

A High Capacity Reversible Watermarking Based on Linear Cellular Automata for 2D Vector Maps

Saleh AL-ardhi ¹, Hassan Aly ² and Vijey Thayananthan ^{1 +}

¹ King Abdulaziz University

² Majmaah University

Abstract. Because of the need to publish digital maps of several programs, for example, traffic information in a secure format to ensure the preservation of copyrights. The 2-D vector map application encounters security problems of copyright protection that constrain the utilization of vector data map in numerous situations. According to recent researches, it has been concluded that for 2-dimensional vector maps, the technically demanding tasks of existing reversible watermarking plans are getting rotation, scaling and translation (RST) invariance property Attacks, embedding distortion control and high capacity. Specifically, to achieve security performance of the copyright information, we propose the first reversible watermarking scheme by is used linear cellular automata for 2D vector maps that are considered as watermarking to building up new coordinates framework and relative coordinates are built as cover information to ensure High Capacity watermarking as well as gives great invisibility. According to analysis and experiments outcomes, our proposed technique gives very efficiently in, invisibility, and reversibility.

Keywords: reversible watermarking, high capacity, LCA, vector map, copyright protection.

1. Introduction

The digital map has the Developments in computer technologies and Geographic Information Systems (GIS), i.e. computer-based systems for managing and displaying locational data related to positions on Earth's surface, increased the amount of digital vector maps that are available on the World Wide Web. Vector map data is very important as a value because when you need collection, processing and storage of such data depend on expensive equipment resources (e.g. electronic distance meters, unmanned aerial vehicles), global navigation systems (e.g. GPS), Geographic Information System (e.g. LBS) because highly accurate, they document attribute and topological information through the use of geometrical shapes, and are small in size compared with GIS raster maps such as satellite image.

Thus, this data generally cannot be freely used without permission from the owners of vector maps. However, due to fast advancement of multimedia network and digital technology, there has been much consideration paid to the data security of the information translated over the network, there are various security mechanisms involved in this case. Watermarking technology among these mechanisms has been greatly utilized for the secret communication, content authentication, and copyright protection, and so forth [1]. As indicated by application purpose, to design watermark scheme in Fig (1) faces the challenges. The first challenges are Robustness: the technique can avoid against for data processing attacks aimed at either removing or distorting it by illegal users, but only the data owners ought to be able to extract the watermark. We can classify Robustness watermarking methods into fragile watermarking and robust watermarking. The fragile watermarking is used to validate the integrity and robust watermarking is used to protect ownership that must robustly survive the attacks. The second is Fidelity: after the insertion of the watermark should not degrade the quality of the map. The third is Invisibility: after the embedding watermark should be

⁺ Corresponding author. Tel.: + 966164041484.
E-mail address: s_ardhi@hotmail.com.

perceptually invisible to data users to avoid discovered the secret information. The fourth is Blindness: not need the original map and the original watermark at hand when you watermark extraction process. [2].



Fig. 1: Basic reversible map watermarking scheme.

The 2D vector maps are a kind of vector graphics having their own qualities. Elements of its graphics are different and its entities are independent, unlike the multimedia data whose relative positions are fixed. Furthermore, 2D vector maps have topology information and geometric information are similarly imperative; In addition, there is little redundancy in the 2D vector maps. This makes watermark embedding more difficult as compared to video, audio images etc. For copyright protection, there have been different works done using the method of digital watermarking [3-7] and also the content authenticity [8-14] of 2-Dimensional vectors contrasted with various other options available, for example, digital signature and encryption method. The watermarking can give extra security for the important data [15] and save you time for the process of retransmission [11].

In many watermarking cases, original maps coordinates are distorted goal to insert hidden data in an irreversible way; it means that they can't access the original information. Be that as it may, because of high-precision nature of 2 Dimensional vector maps, any adjustment to vector map is generally undesired. So, this shows the requirement for the reversible watermarking strategies, since some of the traditional procedures of watermarking brought the stable distortion to the host, so such traditional procedures are not supportive for 2-Dimensional vector maps, which focuses on strict prerequisites on the data integrity and authenticity due to their applications in the fields of product manufacturing, legal authentication, and the medical diagnosis. To conquer these problems, reversible watermarking is taken into account to address the problems of such applications. This is also known as lossless data hiding, and it can thoroughly recover the original information after the process of extraction. Note that it does not have any negative effect on availability of the original carrier, An important principle of the reversible watermarking relies upon the reversible algebraic or geometric operations, for instance, histogram shifting, difference expansion, the space feature, and the lossless compression.

