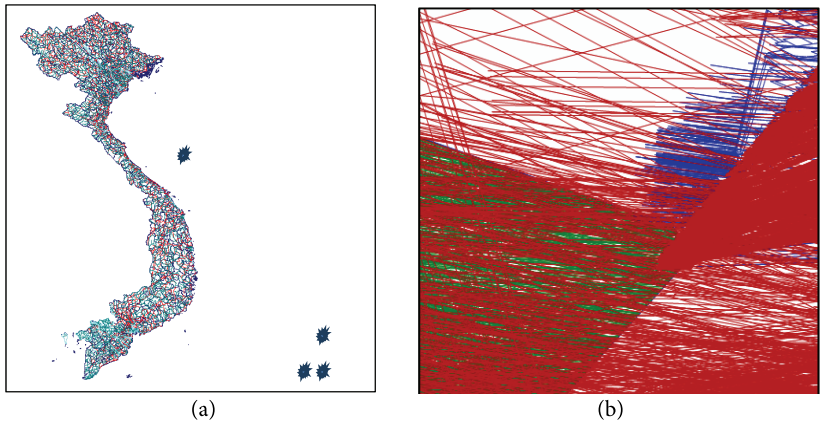


(a) (b)

Fig. 11. Experimental result with full scaling layers 1:5,000. (a) Original map and (b) encrypted map.

4.3 Decryption Error

We used 1:5,000, 1:10,000 and 1:100,000 scaling maps in our experiments. Firstly, we experimented with each polyline layer and polygon layer. Then, we tested the full map of 1:5,000 scale. Experimental results show that all maps are changed as given by Figs. 9–11. Unauthorized users cannot see anything on the map because we changed polyline/polygon through many processes: multiplication, DWT, IDWT, DFT and IDFT.

Table 1. The error between original coordinates and decrypted coordinates

Original coordinates	Decryption coordinates	Error
144.13246	144.132460000128	1.27982957565109E-10
-37.32808	-37.3280800000347	3.47029072145233E-11
…	…	…
-118.749289	-118.749289001048	2.83009171653248E-10
34.817734	34.8177340000613	6.12985218140238E-11

Table 2. The max, min error between original map and decryption map

Size (kB)	Total object	Total point	Max error	Min error	Average error
332	76	20920	4.58763E-07	0	1.75211E-08
449	147	28162	3.70411E-07	0	1.72524E-08
751	20	47947	2.88627E-07	0	4.00043E-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

In this algorithm, only coordinates of points (vertices) in objects are changed. The encrypted map still have the same size as the original map. However, Chaotic map is used to generate random number and key values from user's key hashing SHA-512 bits, it make 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 be no problem when the decryption errors values are approximately zero, as given by Table 1. Then, many maps are tested to find the maximum error and calculate the average error, as shown in Table 2.

4.4 Distance Measure

We use Eq. (7) to calculate the difference between encrypted map and original map:

$$D(E', L) = \sum_{i=1}^{N} d(P_{ij}) \tag{7}$$

where L is an original map and $E'(L)$ is corresponding encrypted map and N is total object in original map. $d(P_{ij})$ is distance between corresponding objects in $E'(L)$ and L, which is computed by

$$d(P_{ij}) = \sum_{j=1}^{N_{ij}} \sqrt{(|x_j' - x_j|^2 + (|y_j' - y_j|^2)} \tag{8}$$

where N_{ij} is the total number of points in object P_{ij}.

We used polyline map, polygon map to experiment with different passwords K_1 and K_2. Then we calculate $D(E', L)$ distance of each experimental time, as show in Table 3.

Table 3. Experimental distance measure

Total number of points	Distance	
	User key K_1	User key K_2
798	35,682	39,065
1249	53,018	50,682
2457	200,133	178,431
2967	233,311	200,644
3900	318,270	404,741

