Fragile Watermarking based on Linear Cellular Automata using Manhattan Distances for 2D Vector Map

Saleh Al-Ardhi*[1], Vijey Thayananthan*[2], Abdullah Basuhail[3]
Faculty of Computing and Information Technology (FCIT)
King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—There has been a growing demand for publishing maps in secure digital format, since this ensures the integrity of data. This had lead us to put forward a method of detecting and locating modification data that is extremely accurate and simultaneously guarantees that the exact original content is recovered. More precisely, this method relies on a fragile watermarking algorithm that is developed in accordance with a frequency manner and, for every spatial feature, it can embed hidden data in 2D vector maps. The current paper proposes a frequency data-hiding scheme, which will be examined in accordance with Linear Cellular Automata Transform using Manhattan distances. Various invertible integer mappings are applied in order to find out the Manhattan distances from coordinates. To begin with, the original map is transformed into LCA, after which the watermark insertion process is carried out to transform the coefficient of the transformation result frequency into LSB. Lastly, a watermarked map is created by applying the inverse LCA transform, meaning that a LCA-transformed map is produced. Findings indicate that the suggested method is effective since in terms of invisibility and the capacity to allow for modifications. The methods also allow the detection of modification data, the addition and removal of some features, and enable the exact original content from the 2D vector map to be included.**

***Keyword*—Reversible watermarking; fragile watermarking; linear cellular automata; Manhattan distances; vector map**

I. Introduction

The digital map is very a very accurate, automated procedure, which is highly beneficial. It also has lossless scaling in comparison to paper maps [1]. Other advantages of it include easy storage and simple distribution, as well as easy data manipulation. It generates a higher need for map producers to ensure that publication maps are subject to high security services as a means of protecting the integrity of the map. It also requires distortions resulting from these security services to be eliminated from the map. Fragile watermarking is deemed to be a highly effective technique for carrying out authentication and integrity for vector map verification. In comparison to conventional methods (such as digital signatures), fragile watermarking allows for both the detection and locate of any modifications made to the original content.

The integrity of the data refers to the authenticity of the data, that is, whether the data has been manipulated with a common or malicious data processing. A digital watermarking technology is used to embed hidden information in a digital map in order to indicate the author of the content [2-7], and authenticate the integrity of the content [8-12]. To remove the distortions introduced by authentication and tamper detection ability, fragile watermarking for digital maps can be included in frequency watermarking [13]. The watermark in frequency watermarking is different from the space domain present in transform-domain embedding methods, since the watermark is not inserted by adapting the vertices' coordinates, but rather by changing their transform coefficients. Fragile watermarks are typically applied to safeguard integrity and authenticity of data content. When this is modified, the watermark becomes damaged, and jeopardizes the data integrity, meaning it cannot be ensured that the data is authentic A fragile watermark thus takes advantage of reversible watermarking techniques in order to insert the authentication data. And this not only enables the location of malicious attacks, but allows for the original content to be recovered [14]. In the present research paper, we will propose a reversible method for conducting fragile watermarking in vector maps. The proposed technique will be developed in accordance with Linear Cellular Automata Transform [15], with Manhattan Distances being employed. This is a new approach that is yet to be used for the first time in research. The Manhattan distances used here will be those between the adjacent vertices, and they will be applied to cover data, with the aim of creating a distance- based scheme [16] that can allow for the coordinates in which the watermark is embedded to be located. Linear Cellular Automata Transform will be used to enhance the ability and invisibility of the final, resulting maps. This suggested method will be able to accurately locate and detect any features that have been tampered with once data has been manipulated. It will also be possible to recover the original vector map in its exact form by extracting the hidden data in cases when there has been no attack.