There are various multimedia watermarking techniques are available that depend on the cellular automata (CA). The complex 2D cellular automata were represented as a strategy for watermarking both images in . The cellular automata are a discrete model of calculation that could be utilized to figure out any computable operation. CA is broadly utilized as a part of various applications, for example, pattern recognition, random number generation, and music in art. The use of CA in the territory of computerized picture handling incorporates the processes of encryption, compression the authentication, scrambling, and image enhancement. Because the contrasts between vector graphics and raster images vary from a data element to the data correlation, the cellular automata methods can't be directly utilized for 2D vector maps.

For this, we propose the first watermarking schema using reversible linear cellular automata method that is based on standards of cellular automata, so that it can be used for 2D vector maps. It has been proposed as a solution to the problem of copyright protection of 2D vector maps. To protect intellectual property, as reversible watermarking can be used to recover original map after the extraction process, so it is possible to make new reversible watermarking plans for the 2D vector maps. In this research paper, first of all, we put forward the linear cellular automata to enhance the cellular automata strategy and utilize it to build a reversible watermarking plan for 2D vector maps. Some major advantages in our proposed include:

- 1) The capacity of data can be too large, about $2S(S \geq 1)$ bit/vertex, where S represents the number of watermarking bits that each for each new coordinate carriers.
- 2) Another advantage is that the invisibility, reversibility and the computational complexity are great.
- 3) It provides various transform patterns designs verifying the Linear cellular automata bases. Hence, our technique can recuperate weak point that has just a single transform plane in the DCT and DFT domain strategies. Utilizing the LCA with different cellular automata bases and the rule number, we can be able to access the channels embedding data.
- 4) Due to the complexity of LCA, it is difficult to guess or find the position embedded watermark.
- 5) It effectively robust the 2-dimensional vector map against the geometric attacks such as rotation, scaling, translation.

We arrange the remaining part of this paper as follows: in Section 2 explain Reversible Linear Cellular automata in Section3 explain our reversible Robust watermarking scheme in detail. We present our experimental results and analysis of the algorithm in Section 4. Conclusions are summarized in Section 5.

2. Reversible Linear Cellular automata

This paper utilizes Cellular Automata that is finite state machine made by the finite number of cells also known as memory units. In one case, the cell can be zero or one. From a certain initial state, the status of network cells changes from the generation to the next, or remains as the same, based on a transition function (or rule) that depends on the state of the selected cell and the situations of its neighbors in the previous generation. The transfer function is applied at the same time to each cell in the network. The Linear cellular automata have an imperative property: The evolution of LCA can be interpreted by methods of the matrix algebraic tools, the following configuration of the linear cellular automata can be created by increasing the present setup by the fixed matrix under modulo-2 addition such

$$(C^{t+1})^T = M_n.(C^t)^T \pmod{2}$$

Here, M_n is the characteristic matrix and $(C^t)^T$ is outcome of the of $C^t.M_n$ is a circulant matrix[61].

The reversibility theorem is explained as: Let M_n is the characteristic matrix of CA An . As discussed in the above, it is a n th order pentadiagonal matrix that has non-zero coefficients equal to one. It derives the following outcomes: Proposition 1. Determinant of M_n satisfies:

$$|M_n| \pmod{2} = \begin{cases} 1, & \text{if } n = 5k \text{ or } n = 5k + 1, \text{ with } k \in \mathbb{N} \\ 0, & \text{otherwise} \end{cases}$$

An example for LCA : Let us consider the LCA A5 and its inverse with $n = 5$ and $0 \leq t \leq 1$. If $n = 5k$, then $k = 1$

$$(C^{t+1})^T = (M_5 = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}).(C^t)^T = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \pmod{2}$$

