International Journal of Multidisciplinary Research and Growth Evaluation
ISSN: 2582-7138
Received: 25-07-2021; Accepted: 11-08-2021
www.allmultidisciplinaryjournal.com
Volume 2; Issue 4; July-August 2021; Page No. 823-826

Vertices random selection for vector map data in encryption process

Giao N Pham
Department of Computing Fundamentals, FPT University, Hanoi, Vietnam

Corresponding Author: **Giao N Pham**

Abstract
Nowadays, a large volume of valuable vector map dataset has been distributed illegally by pirates, hackers, or unauthorized users. Thus, the problem focuses on how to protect the copyright of vector map data for storage and transmission. This paper presents an encryption algorithm for vector data. In proposed algorithm, vector map is separated to select polyline/polygon layer. We use three simplification algorithms to define the feature points in each object. After that, we select randomly vertices of objects based on the ratio of the feature points and encrypt them with key values. Experimental results show the high efficiency visualization by low complexity, high security performance and cryptography. In addition, experiments also show unique performance, decryption error approximate zero and computation time is better than the exist algorithms.

Keywords: Vector Map, Data security, Secret Communication, Chaotic-map

1. Introduction
In vector map security, conventional approaches encrypt whole data, so the cryptography of data files and profiles increase complexity, and these methods take a long time. Furthermore, the decryption data often occur loss data and it take a long time to processing time, because authors use complex computation process on large data. It is not also flexible for various data types. Specially, database management system based on security technique is vulnerable by the conversion between data formats. Moreover, current security techniques focus on access of users via internet but the network security technique cannot preserve the security in case of data leakage on off-line or loophole exposure of network administration. So, the security technique for vector map had to preserve the security in various formats of vector map data, reduce complex computation and encrypted data volume.

To solve that problem, researchers gave watermarking schemes and encryption methods focus on different domains. Looking for the recent security techniques of vector map, the network security techniques for secure transmission or storage and copyright protection of vector map data have been mainly researched [1-6]. Researchers worked data encryption based on vector map database files or data profile using the cryptography and worked the watermarking of vector map for copyright protection [7, 8]. In fact, the watermarking is only useful for identifying ownership, copyright and prevent illegal distribution while providers desiderate unauthorized users or pirate cannot see and attack to extract the content of vector map in the most cases. Thus, the data encryption is necessary to protect vector map.

To compensate for limitations in the conventional approaches described above, a new selective encryption algorithm is proposed in our paper for multimedia applications, storage and transmission for vector map security. In proposed algorithm, vector map is separated to select polyline/polygon layer. We use three simplification algorithms to define the feature points in each object. After that, we select randomly vertices of objects based on the ratio of the feature points. For generating random numbers and key values, we combine SHA-512-bit hashing [9] and 2D Chaotic-map [10]. Finally, the selected vertices are changed their position by mixing with these key values. Main advantages of our algorithm are randomly vertices selection and transforming processes but it still meets requirements of security by random processes, and this algorithm can be implement to many type of vector map formats. In experimental results, we verify the high efficiency visualization by low complexity, high security performance by random processes and cryptography. In addition, experiments also show unique performance, decryption error approximate zero and computation time be very short.

