4. **The CEW results.**

TABLE 2. **The results of the commutativity.**

Comparison Result	Yonkers	LianYunGang	DaFeng
RMSE	0°	$0m$	$0m$

larger. Besides, all the three datasets are conducted and named in the same way, with RMSE values between them are listed in Table 2.

According to the RMSE values in Table 2, all the RMSE values are 0. RMSE = 0 implies that all the vertices of the compared maps are the same, meaning that the maps under watermarking at first and then encrypting is consistent with the maps under encrypting at first and then watermarking. Thus, the commutativity between the watermark embedding and the encryption is verified.

Furthermore, the watermark extraction results will be the main investigation object to evaluate the commutativity between watermark extraction and decryption. Table 3 shows the BER values of watermark extracted from the decrypted-watermarked (D-Wed) maps and those from the CEWed maps.

From Table 3, either the D-Wed maps or the CEWed maps, the BER values between the extracted watermark and the original watermark are always 0, meaning that the watermark is correctly extracted from the D-Wed maps and the CEWed maps. Hence, the commutativity between the watermark extraction and the decryption is proved. Combine the experiments above, the commutativity of the proposed CEW scheme is verified successfully.

TABLE 3. **The results of extracted watermark.**

Type	Data	BER
D-Wed maps	Yonkers	0.00
	LianYunGang	0.00
	DaFeng	0.00
CEWed maps	Yonkers	0.00
	LianYunGang	0.00
	DaFeng	0.00

C. IMPERCEPTIBILITY

Imperceptibility refers to the difference between the D-Wed maps and the original maps for a CEW method. That is, the precision of vector maps should not be deteriorated after processes of watermarking embedding and decryption for a CEW method. Thus, the coordinate difference between the decrypted-watermarked data and the original data is used to evaluate the imperceptibility.

To evaluate the imperceptibility, three CEWed maps are decrypted to obtain the decrypted-watermarked maps, which are shown in Figure 5. It can be seen that there is no visible difference between the D-Wed maps and the original maps.

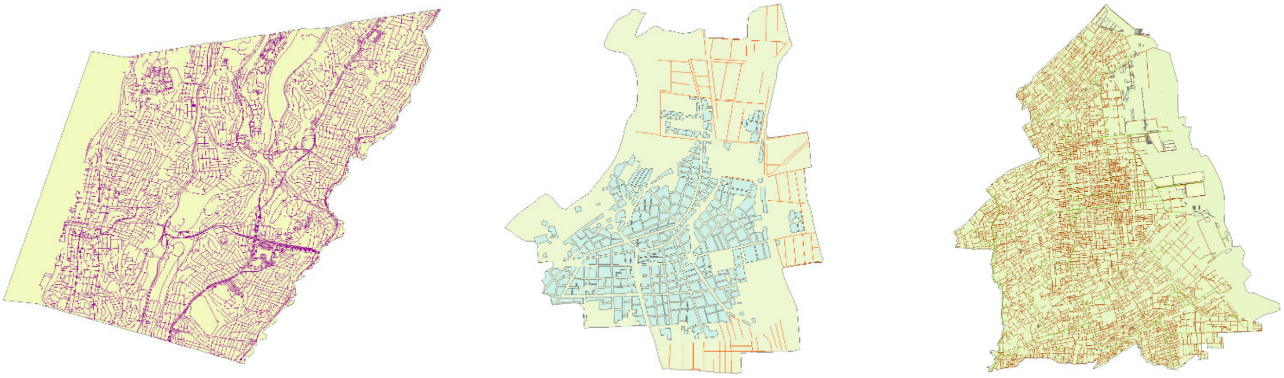

(a) D-Wed Yonkers (b) D-Wed LianYunGang (c) D-Wed DaFeng

FIGURE 5. The D-Wed maps.

TABLE 4. The results of imperceptibility.

Experimental Dataset	Vertices Number	Unchanged Vertices Number	RMSE			
			Proposed	Jang’s	Li’s	Yang’s
Yonkers	37584	37584	0’’	0.0006’’	0’’	0.0008’’
LianYunGang	2779	2779	0*m*	0.215*m*	0 *m*	0.822*m*
DaFeng	38373	38373	0*m*	0.361*m*	0 *m*	0.936*m*

It is noted that the method of embedding watermark based on feature invariants does not change any coordinate, so that embedding watermark and data decryption do not affect the visualization of the D-Wed maps.

