A Watermarking Detection Algorithm Based on Statistical Properties for Vector Geographic Data

Yingying Wang 1, Zhong Yang 1, Nan Xu 1, Jiming Yu 2, Chensong Yang 3,∗

1 College of Intelligent Science and Control Engineering, Jinling Institute of Technology, Nanjing 211169, China
2 College of Electrical Engineering, Jinling Institute of Technology, Nanjing 211169, China
3 Institute of Field Engineering, Army Engineering University of PLA Nanjing 210007, China
Correspondence: ycsdongshang@163.com

Abstract. The watermarking detection algorithm for vector geographic data was designed based on the detected watermark bits characteristic. Firstly, the statistical characteristic of extracted watermark bits was analyzed; the probability-distribution function of detected watermark bits corresponding to the certain index was established on the condition that there was no watermark in detected data. And then a watermarking self-detection algorithm is present for vector geographic data. Finally, the experiments were made to verify the robustness and detection efficiency of the proposed algorithm. The experimental results show that we can judge whether there is watermark in detected data or not without the participation of the original data and the original watermark, and the proposed watermarking detection is more efficient than the common watermarking detection. The proposed algorithm has good performance in the aspects of self-detection, robustness and detection efficiency.

1. Introduction

Vector geographic data are increasingly used in a lot of applications and become the fundamental data representation structure. Correspondingly, it is important for vector graphic data to protect the copyright and authentication. Generally, the digital watermark technique for vector geographic data contains three parts, watermark generating, embedding and detecting. The first is selecting watermark, such as images, characters, binary sequences, voices and so on, preprocessing is done when necessary. The second is hiding the watermark into the cover data which is ready for embedding the watermark. There may be the watermark in the cover data. According to the different embedded domains, the digital watermarking is categorized as spatial and transform domain watermarking. The formal is hiding the watermark into the spatial domain component, the latter is into the discrete cosine transform (DCT) [1], discrete Fourier transform (DFT) [2], discrete wavelet transform (DWT) domain [3]. So far, there are many researches on digital watermark embedding for vector graphic data [4]. The third includes extracting and detecting the watermark. Mostly, extracting is the inverse operation of the embedding process to extract the embedded watermark, and detecting is using various methods to judge the specific content of the watermark. If the original data is not available for detection, it is called blind detection. The opposite is informed detection. Comparing with the researches on embedding, the detection algorithms are few [5]. But the detection algorithm is an important part of
digital watermark technique, which affects the robustness, reliability and practicability of the watermark system. A better detection algorithm is useful for improving the robustness and capability of the watermark system. Therefore, we focus on the detection algorithm and a watermark detection algorithm based on the statistic characteristics is designed.

2. Proposed watermarking algorithm

2.1. Watermarking generating

The watermark can be classified as meaningful and meaningless. Meaningful watermarks have specific meanings like images, characters, voice and so on. In contrast, meaningless watermarks have no specific meanings like chaotic sequence, pseudo random binary sequence and so on. In consideration of the characteristics of vector geographic data and the robustness of algorithm, the pseudo random binary sequence is chosen as the watermark. Let the watermark be \( W \). It is present in the expression below.

\[
W = \{w[i], 0 \leq i < N\}
\]  

where \( w[i] \) is watermark bit, \( i \) is watermark bit index, \( N \) is the length of watermark and \( w[i] \in \{0, 1\} \). \( w[i] \) has the followed statistical characteristics, \( P(w[i] = 0) = \frac{1}{2} \), \( P(w[i] = 1) = \frac{1}{2} \) which means the probability of \( w[i] \) equal to 0 or 1 is the same.

