

Design and Realization of the Enhanced Compression Method aimed at the vector map

Xin Xu^{1,2,*}, **Jinguang Gu**², **Hongbing Zhu**^{2,3}, **Yimin Qiu**²

¹*Division of Computer Integration Technology, School of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai 200240, China*

²*School of Computer Science and Technology,*

Wuhan University of Science and Technology, Wuhan 430065, China

³*Department of Computer Science, Hiroshima Kokusai Gakuin University, Hiroshima-shi, 739-032 Japan*

Abstract: This paper discussed several classical algorithms for vector data compression firstly, and put forward an enhanced vector data compressing algorithm, basing on the analyses of the problems caused by applying these algorithms to the vector map. This algorithm integrates the merits of the classical algorithms, thus can tackle the problem of classical compressing algorithms, such as the distortion of the overall direction in James Algorithm, and the terrain distortion in Douglas - Peucker Algorithm and Light Column Algorithm. An evaluation was developed to test the reliability of this algorithm, and MapInfo data was adopted to show the vector map with the precision of 1:10000, from which the 3rd trunk road of Qingshan District in Wuhan was selected for experiment. Result shows that the proposed algorithm has the least difference in the comparison of the total length and average coordinate of the curve, with corresponding data before compression. And, it has the most similarity in its exterior compared to other classical compressing algorithms. Thus this algorithm could be effective in compressing the vector map. And the experimental research results will hopefully serve as useful feedback information for improvements for vector map compressing in personal navigation system.

Keywords: Vector data compression, Compressing algorithm, Reliability evaluate, Geographic information systems

1. Introduction

Since the idea "Digital Earth" [1] was proposed, an ever-increasing number of digital map's applicants can be seen in the market [2]. Particularly in the areas of navigation, the application of cartographical was no longer confined to specific areas, such as vehicle and maritime transport systems. Currently, there are also many positioning navigation products for personal use. Nevertheless, the trend of micromation and integration for navigation products will inevitably result in more limitation in its processing speed and storage capacity. Therefore, low cost storage and rapid transmission of cartographical is needed. Whereas whether or not the cartographical can be entitled low cost storage and rapid transmission is subjected to the efficient compression of the cartographic [3].

Within Geographic Information Systems (GIS), two types of generalisation have been developed over the past years, namely cartographical and model-based generalisations [4]. Cartographical generalisation can be defined as a geometrical simplification in a scale reduction process while model generalisation is mainly oriented to a structural-based filtering. These two generalisation approaches are closely related, often model generalisation being a pre-process of cartographic generalisation.

Cartographic generalisation is a constraint - based process used by cartographers to reduce the complexity of a map in a scale reduction process. It involves intensive human knowledge obtained through professional cartographic expertise and practise. Since the seminal Douglas-Peucker algorithm for line simplification [5], automatic generalisation has long been a research effort by both scientific researchers and cartographic practitioners [6].

* Corresponding author.
Email address: xuxin0336@163.com.

Cartographic can be divided into two categories: vector structures and grid structure [7]. And they have their own advantages and disadvantages in expressing different geographical phenomenon. The compressing of grid map is not involved in this paper, for it is similar to image compression. On the other hand, the definition of vector data compression can be simply defined as the process of eliminating redundancy. To quote ZHENG Hai-ying [8], "vector data compression was extracting a subset from data sets with possibly little data, and at the same time reflecting the original appearance of it as far as possible." Depending on the nature of the application, two classes of compression methods exist: lossless methods and lossy methods. In the lossless methods, data is compressed in such a way that upon reconstruction (decompression), the original data is exactly restored. In the lossy methods, an error measure between the original and the reconstructed data is used to allow a tolerable data distortion. Most vector map compression techniques, including the algorithm introduced in this paper, are of this last category, and eventually a combination of the two categories.

Structural tables, namely geometry data, were usually used to store the graphical and topological information of vector map in current applications of GIS, such as MapInfo data. Other data, attribute data for example, was saved in other files. Moreover, attribute data could be saved in files or database, while geometry data was saved in files in most of the time. Still, it organized the files through different layer according to their geographical information. Generally, geometry data, compared to attribute data, took the most of the vector map's storage. For instance, MapInfo vector map's layer, which was saved in file format, generally had two textual file, namely *.MAP for geometry data's storage and *.TAB for attribute data's storage. And this paper will focus on the compression of the *.MAP file.