The objective indices including RMSE is used to measure the quality of the D-Wed maps to further verify the imperceptibility of the proposed CEW algorithm. The imperceptibility of our method and two conventional schemes of Li and Zhu [44] and Yang *et al.* [12] are evaluated. They are tested using the same watermark length as our scheme. In addition, the other algorithm combining watermarking and encryption is used as the comparison algorithm, referred to the method in [41]. The experimental results with respect to the imperceptibility are shown in Table 4.

As shown in Table 4, all of the RMSE values of the proposed method are 0. And the number of unchanged vertices is the same as that of the original vertices in three datasets. That is, all the vertices of the dataset in the proposed method keep invariant from the D-Wed maps. However, all the other methods have some distance errors between the watermarked maps and the original maps. This is due to that embedding watermark does not change any coordinate and decryption is also completely reversible. Therefore, the proposed method is a lossless watermarking and decryption that has good imperceptibility and no loss of data accuracy.

D. WATERMARK ROBUSTNESS

As a robust watermarking algorithm, the watermark should survive such acceptable operations as rotating, scaling, translation (RST), projective transformation, and so on. In the experimental design, “GEOMARK” is selected as the copyright information, and then we set up the following attacks. In the RST experiment, we rotated a vector map from 36° to 360° at the gap of 36°, scaled the map by 0.1 to 10 times, and translated the map from 10% to 100% of the data width at the gap of 10%, respectively. In the projection transformation attacks, eight map projections are selected. They come from four categories: the equal area projection, the conformal projection, the equidistant projection, and the compromise projection. Then we project the vector map to different projected coordinate systems. The threshold of BER is set to 0.3, which means if the BER is less than 0.3, the corresponding watermark extracting succeed, that is, the extracted copyright information is “GEOMARK”, otherwise failed. The BER results are shown in Table 5 to Table 8. The symbol “√” means the watermarking algorithm is robust against the attack while “×” means the algorithm is not robust.

TABLE 5. The results of rotation attacks with different methods.

Data	Rotation angle / degree	Proposed	Jang’s	Li’s	Yang’s
Yonkers	36	0.00 √	0.00 √	0.00 √	0.52 ×
	72	0.00 √	0.00 √	0.00 √	0.51 ×
	108	0.00 √	0.00 √	0.00 √	0.52 ×
LianYun Gang	144	0.00 √	0.00 √	0.00 √	0.56 ×
	180	0.00 √	0.00 √	0.00 √	0.00 √
	216	0.00 √	0.00 √	0.00 √	0.52 ×
DaFeng	252	0.00 √	0.00 √	0.00 √	0.51 ×
	288	0.00 √	0.00 √	0.00 √	0.52 ×
	324	0.00 √	0.00 √	0.00 √	0.56 ×
	360	0.00 √	0.00 √	0.00 √	0.00 √

TABLE 6. The results of scaling attacks with different methods.

Data	Scale factor	Proposed	Jang's	Li's	Yang's
Yonkers	0.1	0.00 √	0.00 √	0.00 √	0.51 ×
	0.2	0.00 √	0.00 √	0.00 √	0.47 ×
	0.4	0.00 √	0.00 √	0.00 √	0.42 ×
LianYun Gang	0.6	0.00 √	0.00 √	0.00 √	0.51 ×
	0.8	0.00 √	0.00 √	0.00 √	0.53 ×
	2	0.00 √	0.00 √	0.00 √	0.46 ×
DaFeng	4	0.00 √	0.00 √	0.00 √	0.50 ×
	6	0.00 √	0.00 √	0.00 √	0.50 ×
	8	0.00 √	0.00 √	0.00 √	0.52 ×
	10	0.00 √	0.00 √	0.00 √	0.54 ×

TABLE 7. The results of translation attacks with different methods.

Data	Translation / (%)	Proposed	Jang's	Li's	Yang's
Yonkers	10	0.00 √	0.00 √	0.00 √	0.00 √
	20	0.00 √	0.00 √	0.00 √	0.00 √
	30	0.00 √	0.00 √	0.00 √	0.00 √
LianYun Gang	40	0.00 √	0.00 √	0.00 √	0.00 √
	50	0.00 √	0.00 √	0.00 √	0.00 √
	60	0.00 √	0.00 √	0.00 √	0.00 √
DaFeng	70	0.00 √	0.00 √	0.00 √	0.00 √
	80	0.00 √	0.00 √	0.00 √	0.00 √
	90	0.00 √	0.00 √	0.00 √	0.00 √
	100	0.00 √	0.00 √	0.00 √	0.00 √

TABLE 8. The results of projection transformation attacks with different methods.

