A Zero-watermarking Algorithm of SVG Vector Map Based on Stroke-width Characteristic

Lin Zhou[a], Ying Huang[b, *], Zhanlong Chen[c], Xin Li

Faculty of Information Engineering, China University of Geosciences, Wuhan, China
[a]20890590@qq.com, [b]hying@cug.edu.cn, [c]33184569@qq.com

*Corresponding author

Abstract—**This paper proposes a zero-watermarking algorithm of SVG vector map based on stroke-width characteristic. The algorithm takes the stroke-width as an important characteristic. By statistics, calculating and simplifing the stroke-width probability in SVG vector map, building a K-d characteristic vector. With the Arnold transformation and specific operation, the stroke-width characteristic vector is embedded in the original watermark image, that is the zero-watermarking. Because the algorithm doesn't modify any coordinates, common operation such as geometry editing, geometry transformation, data compression and coordinate precision reduction doesn't affect the watermark extraction. Because of the simplification of stroke-width probability, the deletion of a small amount of entities has no influence on the extraction of the watermark yet.**

Keywords-SVG(Scalable Vector Graphics); Stroke-width Characteristic; Vector Map; Zero-watermarking

I INTRODUCTION

The Internet has deeply changed the way maps are nowadays designed and used [1], digital maps published on the Internet have become the main form of map. The first web maps are simple static images. They can be progressively displayed within specific viewers and explored the geographical space by panning and zooming in and out [2]. But they often occupy large storage space with poor interactivity and scaling distortion.

The solution to improve web mapping interactivity, avoid scaling distortion and reduce web map storage space is to use vector web maps. There are many formats for vector web maps. SVG (Scalable Vector Graphics) is one of the most common vector web map formats. It is a language for describing two-dimensional graphics in XML and allows for three types of graphic objects: vector graphic shapes (e.g., paths consisting of straight lines and curves), images and text [3]. Drawings in SVG can be interactive, dynamic and small. By leveraging and conforming to other standards, SVG becomes more powerful and makes it easier for users to learn how to incorporate SVG into their Web sites.

On the Internet, SVG vector map is easy to be downloaded and used illegally. Digital watermarking technology is an effective solution for copyright protection and products use tracking of geographic information. Recently, the study of vector map watermarking technology is hot [5-12]. According to watermark embedding methods,

watermarking algorithm can be divided into spatial domain algorithm and frequency domain algorithm. Spatial domain algorithm embeds watermark information into the map by modifying or adding map pixels or vertices. Frequency domain algorithm embeds watermark information into the transform coefficient of map pixels or vertices. By the impact on the map, watermarking algorithm can be divided into perturbation algorithm and zero-perturbation algorithm. Perturbation algorithm stores watermark information in the map. Zero-perturbation algorithm namely zero-watermarking stores watermark information in the registration authority. Perturbation watermarking has some interference in map content and may affect the application effect of the map [13].

In SVG vector maps, the styles are encoded within the data file. Then in this paper, a zero-watermarking algorithm is designed based on the stroke-width characteristic of the styles. This algorithm is robust and has no perturbation on map content.

II DESIGN OF ZERO-WATERMARKING ALGORITHM BASED ON STROKE-WIDTH CHARACTERISTIC

When designing a symbol, cartographers with different knowledge background may give different structures. But the appearance, color and stroke-width of a symbol are specific by the national mapping standards. In SVG, stroke-width is the property to specify the width of the stroke on the given graphical element. The value of the property can have different unit. But the unit is optional, if a unit is not provided, the value represents a distance in the current user coordinate system. In this paper, the unit of the value is "px".

A. Watermark Image Scrambling

In order to increase security, watermark image generally need to be scrambled before embedding. Common scrambling method is Arnold transform also called cat face transform. Given the image size N * N, i, j denotes the ith row and the jth column of the image respectively, the colour of the (i, j) pixel will appear in the (i', j') pixel after Arnold transform. Equation (1) is used to calculate the value of i' and j'.

$$\begin{vmatrix} i' \\ j' \end{vmatrix} = \begin{vmatrix} 1 & 1 \\ 1 & 2 \end{vmatrix} \begin{vmatrix} i \\ j \end{vmatrix} \bmod N, 0 \leq i, j \leq N-1 \qquad (1)$$

Arnold transform is cyclical, i.e., after a certain number

of transform, the image can be back to the original state. The scrambling period T is related to the image size N, e.g. the scrambling period of the 40 * 40 image is 30, the scrambling period of the 64 * 64 image is 48. In this algorithm, the size of watermark image is 40*40, and the color depth is 8 bit.

