Journal of Korea Multimedia Society Vol. 20, No. 8, August 2017(pp. 1312-1320)
https://doi.org/10.9717/kmms.2017.20.8.1312

Selective Encryption Algorithm for Vector Map using Geometric Objects in Frequency Domain

Ngoc-Giao Pham[†], Ki-Ryong Kwon[††], Suk-Hwan Lee[†††], Chong-Ho Woo[††††]

ABSTRACT

Recently, vector map data is developed and used in many domains widely. In the most cases, vector map data contains confidential information which must be kept away from unauthorized users. Moreover, the production process of vector maps is considerably complex and consumes a lot of money and human resources. Therefore, the secured storage and transmission are necessary to prevent the illegal copying and distribution from hacker. This paper presents a selective encryption algorithm using geometric objects in frequency domain for vector map data. In the proposed algorithm, polyline and polygon data in vector map is the target of the selective encryption process. Experimental results verified that proposed algorithm is effectively and adaptive the requirements of security.

Key words: Vector Map Data, Selective Encryption, Geometric Object and Discrete Cosine Transformation

1. INTRODUCTION

Vector map is a vector-based collection of Geographic Information System (GIS) data about earth at various levels of detail. Vector map is created and developed by the merging system of cartography, statistical analysis, and database technology based on vector model [1]. Vector map data is used in many domains but the producing process of vector maps consumes a lot of money and human resources. And any company can buy it, make illegal copies and distribute them easily many times without taking any permission from original providers. So the protection for vector map is necessary to prevent the illegal duplication and distribution of it.

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 [2-10]. Researchers worked based on the vector map database files using the cryptography and the watermarking vector map for copyright protection. In fact, the watermarking is only useful for identifying ownership, copyright while providers desiderate unauthorized users or pirate cannot see and attack to extract the content of vector map in the most cases. Thus, data encryption is necessary to protect vector map. But the full encryption techniques often encrypt the entire data, which includes data that does not need to be encrypted. This lead to the full encryption process

※ Corresponding Author : Chong-Ho Woo, Address: (48513) 45, Yongso-ro, Nam-gu, Busan, Pukyong National University, Daeyon Campus, # A.13, No. 2309 TEL : +82-51-629-6250, FAX : +82-51- 629-6230, E-mail : chwoo@pknu.ac.kr
Receipt date : Apr. 4, 2017 Revision date : Jun. 9, 2017 Approval date : Jun. 27, 2017
[†] Dept. of IT Convergence and Application Engineering, Pukyong National University
(E-mail : ngocgiaofet@gmail.com)
[††] Dept. of IT Convergence and Application Engineering, Pukyong National University
(E-mail : krkwon@pknu.ac.kr)
[†††] Dept. of Information Security, Tongmyong University
(E-mail : skylee@tu.ac.kr)
[††††] Dept. of Computer Engineering, Pukyong National University
※ This research was supported by a Research Grant of Pukyong National University (2016 year).

has to compute complexity and spend long time for both the encryption and decryption process. Moreover, the data decryption process often occurs loss data and decryption errors by the complex computation on large data. Consequently, selective encryption is necessary for vector map data security.

For meeting above requirements, we present a novel selective encryption algorithm for vector map data in this paper. In proposed algorithm, polyline and polygon objects are selected to perform selective encryption in the frequency domain of discrete cosine transformation (DCT domain). To clarify the proposed algorithm, we look into the vector map security techniques and discuss the related works in Sec. 2. In Sec. 3, we explain the proposed algorithm in detail. The experimental results, performance evaluation and conclusion will be described in Sec. 4 and Sec. 5.

2. Related Works

2.1 Vector map security

Bertino et al. [5], Chena et al. [6], and Rybalov et al. [7] presented approaches to the definition of an access control system for spatial data on the Web. But access control and management on Web or database do not maintain security in the outflow of an authenticated user. Relating storage and transmission of vector map, data should be encrypted before storing and transmitting. Wu et al. [8] proposed a compound chaos-based encryption algorithm for vector map data by considering the storage characteristics and the parameters of chaos-based systems. This algorithm is not available to various data formats and object indexing. Li et al. [9] encrypted the vector dataset in external Oracle DBMS by using DES and an R-Tree spatial index. This algorithm encrypts the spatial index when the GIS dataset is transmitted to the client and designs the key management of public and private keys on a PKI system. Yasser et al. [10] also described better encrypting algorithm which combined AES and RSA cryptography with a simple watermarking technique for the copyright protection of vector maps in on/off line service. This algorithm encrypts all parts of a shape-file using an AES block cipher operator of 256 bits. Bang et al. [11] proposed an encryption method for vector map data based on Chao-tic map; however this method only selects some objects and encrypts them by the common secret key. This method is very weak and simple because it did not encrypt all contents of vector maps. Moreover, this method did not show the performance evaluation, the security evaluation.