The purpose of this research is to create a successful method to ensure that the integrity of geospatial data can be protected in a more effective and efficient manner than is currently possible and then has been proposed by previous research into the topic. The research results are expected to heighten confidence in the development of digital maps created in computerized environments. In order to enhance content authentication in vector maps, the present research paper puts forward a new and innovative approach to vector map watermarking. This method will rely on the LCA transformation algorithm. It is important to note that the cellular automata transform (CAT) algorithm has been used frequently in the past in cases of multimedia watermarking [17-18], however there is yet to be any research that has used vector maps as the embedded media. The primary advantages of our new approach are that there is a high degree of reversibility and invisibility, as well as low computational complexity, and insertion outcomes [19] that are of advanced quality. Another key point here, is that the approach proposed in the current paper provides various features that enable and promote data origin authentication, primarily resulting from the scrambling technique [20]. What's more, as the approach can only be used on a single transform plane, it is very unique in comparison to current frequency domain watermarking methods. Additionally, it provides multi-frequency domains that can allow for successful DW. The rest of the paper will be organized in the following way: Section 2 will focus on exploring Linear Cellular Automata. Section 3 will discuss the reversible fragile watermarking scheme that is used in our technique in great depth. Section 4 is where we will present the experimental findings and algorithm analysis. Finally, section five will present summarized conclusions.

II. Linear Cellular Automata

Linear cellular Automata Transform can be described as an important algorithm that is applied to represent a certain dynamic given in a discrete time and a frequency domain. Cells are organized to create a regular lattice structure, and it is imperative that each of these cells possesses a finite number of states. As a whole, LCAT is applied as a means of working out the discrete transformation in a fast and efficient manner, and can often be useful in lowering the number of complexities. In general, the following equation can be used to explain the LCAT formulation (1) [21].

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

When M_n indicates the following local transition matrix, if $n = 5k$, then:

$$M_n = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \ldots\ldots\ldots\ldots 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \ldots\ldots\ldots\ldots 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \ldots\ldots\ldots\ldots 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \ldots\ldots\ldots\ldots 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \ldots\ldots\ldots\ldots 0 & 0 & 0 \\ & & & & \ldots.. & & \\ & & & & \ldots. & & \\ 0 & 0 & 0 & 0 & 0 \ldots\ldots\ldots\ldots 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \ldots\ldots\ldots\ldots 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \ldots\ldots\ldots\ldots 1 & 1 & 1 \end{pmatrix}$$

If we use M_n to represent the transition matrix for the cellular automaton An. As previously stated, the matrix is an nth order and is pentadiagonal in nature. This matrix has non-zero coefficients that equate to 1. In cases where M_n is representative of the pentadiagonal matrices, $(C^t)^T$ is representative of the linear matrix tht is made up of a number of random parts. However, it is important to point out that the inverse formulation for LCAT is expressed using the following equation (2).

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

The formula for the transition matrix of the inverse cellular automaton of An is as follows: if $n = 5k$ then:

$$M_n^{-1} = \begin{pmatrix} M_5^{-1} & B & B & \cdots & B \\ B^T & M_5^{-1} & B & \ddots & \vdots \\ B^T & A^T & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & M_5^{-1} & B \\ B^T & \cdots & B^T & B^T & M_5^{-1} \end{pmatrix}$$

Where

$$M_5^{-1} \begin{vmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{vmatrix} (\text{mod } 2)\ ,\ \text{B} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The size of the transition matrix starts from 5 elements as defined in the following equation (3):

$$|M_n\,|mod2 = \begin{Bmatrix} 1, \ \ if\ n = 5k\ \ or\ \ n = 5k+1, with\ k \in N \\ 0, other\ wise \end{Bmatrix} \qquad (3)$$

A. Linear Cellular Automata Transform

In order to improve the copyright protection performance features of vector maps, the present research paper suggests that fragile digital watermarking domain transformation is used to serve as proof of copyright, and this watermarking is achieved through LCAT data transformation algorithm. The CAT algorithm is commonly employed throughout the multimedia watermarking field [22,23], but has never been inserted into a vector map in embedded media form before. In the current paper, the watermark is inserted as a form of copyright indicator that is visible on the vector map and this is achieved by using a transformation domain on the vertices' coordinates. The procedure for inserting the watermark is a process in which the coefficient of the transformation result frequency has to be implemented into the vector map data. The map coordinates have to be adapted into a Linear Cellular Automata Transform ($LCAT$) in order to transform the vector map into a domain frequency signal. The key principle underpinning the process is that the coordinate v_{x1} on the original map is transformable by applying LCAT. The following equation can be used to explain the LCAT(4):