The Calculation is:

$$\begin{aligned} \text{Row}_1 &= (1 * 0) + (1 * 0) + (1 * 1) + (0 * 1) + (0 * 0) = 1 \pmod{2} = 1 \\ \text{Row}_2 &= (1 * 0) + (1 * 0) + (1 * 1) + (1 * 1) + (0 * 0) = 2 \pmod{2} = 0 \\ \text{Row}_3 &= (1 * 0) + (1 * 0) + (1 * 1) + (1 * 1) + (1 * 0) = 2 \pmod{2} = 0 \\ \text{Row}_4 &= (0 * 0) + (1 * 0) + (1 * 1) + (1 * 1) + (1 * 0) = 2 \pmod{2} = 0 \\ \text{Row}_5 &= (0 * 0) + (0 * 0) + (1 * 1) + (1 * 1) + (1 * 0) = 2 \pmod{2} = 0 \end{aligned}$$

The Result is:

$$(C^{t+1})^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The inverse cellular automaton of An is the following:

$$(C^t)^T = M_5^{-1} \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix} . (C^{t+1})^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \pmod{2}$$

The Calculation is:

$$\begin{aligned} \text{Row}_1 &= (0 * 1) + (0 * 0) + (1 * 0) + (1 * 0) + (0 * 0) = 0 \pmod{2} = 0 \\ \text{Row}_2 &= (0 * 1) + (0 * 0) + (0 * 0) + (1 * 0) + (1 * 0) = 0 \pmod{2} = 0 \\ \text{Row}_3 &= (1 * 1) + (0 * 0) + (1 * 0) + (0 * 0) + (1 * 0) = 1 \pmod{2} = 1 \\ \text{Row}_4 &= (1 * 1) + (1 * 0) + (0 * 0) + (0 * 0) + (0 * 0) = 1 \pmod{2} = 1 \\ \text{Row}_5 &= (0 * 1) + (1 * 0) + (1 * 0) + (0 * 0) + (0 * 0) = 1 \pmod{2} = 0 \end{aligned}$$

The Result is:

$$(C^t)^T = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \text{ is equal original } (C^t)^T = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

3. The Proposed Watermarking Scheme

Based on the section [2] we will be using to reversible Linear cellular automata for apply watermarking schema, we introduce a watermarking scheme in two stages. First, how watermarks were embedded into a vector map for each spatial features, as shown in Fig. 1. Second, how we extract the watermarks and recover the original vector map, as shown in Fig. 2.

With a specific end goal to enhance the quality of the vector map that is watermarked, we embedding watermarks into the LSD planes such as least-significant-digit planes of its coordinates. At this point, it is important to note that minimum and maximum coordinates of every axis don't convey watermark bits.

3.1. Watermark generation procedure

Now, considering that original watermark $W = \{w_1, w_2, \dots, w_n\}$ is in the binary sequence with length of n , so the generation of this watermark proceeds and Scramble the elements of W by utilizing the technique explained in :

Step 1: Given an initial value Mn is the characteristic matrix of the cellular automaton An , it is an n th order pentadiagonal matrix whose non-zero coefficients are equal to 1. The next configuration of the LCA can be generated by multiplying the original watermark by a fixed matrix under *modulo - 2* addition, that is:

$$l = (w \times Mn) \% 2 \quad (1)$$

Step 2: Given an embedding strength S , the watermark sequence $w^* = \{w_1^*, w_2^*, \dots\}$ is coded from L according to

$$w_i^* = l_{(i-1)xs+1} \times 2^{s-1} + l_{(i-1)xs+2} \times 2^{s-2} + \dots + l_{ixs} \times 2^0 \quad (2)$$

For example, if $S = 1$, we can get $w_1^* = l_1, w_2^* = l_2, \dots, w_n^* = l_n$, if $S = 4$, we can get $w_1^* = 4(l_1, l_2, l_3, l_4)$, if $S = 8$, we can get $w_1^* = 8(l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8) \dots$

Here, w^* is utilized as watermark for embedding

3.2. Watermark embedded procedure

Scan the vertices of M and select two reference vertices v_{f1} and v_{f2} . For the purpose of security, we select v_{f1} and v_{f2} under control of the private key and embed the scrambled watermark sequence W^* into vertices utilizing the technique explained in. The steps given below shows how the watermark embedding into each vertices; given watermark information $W \in \{0, 1, \dots, 2^s - 1\}$, host information v_x , embedding strength S , embedding position P , the process of embedding given below

Step 1: As per embedding position P , we regulate position of the decimal point of v_x , and carry out an S -bit right-shift to v_x

$$v_x = (v_x \times 10^p) \quad (3)$$

Step 2: Attain integer part I and the decimal part F of v_x , i.e.....,

$$v_x = (I \times F) \quad (4)$$

$$I = LTrim(v_x), F = RTrim(v_x) \quad (5)$$

At this point, $LTrim(v_x)$ presents a function to trim, integer part of X and $RTrim(v_x)$ presents function to trim, such as fractional part of the v_x .