Specifically, vector map compression is based on the principle of selecting a reduced set of dominant points (DPs) on the curve from the map according to an adequate strategy. Currently, there are many classical algorithms for compressing vector data. However, they have their own problems when applying the compression to vector map. As a result, an enhanced vector data compressing algorithm was put forward in this work, whereafter, the reliability of this algorithm was tested through experiment. The result of the experiment shows that the enhanced vector data compressing algorithm introduced in this paper could efficiently tackle the problems caused by the classical compressing algorithms, especially effective when aiming at the

vector map. Besides, it will hopefully serve as useful feedback information for related works.

The rest of this paper is organized as follows. Section 2 is an overview of some previous works for curve simplification. Section 3 presents the main limitation of those previous works (Section 3.1), and proposes a novel algorithm for curve simplification (Section 3.2). The improvement step for the curve simplification algorithm is also reported in Section 3.2. Section 4 tests the reliability of algorithm, and proceeds to an overall discussion on obtained results. Section 5 emits some concluding comments. Section 6 announces future planned works.

2. Classical Compressing Algorithms for Curve Simplification

Three types of geographic information have been developed to represent the vector map over the past years, namely point element, curve element and surface element [9]. Point element, similar to point in geometric feature, stands for location of the target in ground. And two-dimensional coordinates was used to express its spatial information. For instance, we can use point element to express hospital, station, pier, shop and so on. Curve element, similar to curve in geometric feature, can stand for road, river, railway and etc. Surface element, similar to polygon or closed region in geometric feature, usually has the character of area and perimeter. For example, we can use surface element to express farmland, grassland, Public Park, Public Square and so on. Among them, compression towards point element will not have significant effect in saving storage space. Surface element compression can be seen as the compression towards a family of curves [10]. Therefore, compressing towards vector map can be treated as the compressing towards curve element, namely curve simplification.

2.1. Definitions of Curve Simplification

Curve simplification [11]: Given a polygonal curve $C_{A,B} = (P_1 = A, P_2, \dots, P_N = B) = P$, where A and B are the curve endpoints and N is the number of samples, the curve simplification of $C_{A,B}$ consists in computing another polygonal curve $C'_{A,B} = P'(A = Q_1, \dots, Q_M = B) \in C_{A,B}$, with $M < N$ and $[P, P'] < \epsilon$, with $\epsilon > 0$, a preset tolerable error of simplification.

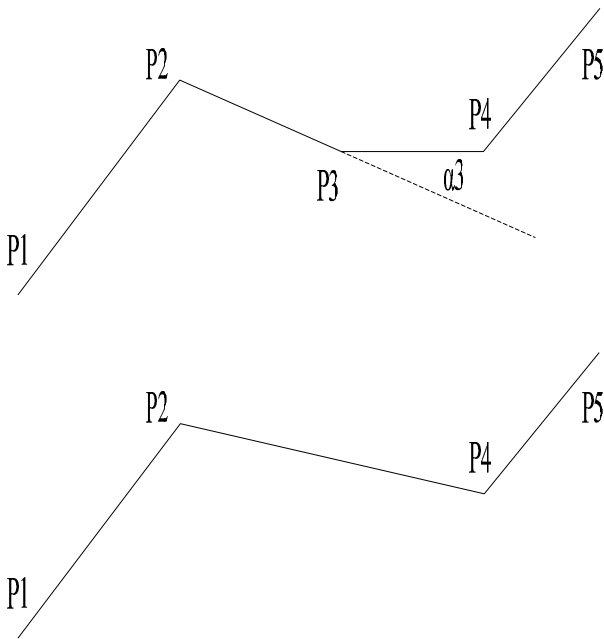


Figure 1. James Algorithm for Curve Simplification.

2.2. James Algorithm

The James algorithm for curve simplification is illustrated on figure 1 [12]. In this method, given a polygonal curve, the James algorithm computes a subset of DPs standing for the simplified curve. The selected DPs have the important property of being perceptually attractive. The key elements of James Algorithm lay in the following steps. For all DPs in each curve, we calculated the angle change α_j between line L_{j1} and line L_{j2} through the point P_j . Then the angle change α_j was compared to the tolerance band α_0 for deciding whether or not the point P_j should be preserved. The figure 1 illustrated once the angle change α_3 was less than the tolerance band α_0 , then point P_3 should not be preserved.