$$T(M) = \sum_{n=0}^{N-1} M_n.v_{x1}\ (mod\ 2) \qquad (4)$$

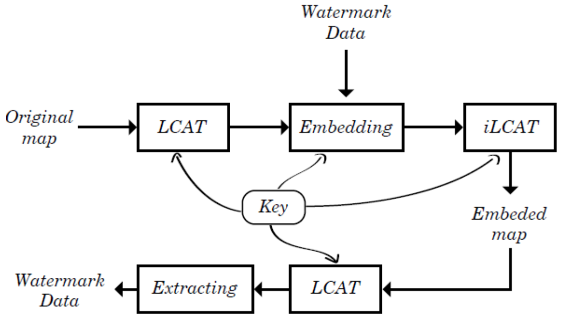


Fig. 1. The Flow Chart of the Linear Cellular Automata Transform Algorithm.

$T(M)$ represents the domain transformation of the original map, whilst v_{x1} represents the digital media value of the original map. Furthermore, N represents the number of vertices that must be modified into a frequency domain. M_n is the transition matrix of Linear cellular automaton.

The key principle of the method is that the $v_{x1}{}'$ coordinate of the original map can be modified using the LCAT, after which the encrypted watermark can be implemented. The following formula explains this method of embedding the watermark (5):

$$v_{x1}{}'' = v_{x1}{}' + \alpha W \quad (5)$$

α here represents the embedding parameter, with W the n representing the watermark bit. As can be seen in the watermark (5) formulation previously outlined, the higher the α, the greater the changes will be on the vector map file. However, the strength of the watermark resistance will be much greater. The α value used in the present research will be as high as 3 bit, which is very much an acceptable level for vector map changes This value also shows a high level of resistance. The numerous stages of the linear cellular automata transform algorithm can be seen in Fig. 1.

In the meantime, the inverse formula for LCAT can be seen in the following equation (6):

$$iT(M) = \sum_{n=0}^{N-1} M_n^{-1}.\, v_{x1}{}''\,(mod\ 2) \quad (6)$$

$iT(M)$ represents the inverse domain transformation value for the original map, whilst $v_{x1}{}''$ represents the original map's transform digital media value.

III. Proposed Watermarking Scheme

Throughout section three, the proposed watermarking method will be introduced in two primary stages. Firstly, we will discuss the method used to insert watermarks into the vector MAP for every spatial feature (see Fig. 2). Secondly, we will discuss the method used to extract the watermarks and for the recovery of the original vector map (see Fig. 3).

A. Watermark Embedded Procedure

A polyline feature refers to a structured group of vertices that come together to create a single or multiple line segments, This happens in such a way that precisely two segments will share the endpoint of every segment (otherwise known as a vertex) v(x, y). When the endpoints are the same, then a polyline will be closed (this is known as a polygon). As a whole, the coordinates of D vector map vertices are in the form of floating-point numbers

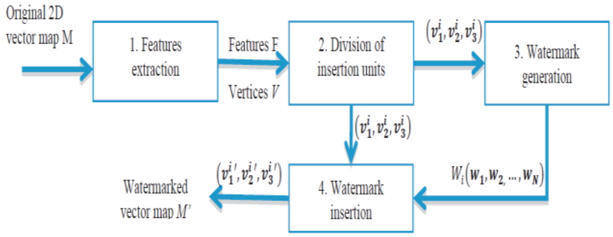


Fig. 2. Watermark Insertion Procedure.

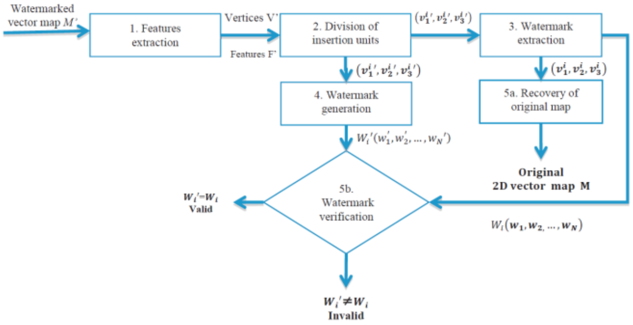

Watermarked vector map M'
1. Features extraction
Vertices V'
Features F'
2. Division of insertion units
$(v_1^{i\prime}, v_2^{i\prime}, v_3^{i\prime})$
3. Watermark extraction
$(v_1^{i\prime}, v_2^{i\prime}, v_3^{i\prime})$
(v_1^i, v_2^i, v_3^i)
4. Watermark generation
5a. Recovery of original map
$W_i'(w_1', w_2', \ldots, w_N')$
Original 2D vector map M
5b. Watermark verification
$W_i'=W_i$ Valid
$W_i(w_1, w_2, \ldots, w_N)$
$W_i' \neq W_i$ Invalid

Fig. 3. Watermark Verification Procedure.