2.2 Vector map data

Vector map data is stored in layers. Each layer contains geometric objects as point, polyline and polygon. Point is used to represent simple objects while polygon and polyline are used for representing complex objects. Thus, polyline and polygon are considered to be very important components of vector map. Beside the geographical information, vector map also includes attribute, display information as header, text and notation. They are attributed data. Fig. 1 shows vector map data model and the components of vector map data. Due to the fact that, attribute data does not contain geographical information. Thus, it does not determine the content of vector maps. We only need to select geometric objects to perform selective encryption process.

3. THE PROPOSED ALGORITHM

3.1 Definition

Fig. 2 shows the general selective encryption for vector map data. Polyline and polygon are extracted from vector map to perform selective encryption. To simple notation, we consider polyline and polygon as an object with a series of vertices which polyline and polygon are denoted as the

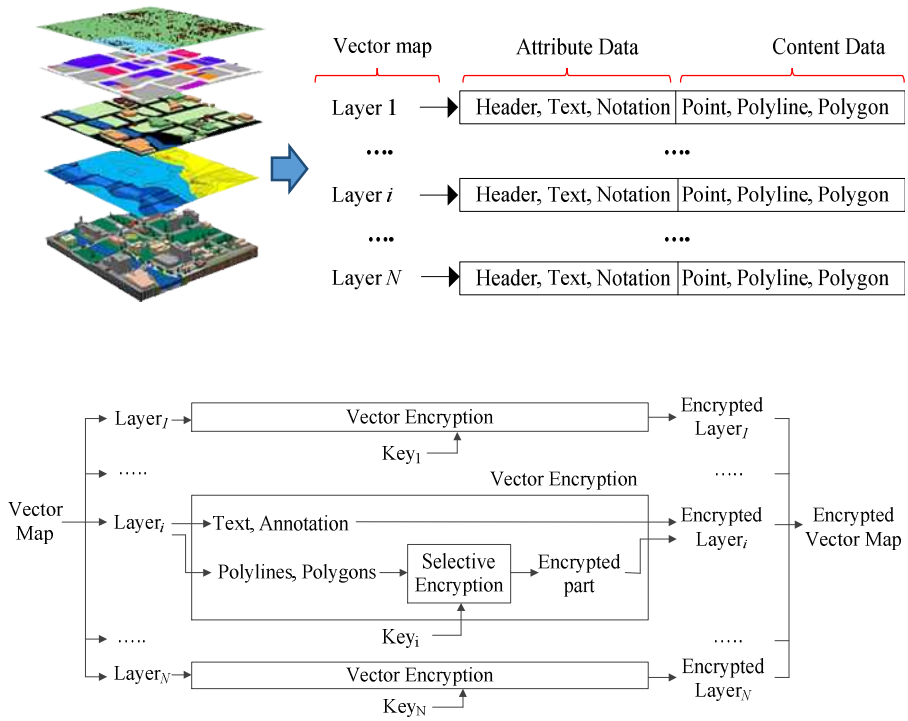

Fig. 2: General selective encryption for vector map data.

same object. Let us look over main notations in the following section and then describe proposed algorithm in detail in next sections. Main notations for describing vector map in our paper are defined as following.

• An original vector map includes a set of layers. Each layer $\mathbf{L}_i$ is a set of objects of polylines and polygons $\mathbf{L}_i = \{\mathbf{P}_{ij} | j \in [1, N_i]\}$

• An object $\mathbf{P}_{ij}$ consists of a series of vertices $\{\boldsymbol{v}_{ijk} | k \in [1, N_{ij}]\}$. Each vertex is represented by a pair of coordinates $\boldsymbol{v}_{ijk} = (x_{ij,k}, y_{ijk})$. $\mathbf{P}_i^{\max}$ is an object with the maximum number of vertices N_i^{max} in $\mathbf{L}_i$. The total number of vertices in a layeris $\mathbf{L}_i$ is N_{vi}.