2.3. Douglas - Peucker Algorithm

The Douglas-Peucker algorithm for curve simplification is illustrated on figure 2 [5]. In this method, given a polygonal curve, the Douglas - Peucker algorithm computes a subset of DPs standing for the simplified curve. The selected DPs have the important property of being perceptually attractive. In this algorithm, a DP is computed as the furthest point, say 1 (figure 2), from the given curve endpoints, say M and N. Further DPs are computed by recursively applying the DP definition for the two resulting sub-segments: $[M \dots 1]$ and $[1 \dots N]$. In this algorithm, the end of process criterion is a threshold, say t_d , on the Euclidian distance d_i . This curve simplification step delivers

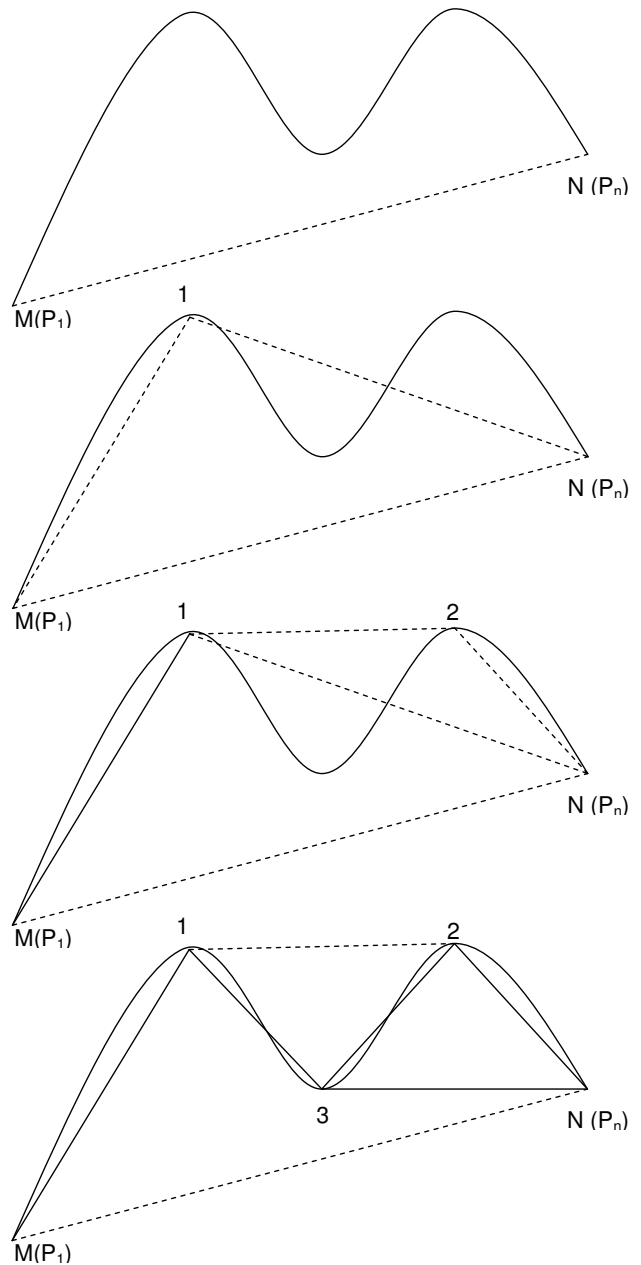


Figure 2. Douglas - Peucker Algorithm for Curve Simplification.

a curve skeleton with fewer points to process [13].

2.4. Light Column Algorithm

The Light Column Algorithm for curve simplification is illustrated on figure 3 [14]. In this method, given a polygonal curve, the Light Column Algorithm computes a subset of DPs standing for the simplified curve. The selected DPs have the important property of being perceptually attractive. The sticking point of Light Column Algorithm lay in the following steps. For all DPs in each curve, a fan-shaped region is defined for each point. Afterward, whether or not the