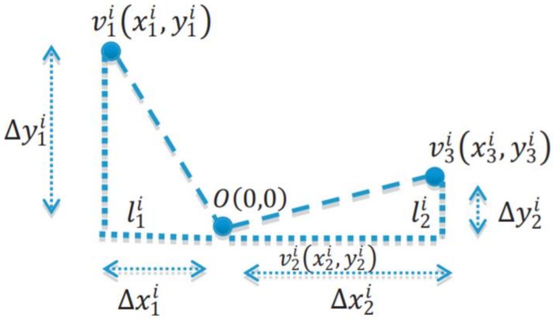


Fig. 4. An Insertion unit.

If we allow $dmax$ to serve as the maximum number of digits following the decimal point, (x, y) can represent the original coordinates. In order to allow for the restoration of the original coordinates (x, y), equation (1) can be used to extract the integer coordinates (xi, yi).

$$(x^i, y^i) = \lfloor (x, y) \times 10^d \rfloor, d \ \leq d_{max} \quad (7)$$

The three consecutive vertices for every feature can be grouped as an insertion unit. The structure of an insertion unit can be seen in Fig. 4. The following formula represents N units in F:

$$(v_1{}^i, v_2{}^i, v_3{}^i)\ \{(x_1{}^i, y_1{}^i), (x_2{}^i, y_2{}^i), (x_3{}^i, y_3{}^i)\},$$
$$i\ \varepsilon\ \{1,2, \ldots \ldots, N\}.$$

2 The following formula shows the relative coordinates $(\Delta x\ , \Delta y)$ for every insertion unit, with $v_2{}^i$ serves as the center point 0(0,0)), resulting from equation (8).

$$\begin{cases}\Delta x_1^i = x_1^i - x_2^i \\ \Delta y_1^i = y_1^i - y_2^i\end{cases} \begin{cases}\Delta x_2^i = x_3^i - x_2^i \\ \Delta y_2^i = y_3^i - y_2^i\end{cases} \quad (8)$$

$$\begin{cases} l_1^i = |\Delta x_1^i| + |\Delta y_1^i| \\ l_2^i = |\Delta x_2^i| + |\Delta y_2^i| \end{cases} \begin{cases} r_1^i = \left\lfloor \frac{|\Delta x_1^i| - |\Delta y_1^i|}{2} \right\rfloor \\ r_2^i = \left\lfloor \frac{|\Delta x_2^i| - |\Delta y_2^i|}{2} \right\rfloor \end{cases} \quad (9)$$

As demonstrated in both Fig. 4 and equation (9), the Manhattan distances $l_1^i\ and\ l_2^i$ refer to the distances between the center point and the two nearest neighbouring vertices, respectively. Moreover, r represents the integer mean difference value, and this will not be impacted by modifications at all during the embedding process. In a given pair, the following formula shown in equation (10) can be applied to work out the difference (di) and integer-mean (mi) of the two Manhattan distances.

$$\begin{cases} d^i = l_1^i - l_2^i \\ m^i = \left\lfloor \frac{l_1^i + l_2^i}{2} \right\rfloor \end{cases} \tag{10}$$

The primary purpose of calculating the Manhattan distances is to allow for the location of the implemented watermark bits to be detected through the modification of the difference di (see equation (5) and Fig. 5).

$w_k \in W(k = 1,2, \dots, NW)$

This is based on the premise that the insertion unit fulfills the two conditions required of embedded data. W represents the embedded data, and may be in the form of cryptographic hash value for the host vector map, the purpose of which is to verify the integrity of data or to highlight any secret data.

1) The coordinates acquired by working out the difference di between the vertex points are subsequently converted into LCAT.