Data	Map Projection	Proposed	Jang's	Li's	Yang's
Yonkers	Equal-Area Cylindrical	0.00 √	0.62 ×	0.27 √	0.45 ×
	Gall Orthographic	0.00 √	0.63 ×	0.36 ×	0.52 ×
	Mercator	0.00 √	0.62 ×	0.34 ×	0.45 ×
LianYun Gang	Lambert Conformal Conic	0.00 √	0.58 ×	0.27 √	0.53 ×
	Equidistant Azimuthal	0.00 √	0.62 ×	0.35 ×	0.49 ×
	Equidistant Cylindrical	0.00 √	0.57 ×	0.36 ×	0.48 ×
DaFeng	Robinson	0.00 √	0.62 ×	0.29 √	0.53 ×
	Winkel I	0.00 √	0.58 ×	0.34 ×	0.51 ×

Overall, all BER values of the methods vary with different types of attacks from Table 5 to Table 8 except the proposed method. The BER values of the proposed method always maintain 0.00, which means that it can resist RST and projection transformation attacks completely. As for Jang's method and Li's method, the BER values keep 0.00 from Table 5 to Table 7, but all of the values for Jang's method and more than half of the values for Li's method are larger than the threshold of BER, 0.3, in Table 8. This shows Jang's method and Li's method are robust to RST attacks but cannot resist projection transformation attacks. For Yang's method, Only the BER values in Table 7 are 0.00, and most of them in other tables are larger than the threshold. Therefore, Yang's method is not robust among the four methods, and it can only resist translation attacks. In the proposed method, the RST and projection transformation attacks only change the coordinate of the vertices of the polylines and polygons, and they do not change the constructed feature invariants. Thus, the watermark can be extracted directly in the arbitrarily rotated, scaled, translated and projected map using the proposed method. Especially, the proposed algorithm is able to resist the attack of projection transformation, while its counterparts cannot. Thus, the robustness of the proposed CEW scheme is high enough to resist the attacks of rotation, scaling, translation and projection transformation comparing with other watermarking algorithms.

TABLE 9. Distribution index σ of each dataset.

Experimental Dataset	σ	
	original data	encrypted data
Yonkers	796.96	35.49
LianYunGang	53.41	8.46
DaFeng	736.88	21.90

E. SECURITY OF ENCRYPTION

The encryption algorithm with good security that can make the encrypted data have nothing to do with the original data, and any useful information cannot be obtained from the encrypted data. The security of the proposed CEW scheme will be verified by three aspects: the spatial distribution, the data difference and the key sensitivity.

1) THE SPATIAL DISTRIBUTION

The spatial distribution measures the aggregation degree of spatial objects in a certain area. Generally, the spatial distribution of a vector map is large because of uneven distribution. Here the distribution index σ is used to evaluate the spatial distribution. In the calculation of σ, the vector layer is tiled into grid with the size of $m \times n$, and there are vertices $[N_1, N_2, N_3 \ldots, N_{m\times n}]$ in each tile. $\overline{N}$ represents the mean value of $[N_1, N_2, N_3 \ldots, N_{m\times n}]$. σ is expressed as the standard deviation of those vertices:

$$\sigma = \sqrt{\frac{1}{m \times n - 1} \sum_{i=1}^{m\times n} (N_i - \bar{N})^2} \tag{10}$$

By comparing the difference of σ, the uniformity of vertex distribution is verified.

According to Equation (10), the distribution index σ of the datasets are listed in Table 9.

From Table 9, it is obvious that σ of the encrypted data has been largely reduced after data encryption. The magnitude of change rate of σ ranges from $736.88/21.90 \approx 33.65$ to

TABLE 10. The data difference *D* between original and encrypted for each dataset.

Experimental Dataset	MAE
Yonkers	0.6784°
LianYunGang	11523*m*
DaFeng	12149*m*

$53.41/8.46 \approx 6.31$. The dramatic reduction on σ means that the uneven spatial distribution is changed to the distribution with a quite even level. Obviously, the original vector data is distributed irregularly, making it easier for cracker to infer the spatial characteristics. The spatial distribution becomes more even after encryption. Its characteristics then become harder to deduce. Thus, the reduction on σ demonstrates the algorithm security to keep from cracking through the features of spatial distribution.

2) THE DATA DIFFERENCE

The data difference represents the mean error between two different data. The larger difference means more difference between the encrypted maps and the original maps. Here MAE (Mean Absolute Error) is used to measure the difference between the original vertices and the encrypted vertices, which is calculated by:

$$MAE = \frac{1}{N_v}\sum_{i=1}^{N_v}\left(\left|x_i' - x_i\right| + \left|y_i' - y_i\right|\right) \tag{11}$$

where the variables are with the same definition of variables as in Equation (9). From the definition of MAE, it measures how much error between the encrypted map and the original one. Larger MAE means that each encrypted vertex is far away from the original one, increasing the difficulty to deduce the original vertex position from the encrypted vertex. Thus, the larger MAE implies better encryption effect for encryption.