Step 3: Perform the S-bit left-shift on F , and embed the watermark W as per:

$$F^W = (F + W) \quad (6)$$

Step 4: For any watermark W , embed W into mantissa part of the j-th element v_x , of v_x can be attained by means of bit substitution, then position of the watermarked vertex $V_{x,y}(v_x, v_y)$.

$$F^W = ((F + W) \times Mn) = 1\%2 \quad (7)$$

$$F^W = ((\text{mantissa}(52 \text{ bit}) + (s = 4)) = 56 \text{ bit} \times M56) = 1\%2 \quad (8)$$

Here, Mn is characteristic matrix of the CA An. This is a nth order pentadiagonal matrix and its non-zero coefficients are equal to one. Using the mantissa part of j-th element = 52 bit, the embedding strength $S=4$

Step 5: moving forward, create field in (Feature attributes) for F^W . It can be utilized to store its last bit and its removed unit 52 bit. It is mentioned in the Fig. 2:

$$F^W = (\text{mantissa}(56 \text{ bit})) - (\text{last bit} = 4) = (\text{mantissa}(52 \text{ bit})) \quad (9)$$

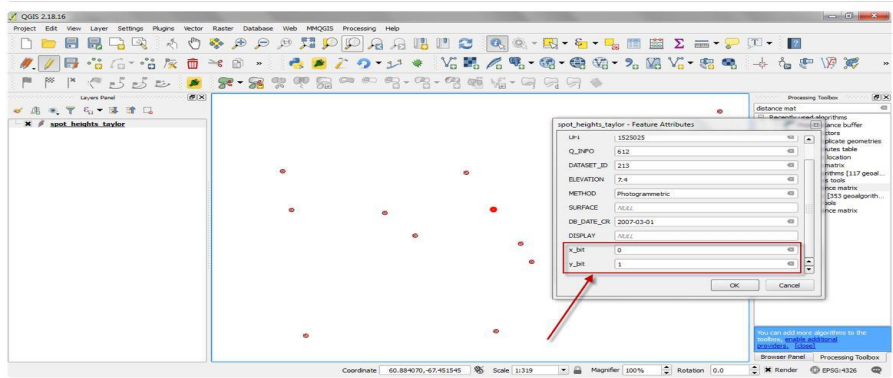


Fig. 2: Feature attributes in properties of map.

Step 6: In this step, we use the LCA transform to LSD planes of its coordinates:

$$F^{W'} = ((F^W) \times Mn) = 1\%2$$

Step 7: In this step, we calculate host data for the v_x after the watermark embedding process:

$$v_x^w = ((I + F^{W'}) \times 10^{-p}) \quad (10)$$

In Fig.3 shown the Watermark embedding principle. This implements shift tasks in binary, decimal and arithmetic and extends original information and them implants a watermark to its integer part; from that point onward, the reverse shift in decimal is utilized to get the watermarked information.

3.3. Watermark extraction procedure

(1) Scan the vertices of M' and select two reference vertices v_{f1}' and v_{f2}' ; under the control of

The private key k .

(2) Extract the watermark sequence w^* from the embedded vertices and recover the original content of M' utilizing the technique explained in:

Step 1: Obtain watermarked host data such as v_x^w .