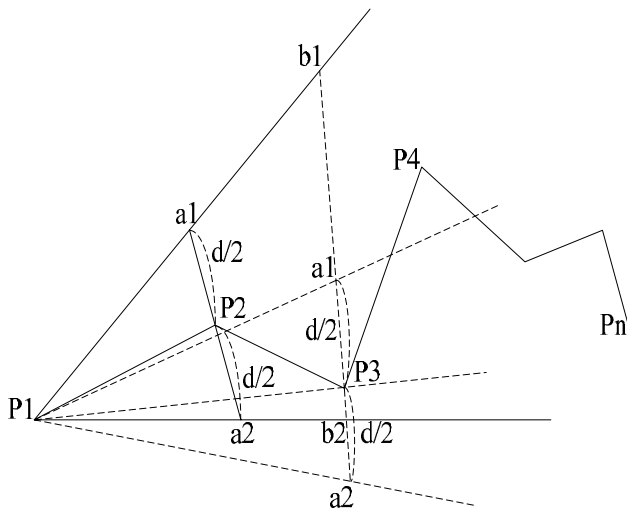


Figure 3. Light Column Algorithm for Curve Simplification.

point should be preserved can be attributed to whether or not the point is in the fan-shaped region. The figure 3 shows the implementation procedure. We draw a straight line perpendicular to the line L_{P_1, P_3} , which intersect the edge of the fan-shaped region defined by Light Column Algorithm at b_1 and b_2 . Then, a_1 and a_2 , which is located in this straight line and $d/2$ apart from the point P_2 , is connected separately to P_1 for defining a new sector. Besides, a_1 or a_2 should be replaced by b_1 or b_2 unless they are inside the sector. As a result, the sector is updated. Afterward, if P_4 is inside this sector, P_3 should be replaced by P_1 .

3. Enhanced vector data compressing algorithm

3.1. Limitation of classical algorithms

All the discussed classical compressing algorithms could be successfully applied to vector data. However, they had their own problems when applying the compression to vector map.

In James Algorithm, due to its' lack of consideration of the curves' global features, the overall direction of the curve may be distorted when there was a continuous small angle change. As shown in figure 4, we calculated the angle change α_2 between line $L_{1,2}$ and line $L_{2,3}$ through the point 2. Then the angles change α_2 was less than the tolerance band α_0 . As a result, the point 2 should not be preserved. And similar statements could be made about the points 3, 4, 5 and 6. At last the curve $C_{1,2,3,4,5,6,7}$ could be simplified to line $L_{1,7}$ with a distortion of the overall direction.

Douglas - Peucker Algorithm, which is the most popular method to reduce the number of vertices in a digital curve, can keep the overall direction well. The

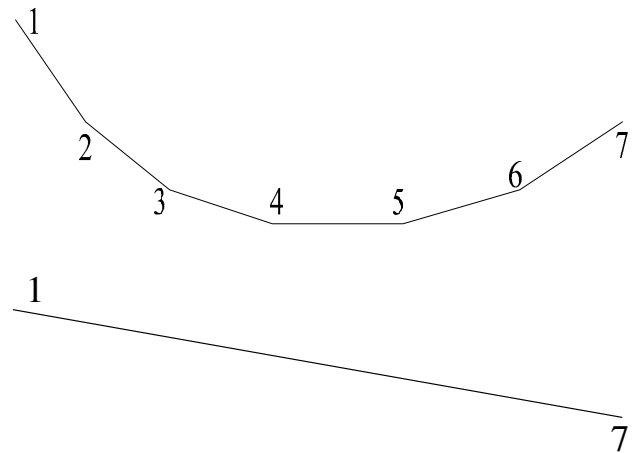


Figure 4. Distortion of the overall direction.

algorithm iteratively selects new points for inclusion in the thinned output curve based on their deviation from a baseline connecting two neighboring points already chosen for inclusion. A review of relevant GIS literature, in journals and online, indicates that implementations often incorporate an error in the method used to calculate the distance between baselines and intermediate data points. This problem exists because the original Douglas-Peucker paper is somewhat ambiguous in its definition of the distance criterion for selection of a point [5]. On pages 116 and 117 of their paper we see: (1) "The perpendicular distance of C from the segment A-B," (2) "distance from the straight line segment", (3) "the furthest point from the straight segment", (4) "maximum perpendicular distance from the segment", (5) "the greatest perpendicular distance between it and the straight line defined by the anchor and the floater". Of these definitions, (2) and (3) are correct, but (5) seems to be the most widely used by programmers [15]. We need the distance from the segment, not the distance from the line or the perpendicular distance from the segment. To see the effect of this mistake, consider the situation shown in figure 5. When the curve witnessed a significant angle change, some characteristic bending points (point P in figure 5) might be lost after compression and resulted in terrain distortion.

