JCSE Journal of Computing Science and Engineering

Regular Paper

*Journal of Computing Science and Engineering,*
*Vol. 11, No. 4, December 2017, pp. 152-159*

# Selective Encryption Algorithm for 3D Printing Model Based on Clustering and DCT Domain

**Giao N. Pham and Ki-Ryong Kwon***
Department of IT Convergence & Application Engineering, Pukyong National University, Busan, Korea
**ngocgiaofet@gmail.com, krkwon@pknu.ac.kr**

**Eung-Joo Lee**
Department of Information & Communication Engineering, Tongmyong University, Busan, Korea
**ejlee@tu.ac.kr**

**Suk-Hwan Lee**
Department of Information Security, Tongmyong University, Busan, Korea
**skylee@tu.ac.kr**

## Abstract
Three-dimensional (3D) printing is applied to many areas of life, but 3D printing models are stolen by pirates and distributed without any permission from the original providers. Moreover, some special models and anti-weapon models in 3D printing must be secured from the unauthorized user. Therefore, 3D printing models must be encrypted before being stored and transmitted to ensure access and to prevent illegal copying. This paper presents a selective encryption algorithm for 3D printing models based on clustering and the frequency domain of discrete cosine transform. All facets are extracted from 3D printing model, divided into groups by the clustering algorithm, and all vertices of facets in each group are transformed to the frequency domain of a discrete cosine transform. The proposed algorithm is based on encrypting the selected coefficients in the frequency domain of discrete cosine transform to generate the encrypted 3D printing model. Experimental results verified that the proposed algorithm is very effective for 3D printing models. The entire 3D printing model is altered after the encryption process. The decrypting error is approximated to be zero. The proposed algorithm provides a better method and more security than previous methods.

**Category:** Privacy and Security

**Keywords:** 3D printing data; 3D printing security; Selective encryption; DCT; Clustering

## I. INTRODUCTION

In recent years, three-dimensional (3D) printing has been widely used in many areas of life such as healthcare, industry, automotive and many other sectors [1-3]. The 3D printing technology makes a revolution in the industry so that it allows users to turn any digital file into a 3D physical product. Due to the fact that the benefits of 3D printing are enormous in all domains and the price of a 3D printer is not high, the individual user can buy a 3D

Open Access http://dx.doi.org/10.5626/JCSE.2017.11.4.152 http://jcse.kiise.org



**Received** 13 November 2017; **Revised** 16 November 2017; **Accepted** 21 November 2017
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

printer and download 3D printing models on the Internet to print real objects without any permission from the original providers. Moreover, some special models and anti-weapon models must be secured from un-authorized users. Thus, 3D printing models should be encrypted before being stored and transmitted in order to ensure the access and to prevent illegal copying.

In fact, the traditional encryption strategies such as DES and AES do not work for 3D printing models, because these methods are only useful to encrypt bit data such as text and music while the data format of 3D printing models is geometric data. Moreover, the requirements for encryption techniques must reduce computational complexity while still keeping or increasing security.

For meeting issues above, we propose a selective encryption algorithm for 3D printing models in this paper. The data format of 3D printing models is the 3D triangle mesh. The main content of the proposed algorithm is to extract facets from 3D triangle mesh for clustering into groups. All vertices of sides in each group will be used to construct a matrix, and this matrix is then transformed to the frequency domain of discrete cosine transform (DCT) [4]. In DCT domain, DC coefficients will be encrypted by a key value and then performed the inverse DCT in order to generate the encrypted 3D triangle mesh. To clarify the proposed algorithm, we organize our paper as follows. In Section II, we look into previous encryption techniques for 3D models and explain the relation of 3D triangle mesh to the proposed algorithm. In Section III, we show the proposed algorithm in detail. Experimental results and the evaluation of proposed algorithm will be shown in Section IV. Section V shows the conclusion.