Step 2: As per embedding position P , regulate position of decimal point of the host data, such as:

$$v_x^w = (v_x \times 10^p) \quad (11)$$

Step 3: Now, calculate integer part I and decimal part F^W of v_x^w ,

$$v_x^w = (I + F^{W'}) \quad (12)$$

$$I = LTrim(v_x^w), F^{W'} = RTrim(v_x^w) \quad (13)$$

Step 4: In this step, we use the Reversible LCA transform to LSD planes of its coordinates:

$$F^W = ((F^{W'}) \times \text{Reversible M52}) = 1\%2$$

Step 5: insert in the last bit the value Store into field in (Feature attributes) for:

$$F = ((F^W (\text{mantissa}(52 \text{ bit})) + (s = 4)) = 56 \text{ bit} \times \text{Reversible M56}) = 1\%2 \quad (14)$$

Then:

$$w^* = ((\text{mantissa}(52 \text{ bit}) - (56 \text{ bit})) = \text{Last 4 bit is watermarking}) \quad (15)$$

Last bit is w^* after the watermark extraction process.

Step 5: Recover the host data such as:

$$v_x^w = (I + F) \quad (16)$$

$$v_x^w = ((I + F) \times 10^{-p}) \quad (17)$$

As per above Equation, it can be conclude that $\text{sign}(v_x) = \text{sign}(v_x^w)$.

$$w^* = w \quad (18)$$

(3) Descramble the elements of W^* , to acquire the original embedded watermark sequence W , utilizing the technique.

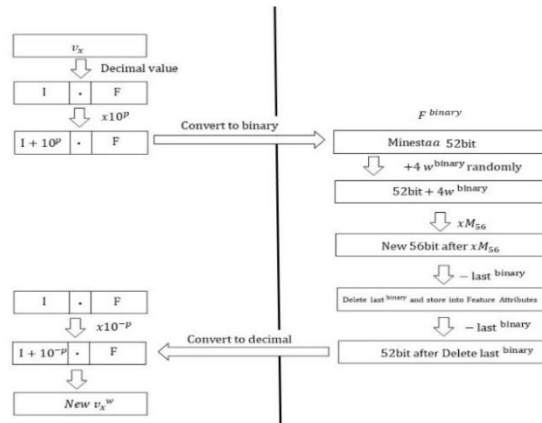


Fig. 3: The process of watermark embedding.

In Fig.4 explain the principle of watermark extraction. It implements the shift operations in decimal arithmetic and in binary, and expands the original data and they extracted a watermark to the integer part of it; after that, a reverse shift operation in decimal arithmetic is used to obtain the watermarked data:

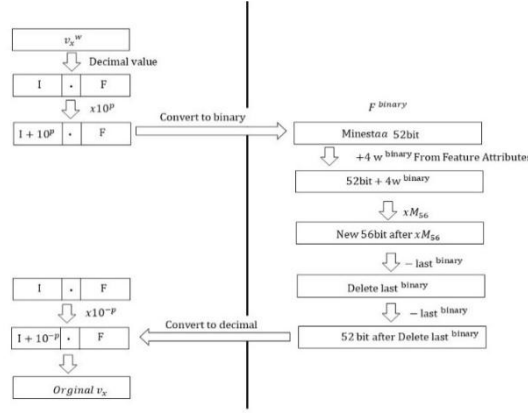
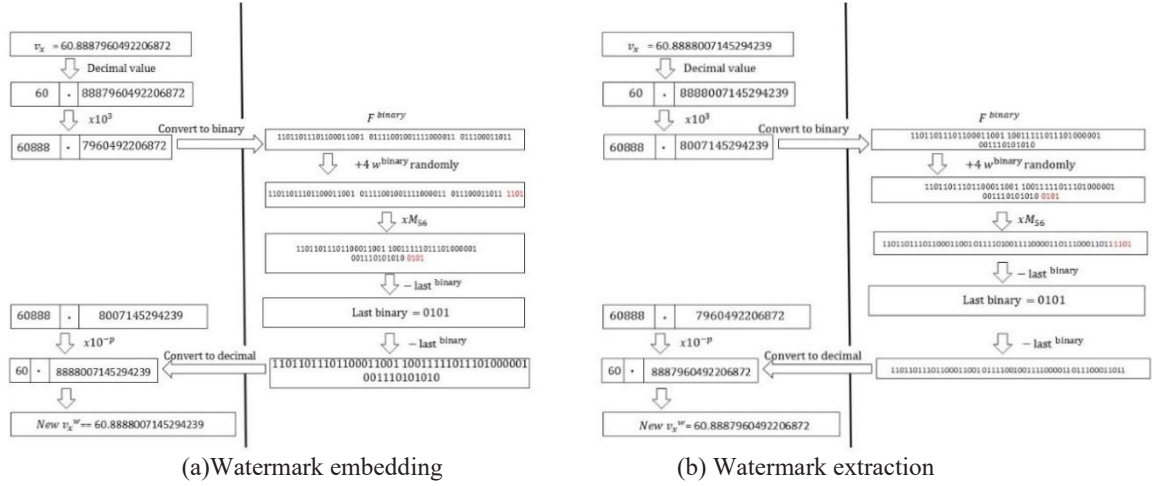


Fig. 4: The process of watermark extracting.