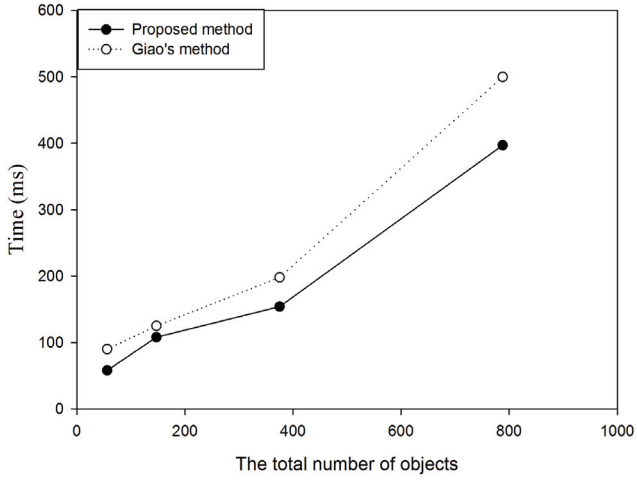


Fig. 12. Computation time according to the total number of objects.

4.5 Computation Time

The computation time of proposed methods depends on many factors. Moreover, it also depends on random processes, size of the map. Therefore, to measure the computation time we need to use specific mathematical model. In this section we just show the dependence of computation time and we compared result with Giao's algorithm [20,21] under same the total number of objects, as shown in Fig. 12. Analyzing the detail results, we verify that the computation time of our method have lower than those of Giao's method.

4.6 Security

Cryptographic security: Our proposed method use threshold values to select the significant objects, this method encrypted 60%-70% of data with the key values and random coefficients created by using Chaotic map and secret key.

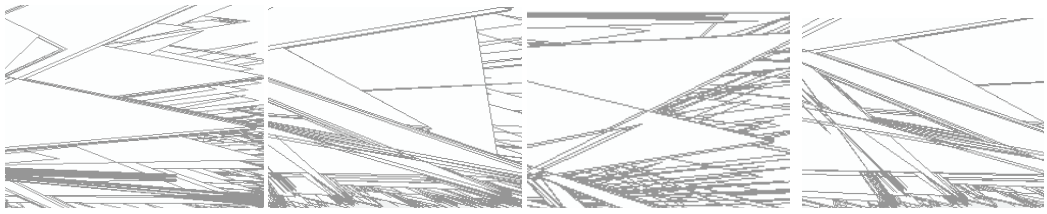
Key sensitivity analysis: A highly key sensitive encryption algorithm protects the encrypted data against various cryptanalytic attacks. While developing a cryptosystem, it is assumed that an intruder knows the encryption structure and a-priori probability of used key $k \in K$. As per the Kerckohoff's principle [22], only secrecy of the used key is required. Even a strong or well-designed cryptosystem can be broken easily if the key is poorly chosen or key space is too small. Thus, a good cryptosystem should satisfy the following two conditions to verify the key sensitivity and key space:

- The key space should be discretized in such a way that two ciphertexts encrypted by two slightly different keys $k_1, k_2 \in K$ should be completely different.
- With the generated keys, the ciphertext should be responding to slight changes, signals, or influences.

The original layer is encrypted by using the slightly different keys, and we analysis the difference between the obtained encrypted layers. The different layers are evaluated to verify the condition that, "layer is encrypted with slightly different keys should be completely different".

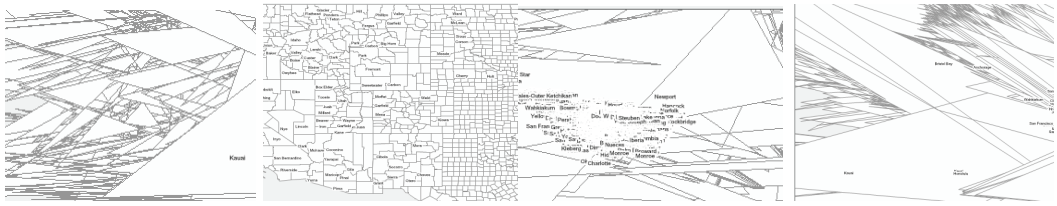
For security evaluation, the slightly different keys are generated by modifying the first key in each key set *a, b* and modifying coefficient μ in Eq. (3). And then, the method change one of them when keeping other values in the modified key. So, when test the key sensitivity, algorithm use the original key with three components. For the original key K_1: $(3.52, 0.34, 0.62)$ (three parameters (μ, a_1, b_1) represent for key K), the modified keys are expressed as K_2: $(3.42, 0.34, 0.62)$, K_3: $(3.52, 0.44, 0.62)$ and K_4: $(3.52, 0.34, 0.52)$. We use K_1 to generate an encrypted layer of the original layer (Fig. 9(c)) and Fig. 13(a) show image of this layer. After that, we continue to generate other encrypted layers with the slightly modified keys K_2, K_3, and K_4. Fig. 13(b)–(d) indicates the corresponding encrypted layers. It is observed that layers encrypted with slightly different keys are completely incomprehensible. This verifies that, "ciphertexts generated using slightly different keys are completely different from each other".