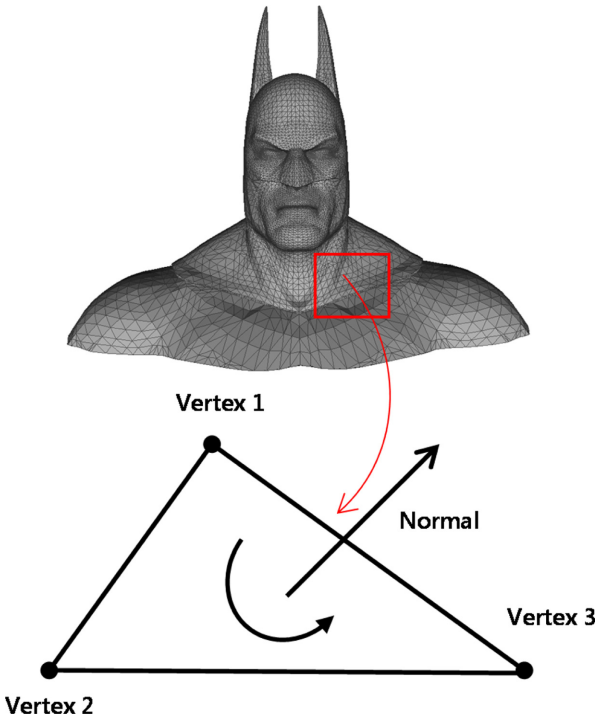


**Fig. 1.** Structure of 3D triangle mesh.

## II. RELATED WORK

### *A. 3D Model Encryption*

The purpose of encryption for 3D printing models is to alter and distort the shape of 3D printing models in order to prevent illegal copying or un-authorizd user from viewing the shape of 3D printing models for designing again. In addition, dangerous weapons should be secured with the un-authorized user. So, the target of encryption methods for 3D printing models is to change the shape of facets which are main components in 3D printing models.

Currently, there are no encryption method for 3D printing models. Some encryption techniques are only proposed for 3D CAD model. Eluard et al. [5] proposed a method to encrypt 3D objects based on geometry-preserving. This algorithm introduces a geometry-preserving paradigm that heavily distorts 3D objects while preserving some intrinsic geometrical property, thereby avoiding a global corruption of the whole 3D scene. The key idea of this method only permutes some facets of a 3D object. It did not alter the entire shape of a 3D object, and it is not effective to the various formats of 3D printing models. Moreover, the reconstruction cannot fully restore the encrypted 3D objects, and the security of this method is very low. Cai et al. [6-8] proposed an encryption approach for CAD models, which is based on geometric transformation encryption mechanisms on the features of CAD models. The key content of this approach is centered on an Enhanced Encryption Transformation Matrix, which is characterized as parametric, randomized and self-adaptive for feature encryption. This method only changes a little the shape of 3D CAD models. Consequently, the previously proposed methods cannot provide a response to the secured storage and transmission for 3D printing models.

### *B. 3D Triangle Mesh-Based Encryption*

At this time, 3D printing technology often uses 3D triangle mesh [9, 10] to print real objects. A 3D triangle mesh is a set of facets. Each facet contains three vertices (a triangle) and a normal vector (see Fig. 1). Each vertex is presented by three coordinates x, y, and z. Thus, to alter the shape of 3D printing models, we only alter the shape of facets. So, a facet is the target of the encryption process. Due to the fact that the normal vector of a facet only presents the direction of a facet, it does not determine the shape of a facet. So, we only need to encrypt three vertices of each facet to encrypt a 3D printing model.