The followings are main notation for describing the encryption process:

• $\mathbf{r}_i = \{r_{ik} | k \in [1, N_i^{max}]\}$ is random vector for randomizing vertices, created by key value $\mathbf{K}_i$

• Randomized object is $\mathbf{P'}_{ij} = \{\boldsymbol{v'}_{ijk} | k \in [1,\ N_{ij}]\}$ with $\boldsymbol{v'}_{ijk} = (x'_{ij,k}, y'_{ijk})$

• Encrypted DCT coefficients of $\mathbf{P}_{ij}$ is $E_K(\mathbf{P}_{ij})$

• Encrypted layer is a set of encrypted objects $E_K(\mathbf{L}_i) = \{E_K(\mathbf{P}_{ij}) | j \in [1, N_i]\}$

3.2 Selective encryption

Fig. 3 shows the proposed algorithm. First of all, we find N_i^{max} to generate random coefficients for random vector $\mathbf{r}_i$. After vertices randomization, all coordinates of objects in a layer are arranged into 2D array $A_{m\times n}$ to perform 2D-DCT [12]. The size of $A_{m\times n}$ is determined by the arrangement algorithm relating the total number of objects N_i and the total number of vertices N_{vi} in a layer $\mathbf{L}_i$. In DCT domain, we select DC value and some low AC values in 2D-DCT coefficients matrix to encrypt selectivity by random values using $\mathbf{K}_i$. 2D-DCT^{-1} process continuously changes all 2D-DCT coefficients matrix because some DCT coefficients are changed by encrypting selectivity. The purpose of arrangement all coordinates of vertices into 2D array is to increase high security because the size

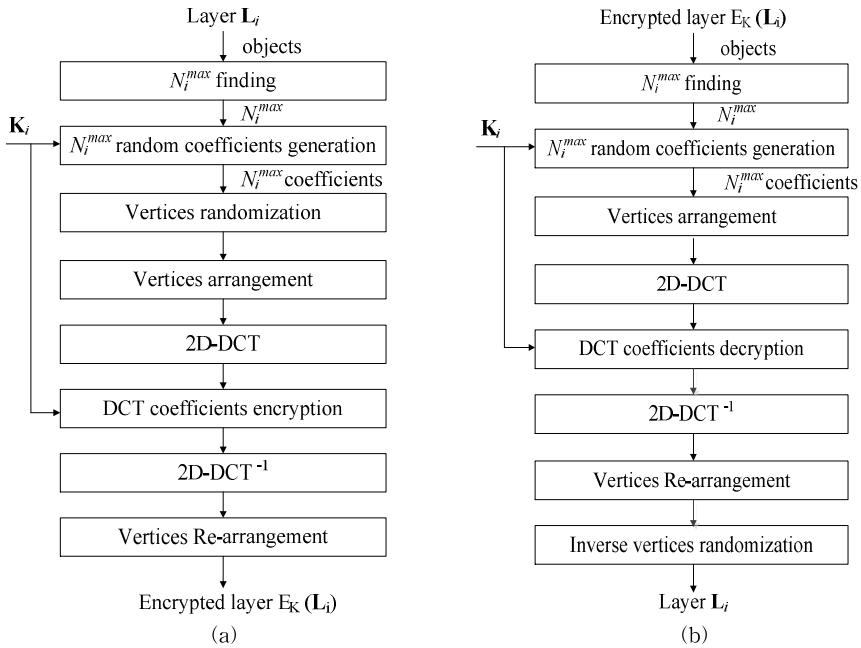

Fig. 3. The proposed algorithm. (a) Selective encryption, and (b) Decryption.

of 2D array is dynamic and decided by private algorithm depending on total number of objects and total number of vertices.

The purpose of finding N_i^{max} is to generate random vector $\mathbf{r}_i$. N_i^{max} is the maximum number of vertices of an object among objects in a layer $\mathbf{L}_i$:

$$N_i^{max} = max_{j\in[1,N_i]} N_{ij} \tag{1}$$

Key $\mathbf{K}_i$ is used as a seed of pseudo-random function $R_k(\ )$ to create N_i^{max} random coefficients of random vector $\mathbf{r}_i$ and encrypt DC values in DCT coefficients. It is randomly generated by the SHA-512 algorithm with user key input [13]. The random vector $\mathbf{r}_i = \{r_{ik} | k \in [1, N_i^{max}]\}$ is used to randomize the vertices of objects with r_{ik} be calculated by the pseudo-random function $R_k(\ )$ using $\mathbf{K}_i$:

$$r_{ik} = k \times R_K(\mathbf{K}_i) | k \in [1, N_i^{max}] \tag{2}$$

In vertices randomization step, we randomize vertices in an object by the randomized function $R_v(\ )$:

$$\mathbf{P}'_{ij} = R_V(\mathbf{P}_{ij}, \mathbf{r}_i) = \{v_{ij,k} \times r_{ik} | k \in [1, N_{ij}]\} \tag{3}$$