Light Column Algorithm calculated through ever-dwindling caliber to detect the point outside the fan-shaped region. Owing to the increasingly stringent conditions of detection, the compressing was not that efficient. Still, similar to Douglas - Peucker Algorithm, when the curve saw a significant angle change, some bending feature points might be lost and resulted in terrain distortion. A simpler example, which could be used to test whether an implementation of the Light Column Algorithm suffers from this mistake, may be seen in figure 6. The modified line segment will miss

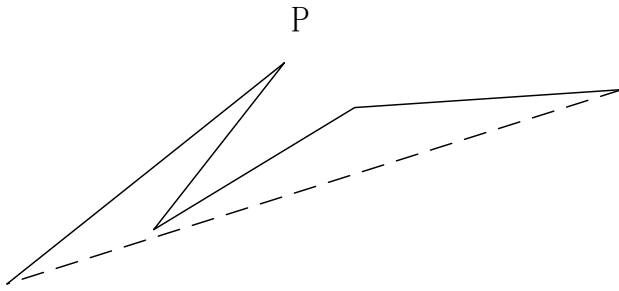


Figure 5. Terrain distortion.

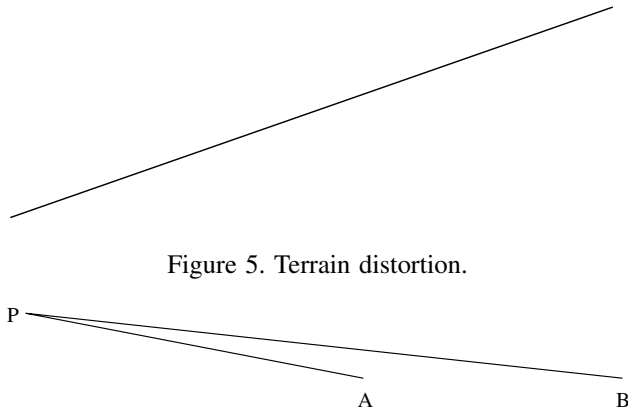


Figure 6. Polyline with three vertices.

the removed point during the line $L_{B,P}$ and $L_{P,A}$.

In spite of the reduction in tolerance band could solve the above problems; however there, at the same time, will inevitably be a significant reduction in the compressing ratio. Accordingly, some improvement on compressing algorithm for vector map was investigated in this paper.

3.2. Description of the enhanced vector data compressing algorithm

In order to solve the limitation of classical algorithms and extract as less data as possible without reduction in accuracy, it was considerable to integrate the merits of the above classical algorithms.

Procedure of this algorithm:

(1) Using James Algorithm to extract the characteristic point with a significant angle change, and compressing the curves by Light Column Algorithm at the same time.

For all points in each curve, take P_i for example, the angle change α_i between line L_{P_{i-1},P_i} and line $L_{P_i,P_{i+1}}$ was calculated. Afterward, whether or not the point P_i witnessed a significant angle change and should be preserved was subjected to whether or not the angle change α_i was more than the tolerance band α_0 .

On the other hand, we drew a straight line perpendicular to the line L_{P_{i-1},P_i} , which intersect the edge of the fan-shaped region defined by Light Column Al-

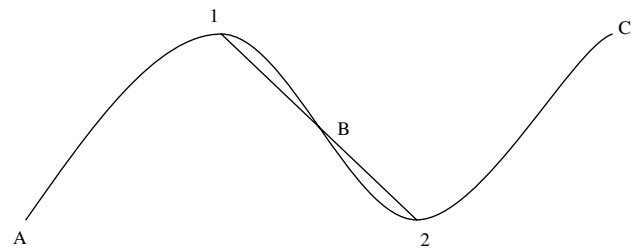


Figure 7. Endpoint redundancy.

gorithm at b_1 and b_2 . Then, a_1 and a_2 , which was located in this straight line and $d/2$ apart from the point P_i , was connected separately to P_{i-1} for defining a new sector. Besides, if the variable flag was equal to the true value, a_1 or a_2 should be replaced by b_1 or b_2 when they were outside the sector. As a result, the sector was updated. Afterward, if P_{i+1} was inside this sector, P_i should be replaced by P_{i-1} , and set flag to true. Else flag should be set to false.