For example, 1: when adding $S = 4$, given $v_x = 60.8887960492206872$, $p = 3$, $w = 1101$, $M_n=56$ the parameters can be calculated as follows. The output for $v_i^w = 60.8887980035400850$, and the differences between the original coordinates and the recovered ones $= 1.96 \times 10^{-6}$



4. Results and Analysis

The shape file format (.shp) of Environmental Systems Research Institute, Inc. (ESRI) is exploited for the scheme. In our experiments, we ran on a PC with CPU 2.3 GHz, RAM 16G, Win10 Professional, QGIS Version 3.0, python language a simple shape file (.shp) format 2D vector map “Coastline map of Taylor Rookery” is used as the original map to test the performance of our scheme. The former vector map contains polyline features with 18 features and 4279 vertices. Because the geometric data structures of a polygon feature and polyline feature are identical, we show the result for polyline features in details. We show the result for point features in details. In the first test case, we demonstrate the quality of our watermarking scheme. For evaluating the embedded vector map subjective quality, we compared the original 2D vector map and the watermarked ones. From Fig. 5a and 5b, we know the invisibility of the watermarked vector map. The average distortion $d(M, M')$ and the maximum distortion $Maxd(M, M')$ were calculated is exploited to measure the watermarked vector map objective quality by Eq. (19).

$$d(M, M') = \frac{1}{N_V} \sum_{i=0}^{N_V} |v_i - v_i^w| \quad (19)$$

$$Maxd(M, M') = \text{Max}(|v_i - v_i^w|), (i = 1, 2, \dots, N_V) \quad (20)$$

where v_i and v_i^w are the corresponding vertices in the original vector map M and the recovered vector map M' , and N_V denotes the total number of vertices in the vector map M . The vector maps were embedded by the watermark embedding procedure technique proposed in Section 2, and then recovered by the

corresponding watermark extraction and data recovery technique. Figure 4 shows the recovered version of Fig. 3. It can be seen from Fig. 4 that the perceived quality is acceptable.

From Fig. 5a and 5b in our experiments, we can obtain that the $Maxd$ values and the d values of the vector maps are all less than 10^{-6} m. In general, the storage precision of the 2D vector map coordinates is obtained 0.1 mm. That is, as long as the differences are very small between the original and watermark coordinates. The watermark scheme can be considered reversible. Thus, precision requirements can be met for most situations. If we want to increase the quality of the vector map of the watermark, we must reduce the distortion by increasing the allowable precision tolerance of the map. It causes reduces the total number of insertion units, thus reducing the capacity of the watermark bits.

5. Conclusions

In our research, we proposed a new watermarking technique was introduced in the 2D digital vector map by takes advantage of linear cellular automata with complex behavior to achieve a high invisibility. Compared with a watermark based on the conventional cellular technique, this is the first watermarking support reversible by is used linear cellular automata for 2D vector maps, no need for the original map in the proposed method to extract, that is, the 4-bit watermark information at least can be included into each vertex, which produces larger watermark capacity. Meanwhile, the experimental results and analyses also indicate that the proposed reversible watermark has a good reversibility, imperceptibility. in future we will do more experiment in point and polygon