In vertices arrangement step, to order the total coordinates of the randomized objects in a layer $\mathbf{L}_i$ into a 2D array we computed the total number of vertices N_{vi} in a layer $\mathbf{L}_i$:

$$N_{vi} = \sum_{j\in[1,N_i]} N_{ij} \tag{4}$$

Because each vertex is presented by a pair of coordinates, thus we have total $2\times N_{vi}$ coordinates in a layer, and we need to determine the size of 2D array $A_{m\times n}$ for arranging total coordinates into it. We firstly verify remainder of division $(2\times N_{vi})$ for N_i. If the remainder is zero, we determine the size of 2D array as follow:

$$m = N_i,\ n = (2 \times N_{vi})/N_i \tag{5}$$

If remainder is different zero, we continue finding a set of divisors of $(2\times N_{vi})$ which are higher than 2 and smaller than N_i. Then, we find maximum divisor among those divisors. Suppose maximum divisor is d_s^{max}, we determine the size of 2D array as follow:

$$m = d_s^{max},\ n = (2 \times N_{vi})/d_s^{max} \tag{6}$$

In 2D-DCT domain, we have 2D - DCT coefficients matrix by forward DCT:

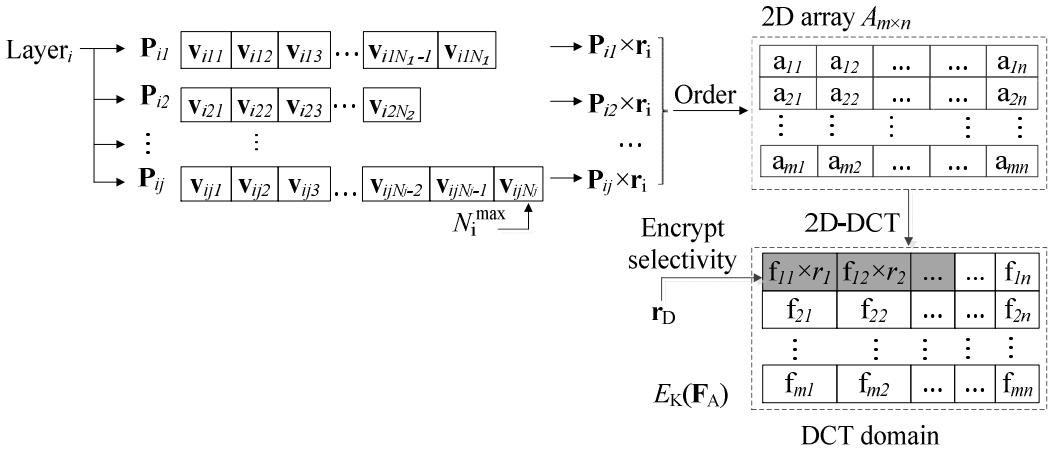

Fig. 4. Selective encryption in DCT domain.

$$\mathbf{F}_A = DCT(A_{m\times n}) = \{f_{st} | s \in [1,m], t \in [1,n]\} \quad (7)$$

And encrypt DC value and some low DCT coefficients in $\mathbf{F}_A$ by $\mathbf{K}_i$. Assume we select D coefficients including DC value for encryption, DC value and some low DCT coefficients are encrypted as equation (8):

$$(f_{1t} \times \mathbf{K}_i) | t \in [1,D] \quad (8)$$

After DC value and some low AC values encryption, we have a 2D array of coefficients $E_K(\mathbf{F}_A)$ as Fig. 4. We perform inverse 2D-DCT for $E_K(\mathbf{F}_A)$ to have $E_K(\mathbf{F^{-1}}_A)=\{f^{-1}{}_{st} | s \in [1,m], t \in [1,n]\}$ and re-order it to get the encrypted object $E_K(\mathbf{P}_{ij})=\{(u'_{ij,k}, v'_{ij,k}) | k \in [1, N_{ij}]\}$ to obtain encrypted layer $E_K(\mathbf{L}_i) = \{E_K(\mathbf{P}_{ij}) | j \in [1, N_i]\}$.