B. Stroke-width Characteristic Extraction

Given K is the amount of stroke-width in a SVG vector map, S_i is the amount of the occurrences of the ith stroke-width in a SVG vector map, the probability of the ith stroke-width occurring probability in the SVG vector map is H_i, and,

$$H_i = \frac{S_i}{\sum_{i=0}^{K} S_i} \qquad (2)$$

First, record the stroke-width in SVG vector map and ordered by descending. Then, calculate each occurring probability of stroke-width using (2). Finally, get the K-d stroke-width characteristic vector $(H_1, ...H_i...)$, $1 \le i \le N$.

C. Zero-watermarking generating process

Zero-watermarking generating process is shown in Fig. 1. First, simplify the stroke-width characteristic vector:

$$H_i' = \lfloor H_i * 100 \rfloor, 0 < i < K \qquad (3)$$

The result H_i' is the smallest integer of vector multiplied by 100. This simplification maps the vector values from (0,1] to (0,100]. Convert each vector into 8 bit binary number and get the K-d binary vector. Cycle the K-d binary vector to form a binary sequence A with the same size as the watermark image. Second, scramble the original watermark image W with Arnold transform, the number of scrambling is K mod T (K is the number of stroke-width, T is the scrambling period of the watermark image), and get the binary sequence B. Finally, execute A XOR B to get zero-watermarking W'.

D. Watermark Extraction

The zero-watermarking extraction is the inverse process of the zero-watermarking generating. The process is shown in Fig. 2. First, extract stroke-width characteristic of the SVG vector map and simplify the stroke-width characteristic vector by (3). Convert the vector into K-d binary vector and cycle the binary vector to form a binary sequence A′ with the same size as the zero-watermarking W′. Second, translate the zero-watermarking W′ into a binary sequence B'. Third, execute A' XOR B' to get the scrabled watermark image. Finally, do Arnold inverse scrambling on the scrabled watermark image to get the watermark image W.

III   DESIGN AND IMPLEMENTATION OF ZERO-WATERMARKING SYSTEM

A. Design of Zero-watermarking System

The zero-watermarking system consists of four parts

(Fig.3): stroke-width statistics function, stroke-width characteristics processor, zero-watermarking extractor and zero-watermarking generator.

In the zero-watermarking generating process, three modules are used: stroke-width statistics function, stroke-width characteristics processor and zero-watermarking generator. The following is the specific work of them:

- The stroke-width statistics function inputs the SVG vector map file which need to be embed zero-watermarking, and outputs the statistics value of each stroke-width.
- The stroke-width characteristics processor receives the output of the stroke-width statistics function as its input. If the input is empty, that means the SVG vector map has no content and needn't generate zero-watermarking. If there is only one stroke-width statistic value, sets the stroke-width characteristics vector $H_1' = 100$. Otherwise, calculate stroke-width characteristic vector using (2), simplify the vector using (3) and convert the vector into binary.
- The zero-watermarking generator builds the binary sequence A, scrambles the original watermarking image by Arnold transform and converts the scrambling result into the binary sequence B. Then, execute A XOR B to get the zero-watermarking.

In the zero-watermarking extraction process, three modules are used: stroke-width statistics function, stroke-width characteristics processor and zero-watermarking extractor. The work of stroke-width statistics function and stroke-width characteristics processor is the same as in the zero-watermarking generating process, The work of the zero-watermarking extractor as follows:

Receive the result from the characteristics processor. If the sum of the stroke-width charateristics vector is less than K, fill the blank places with "1". If the sum of the stroke-width charateristics vector is greater than K, just take the former K vectors to build the binary sequence A'. Converted zero-watermarking into binary sequence B'. Execute A' XOR B' and do inverse Arnold scrambling on the result to get the original watermark image. Where the K value and the zero- watermarking can be taken from the registration authority according to the range of the SVG vector map.