2) The method presented in [20] can be employed in order to encrypt the factors relating to W^* and to identify any patterns in data $W^* = \{w_i^* \mid w_i^* \in \{0, 1\}, i = 0, 1, \dots, l-1\}$.

3) If one is to assume that there is a double floating-point number in the form of a 16-digit coordinate value, and that this is inserted in decimal fraction format, and if W^* is embedded into the two final successive digits, then they will have relatively little impact on accuracy. Precision. What's more, the embedded value is not in line with the w_i^* instead lying somewhere between 0 to 99. If D is presumed to be the integer resulting from the two digits, then the following formula is to be applied:

$$W^* = \begin{cases} if\ w_i^*\ is\ 0\ \ then\ D \leq 50\ \ and\ saved\ at\ the\ positions; \\ \qquad\qquad w_i^* = 1, otherwise \end{cases} \tag{11}$$

4) When the $iLCAT$ is used after the watermark has been embedded, the frequency domain vector map can be restored to the original form.

5) The third and fourth steps can be repeated K number of times in high capacity situations, with blind watermarking and LCAT able to be employed as a means of extract the watermark.

To enable, secret communication, two criteria must be fulfilled in the proposed method:

Criterion 1. To guarantee the ability to recover the original 2D vector map, each watermarked vertice has to remain in the same region as the original vertices. To clarify, this means that the relative coordinates of original vertices (Δx, Δy) have to possess the exact same numbers as the corresponding watermarked vertices (Δx', Δy').

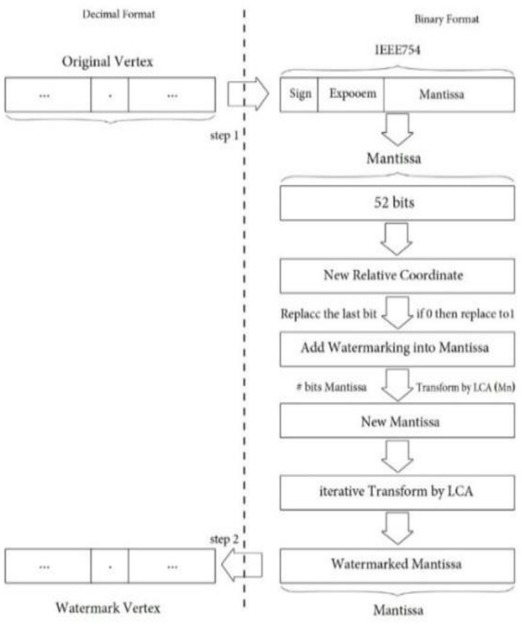


Fig. 5. The Binary Transform Process for Cover Data.

Condition 2. In order to guarantee the high quality of the watermarked 2D vector map, it is crucial to restrict as much as possible the distortion causing by the embedding process. This must be restricted by the map's precision tolerance τ [15]. Euclidean distances can be employed here to work out the extent of the distortions (Eq. (12)).

$$\sqrt{\left(x_1^{i\prime} - x_1^i\right)^2}\ , \sqrt{\left(x_3^{i\prime} - x_3^i\right)^2 \leq \tau} \tag{12}$$

If the embedding of wk is complete, then the modified Manhattan distances $l_1^{i\prime}\ and\ l_2^{i\prime}$ can be acquired from the d' and mi by using equation (13). Subsequently, the vertices coordinate of the watermarked unit $({v_1}^{i\prime}, {v_2}^{i\prime}, {v_3}^{i\prime})$ may be worked out using the formula: $l_1^{i\prime}\ and\ l_2^{i\prime}$ and by subsequently employing equations (14) (15). This is chosen based on the values of $\Delta x_1^i\ \ and\ \Delta y_1^i$

$$\begin{cases} l_1^i = m_i + \left\lfloor \frac{d_i' + 1}{2} \right\rfloor \\ l_2^{i\prime} = m_i - \left\lfloor \frac{d_{i'}}{2} \right\rfloor \end{cases} \tag{13}$$