3.3 Decryption

Following steps in the selective decryption in Fig. 3b, after the 2D-DCT process we receive the encrypted DCT coefficients of object $E_K(\mathbf{F}_A)$. Refer to Eq. (8), in the DCT coefficients decryption we only need to divide DC value and some low AC values for the corresponding key value $\mathbf{K}_i$. After the inverse 2D-DCT and re-arrangement process we get the randomized object $\mathbf{P'}_{ij}$, and the decrypted objectis $\mathbf{P}_{ij}$ recovered by the vertices inverse randomization using the random coefficients of random vector $\mathbf{r}_i$ varying the total number of vertices in an object.

4. EXPERIMENTAL RESULTS

4.1 Visualization

We used the vector maps of Los Angeles city with the different layers in visualization experiments. The proposed algorithm is applied to polylines and polygons in vector map. The data format of vector map data is the shape-file (SHP) format [14]. Experimental results are shown in Fig. 5. The contents of railway map and waterway map (Fig. 5a) is polylines. After the encryption process, all polylines are changed. Similar result is shown in Fig. 5c with polygons. Maps are changed entirely after the selective encryption process. Due to the fact that the proposed algorithm only manipulates with the value of vertices, thus the number of vertices does not change. So, the proposed algorithm does not alter the size of the encrypted file and loss data.

4.2 Security Evaluation

In order to extract information from the perceptual encrypted map, any pirate has to extract all encrypted objects of map without the knowledge of keys. If the randomness of perceptual encryption is high, it will be so difficult to attack encrypted

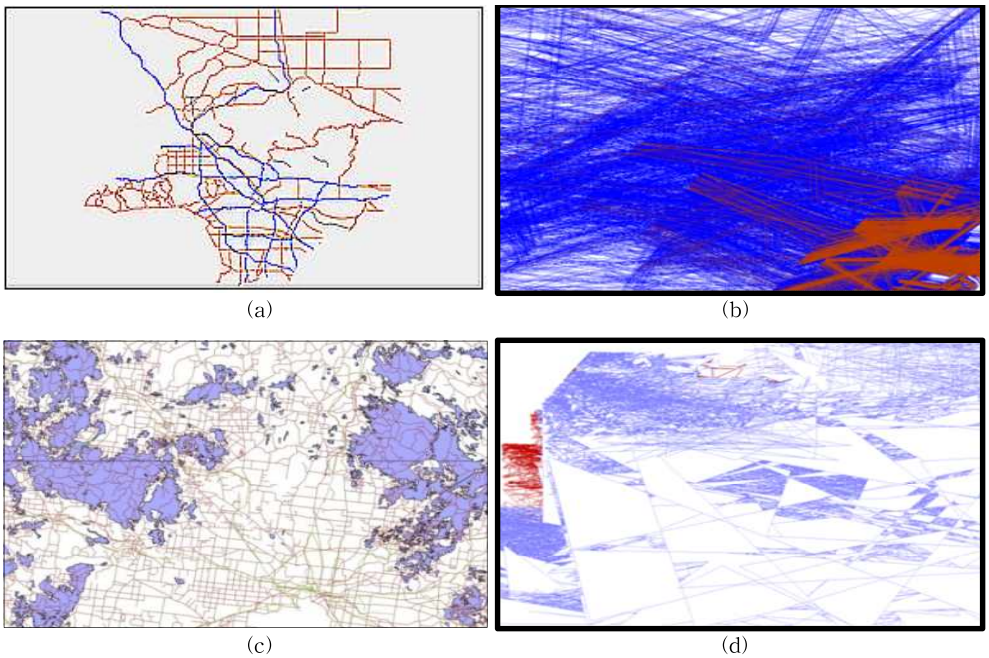

Fig. 5. Visualization experimental results, (a) and (b) Original and encrypted railway map layer, (c) and (d) Original and encrypted nature map layer.

objects. Therefore, we will calculate the entropy of perceptual encrypted map to evaluate the security of proposed method.