B. Experimental Results

To verify the robustness of the proposed algorithm, we do experiments on SVG vector maps under geometric translation, rotation, scaling, data compression, vertex adding, vertex deletion, vertex moving, cutting and coordinate precision decrease. The results were shown in table 1. The evaluation norm is the watermark similarity:

$$Sim(W, W^T) = \frac{\sum_{i=0}^{N} (w_i \otimes w_i^T)}{N} \qquad (4)$$

Where W is the 1-d vector of the original watermark image, W^T is the 1-d vector of the watermark extracted. N

denotes the length of the vector. If W_i and W_i^T are the same, then $W_i \otimes W_i^T = 1$, else $W_i \otimes W_i^T = 0$.

TABLE I.   RESULTS OF EXPERIMENT

Attack Types	Sim(%)
geometric translation	100
geometric rotation	100
geometric scaling	100
data compression (50%)	100
vertex adding (20%)	100
vertex deletion (20%)	100
vertex moving(20%)	100
cutting(1/4 map)	84.2
coordinate precision decrease	100

IV   CONCLUSION

This paper proposes a zero-watermarking algorithm based on the stroke-width characteristics rather than the coordinates of spatial data. This algorithm is robust in the common operations such as geometric edit (vertices increase, deletion), geometric transformation (translation, rotation, scaling), data compression and coordinate precision decrease. As the stroke-width characteristics vector is simplified, the deletion of a small amount of entities doesn't affect the watermark extraction.

REFERENCES

[1] Peterson, M.P., ed.: International Perspectives on Maps and the Internet. Lecture Notes in Geoinformation and Cartography. Springer .2008.

[2] Gaffuri J. Toward web mapping with vector data. Lecture Notes in Computer Science. Springer.2012.

[3] W3C. Scalable Vector Graphics(SVG)1.1 Specification[OL]. http://www.w3.org/TR/2011/REC-SVG11-20110816/intro.html

[4] Sun Jianguo. Research on Digital Watermarking of 2D Vector Map Based on Content Feature[D]. Harbin Engineering University,2009.

[5] Sun Hongrui. Study on Lossless Digital Watermarking Technology and Algorithm for Vector Map[D]. Central South University.2013.

[6] Zhu Changqing, Yang Chengsong, Li Zhongyuan. "An Anti compression Watermarking Algorithm for Vector Map Data" Journal of Zhengzhou Institute of Surveying and Mapping,vol.23, Aug. 2006, pp.281-283.

[7] Zeng Duanyang, Yan Haowen, Zhang Liming, Niu Liting," A blind watermarking algorithm for point layer of vector map," (DOIs), http://www.cnki.net/kcms/detail/11.4415.p.20140918.1011.002.html.

[8] Zhang Yanqun , Wang Qianping,"Complementary wate}rnarking algorithm of vector map based on discrete wavelet transform." Journal of Computer Applications.Vol. 30,Dec.2010, pp.110-115.

[9] Min Lianquan,"A Robust Digital Watermarking in Cartographic Data in Vector Format," Acta Geodaetica et Cartographica Sinica,Vol.37,May.2008,pp.262-267.

[10] Zhong Shangping, Gao Qingshi, "The Feasibility Analysis of Normalized-correlation based Vector Maps Watermarking Detection Algorithm and the Improved Watermarking Algorithm," Journal of Image and Graphics, Vol. 11,Mar.2006,pp.401-407.

[11] Wang Xun , Lin Hai , Bao Hujun," A Robust Watermarking Algorithm for Vector Digital Mapping," Journal of Computer Aided Design & Computer Graphics,Vol.16,Oct.2004, pp.1377-1381.

[12] Li Yuanyuan, Xu Luping, "Copyright protection of the vector map using the digital watermark," Journal of Xidian University, Vol.31, Oct.2004, pp.719-723.