2. The Proposed Method

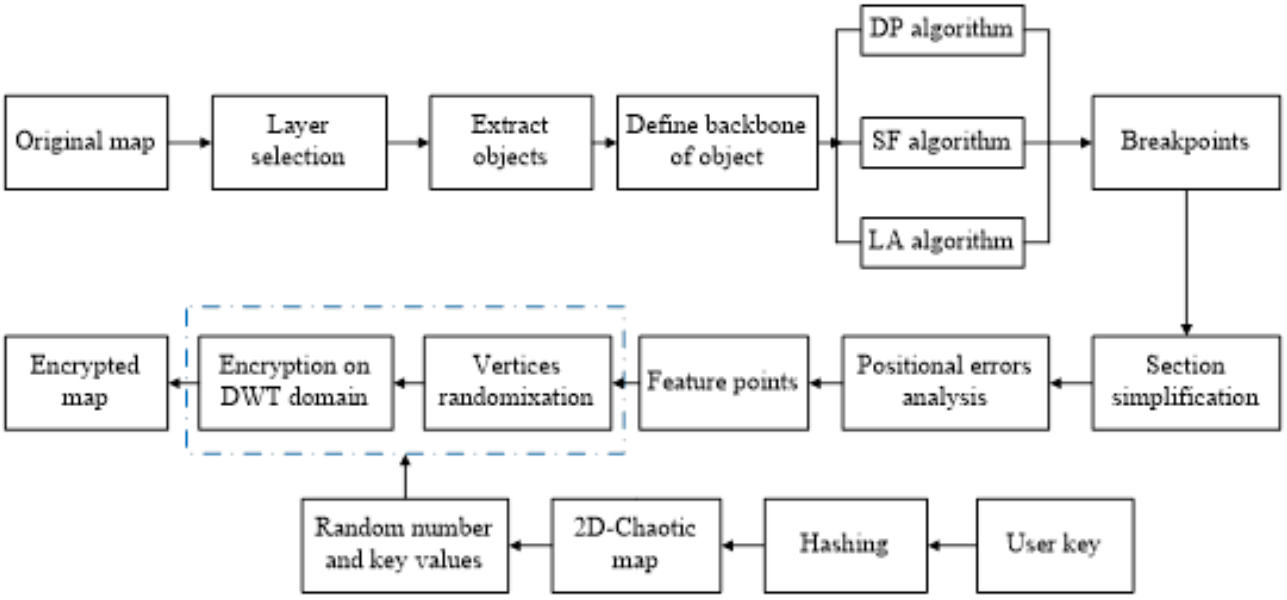


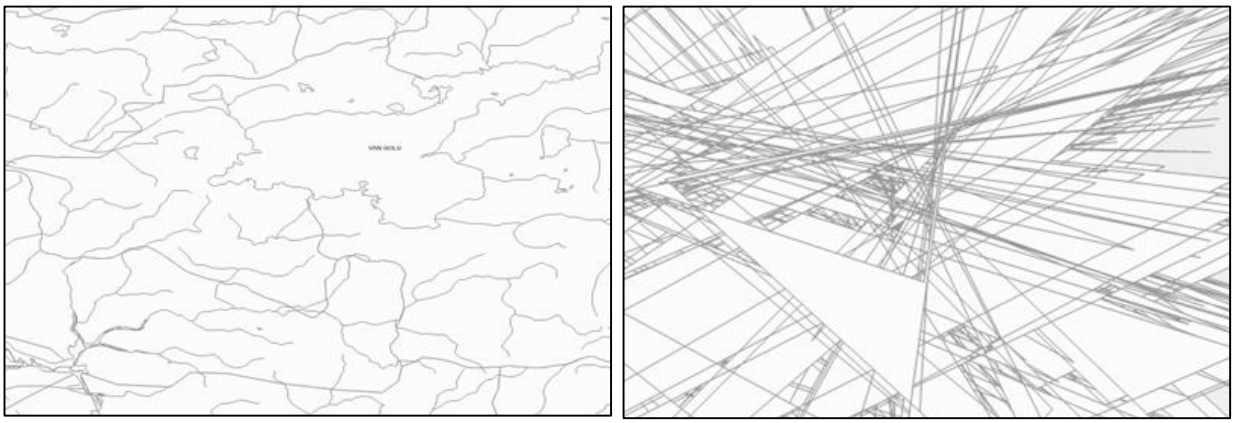
Fig 1: The Proposed Method

In this paper, the proposed method selects randomly vertices for encryption based on simplification algorithms. In Fig. 1, we show the schematic of the algorithm, and the step-by-step is explained in detail, as follows:

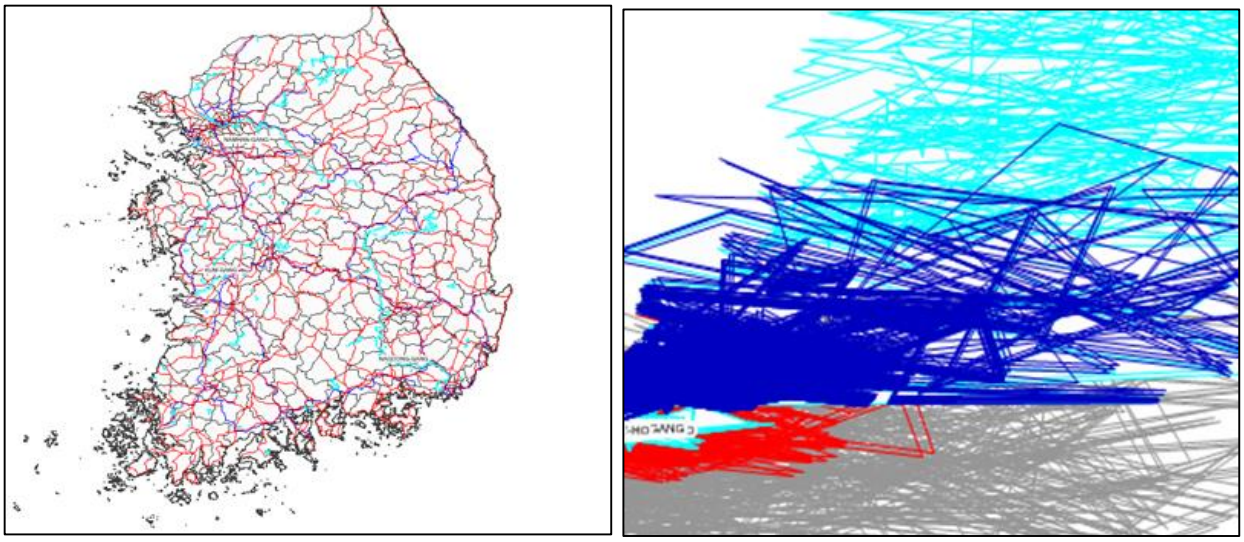
- A vector map $\boldsymbol{M}$ is defined as a set of layers: $\boldsymbol{M} = \{L_i | i \in [1, |M|]\}$ with $|\boldsymbol{M}|$is *the cardinality* in map $\boldsymbol{M}$ (the cardinality of a set is a measure of the "number of elements of the set" in mathematics).
- A layer $\boldsymbol{L}_i$ include many polyline/polygon objects: $\boldsymbol{L}_i = \{O_{ij} | j \in [1, |L_i|]\}$ with $|\boldsymbol{L}_i|$ is the total number of objects in layer and $|\boldsymbol{L}_i|$ is also the cardinality in a layer $\boldsymbol{L}_i$.
- An object $\boldsymbol{O}_{ij}$ is a set of vertices $\boldsymbol{O}_{ij} = \{v_{ijk} | k \in [1, |O_{ij}|]\}$ with $|\boldsymbol{O}_{ij}|$ is the number of the pair coordinates of vertices in object $\boldsymbol{O}_{ij}$. Then, we define a backbone of the object.
- Our method processes all the backbones in layers using Douglas-Peucker (***DP***), Sleeve-fitting (***SF***), and Lang (***LA***) algorithms [11] to define breakpoints of objects. After that, we analyze the positional errors of three exist algorithms to identify feature points.
- Finally, we select vertices based on the ratio of the feature points; generating randomization numbers and key values using SHA-512 hashing and 2D Chaotic-map; and then we encrypt the selected vertices in DWT domain.

3. Experimental Results

For performance evaluation of our encryption algorithm, we used many different scaled maps that contain layers, including rivers, roads, lakes, and countries. Experimental results prove that the encrypted map change absolutely perception of whole maps, as shown in Fig. 2. The proposed method changes whole visual image of map because we encrypted randomly vertices lead to scrambling shape of all objects. In addition, our method changes only vertices position of objects in the layer. The size of input and output map are same, so it is not lost data.



A. B.



A. B.

Fig 2(a): Turkey original water lines layer, **(b)** the encrypted layer, **(c)** Korean original full map (5 layers), and **(d)** the encrypted map.