$$\begin{cases} x \\ y \end{cases} \quad \Delta x_1^{i\prime} = \begin{cases} r_1^i + \left\lfloor \frac{I_1^{i\prime}+1}{2} \right\rfloor, if \Delta x_1^i \geq 0 \\ -r_1^i - \left\lfloor \frac{I_1^{i\prime}+1}{2} \right\rfloor, if \Delta x_1^i\ < 0 \end{cases} \quad \Delta x_1^i \begin{cases} r_1^i + \left\lfloor \frac{I_1^{i\prime}+1}{2} \right\rfloor, if \Delta x_2^i \geq 0 \\ -r_1^i - \left\lfloor \frac{I_1^{i\prime}+1}{2} \right\rfloor, if \Delta x_2^i\ < 0 \end{cases} \tag{14}$$

$$\begin{matrix} x_1^{i\prime} = x_2^i + \Delta x_1^{i\prime} \\ y_1^{i\prime} = y_2^i + \Delta y_1^{i\prime} \end{matrix} \begin{cases} x_1^{i\prime} = x_2^i + \Delta x_2^{i\prime} \\ y_1^{i\prime} = y_2^i + \Delta y_2^{i\prime} \end{cases} \tag{15}$$

B. Watermark Verification Procedure

There are three fundamental processes underpinning the watermark verification procedure. These are the extraction of watermarks, the verification of such watermarks, and recovering of the original map. When the watermarked vector map has been obtained M', the following formula can be used to extract the watermark:

1) The watermark can be extracted from the watermarked map M' into $\{F_1{}' , F_2{}', F_\mathrm{D}{}' \}$ feature groups. These groups are in the form of integers. In every group $F_\mathrm{i}{}'$, three consecutive vertices must be separated to form a watermarked insertion unit.

$$(v_1{}^{i\prime}, v_2{}^{i\prime}, v_3{}^{i\prime}) \{(x_1{}^{i\prime}, y_1{}^{i\prime}), (x_2{}^{i\prime}, y_2{}^{i\prime}), (x_3{}^{i\prime}, y_3{}^{i\prime})\}, i \ \{1,2, \dots \dots, N\}$$

2) For each of the watermarked unit insertions, steps two to six outlined below must be carried out.

3) Work out the units' Manhattan distances $l_1^i{}' \ and \ l_2^i{}'$ by employing equations (13) and (14).

4) Use equation 15 to work out the difference $'$ and the integer $-$ mean mi of $l_1^i{}' \ and \ l_2^i{}'$.

5) After acquiring a set of coordinates for every feature, this is then to be transformed into an LCAT form.

6) To extract the embedded watermark location and watermark bits, equation (16) must be employed.

$$W^* = \begin{Bmatrix} if \ \ D \leq 50 \ \ then \ w_i{}^* \ is \ 0 \\ w_i{}^* = 1, otherwise \end{Bmatrix} \tag{16}$$

1) Private key K must be used here to extract the initial embedded watermark pattern W. To do this, the inverse watermark pattern must be used.

Once the outlined process is complete, then the original difference di for each unit will be obtained. When collaborating these with the integer- mean mi, it is then possible to work out the original coordinates of each unit by applying equations (13) to (15). To work out the watermark W' through the given method, the process above must be used. A group F_i is considered to be authentic when two watermarks W and W 'are equal. The watermark is considered tampered, and thus unauthentic, if they are not equal

IV. Results and Analysis

In the proposed method, the shape file format (.shp) of Environmental Systems Research Institute, Inc. (ESRI) is used. A simple shape file (.shp) vector map of "King Salman road map in Riyadh city" is the original map used to explore and test our proposed method. The map used to be a vector map, and has thirty polyline features and 130 vertices. Given the identical nature of geometric data structures of both polygon and polyline features, the findings of the polyline features will be presented in detail. Experiments have been carried out on computer using the software CPU 2.3 GHz, 16GB RAM, Win 10 Professional, QGIS Version 3.0, python language. When attempting to hide data, the secret bits corresponding to each transform coordinate carried α in LSB, Mn is $matrix\ size\ equal = 35$ and $T = 1$ represent iterative embedding.

It is worth noting that the proposed scheme was implemented to vector maps. Furthermore, in this case, the map employed represents the Riyadh Development Authority. Additionally, to serve as the watermark, an image was utilized (see Fig. 6). During the first test, we attempted to verify the quality of our proposed watermarking technique. To assess the subjective quality of the embedded vector map, comparisons were between the watermarked map and its original vector map counterpart. By adhering to Fig. 7(a) and 7(b), it is evident that the watermarked vector map is invisible.