As for the security experiments, MAE is calculated based on the proposed CEW scheme. According to Equation (11), MAE of the encrypted vertices and the original vertices for each dataset is listed in Table 10.

From Table 10, it is apparent that MAE values between the encrypted maps and the original maps are all very large, either in geographical coordinate system or in projected coordinate system. For example, the MAE of Yonkers are 0.6784°, which is a large error in geographical coordinate system. These results means that each vertex has been moved with a wide range in the coordinate system. Thus, the difference demonstrates the security of the proposed CEW scheme.

3) KEY SENSITIVITY

A good cryptographic system needs to have a dependency on the secret keys to be able to resist brute-force attacks [42]. In other words, the encrypted maps cannot be recovered correctly when the key used is slightly changed in the decryption process. If similar results have been got when using similar keys in encryption or decryption, the cryptographic system is easy to crack by using differential cryptanalysis. Hence, key sensitivity is a vital feature of security. In this section, the key sensitivity is measured from two aspects. One is the differences between the decrypted data with slightly different keys and the original key are evaluated. Another one is the differences among the encrypted data with slightly different keys are evaluated.

TABLE 11. MAE of decrypted data using different secret keys.

Key	MAE		
	Yonkers	LianYunGang	DaFeng
key_2	0.5612°	11467*m*	10442*m*
key_3	0.6198°	14369*m*	9852*m*

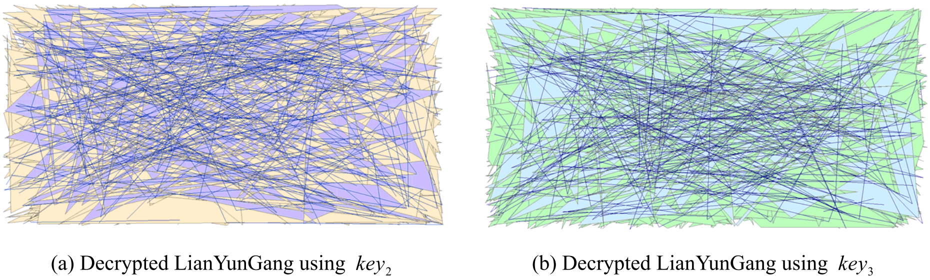
As mentioned in Section 4.1, the initial encryption key is $key_1 = abcdef1234567890$. Two different keys are set as $key_2 = abcdef1234567891$ and $key_3 = abcdef1234567889$ to evaluate the key sensitivity of the proposed encryption method. key_2 and key_3 are only one digit difference with the initial encryption key key_1, and these two different slightly keys are used to replace key_1 in the decryption of LianYunGang, which is already encrypted by key_1. The results are shown in Figure 6. Figure 6(a) shows the decrypted map using the secret key key_2, and Figure 6(b) shows the decrypted map using the secret key key_3. In addition, Table 11 shows the MAE distortion between the decrypted data and the original one with two different keys to measure the differences objectively.

It can be clearly seen from Figure 6 that these decrypted maps have not been recovered to the original ones successfully, and no useful information is obtained from them. Moreover, from Table 11, all the values of MAE are very large. In particular, the MAE value between the decrypted LianYunGang map and the original one is 14369*m*. So, the maps decrypted with slightly incorrect keys are totally different from the maps decrypted with correct key. Therefore, the proposed method is sensitive to the slight change in the keys for the decryption.

To further evaluate the key sensitivity, key_2 and key_3 are used to replace key_1 to encrypt LianYunGang map. The encrypted maps with two keys are shown in Figure 7. Figure 7(a) shows the encrypted LianYunGang map with key_2, and Figure 7(b) shows the encrypted LianYunGang map with key_3. As shown in Figure 7, there are prominent visual differences among the encrypted data using different secret keys.

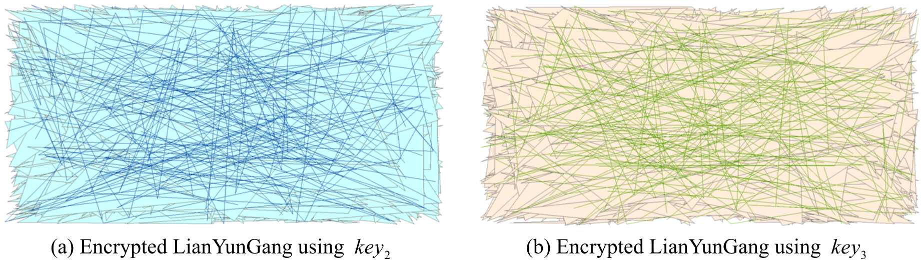
In addition, the values of MAE between the encrypted map and the original map using different secret keys for LianYunGang data are listed in Table 12 to measure the differences objectively. It can be observed from Table 12 that the values of MAE of these encrypted data are very high, although the three keys are almost the same with a tiny difference of the secret key. It indicates that the sensitivity of the secret key in encryption is also verified.