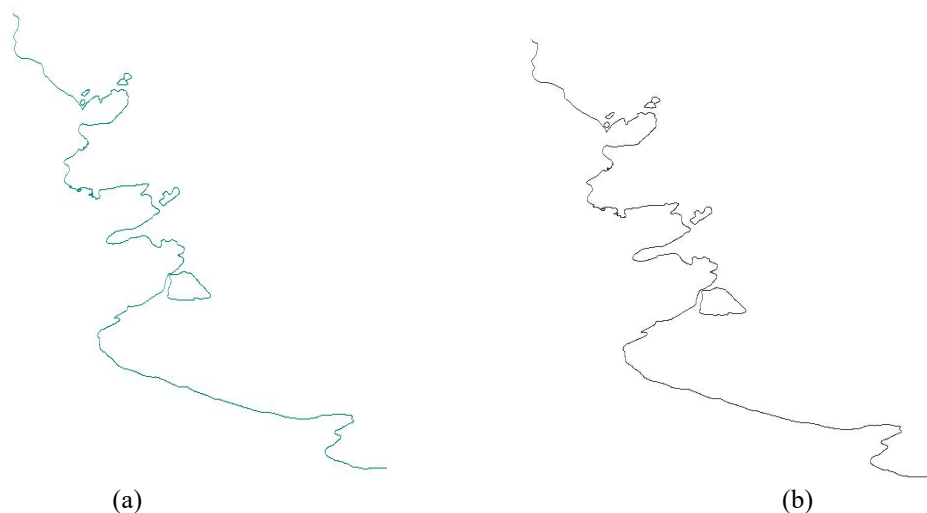


Fig. 5: Experiments on the road map: (a) original vector map; (b) watermarked vector map.

6. References

- [1] C. Lopez, "Watermarking of digital geospatial datasets: a review of technical, legal and copyright issues," *Int. J. Geogr. Inf. Sci.*, vol. 16, (6), pp. 589-607, 2002
- [2] X. Niu, C. Shao and X. Wang, "A survey of digital vector map watermarking," *International Journal of Innovative Computing, Information and Control*, vol. 2, (6), pp. 1301-1316, 2006.
- [3] A. Abubahia and M. Cocca, "A clustering approach for protecting GIS vector data," in *International Conference on Advanced Information Systems Engineering*, 2015, pp. 133-147.
- [4] Z. Peng et al, "Blind watermarking scheme for polylines in vector geo-spatial data," *Multimedia Tools Appl.*, vol. 74, (24), pp. 11721-11739, 2015.
- [5] J. Lafaye et al, "Blind and squaring-resistant watermarking of vectorial building layers," *Geoinformatica*, vol. 16, (2), pp. 245-279, 2012.
- [6] S. Lee and K. Kwon, "Vector watermarking scheme for GIS vector map management," *Multimedia Tools Appl.*, vol. 63, (3), pp. 757-790, 2013.

- [7] S. K. Muttoo and V. Kumar, "Watermarking digital vector map using graph theoretic approach," *Annals of GIS*, vol. 18, (2), pp. 135-146, 2012.
- [8] S. N. Neyman, Y. H. Wijaya and B. Sitohang, "A new scheme to hide the data integrity marker on vector maps using a feature-based fragile watermarking algorithm," in *Data and Software Engineering (ICODSE)*, 2014 International Conference on, 2014, pp. 1-6.
- [9] C. Shao, X. Wang and X. Xu, "Security issues of vector maps and a reversible authentication scheme," in *Doctoral Forum of China*, 2005, pp. 326-331.
- [10] N. Wang, "Reversible fragile watermarking for locating tampered Polylines/Polygons in 2D vector maps," *International Journal of Digital Crime and Forensics (IJDCE)*, vol. 8, (1), pp. 1-25, 2016.
- [11] N. Wang and C. Men, "Reversible fragile watermarking for 2-D vector map authentication with localization," *Comput. -Aided Des.*, vol. 44, (4), pp. 320-330, 2012.
- [12] N. Wang and C. Men, "Reversible fragile watermarking for locating tampered blocks in 2D vector maps," *Multimedia Tools Appl*, vol. 67, (3), pp. 709-739, 2013.
- [13] M. Yue, Z. Peng and Y. Peng, "A fragile watermarking scheme for modification type characterization in 2d vector maps," in *Asia-Pacific Web Conference*, 2014, pp. 129-140.
- [14] L. Zheng and F. You, "A fragile digital watermark used to verify the integrity of vector map," in *EBusiness and Information System Security*, 2009. *EBISS'09*. International Conference on, 2009, pp. 14.
- [15] Z. Xia et al, "Steganalysis of LSB matching using differences between nonadjacent pixels," *Multimedia Tools Appl*, vol. 75, (4), pp. 1947-1962, 2016.