We made effective use of the root mean square error (RMSE) to work out the objective quality of the watermarked vector map using equation (17).

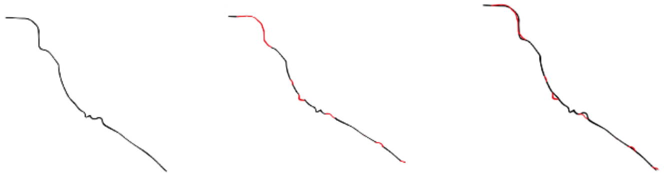
$$RMSE = \sqrt{\sum_{i=0}^{N} \sum_{i=0}^{N} \frac{[I(i,j) - I\prime(i,j]^2}{[I(i,j)]^2}} \tag{17}$$

On both the original map (M) and the watermarked map (M'), $V_{M'}$ and V_M are the corresponding vertices, with N representing the total vertices. The RMSE of watermarked vector map in our experiment is 1.973×10^{-10}. As has been previously discussed, when exploring the watermark embedding procedure in Section 3.1, the features of the 2D map play a major role in influencing the quality of the map. To improve invisibility as much as possible, it is important to choose the original vector map that has the highest correlation.

Fig. 4 presents the correlation between insertion distortion created from the Euclidean distances and the quality of the watermarked vector map worked out through RMSEs. To enhance the quality of watermarked vector map, it is crucial to minimize distortion. This is done by raising the map's precision tolerance. This results in a reduction of the total number of insertion units, and thus lowers the watermark's ability to be embedded.

Fig. 7 shows the findings of the experiment. The vector map presented in Fig. 7(a) has been subjected to the proposed watermarking scheme. To assess the subjective quality of the embedded vector, a human visual system (HVS) was employed. Through this, comparisons were made between the original vector maps and their watermarked counterparts. As shown in Fig. 7(a) and 7(b), there is a high level of invisibility in the watermarked vector maps. Fig. 7(c) shows the subset of the watermarked vector map and how this overlaps the original vector map, which allows for the differences between the two maps to be seen. Such variations indicate changes to the position of coordinates for original map features caused during the insertion process. It is possible to recover the original map so long as there have been no modifications to the watermarked map and the two watermarks must be identical.

Fig. 6. Example of Watermark.



[a] original vector map

[b] watermarked vector map

[c] the difference of them

Fig. 7. Watermark Imperceptivity Proof.

Original Watermark Extracted Watermark

Fig. 8. Well Extracted Watermark.

The capacity of the suggested scheme to identify tampering and to localize these was revealed in the second test. Modifications were made to certain areas (such as the coordinates of vertices, the addition of vertices and removal of some vertices). Fig. 7(a) shows the vector map following the process of watermark embedding. It has been modified by employing QGIS. One such modification that was made here is the removal of some features. Subsequently, we assessed the integrity of the manipulated vector maps through the watermark verification procedure. Fig. 8 shows the output of the watermark verification process. The dashed line shows the exact point where tampering occurred.

V. Conclusions

The proposed scheme of reversible fragile watermarking relies on the implementation of Manhattan distances using certain features as computation units. The watermark can then be inserted into the 2D vector map. Not only is the process beneficial in clarifying the map's integrity, but it can also precisely detect any modifications to map features. What's more, the embedding of watermark information accounts for the map's error tolerance. There is still a high level of practical value in the map following watermark insertion. When data must be very accurate, it is possible to recover the original vector map by conducting integrity verification. The findings of the experiments indicate that it is possible to accurately recover the original vector map following the extraction of watermark, so long as there have been no modifications made to the data. In terms of invisibility, the findings of the test case show that the quality of the relevant data cover is a crucial factor in improving the method's performance. Cover data that is highly correlated could lead to high capacity and invisibility. In future, we will investigate the scheme in more depth and examine how the scheme can be applied to point features. We will also investigate methods of improving the scheme's capacity through the iterative embedding on highly correlated data set.