Then the first step of compression was completed, and the point with a significant angle change was marked at the same time.

(2) Cutting the compression curve through the point with a significant angle change, and then compressing them separately by Douglas - Peucker Algorithm.

There were some problems, which had been discussed in reference [16], should be pay attention to in using Douglas - Peucker Algorithm to compress the vector map.

Closed curve should be divided into two curve end to end, and then compressing them separately by Douglas - Peucker Algorithm. The principle for selecting the split points lay in three criterions.

- a The split point should be one of the DPs.
- b The other split point should be the point with the greatest distance between it and the split point.
- c The split curve should be the longest one of DPs' connection.

Redundant data may follow the compression to a family of curves linked end to end, for the initial point and the endpoint were extracted as the characteristic point for each curve. As shown in figure 7, point 1 and 2 respectively stand for the nearest vertices to endpoint B, therefore the curve $C_{1,B}$ and $C_{B,2}$ can respectively be replaced by subtense $L_{1,B}$ and $L_{B,2}$. However, the entire curve $C_{1,B,2}$ can be replaced by subtense $L_{1,2}$ with reduction of point B.

Accordingly, additional procedure is needed: detecting whether or not the endpoint (point B in figure 7) can be deleted.

- a If the two succession curves were the public edge of a polygon, then the public point should be preserved.
- b Else if the curve between the two points which

were extracted from the two succession curves satisfied the criterion of Douglas - Peucker's tolerance band, then the joint point can be deleted.

c Else the point should be preserved.

In this program, depth-searching algorithm was adopted to find the public edge of the polygon [17].

After these two steps of compression above, the curve was compressed. Also, the procedure associated to this strategy was given in Algorithm 1, where $C_{A,B}$ is the curve needed to be simplified and J, G and D were the tolerance band of James, Light Column and Douglas-Peucker Algorithm. Besides, α_k stood for the angle change between line $L_{k-1,k}$ and line $L_{k,k+1}$ through the point P_k ; $LCA(P_i, P_k, G)$ stood for a sector defined by points P_i, P_k and tolerance band G; $DP(C_{A,B}, D)$ stood for procedure of the Douglas-Peucker Algorithm.

Algorithm 1: CurveSimp($C_{A,B}, J, G, D$)

Input: J, G and D were the tolerance band of James, Light Column and Douglas - Peucker Algorithm, $C_{A,B}$ is the curve needed to be simplified.

Output: P' is the set of dominant points after compression, M is the number of the points.

```

1  $P' \leftarrow \emptyset \cup P_1, i \leftarrow 1, k \leftarrow 2, flag \leftarrow false;$ 
2 while  $P_k \in C_{A,B}$  do
3   if  $\alpha_k \geq J$  then
4      $P' \leftarrow P' \cup P_k;$ 
5   else
6      $LCA(P_i, P_k, G);$ 
7     if  $flag \neq false$  then
8        $a_1 \leftarrow b_1, a_2 \leftarrow b_2;$ 
9     end
10    if  $P_{k+1} \in LCA(P_i, P_k, G)$  then
11       $P_k \leftarrow P_i, i \leftarrow i + 1, flag \leftarrow true;$ 
12    else
13       $P' \leftarrow P' \cup P_k, flag \leftarrow false;$ 
14    end
15  end
16   $k \leftarrow k + 1;$ 
17 end
18 if  $P_1 == P_N$  then
19   CurveSimp( $C_{A,C}, J, G, D$ );
20   CurveSimp( $C_{C,B}, J, G, D$ );
21 end
22  $M \leftarrow |P'|;$ 
23  $DP(C_{A,B}, D);$ 
24 return ( $P', M$ );
```

4. Algorithm Evaluations

The reliability of algorithm can be tested through the comparison of the data, including the total length and average coordinate of the curve, with correspond-

Table 1
Length Comparison

Name	Length km	Difference
Original curve	9.023876126834	0
James Algorithm	8.951782492573	7.98921E-3
Douglas-Peucker Algorithm	9.013681028737	1.12979E-3
Light Column Algorithm	9.015937028463	8.7979E-4
This Algorithm	9.016173945937	8.5353E-4

Table 2
Average Coordinates Comparison

Name	Length km	Difference
Original curve	114.385484755 30.6358562641	0 0
James Algorithm	114.385866870 30.6349333413	3.3406E-6 3.0126E-5
Douglas-Peucker Algorithm	114.385768397 30.6353483255	2.4797E-6 1.6580E-5
Light Column Algorithm	114.385682864 30.6354687928	1.7319E-6 1.2648E-5
This Algorithm	114.385515924 30.6356739586	2.7249E-7 5.9507E-6

ing data before compression. The less difference in corresponding data, the higher fidelity was.