The sensitivity of key has been verified in the above experiments. Hence, the integrity of key must be ensured in



(a) Decrypted LianYunGang using key_2 (b) Decrypted LianYunGang using key_3

FIGURE 6. Decrypted data using different secret keys.



(a) Encrypted LianYunGang using key_2 (b) Encrypted LianYunGang using key_3

FIGURE 7. Encrypted data using different secret keys.

TABLE 12. MAE of encrypted data using different secret keys.

Encrypted data comparing with	MAE
key_1 and key_2	13049m
key_2 and key_3	14050m
key_3 and key_1	14001m

encryption or decryption, otherwise, the result will be wrong. However, the distribution or management of secret keys are not the focus of this paper and can be solved by classical cryptographic technology. For example, the digital signature can be used to check the key integrity and asymmetric encryption can be used to distribute the key, which will not be detailed explained here.

V. DISCUSSION

A. THE EFFECTIVENESS OF THE APPROACH

1) COMMUTATIVITY

The proposed method has full use of the inherent features of vector map for constructing two feature invariants, which are further utilized for design of the CEW algorithm. Both simulation and experiments are presented to verify the commutativity of the proposed CEW method in Figure 3, Table 2 and Table 3. Therefore, the proposed method has solved the aforementioned challenges of CEW technique for vector map.

2) LOSSLESS DECRYPTION AND LOSSLESS WATERMARKING

The second feature of the proposed method lies in lossless decryption [43] and lossless watermarking. The lossless decryption allows the original data to be perfectly recovered from the encrypted data without any attacks. The lossless watermarking means that embedding watermark will not damage the coordinates of the data. The lossless decryption and the lossless watermarking are achieved by the proposed CEW method, as shown in Figure 5 and Table 4. Thus, both the security and the precision are guaranteed for vector map. This feature shows the advantages of the proposed CEW method in security protection for vector map, especially for the high-precision vector map.

B. APPLICABILITY OF THE APPROACH

It is worth noting that the proposed encryption method in this paper is not fixed on the perceptual stream cipher, because the feature invariants proposed are independent of the coordinate values of the objects. Actually, the proposed CEW algorithm can be used for arbitrary encryption methods based on coordinate values, including the traditional AES/DES encryption methods, without changing the extracted feature invariants.

There are still some points to be studied further in the future for a wide use of the proposed CEW method in practice, including the security of the method with different encryption algorithms, the working efficiency of the method, the integration of the method with the features of vector map, and the commutativity between the method and other the watermarking algorithm.

C. THE CHARACTERISTICS OF THE PROPOSED WATERMARKING METHOD

1) WATERMARK CAPACITY

The watermark capacity refers to the maximum amount of watermark that can be embedded in the host data. The watermark capacity directly determines whether the copyright information can be successfully embedded. The proposed method selects polyline and polygon objects as the embedding target, and every two adjacent objects can be embedded one bit. There, the watermark capacity of the proposed method is half of the number of the objects in

TABLE 13. The experimental results of watermark length.

Data	Attacks	Watermark length					
		64	100	200	300	400	500
Yonkers	Rotate 36°	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Scale 0.1 times	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Translate 10%	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Mercator Projection	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
LianYunGang	Rotate 72°	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Scale 0.8 times	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Translate 50%	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Robinson Projection	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
DaFeng	Rotate 288°	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Scale 10 times	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Translate 100%	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √
	Winkel I Projection	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √	0.00 √

vector, and the unit is bit. In the experiments, we set the watermark length as 100bits, which means that if a vector data needs to be embedded a watermark, it must contain at least 200 objects. That is to say, for those data whose number of objects is far less than 200, the proposed method is not applicable. Thus, there is still room in improving the watermark capacity of the proposed method. For example, spreading the characteristics between objects, or finding the characteristics of a single object to embed a watermark bit losslessly are some feasible approaches. This will be further studied in our future research.