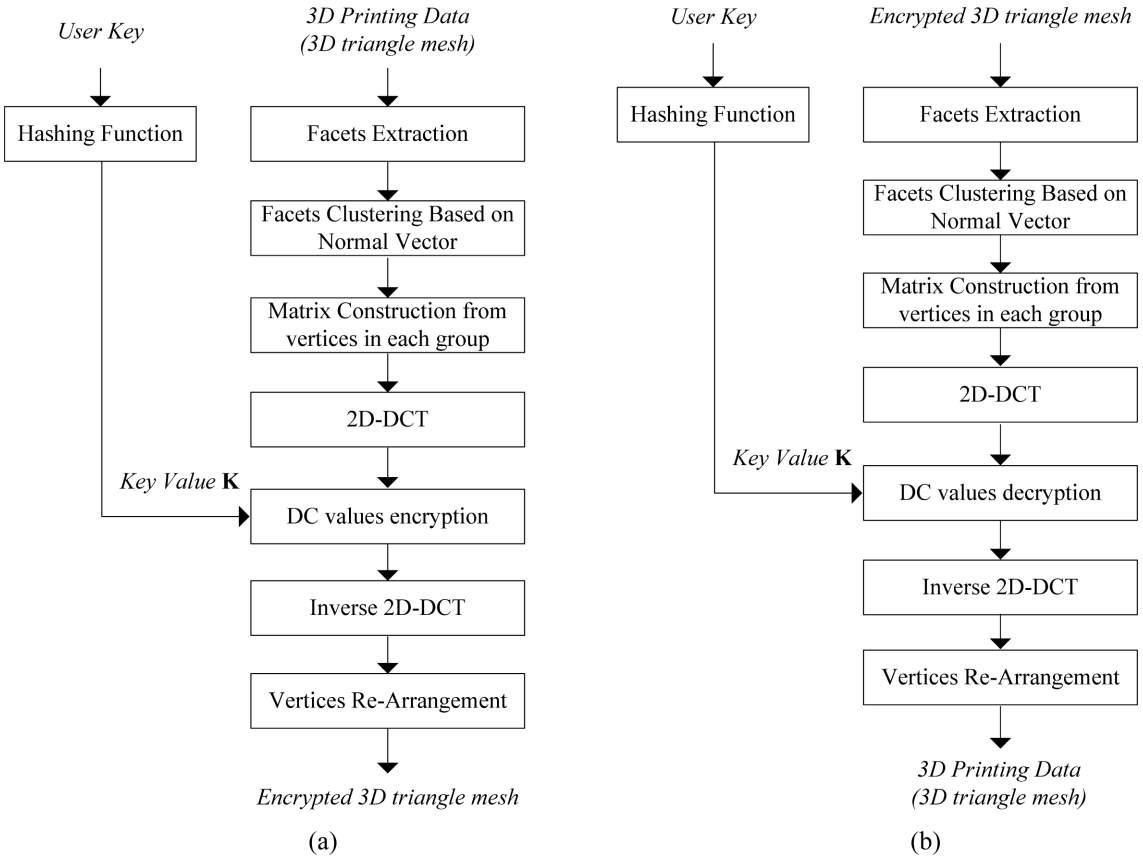


**Fig. 2.** The proposed algorithm. (a) Encryption process and (b) decryption process.

## III. THE PROPOSED ALGORITHM

### A. Overview

The proposed algorithm is described in Fig. 2(a). Facets are firstly extracted to divide into groups by the clustering algorithm based on normal vectors. All vertices of facets in each group will be used to construct a two-dimensional (2D) matrix. These 2D matrixes are then transformed to the frequency of DCT. In DCT domain, DC coefficients are selected and encrypted by a secret key value. The secret value is generated by a hashing function with a user's key input. After the DC coefficients undergo encryption process in DCT domain, DCT coefficients are performed inverse DCT to change all coefficients one more time. Finally, the coefficients of the inverse DCT process will be re-arranged to generate the encrypted 3D triangle mesh. The encrypted 3D triangle mesh is a set of the encrypted facets. The decryption process is shown in Fig. 2(b).

### B. Selective Encryption

As mentioned above, a 3D triangle mesh contains a set of facets. Each facet includes three vertices. In brief, we consider a 3D triangle mesh $\mathbf{M} = \{\mathbf{F}_i | i \in [1, |\mathbf{M}|\}$ where $|\mathbf{M}|$ is the cardinalities of a 3D triangle mesh; and $\mathbf{F}_i = \{v_{i1}, v_{i2}, v_{i3}$ and $\mathbf{n}_i\}$ indicates the $i^{th}$ facet with three vertices $\{v_{i1}, v_{i2}, v_{i3}\}$ and the normal vector $\mathbf{n}_i(nx_i, ny_i, nz_i)$. Due to the fact that the normal vector of a facet does not determine the shape of 3D triangle mesh, we briefly consider the facet $\mathbf{F}_i = \{v_{i1}, v_{i2}, v_{i3} | i \in [1, |\mathbf{M}|\}$.

After the facet extraction process, we have $|\mathbf{M}|$ facets, which are divided into groups by the clustering algorithm. Assume that $|\mathbf{M}|$ facets are divided into $G$ groups with $G = \{\omega_g | g \in [1,|G|]\}$. Each group $\omega_g$ has $N_g$ facets and groups' total facets equal $|\mathbf{M}|$ as shown in Eq. (1).

$$|\mathbf{M}| = \sum_{g}^{|G|} N_g \tag{1}$$

All vertices of facets in each group $\omega_g$ will be used to construct a 2D matrix $\mathbf{M}^g_{3,3\times N_g}$, as shown in Fig. 3. The principle of arrangement to construct a 2D matrix: each three coordinates of each vertex is a column of matrix according to the increasing order of index of each vertex. For convenience, we use $\mathbf{F}^g_i$ sign to replace $\mathbf{F}_i$ sign after the clustering process in order to identify that the facet $\mathbf{F}_i$ is located in the group $\omega_g$. Thus, the matrix $\mathbf{M}^g_{3,3\times N_g}$ can be presented by Eq. (2).