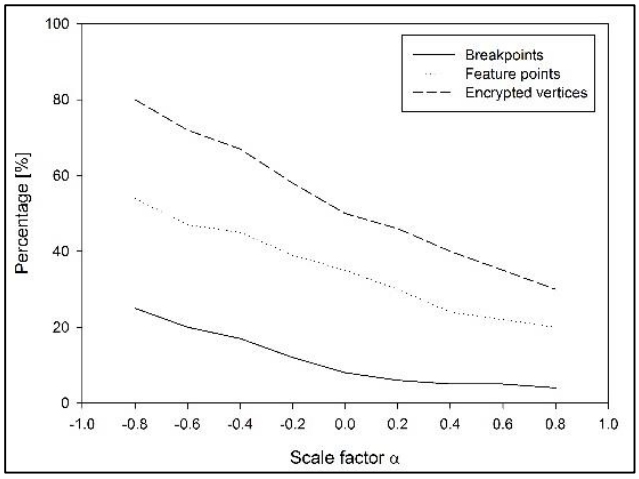


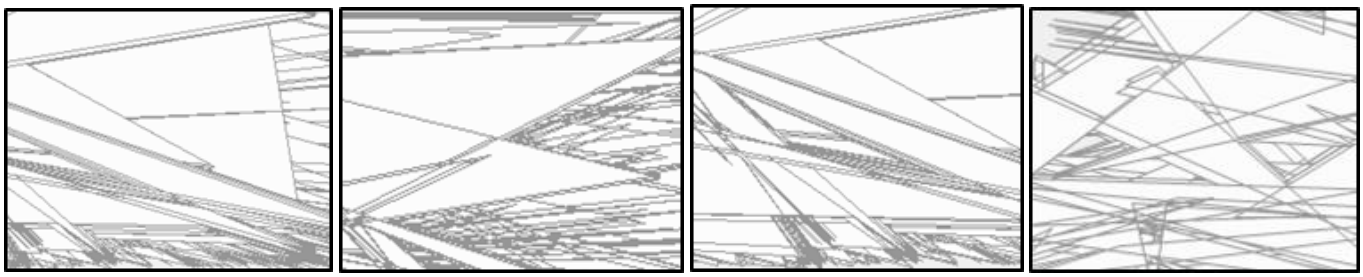
Fig 3: Ratio of breakpoints, feature points, and encrypted vertices of my method on the scale factor (Calculation with Turkey layer).

The breakpoints in our method are defined based on the scale factor α that control the threshold parameters of three simplification algorithms. The feature points are defined by using the breakpoints. After that, we select vertices based on ratio of feature points in object for encryption. Fig. 3 shows the ratio of breakpoint number, feature point number and encrypted vertices calculated by formulas: number of breakpoints/ number of vertices of object's backbone, number of feature points/ number of vertices of object's backbone, number of encrypted vertices/ number of vertices of object.

In this algorithm, we only changes coordinates of points (vertices) in objects. The encrypted map is still same size with the original map. However, we use 2D Chaotic-map to generate random numbers and key values from user's key hashing SHA-512 bits, it makes these values not absolutely similar in encryption and decryption step. Meaning of this issues come from the problem of system calculation when it stored real numbers in memory. With storing vertices in double type, it seems no problem when the decryption errors values are approximately zero and we tested with many maps to find max error and calculate average error, as shown in Table 1.

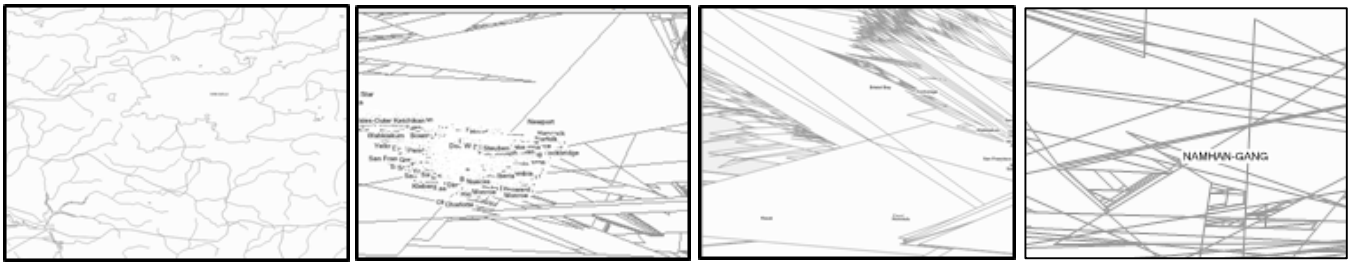
Table 1: The max, min error between original map and decryption map

Size (kb)	Total Object	Total Point	Max error	Min error	Average error
449	147	28162	3.70411E-07	0	1.72524E-08
965	7011	37209	1.13562E-07	0	2.92078E-10
1246	375	79499	6.41549E-07	0	6.00142E-08
1730	13960	61798	7.83001E-08	0	1.76301E-09
2246	575	88948	8.31949E-07	0	7.0112E-08



(a) Encrypted layer E_1 (b) Encrypted layer E_2 (c) Encrypted layer E_3 (d) Encrypted layer E_4

Fig 4: Key sensitivity analysis for encryption process



(a) Correct key (K_1) (b) Wrong key (K_2) (c) Wrong key (K_3) (d) Wrong key (K_4)