2) WATERMARK LENGTH

The watermark used in the paper is a binary sequence with the length of 100 bits. This section will discuss the relationship between the watermark length and the robustness in the proposed method. Table 13 shows the experimental results of the proposed method with different watermark lengths. In the experiment, three experimental maps are tested with RST and projection transformation attacks. Overall, as the watermark length increases from 64 to 500, the BER value is always 0. Thus, varying the watermark length does not affect the robustness result in the above experiments. This is because the proposed CEW method is totally resistant to the above attacks. Therefore, no matter the watermark length is shorter or longer, the watermark can be extracted without error and BER keeps 0. However, considering the mapping relationship between watermark and copyright, the watermark length of 100 can provides 2^{100} possibilities of different copyright, which is sufficient in normal cases.

VI. CONCLUSION

In this paper, a novel commutativity encryption watermarking method is proposed for secure vector map. The proposed CEW method consists of a lossless watermarking algorithm and a coordinate encryption algorithm, which are proposed based on feature invariants and perceptual stream cipher, respectively. The key is to propose two feature invariants, i.e., the sum of inner angles and the storage direction of two adjacent objects, for vector maps according to their inherent characteristics. In this way, the proposed CEW method can achieve the commutativity of watermark embedding and encryption, and also the commutativity of watermark extraction and decryption. Compared with the existing methods, the proposed method has better robustness and security. In addition, the quality of the protected vector map is not degraded by embedding watermark or decryption by using the proposed CEW method. The experimental results have been given to verify the improved performance of the proposed method. It is noted that the proposed method is not limited to the stream cipher based encryption method. Instead, it can be applied to other encryption methods with coordinate values without changing the constructed feature invariants. Therefore, the proposed CEW method offers good application values of security protection for vector maps.

REFERENCES

[1] F. Peng, Z.-X. Lin, X. Zhang, and M. Long, "Reversible data hiding in encrypted 2D vector graphics based on reversible mapping model for real numbers," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 9, pp. 2400–2411, Sep. 2019.

[2] A. Abubahia and M. Cocea, "Advancements in GIS map copyright protection schemes—A critical review," *Multimedia Tools Appl.*, vol. 76, no. 10, pp. 12205–12231, May 2017.

[3] C. Zhu, "Research progresses in digital watermarking and encryption control for geographical data," *Acta Geodaetica Cartograph. Sinica*, vol. 46, no. 10, pp. 1609–1619, Oct. 2017.

[4] M. Cancellaro, F. Battisti, M. Carli, G. Boato, F. G. B. De Natale, and A. Neri, "A joint digital watermarking and encryption method," *Proc. SPIE*, vol. 6819, Mar. 2008, Art. no. 68191C.

[5] K. Datta and I. S. Gupta, "Partial encryption and watermarking scheme for audio files with controlled degradation of quality," *Multimedia Tools Appl.*, vol. 64, no. 3, pp. 649–669, Jun. 2013.

[6] C. Yang, C. Zhu, Y. Wang, T. Rui, J. Zhu, and K. Ding, "A robust watermarking algorithm for vector geographic data based on qim and matching detection," *Multimedia Tools Appl.*, vol. 79, nos. 41–42, pp. 30709–30733, Aug. 2020.

[7] Y. Wang, C. Yang, N. Ren, C. Zhu, T. Rui, and D. Wang, "An adaptive watermark detection algorithm for vector geographic data," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 1, pp. 323–343, 2020.

[8] C. Li, S. Li, G. Chen, and W. A. Halang, "Cryptanalysis of an image encryption scheme based on a compound chaotic sequence," *Image Vis. Comput.*, vol. 27, no. 8, pp. 1035–1039, Jul. 2009.

[9] J. Li, Y. Feng, and X. Yang, "An improved image encryption scheme based on line maps," in *Proc. 5th Int. Conf. Inf. Assurance Secur. (IAS)*, vol. 1, Aug. 2009, pp. 605–608.

[10] G. N. Pham, S. T. Ngo, A. N. Bui, D. V. Tran, S. H. Lee, and K. R. Kwon, "Vector map random encryption algorithm based on multi-scale simplification and Gaussian distribution," *Appl. Sci.*, vol. 9, no. 22, p. 4889, Nov. 2019.

[11] J. Lafaye, J. Béguec, D. Gross-Amblard, and A. Ruas, "Blind and squaring-resistant watermarking of vectorial building layers," *Geoinformatica*, vol. 16, no. 2, pp. 245–279, Apr. 2012.

[12] C. Yang, C. Zhu, and D. Tao, "A blind watermarking algorithm for vector geo-spatial data based on coordinate mapping," *J. Image Graph.*, vol. 15, no. 4, pp. 2–5, Apr. 2010.

[13] X. Niu, C. Shao, and X. Wang, "A survey of digital vector map watermarking," *Int. J. Innov. Comput. Inf. Control*, vol. 2, no. 6, pp. 1301–1316, Dec. 2006.