2.2. Watermark embedding

The basic idea of the proposed algorithm is that the watermark bits are embedded in the different coordinates repeatedly using the “one to many” mapping between watermark bit indexes and coordinates. In other words, a watermark bit is embedded in the different coordinates. Therefore, a watermark bit index may be corresponding to several coordinates. It is more robust against the vertex deletion attacks. And the Figure 1 depicts “one to many” relationship. The embedding model is shown in (2).

\[
D \oplus W = \{x_k \oplus w[f(x_k, y_k)]\}
\]  

where \( D \) is coordinates set, \((x_k, y_k)\) is the kth coordinate in the set, \( f() \) is the relationship between watermark bit index and coordinates, \( 0 \leq f() < N \), \( N \) is the length of the watermark. \( \oplus \) is the rule of watermarking embedding, quantization is used to embed watermark for the blind detection [6].

![Figure 1. The example of the “one to many” relationship between the watermark bit index and vertexes.](image)

2.3. Watermark detection

On account of pseudo random binary sequence, the watermark detection includes extraction and detection. Firstly, the inverse operation is done to get the extracted watermark bits. Then the statistical properties of the extracted watermark bits are analyzed. The final is detection.

Because the same watermark bit is embedded into the vector geographic data repeatedly, in the extraction phrase there are many extracted watermark bits corresponding to the same watermark bit index. Let the extracted watermark bits be the set, \( V \), \( V = \{V_i, 0 \leq i < N\} \), the \( V_i \) is a set which
corresponding to the $ith$ watermark bit index, \( V_i = \{ w'_i[k], 0 \leq k < L_i \} \), where \( w'_i[k] \) is the $kth$ extracted watermark bit corresponding to the index $i$ and $L_i$ is the amount of the extracted watermark bits corresponding to the index $i$.

The watermark bit extraction is the inverse progress of embedding. For the $kth$ vertex($x_k,y_k$), let extracted watermark bit be $b$, the watermark bit extraction expressions are present:

\[
\begin{align*}
  b = 0 \quad & \text{if} (x \% r) < \left(\frac{r}{2}\right) \\
  b = 1 \quad & \text{if} (x \% r) \geq \left(\frac{r}{2}\right)
\end{align*}
\]

(3)

We can also get watermark bit index corresponding to the watermark bit $b$ according to the function $f(x_k,y_k)$. Then we can get the extracted watermark bits set $V$ after extracting the watermark bits of all vertexes.

The set $V$ consists of many subsets $V_i$. There may be many watermark bits in set $V_i$ because of the “one to many” relationship between watermark bit index and vertex. Let $X = \sum (V_i) = \sum_{k=0}^{L_i} w_i[k]$. $X$ obeys binomial distribution present as expression (4).

\[
P(X = m) = B(m, L_i, 0.5) = \left(\binom{L_i}{m}\right) (0.5)^m
\]

(4)

Hence, the watermark bit corresponding to index, $i$, can be determined by the method below.

- Let $P(X \leq m_0) = \mu$, and $\mu$ is very small, which means that the probability of $X \leq m_0$ is very small. When $X \leq m_0$, the original hypothesis is fake. It means that the watermark bit corresponding to the $ith$ index is equal to 0.
- Let $P(X \geq m_1) = \mu$, and $\mu$ is very small, which means that the probability of $X \geq m_1$ is very small. When $X \geq m_1$, the original hypothesis is fake. It means that the watermark bit corresponding to the $ith$ index is equal to 1.

$\mu$ is 0.005 in the following experiments. It is quick to detect the single watermark bit without the original watermark and the original data. We call this kind of watermark detection as self-detection.

### 3. Experimental Results

To evaluate performance of the proposed detection algorithm, 2-D vector maps with the scale 1:1000000 and unit meter were used in experiments. The experiment exploits on Visual C++ and ArcGIS platform and was made to detect the robustness and detection efficiency of the algorithm.