From equations in Sec. 3, we can see that encrypted object be dependent on secret key $\mathbf{K}$ and the selected number of DCT coefficients D. Both secret key $\mathbf{K}$ and D are random variables. Thus, entropy H_{Pi} of encrypted object $\mathbf{P}'_i$ is the sum of entropies of random variables above:

$$H_{\mathbf{P}'i} = H(\mathbf{K}) + H(D) \tag{9}$$

And thus, the entropy H_{M} of the encrypted map from original map $\mathbf{M}$ will be the sum of entropies of encrypted object $\mathbf{P}'_i$:

$$H_{\mathbf{M}} = \sum H_{\mathbf{P}'i} \tag{10}$$

Fig. 6 shows the increasing of entropy of map according to the number of the selected DCT coefficients. We only select some DCT coefficients but the entropy of the encrypted map is very high.

We choose Yasser's method [10] and Bang's method [11] for comparison because the issues presented in those papers are similar to those in our method. Previous methods also used the shape-file format in their experiments. In Yasser's method, Yasser used only the AES-256 cipher operator to encrypt the entire data stream of a shape-file. This is the conventional work. The shape-file is converted to data stream and it is then encrypted by the AES-256 function. Thus, the entropy of this method is dependent on the length of the secret key and the length of data stream. In Bang's method,

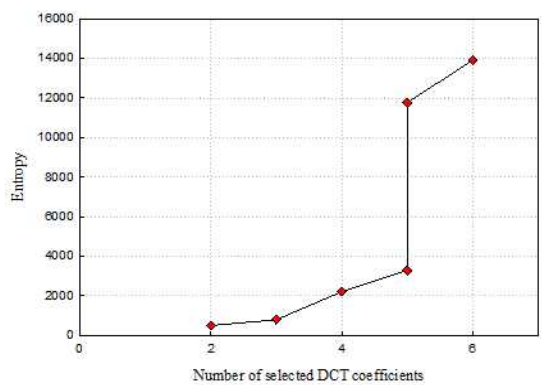

Fig. 6. The entropy of proposed algorithm.

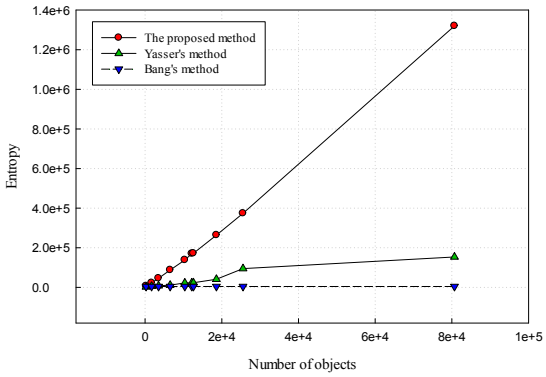

Fig. 7: Entropy according the number of objects.

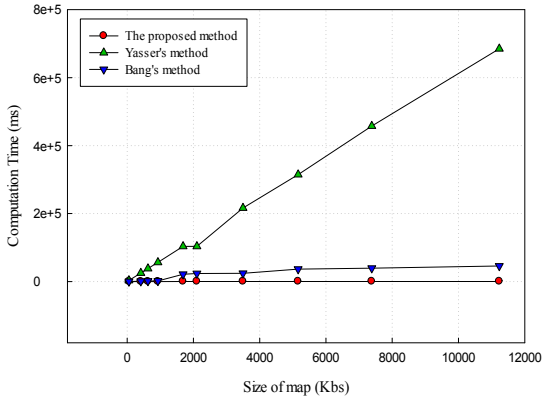

Fig. 8. Computation time.

about 70% polylines/polygons are selected for the encryption process. This mean 30% polylines/polygons are not encrypted and they are could be extracted without the secrete key and the decryption process. Moreover, Bang used a common secret key to encrypt all DC values in a vector map in DWT, DFT domains. The length of the secret key is 512 bits. Thus, the entropy of this method is only dependent on the length of the secret key. Fig. 7 shows the difference between the entropy of our method and the entropy of Yasser's method, the entropy of Bang's method according to the number of objects. The entropy of our method is much higher than Yasser's method and Bang's method.

4.3 Computation Time Evaluation

In our experiments, we used an Intel Core i7 Quad 3.5 - GHz, 8 GB of RAM, Windows 7 64-bits, and C# on Visual Studio 2013. From Sec. 3, we can see that the computation time of the proposed method is dependent on the number of objects in vector maps. In Yasser's method, Yasser performed full encryption for vector maps, and the computation time of Yasser's method is dependent on the size of vector maps which includes both notation, header, text and all geometric objects. In Bang's method, the computation time is dependent on the computation time of the selection process, the number of the selected objects and the computation time of DWT/DFT and inverse DWT/DFT processes. Fig. 8 shows the computation times of the proposed method, Yasser's method and Bang's method according to the size of vector maps. Our method's time is much shorter than previous methods.