[14] S.-H. Lee and K.-R. Kwon, "Vector watermarking scheme for GIS vector map management," *Multimedia Tools Appl.*, vol. 63, no. 3, pp. 757–790, Apr. 2013.

[15] X. Xi, X. Zhang, Y. Sun, X. Jiang, and Q. Xin, "Topology-preserving and geometric feature-correction watermarking of vector maps," *IEEE Access*, vol. 8, pp. 33428–33441, 2020.

[16] F. Peng, Q. Long, Z.-X. Lin, and M. Long, "A reversible watermarking for authenticating 2D CAD engineering graphics based on iterative embedding and virtual coordinates," *Multimedia Tools Appl.*, vol. 78, no. 19, pp. 26885–26905, Oct. 2019.

[17] Y. Qiu, H. Duan, J. Sun, and H. Gu, "Rich-information reversible watermarking scheme of vector maps," *Multimedia Tools Appl.*, vol. 78, no. 17, pp. 24955–24977, Sep. 2019.

[18] X. Xi, X. Zhang, W. Liang, Q. Xin, and P. Zhang, "Dual zero-watermarking scheme for two-dimensional vector map based on delaunay triangle mesh and singular value decomposition," *Appl. Sci.*, vol. 9, no. 4, p. 642, Feb. 2019.

[19] R. Schmitz, S. Li, C. Grecos, and X. Zhang, "Towards more robust commutative watermarking-encryption of images," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2013, pp. 283–286.

[20] F. Battisti, M. Cancellaro, G. Boato, M. Carli, and A. Neri, "Joint watermarking and encryption of color images in the fibonacci-Haar domain," *EURASIP J. Adv. Signal Process.*, vol. 2009, no. 1, pp. 938515-1–938515-13, Sep. 2009.

[21] M. Li, D. Xiao, Y. Zhu, Y. Zhang, and L. Sun, "Commutative fragile zero-watermarking and encryption for image integrity protection," *Multimedia Tools Appl.*, vol. 78, no. 16, pp. 22727–22742, Aug. 2019.

[22] J. Dittmann, A. Lang, M. Steinebach, and S. Katzenbeisser, "ECRYPT-European network of excellence in cryptology. Aspekte der Sicherheit von Mediendaten," in *Proc. Sicherheit, Sicherheit-Schutz und Zuverlässigkeit*, 2005, pp. 189–192.

[23] L. Jiang, Z. Xu, and Y. Xu, "Commutative encryption and watermarking based on orthogonal decomposition," *Multimedia Tools Appl.*, vol. 70, no. 3, pp. 1617–1635, Jun. 2014.

[24] Z. Xu, L. Xiong, and L. Jiang. (Dec. 2015). *Review of Commutative Encryption and Watermarking*. [Online]. Available: http://www.paper.edu.cn/releasepaper/content/4668462

[25] B. Guan, D. Xu, and Q. Li, "An efficient commutative encryption and data hiding scheme for HEVC video," *IEEE Access*, vol. 8, pp. 60232–60245, 2020.

[26] X. Zhang, "Commutative reversible data hiding and encryption," *Secur. Commun. Netw.*, vol. 6, no. 11, pp. 1396–1403, Nov. 2013.

[27] S. Lian, Z. Liu, R. Zhen, and H. Wang, "Commutative watermarking and encryption for media data," *Opt. Eng.*, vol. 45, no. 8, pp. 080510-1–080510-3, Aug. 2006.

[28] M. Cancellaro, F. Battisti, M. Carli, G. Boato, F. G. B. De Natale, and A. Neri, "A commutative digital image watermarking and encryption method in the tree structured Haar transform domain," *Signal Process., Image Commun.*, vol. 26, no. 1, pp. 1–12, Jan. 2011.

[29] S. Lian, Z. Liu, Z. Ren, and H. Wang, "Commutative encryption and watermarking in video compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 6, pp. 774–778, Jun. 2007.

[30] L. Jiang, "The identical operands commutative encryption and watermarking based on homomorphism," *Multimedia Tools Appl.*, vol. 77, no. 23, pp. 30575–30594, Dec. 2018.

[31] S. Lian, "Quasi-commutative watermarking and encryption for secure media content distribution," *Multimedia Tools Appl.*, vol. 43, no. 1, pp. 91–107, May 2009.

[32] G. Boato, N. Conci, V. Conotter, F. G. B. De Natale, and C. Fontanari, "Multimedia asymmetric watermarking and encryption," *Electron. Lett.*, vol. 44, no. 9, pp. 601–603, Apr. 2008.