In order to evaluate the reliability, the implementation of the algorithm was put forward in Windows XP platform, using the Visual C++ 6.0 as the Integrated Developing Environment. Besides, MapInfo data was adopted to show the vector map with the precision of 1:10000, from which the 3rd trunk road of Qingshan District in Wuhan was selected for experiment. Based on the requirement of the application, threshold value was set to 2 meters for corresponding tolerance band. An improvement on the result shown below could be made by based on the data provided in Table 1 and Table 2.

As shown in Table 1, the enhanced vector data compressing algorithm introduced in this paper had the least difference (8.5353E-4 in Table 1) in the comparison of the length value after the compression.

Similar statements could be made about Table 2, where the corresponding data of average coordinates had the least difference (2.7249E-7 difference in longitude and 5.9507E-6 difference in latitude in Table

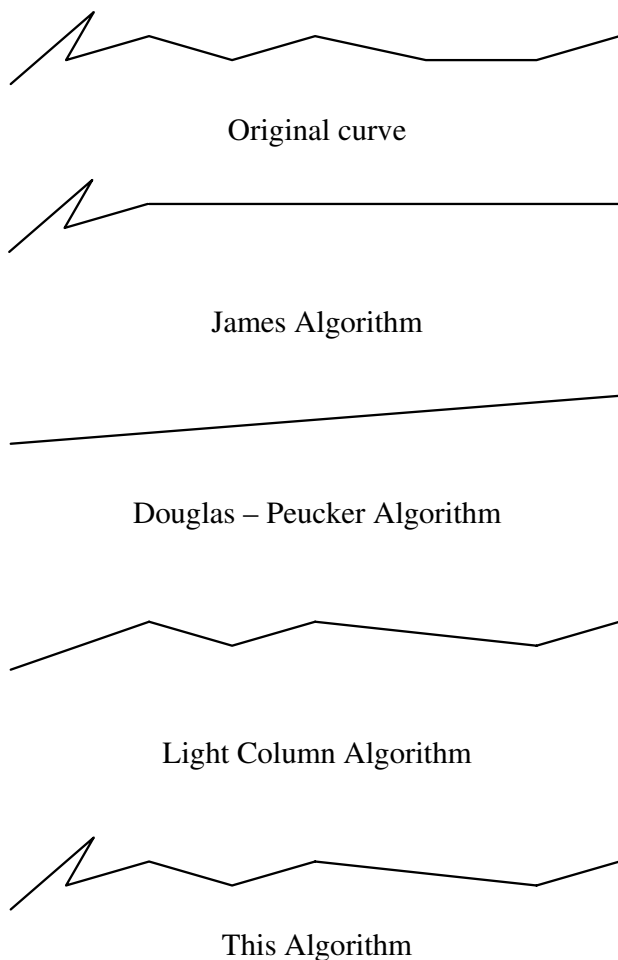


Figure 8. exterior comparison.

2) after the compression by the enhanced vector data compressing algorithm.

Also, as shown in figure 8, it can be seen that the algorithm proposed in this paper had the most similarity in its exterior compared to other classical compressing algorithms.

Furthermore, the enhanced vector data compressing algorithm could process during the digitalization of the map with linear temporal complexity and linear spatial complexity in most of the time [18]. Thus, it had a fast compression speed.

5. Conclusion

Form the above discussion, we may reasonably come to the conclusion that the enhanced vector data compressing algorithm introduced in this paper could efficiently tackle the problems caused by the classical compressing algorithms, especially effective when aiming at the vector map in personal positioning navigation products. Moreover, this algorithm can also be successfully applied in other vector maps applicants with high fidelity and fast processing speed. And

the experimental research results will hopefully serve as useful feedback information for improvements for vector map compressing.