5. CONCLUSION

In this paper, we proposed the selective encryption algorithm for vector map data based on DCT domain. Experimental results showed that the proposed algorithm is very effective with a large volume of vector map dataset. Comparing to previous algorithms, the proposed method has high security and it can be applied to the security of map service on/off-lines. Furthermore, our algorithm can be applied to various vector contents such as CAD and 3D content fields.

REFERENCE

[1] K.E. Foote and M. Lynch, *Geographic Information Systems as an Integrating Technology: Context, Concepts, and Definitions*, Online Book, 2009.

[2] S.H. Lee and K.R. Kwon, "Vector Water-Marking Scheme for GIS Vector Map Management," *Journal of Multimedia Tools and Applications*, Vol. 63, No. 3, pp. 757-790, 2011.

[3] V. Solachidis and I. Pitas, "Watermarking

Polygonal Lines using Fourier Descriptors," *Journal of IEEE Computer Graphics and Applications*, Vol. 24, No. 3, pp. 44-51, 2004.

[4] V.R. Doncel, N. Nikolaidis, and I. Pitas, "An Optimal Detector Structure for the Fourier Descriptors Domain Watermarking of 2D Vector Graphics," *Journal of IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 5, pp. 851-863, 2007.

[5] E. Bertino and M.L. Damiani, "A Controlled Access to Spatial Data on Web," *Proceeding of Conference on Geographic Information Science*, pp. 369-377, 2004.

[6] S.C. Chena, X. Wangb, N. Rishea, 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, 2003.

[7] N.B. Rybalov and O.I. Zhukovsky, "Access to the Spatial Data in the Web-Oriented GIS," *Proceeding of Siberian Conference on Control and Communications*, pp. 104-107, 2007.

[8] F. Wu, W. Cui, and H. Chen, "A Compound Chaos-Based Encryption Algorithm for Vector Geographic Data Under Network Circumstance," *Journal of Cardholder Information Security Program*, Vol. 1, pp. 254-258, 2008.

[9] G. Li, "Research of Key Technologies on Encrypting Vector Spatial Data in Oracle Spatial," *Proceeding of International Conference on Industrial Electronics and Computer Science*, pp. 1-4, 2008.

[10] D. Yasser, 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.

[11] N.V. Bang, S.H. Lee, K.S. Moon, and K.R. Kwon, "Encryption Algorithm Using Polyline Simplification for GIS Vector Map," *Journal of Korea Multimedia Society*, Vol. 19, No. 8, pp. 1453-1459, 2016.

[12] G. Strang, "Discrete Cosine Transform," *Journal of Society for Industrial and Applied Mathematics*, Vol. 41, No. 1, pp. 135-147, 1999.

[13] RSA Laboratories, PKCS: Password-Based Cryptography Standard, #5, v2.1, 2006.

[14] Environmental Systems Research Institute, *A White Paper: ESRI Shape-File Technical Description*, CA 92373-8100, 1998.

Ngoc-Giao Pham

received the Degree of Engineering in School of Electronic & Telecommunica-tion in Hanoi University of Science & Technology (HUST) in 2011, and Master degree from Pukyong National University (PKNU), Busan, South Korea in 2014. Currently, he is Ph.D candidate in PKNU. His research interests include digital image processing applica-tion, GIS visualization, multimedia data security and smart system.

Suk-Hwan Lee

received a B.S., a M.S., and a 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

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 Dept. of IT Convergence & 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 was the General President of Korea Multimedia Society on 2015-2016, also was a director of IEEE R10 Changwon section on 2012-2016. His research interests are in the area of digital image processing, multimedia security and watermarking, bioinformatics, weather radar information processing.

Chong-Ho Woo

received the B.S., M.S., and Ph.D. degrees in Computer Engineering from Kyungpook National University in 1978, 1981, and 1990 respectively. He is currently a professor in Dept. of Computer Engineering at the Pukyong National University. He has researched Lehman College CUNY, New York in 2008-2009 and LSU Baton Rouge, Louisiana in 2001-2002 as visiting professor. His research interests include Digital Image Processing, Embedded Systems, and IoT with WiFi Module.