#### 3.1. Robustness

The basic experimental idea is following. The watermark is embedded in the vector map using the previous algorithm. Then the attacks in table 1 are made to the map with the watermark. Finally, it is detected whether there is watermark in the map or not after being attacked. The results are shown in Table 1. Neither the original vector map nor the original watermark does participate in the detection progress. In the table 1 “√” indicates there is the watermark in data after attacking, “×” indicates the opposite result.

| Attacks          | Attack intensions (%) / Detection results |
|------------------|-----------------------------------------|
| Data compression | Compression percentage: 92.1 83.4 75.9 64.7 58.6   |
|                  | Detection result: √ √ √ √ √               |
| Data adding      | Adding percentage: 10 20 30 40 50        |
|                  | Detection result: √ √ √ × ×              |
| Data deleting    | Deleting percentage: 10 20 30 40 50       |
|                  | Deletion result: √ √ √ × ×               |
| Data cropping    | Cropping percentage: 91.4 75.5 48.7 35.5 21.7 |
|                  | Deletion result: √ √ √ √ √                |
| Feature deleting | Deleting percentage: 93.4 81.3 69.6 51.2 39.7 |
|                  | Deletion result: √ √ √ √ √                |
The experimental results in Table 1 indicate that the above algorithm is robust against common attacks, such as data compression attack, clipping attack, deleting features attack and so on. The robustness of the proposed algorithm is good without the help of original watermark. And it is also against the weak adding coordinates attack. To judge whether there is watermark in detected data or not, by using the proposed algorithm it is no need for the original watermark when there are plentiful vertexes in the map.

3.2. Detection efficiency

The comparative experiments between the common detection algorithm (algorithm 1) and the proposed detection algorithm (algorithm 2) are made to verify the algorithm 2 is efficient. The steps of the comparative experiment are as following. Firstly, the watermarks were embedded into the 6 experimental maps that have different data payloads. Then the watermarks were detected separately by using the two algorithms, while each detecting time is recorded. The comparative experimental results are shown in Table 2.

Table 2. The comparative experimental results of detection efficiency

| No. of experimental map | Data size(KB) | Detecting time(s) |
|-------------------------|---------------|-------------------|
|                         |               | Algorithm 1 | Algorithm 2 |
| 1                       | 479           | 0.25        | 0.015       |
| 2                       | 765           | 0.36        | 0.015       |
| 3                       | 1002          | 0.45        | 0.015       |
| 4                       | 1343          | 0.61        | 0.015       |
| 5                       | 1487          | 0.67        | 0.015       |
| 6                       | 1747          | 0.77        | 0.015       |

4. Conclusions

After analyzing the statistic property of detecting watermark bits, it was proposed a watermark detection algorithm without the original vector map and the original watermark. Then some experiments were made to test the robustness and the detecting efficiency. The experimental results depict that the proposed detection algorithm has good performance on robustness and detection efficiency. In the experimental phrase, it was found that the algorithm is not robust against the strong adding coordinates attack and is not suit for the vector map with little data payload, which is the following work we will focus on.

References

[1] Voigt M, Yang B, Busch C. Reversible watermarking of 2D-vector data, Proceedings of the 2004 Workshop On Multimedia and Security, Magdeburg, Germany, September 20-21, 2004; ACM: New York, USA.

[2] Urvoy M.; Goudia D.; Autrusseau F. Perceptual DFT watermarking with improved detection and robustness to geometrical distortions. *IEEE Transactions on Information Forensics and Security* 2014, 9, 1108–1119. DOI: 10.1109/TIFS.2014.2322497.

[3] Benoraira A.; Benmahammed K.; Boucenna, N. Blind image watermarking technique based on differential embedding in DWT and DCT domains, *EURASIP Journal On Advances In Signal Processing* 2015, 2015,1-11, DOI: 10.1186/s13634-015-0239-5.

[4] Peng, Z.; Yue, M.; Wu, X.; Peng, Y. Blind watermarking scheme for polylines in vector geo-spatial data. *Multimedia Tools and Applications*, 2015, 74, 11721-11739, DOI:10.1007/s11042-014-2259-9.

[5] Yang, C. S.; Zhu, C. Q.; Wang, Y. Y. Self-detection watermarking algorithm and its application to vector geo-spatial data. *Geomatics and information science of Wuhan University* 2011, 36, 1402-1405, DOI: 10.13203/j.whugis2011.12.012.
[6] Chen, B.; Wornell, G.W. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. IEEE Transactions on Information Theory 2001, 47, 1423-1443, DOI: 10.1109/18.923725