[13] Sun Shenghe, Lu Zheming, Niu Xiamu, et al. Digital watermarking Technique and Application [M]. Beijing:Science Press,2004.
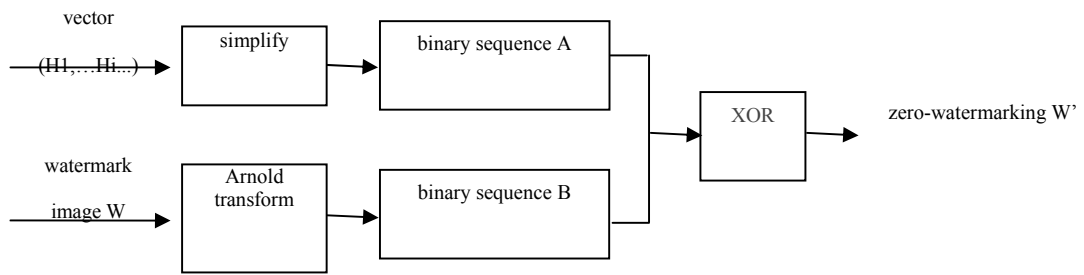
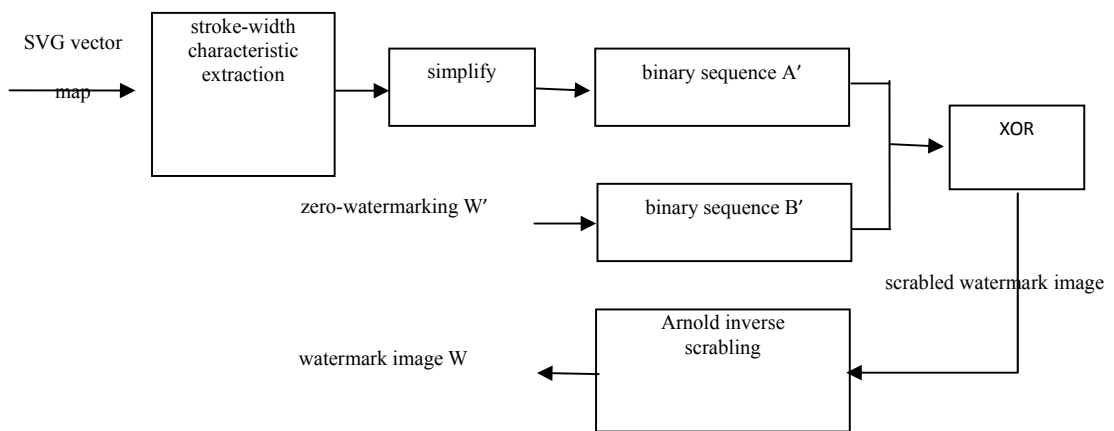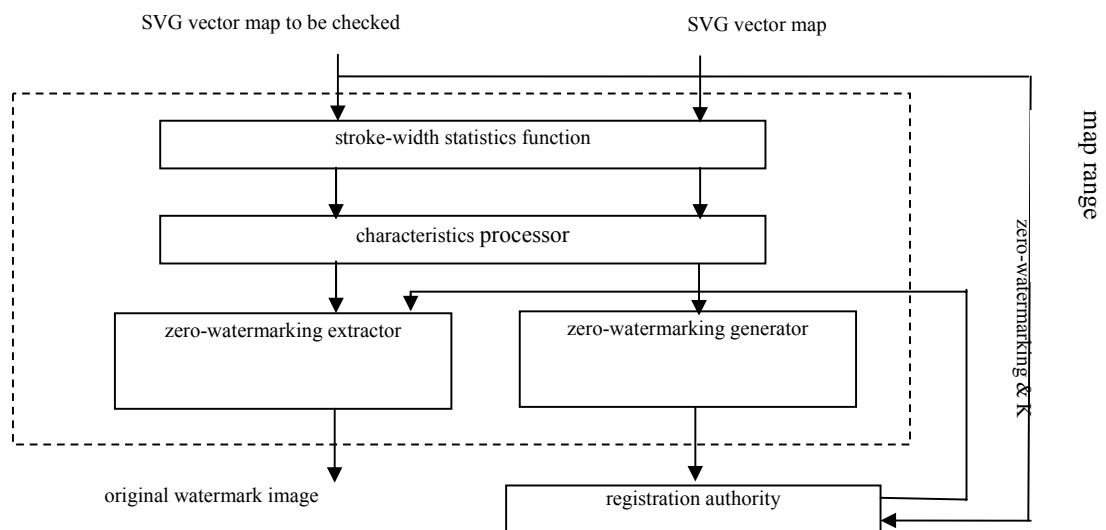Figure 1. Zero-watermarking generating process

Figure 2. Watermarking extraction

Figure 3. Zero-watermarking system of SVG vector map