6. Future works

In this paper, we proposed a vector map compressing algorithm aiming at the personal positioning navigation products. Beyond the important results of the proposed algorithm as far as compression is applied to the project, which grant it a good ranking amongst competing methods, the study showed that this algorithm can be entitled low cost storage and rapid transmission to vector map compression. Currently, the study focused on the vector map compression mainly in the monitoring center which could locate the end user's position, not in the receiving terminal. Then the compressed vector map was sent from the monitoring center by network to help locating the receiving terminal. As a result, the end user cannot get their position without the help of the network. Nevertheless, there are needs for specific end users to know their position in all circumstances, such as fieldwork staff, underground workers and etc. Accordingly, it is necessary to apply the compression to the receiving terminal.

Acknowledgment

This work was partially supported by a grant from the NSF (Natural Science Foundation) of Hubei Prov. under grant number 2007ABA296 and NSF of educational agency of Hubei Prov. under grant number Q200711004, B20071106 and T200801. It was partially supported by China Postdoctoral Science Foundation under grant number 20060400275 and Jiangsu Postdoctoral Science Foundation under grant number 0601009B.

References

- [1] D. Li, "Digital Earth and 3S," in *Proceedings of the Fifth Chinese GIS Association Annual Meeting*. Beijing: Chinese GIS Association, 1999.
- [2] L. Li, H. Mao, C. Li, L. Liu, and Y. Chen, "Application of the Digital Map based on GIS," *Areal Research and Development*, vol. 6, pp. 52–54, 2003.
- [3] S. Zhong and Q. Gao, "An Efficient Lossless Compression Algorithm for a kind of Two-Dimension Vector Maps," *Journal of System Simulation*, vol. 16, no. 10, pp. 2189–2194, 2004.
- [4] J. C. Muller, R. Weibel, and J. P. Lagrange, "Generalization: State of the Art and Issues," in *GIS and Generalization: Methodology and Practice*. New York: Taylor and Francis, 1995, pp. 3–17.

- [5] D. H. Douglas and T. K. Peucker, "Algorithm for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature," *The Canadian Cartographer*, vol. 10, no. 2, pp. 116–117, 1973.
- [6] B. P. Buttenfield and R. B. McMaster, "Making Rules for Knowledge Representation: Map Generalization," *Longman Scientific and Technical*, pp. 121–135, 1991.
- [7] L. Wu, *Geographical Information System: Principles and Applications*. Beijing: Science Press, 2002.
- [8] H. Zheng, *Theory of Mapping and Cartography Based Computer*. Zhengzhou: Institute of Surveying and Mapping, 1997.
- [9] W. Fu, "Expression of Knowledge and Organization of Knowledge base in Geographical Expert System," *Journal of Applied Science*, vol. 15, no. 4, pp. 482–489, 1997.
- [10] L. Wu and W. Shi, *Theory and Algorithm of Geographical Information System*. Beijing: Science Press, 2003.
- [11] B. Boucheham and Y. Ferdi, "Recursive Versus Sequential Multiple Error Measures Reduction: A Curve Simplification Approach to ECG Data Compression," *Computer Methods and Programs in Biomedicine*, vol. 81, pp. 162–173, 2006.
- [12] Z. Zhu. (2006) Geographical Information System Technology. National Taiwan University. [Online]. Available: <http://www.geog.ntu.edu.tw/course/gistech/>
- [13] M. Visvalingam and J. Whyatt, "Line Generalization by Repeated Elimination of Points," *Cartographic Journal*, vol. 30, no. 1, pp. 46–51, 1993.
- [14] P. Hu, X. Huang, and Y. Hua, *Tutorial of Geographical Information System*. Wuhan: Wuhan University Press, 2002.
- [15] K. Ebisch, "A Correction to the Douglas - Peucker Line Generalization Algorithm," *Computers and Geosciences*, vol. 28, pp. 995–997, 2002.
- [16] L. Wu and W. Shi, *Geographical Information System: Principles and Algorithms*. Beijing: Science Press, 2003.
- [17] J. Wang and G. Jiang, "Researching and Realization of the Quick Compression Method aimed at the Non-Topology Vector Data," *Journal of Surveying and Mapping*, vol. 32, no. 2, pp. 173–177, 2003.
- [18] X. Liu and S. Li, "Study on Subsection Douglas Algorithm with the Goniometry in Generalization," *Journal of Surveying and Mapping*, vol. 28, no. 2, pp. 51–52, 2005.