With another provision of key sensitivity, the encrypted layer is decoded with key K_2, K_3 instead of the correct key K_1, as shown in Fig. 14. The layer is decoded with incorrect keys that completely incomprehensible, and do not leak any information about the original layer.



(a) (b) (c) (d)

Fig. 13. Key sensitivity with original layer (Fig. 9(c)) using different keys. (a) Key K_1, (b) key K_2, (c) key K_3, and (d) key K_4.



(a) (b) (c) (d)

Fig. 14. Key sensitivity analysis for decryption process. (a) Proposed encrypted map, (b) decypted map with K_1, (c) decrypted map with K_2, and (d) decrypted map with K_3.

4.7 Algorithm Comparison

We compare the properties of the proposed method with the existing encryption method [20,21]—authors use DCT transform for GIS vector map encryption. In these papers, authors select all vertices in a layer and encrypt them by random algorithm in DCT domain, they did not consider important part in each layer. So, this is full encryption methods and still need very long computation time (we proved it in Sections 4.1 and 4.5). Our method only select the significant objects (that is defined by owner based on threshold values) for encryption, it would be very difficult to break the encryption algorithm or try to predict the encrypted part. Furthermore, authors only used a set of random value to encrypt coefficients in DCT domain and they didn't clearly explain in security evaluation part. We think that it should be broken easily because the key is poorly chosen and key space is too small.

5. Conclusion

In this paper, a new method is proposed which aim to reduce the ratio of encrypted data in GIS vector map but still assure the performance and the high security. This considers how to select significant objects in a layer by using threshold values. After that, the selected vertices are encrypted with random numbers, key values generating from chaotic map and hybrid transforms. We 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 GIS vector map data can be protected by this algorithm. The algorithm can be used in many kind of application or standard vector map because it is proposed to encrypt randomly vertices of important/complex objects (polygons and polylines), so it can be applied for any vector map data and GIS database on on/off-lines server.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2014R1A1A4A01006663 and NRF-2016R1D1A3B03931003) and the ICT R&D program of MSIP/IITP (R0126-15-1112, Development of Media Application Framework based on Multi-modality which enables Personal Media Reconstruction).

References

[1] K. E. Foote and M. Lynch, "Geographic information systems as an integrating technology: context, concepts, and definitions," 2014 [Online]. Available: http://www.colorado.edu/geography/gcraft/notes/intro/intro_f.html.

[2] M. F. Goodchild, "Twenty years of progress: GIScience in 2010," *Journal of Spatial Information Science*, vol. 2010, no. 1, pp. 3-20, 2010.

[3] The GIS spatial data model, 2015 [Online]. Available: https://courses.washington.edu/gis250/lessons/introduction_gis/spatial_data_model.html.

[4] GIS digital vector maps, 2015 [Online]. Available: http://www.mapmart.com/Products/DigitalVectorMapping.aspx.

[5] Vector data, 2015 [Online]. Available: https://docs.qgis.org/2.8/en/docs/gentle_gis_introduction/vector_data.html.

[6] Advantage of vector map, 2015 [Online]. Available: https://maxdisruption.wordpress.com/2011/01/14/advantage-and-disadvantages-of-vectorraster-data/.

[7] E. Bertino and M. L. Damiani, "A controlled access to spatial data on web," in *Proceedings of 7th AGILE Conference on Geographic Information Science*, Crete, Greece, pp. 82-91, 2004.

[8] S. C. Chen, X. Wang, N. Rishe, and M. A. Weiss, "A web-based spatial data access system using semantic R-trees," *Journal of Information Sciences*, vol. 167, no. 1-4, pp. 41-61, 2004.

[9] R. Ohbuchi, H. Ueda, and S. Endoh, "Robust watermarking of vector digital maps," in *Proceedings of IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, 2002, pp. 577-580.