[33] A. Boho, G. Van Wallendael, A. Dooms, J. De Cock, G. Braeckman, P. Schelkens, B. Preneel, and R. Van de Walle, "End-to-end security for video distribution: The combination of encryption, watermarking, and video adaptation," *IEEE Signal Process. Mag.*, vol. 30, no. 2, pp. 97–107, Mar. 2013.

[34] R. Schmitz, S. Li, C. Grecos, and X. Zhang, "Towards robust invariant commutative watermarking-encryption based on image histograms," *Int. J. Multimedia Data Eng. Manage.*, vol. 5, no. 4, pp. 36–52, Oct. 2014.

[35] H. M. Heys, "Statistical cipher feedback of stream ciphers," *Comput. J.*, vol. 60, no. 12, pp. 1839–1851, Dec. 2017.

[36] W. Si and C. Ding, "A simple stream cipher with proven properties," *Cryptogr. Commun.*, vol. 4, no. 2, pp. 79–104, Jun. 2012.

[37] Y. Tsunoo, T. Saito, H. Kubo, and T. Suzaki, "A distinguishing attack on a fast software-implemented RC4-like stream cipher," *IEEE Trans. Inf. Theory*, vol. 53, no. 9, pp. 3250–3255, Sep. 2007.

[38] R. L. Rivest, *The RC4 Encryption Algorithm*. Bedford, MA, USA: RSA Data Security, 1992.

[39] A. K. Bhateja and M. Din, "ANN based distinguishing attack on RC4 stream cipher," in *Proc. 7th Int. Conf. Bio-Inspired Comput., Theories Appl. (BIC-TA)*, vol. 202, 2013, pp. 101–109.

[40] Z. Peng, M. Yue, X. Wu, and Y. Peng, "Blind watermarking scheme for polylines in vector geo-spatial data," *Multimedia Tools Appl.*, vol. 74, no. 24, pp. 11721–11739, Dec. 2015.

[41] B.-J. Jang, S.-H. Lee, E.-J. Lee, S. Lim, and K.-R. Kwon, "A cryptomarking method for secure vector map," *Multimedia Tools Appl.*, vol. 76, no. 14, pp. 16011–16044, Jul. 2017.

[42] H. Qiu, G. Memmi, and H. Noura, "An efficient secure storage scheme based on information fragmentation," in *Proc. IEEE 4th Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)*, Jun. 2017, pp. 108–113.

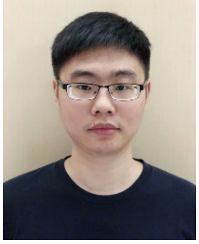
[43] J. Zhou, J. Li, and X. Di, "A novel lossless medical image encryption scheme based on game theory with optimized ROI parameters and hidden ROI position," *IEEE Access*, vol. 8, pp. 122210–122228, 2020.

[44] A.-B. Li and A.-X. Zhu, "Copyright authentication of digital vector maps based on spatial autocorrelation indices," *Earth Sci. Informat.*, vol. 12, no. 4, pp. 629–639, Dec. 2019.

NA REN received the B.S. and M.S. degrees from Shaanxi Normal University, in 2004 and 2008, respectively, and the Ph.D. degree from Nanjing Normal University, in 2011. She is currently an Associate Professor with the School of Geography, Nanjing Normal University. Her research interests include digital watermarking, data security, and GIS.

CHANGQING ZHU received the B.S. degree in mathematics from the Zhengzhou Institute of Surveying and Mapping, in 1982, the M.S. degree in mathematics from Zhengzhou University, in 1990, and the Ph.D. degree in cartography from the Zhengzhou Institute of Surveying and Mapping, in 1997. He was a Professor with the Zhengzhou Institute of Surveying and Mapping. He is currently a Professor with the School of Geography, Nanjing Normal University. His research interests include digital watermarking, data security, and GIS.

DEYU TONG (Member, IEEE) received the Ph.D. degree in cartography and geographic information system from Nanjing Normal University, Nanjing, China, in 2018. He is currently a Lecturer with the Nanjing University of Finance and Economics, Nanjing. His research interest includes security issues on geographic data.



WEITONG CHEN received the B.S. degree in geography and the M.S. degree in marine geography from Nanjing Normal University, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree in cartography and geography information system with the School of Geography. His research interests include digital watermarking, geo-data security, and GIS.

QIFEI ZHOU received the B.S. degree in geographic information system from Nanjing Normal University, in 2015, where he is currently pursuing the Ph.D. degree in cartography and geographic information system with the School of Geography. His research interests include digital watermarking and transparent encryption.