References

[1] K.-T. Chang, Introduction to geographic information systems, McGraw-Hill, 2012.Li, W. Zhou, B. Lin, and Y. Chen, Copyright protection for GIS vector data production, Proceedings of SPIE, 2008, vol. 7143, p. 71432X–71432X–9.

[2] A. Li, Y. Chen, B. Lin, W. Zhou, and G. Lü, "Review on Copyright Marking Techniques of GIS Vector Data," in 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008, pp. 989–993.

[3] Kim J, Won S, Zeng W, Park S (2011) Copyright protection of vector map using digital watermarking in the spatial domain. In: 7th international conference on digital content, multimedia technology and its applications, pp 154–159 67.

[4] Kitamura I, Kanai S, Kishinami T (2001) Copyright protection of vector map using digital watermarking method based on discrete fourier transform. In: International symposium on geoscience and remote sensing, vol 3, pp 1191–1193.

[5] A. Li, B. Lin, Y. Chen, and G. Lü, "Study on copyright authentication of GIS vector data based on Zero-watermarking," The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences., vol. XXXVIII Pa, pp. 1783–1786, 2008.

[6] S. Tao, X. Dehe, L. Chengming, and S. Jianguo, "Watermarking Gis Data For Digital Map Copyright Protection," ICC, pp. 1–9, 2009.

[7] L. Zheng and F. You, "A Fragile Digital Watermark Used to Verify the Integrity of Vector Map," in E-Business and Information System Security, 2009. EBISS'09. International Conference on, 2009, pp. 1–4.

[8] N. Wang and C. Men, "Reversible fragile watermarking for 2-D vector map authentication with localization," Computer-Aided Design, Nov. 2011.

[9] X. Wang, C. Shao, X. Xu, and X. Niu, "Reversible Data-Hiding Scheme for 2-D Vector Maps Based on Difference Expansion," IEEE Transactions on Information Forensics and Security, vol. 2, no. 3, pp. 311–320, Sep. 2007.

[10] Neyman S, Sitohang B,C ahyono F (2013) An improvement technique of fragile watermarking to assurance the data integrity on vector maps. In: International conference on computer, control, informatics and its applications, pp 179–184 107.

[11] Zheng L, Li Y, Feng L, Liu H (2010) Research and implementation of fragile watermark for vector graphics. In: 2nd international conference on computer engineering and technology, vol 1, pp 522– 52.

[12] Wang Q , Zhu C (2012) Fragile watermarking algorithm for vector geographic data exactau then citation. J Geom Sci Technol

[13] eng F, Guo RS, Li CT, Long M (2010) A semi-fragile watermarking algorithm for authenticating 2d cad engineering graphics based on log-polar transformation. Comput Aided Des 42(12):1207–1216

[14] AL-ardhi S , Thayananthan V, Basuhail A (2020) Copyright Protection and Content Authentication Based on Linear Cellular Automata Watermarking for 2D Vector Maps. In: Arai K., Kapoor S. (eds) Advances in Computer Vision. CVC 2019. Advances in Intelligent Systems and Computing, vol 943. Springer, Cham.

[15] Neyman SN, Sitohang B, Sutisna S (2013) Reversible fragile watermarking based on difference expansion using manhattan distances for 2d vector map. Procedia Technol 11:614–620.

[16] R. Shiba, S. Kang and Y. Aoki: An image watermarking technique using cellular automata transform. In TENCON 2004 IEEE Region 10 Conference (2004), pp. 303–306.

[17] A. Dalhoum et al.: Digital image scrambling using 2D cellular automata. IEEE Multimedia, 2012.

[18] R.Shiba, S.Kang and Y.Aoki, "An Image Watermarking Technique using CAT", 2004 IEEE region 10 Conference, Vol.1, pp.303-306, 2004.

[19] N, Zhang H, Men C (2014) A high capacity reversible data hiding method for 2D vector maps based on virtual coordinates. Comput Aided Des 47:108–117.

[20] A. Martı and G. Rodrı: Reversibility of linear cellular automata. Applied Mathematics and Computation 217 (21) (2011), 8360–8366.

[21] S. Wolfram, Theory and Applications of Cellular Automata, World Scientific Publishing Company, Singapore, 1986.

[22] S. Wolfram, Cryptography with Cellular Automata, Springer- Verlag, Beilin, 1986.