Fig. 5 Key sensitivity analysis for decryption process

For security evaluation, we generate slightly different keys by modifying first key x_1, parameter μ_1 used to generate key values in Chaotic-map equation and scale factor α. And then, we change one of them when keeping other values in the modified key. So, when test key sensitivity, we use the original key with three components $K: (x_1, \mu_1, \alpha)$. We generate key $K_1: (0.62, 2.9, -0.6)$, the modified keys are expressed as $K_2: (0.75, 2.9, -0.6)$, $K_3: (0.62, 3.1, -0.6)$, $K_4: (0.62, 2.9, 0.1)$. We use key values that are created from K_1 to encrypt the original layer (***TU*** map) and generate the first encrypted map. The encrypted layer E_1 for this case is shown in Fig. 8(a). The original layer is then encrypted with key sets are generated from slightly modified key K_2, K_3 and K_4. Fig. 4(b)-(d) perform the corresponding encrypted layers. It is observed that layers encrypted with slightly different keys are completely incomprehensible. This mean, "cipher texts generated using slightly different keys are completely different from each other". With the second provision of key sensitivity, encrypted layer is decoded with key K_2, K_3, K_4 instead of the correct key K_1, as shown in Fig. 5. The layer is decoded by using incorrect keys, it is completely incomprehensible, and do not leak any information about the original layer. In contrast, decryption using actual key retrieves the layer correctly. This depicts the high key sensitivity of the proposed cryptosystem, and also verifies the second condition of Kerckhoff's principle [12], "decryption using a wrong decryption key should not reveal any information".

4. Conclusion

In our paper, we created a new method which aim to reduce ratio of encrypted data in vector map but still assure performance and high security. This considers how to select randomly vertices of object in a layer by using simplification algorithms. After that, the selected vertices are encrypted with random numbers and key values generating from 2D Chaotic map. We also confirm that: Human perception do not see any information in encrypted map, poor error in decryption step, computation time is less than it in the existing methods, high security and a large amount of vector map data can be protected by this algorithm. The algorithm can be used in many kinds of application or standard vector map because we proposed to encrypt randomly vertices of important/complex objects (polygons and polylines), so it can be applied for any vector map data and database on on/off-lines server.

5. Acknowledgments

This work is supported by FPT University, Hanoi, Vietnam

6. Disclosure of conflict of interest

On behalf of all authors, corresponding author declares that there is no conflict of interest to publish this research.

7. References

1. E Bertino, ML Damiani. A Controlled Access to Spatial Data on Web, Proc. Conference on Geographic Information Science, 2004, 369-377.
2. SC Chena, X Wang, N Rishea, MA Weiss, A Web-based Spatial Data Access System using Semantic R-trees, Journal of Information Sciences. 2003-2004; 167(1-4):41-61.
3. E Bertino, B Thuraisingham, M Gertz, ML Damiani. Security and Privacy for Geospatial Data: Concepts and Research Directions, Proc. of the SIGSPATIAL ACM GIS 2008 International Workshop on Security and Privacy in GIS and LBS, 2008, 6-19.
4. NB Rybalov, OI Zhukovsky. Access to the Spatial Data in the Web-Oriented GIS, Proc. Siberian Conference on Control and Communications, 2007, 104-107.
5. M Fuguang, G Yong, Y Menglong, X Fuchun, L Ding. The Fine-grained Security Access Control of Spatial Data, Proc. 18th International Conference on Geoinformatics, 2010, 1-4.
6. F Wu, W Cui, H Chen. A Compound Chaos-Based Encryption Algorithm for Vector Geographic Data under Network Circumstance, Proc. of Cardholder Information Security Program. 2008; 1:254-258.
7. G Li. Research of Key Technologies on Encrypting Vector Spatial Data in Oracle Spatial, Proc. of International Conference on Industrial Electronics and Computer Science, 2010, 1-4.
8. Y Dakroury, IA El-ghafar, A Tammam. Protecting GIS Data Using Cryptography and Digital Watermarking, International Journal of Computer Science and Network Security. 2010; 10(1):75-84.
9. RSA Laboratories. PKCS #5 v2.1: Password-Based Cryptography Standard, 2006.
10. C Pellicer-Lostao, R López-Ruiz. Pseudo-Random Bit Generation Based on 2D Chaotic Maps of Logistic Type and Its Applications in Chaotic Cryptography, Computational Science and Its Applications-ICCSA, 2008, 784-796.
11. W Shi, C Cheung. Performance Evaluation of Line Simplification Algorithms for Vector Generalization, The Cartographic Journal. 2006; 43(1):27-44.
12. Kerckhoffs's principle, http://en.wikipedia.org/wiki/Kerckhoffs's_principle.html, accessed to Mar. 2016.