[10] M. Voigt and C. Busch, "Feature-based watermarking of 2D vector data," in *Proceedings of the SPIE 5020: Security and Watermarking of Multimedia Content*. Bellingham, WA: International Society for Optics and Photonics, 2003, pp. 359-366.

[11] G. Schulz and M. Voigt, "A high capacity watermarking system for digital maps," in *Proceedings of the 2004 Workshop on Multimedia and Security*, Magdeburg, Germany, 2004, pp. 180-186.

[12] C. Wang, Z. Peng, Y. Peng, L. Yu, J. Wang, and Q. Zhao, "Watermarking geographical data on spatial topological relations," *Multimedia Tools and Applications*, vol. 57, no. 1, pp. 67-89, 2012.

[13] S. H. Lee and K. R. Kwon, "Vector watermarking scheme for GIS vector map management," *Multimedia Tools and Applications*, vol. 63, no. 3, pp. 757-790, 2013.

[14] F. Wu, W. Cui, and H. Chen, "A compound chaos-based encryption algorithm for vector geographic data under network circumstance," in *Proceedings of 1st International Congress on Image and Signal Processing*, Sanya, China, 2008, pp. 254-258.

[15] G. Li, "Research of key technologies on encrypting vector spatial data in oracle spatial," in *Proceedings of 2nd International Conference on Information Engineering and Computer Science*, Wuhan, China, 2010, pp. 1-4.

[16] Y. Dakroury, I. A. El-ghafar, and A Tammam, "Protecting GIS data using cryptography and digital watermarking," *International Journal of Computer Science and Network Security*, vol. 10, no. 1, pp. 75-84, 2010.

[17] B. J. Jang, S. H. Lee, and K. R. Kwon, "Perceptual encryption with compression for secure vector map data processing," Digital Signal Processing, vol. 25, pp. 224-243, 2014.
[18] Y. Pomeau and P. Manneville, "Intermittent Transition to Turbulence in Dissipative Dynamical Systems," *Communications in Mathematical Physics*, vol. 74, no. 2, pp. 189–197, 1980.
[19] RSA Laboratories, PKCS #5: Password-Based Cryptography Standard (version 2.1) [Online]. Available: https://germany.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-5-password-based-cryptography-standard.htm.
[20] P. N. Giao, S. H. Lee, and K. R. Kwon, "Selective encryption algorithm for GIS vector map using geometric objects," *International Journal of Security and Its Applications*, vol. 9, no. 2, pp. 61-72, 2015.
[21] P. N. Giao, G. C. Kwon, S. H. Lee, and K. R. Kwon, "Selective encryption algorithm based on DCT for GIS vector map," *Journal of Korea Multimedia Society*, vol. 17, no. 7, pp. 769-777, 2014.
[22] Wikipedia, Kerckhoffs's principle [Online]. Available: http://en.wikipedia.org/wiki/Kerckhoffs's_principle.

Bang Van Nguyen

He received a B.S. degree in School of Electronic & Telecommunication from Hanoi University of Science & Technology (HUST) in 2014. Currently, he is a Master student in Multimedia Communication & Signal Processing Lab in Pukyong National University. His research interests include video processing & application, GIS applications, data security, and smart system.

Suk-Hwan Lee http://orcid.org/0000-0003-4779-2888

He received the B.S., M.S., and Ph.D. degrees in Electrical Engineering from Kyungpook National University, Korea in 1999, 2001, and 2004, respectively. He is currently an associate professor in Department of Information Security at Tongmyong University. His research interests include multimedia security, digital image processing, and computer graphics.

Ki-Ryong Kwon

He received the B.S., M.S., and Ph.D. degrees in electronics engineering from Kyungpook National University in 1986, 1990, and 1994, respectively. He worked at Hyundai Motor Company from 1986–1988 and at Pusan University of Foreign Language from 1996–2006. He is currently a professor in Department of IT Convergence and Application Engineering at the Pukyong National University. He has researched University of Minnesota in USA on 2000–2002 with Post-Doc. and Colorado State University on 2011–2012 with visiting professor. He is currently the President of Korea Multimedia Society. His research interests are in the area of digital image processing, multimedia security and watermarking, bioinformatics, weather radar information processing.