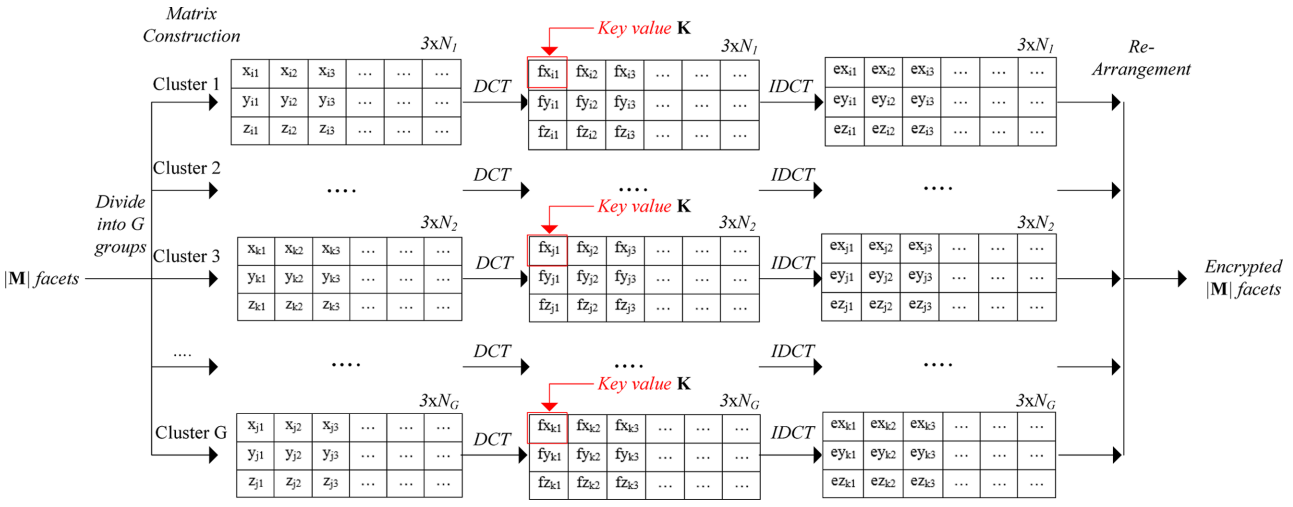


**Fig. 3.** Selective encryption process in DCT domain of a 3D triangle mesh.

$$\mathbf{M}^{g}_{3,3\times N_g} = \{\mathbf{F}^{g}_{i} | i \in [1,|\mathbf{M}|]\} \text{ with } \mathbf{F}_i \in \omega_g$$

$$= \begin{bmatrix} x^g_{i1} & x^g_{i2} & x^g_{i3} & \dots & x^g_{N_g1} & x^g_{N_g2} & x^g_{N_g1} \\ y^g_{i1} & y^g_{i2} & y^g_{i3} & \dots & y^g_{N_g1} & y^g_{N_g2} & y^g_{N_g1} \\ z^g_{i1} & z^g_{i2} & z^g_{i3} & \dots & z^g_{N_g1} & z^g_{N_g2} & z^g_{N_g1} \end{bmatrix} \quad (2)$$

Given *DCT*(.) and *IDCT*(.) are the forward DCT and inverse DCT functions, respectively. And *EC*(.) is the encryption function. The matrix $\mathbf{M}^{g}_{3,3\times N_g}$ is transformed to DCT domain as shown in Eq. (3) with $\mathbf{FM}^{g}_{3,3\times N_g}$ is the matrix of DCT coefficients. In DCT domain, the DC coefficient $fx^g_{i1}$ is encrypted by the key value **K** as shown in Eq. (4). The key value **K** is generated by the SHA-512 hashing algorithm [11] that uses a user's key input. The length of each key value is 512 bits. After DC coefficient encryption, we perform inverse DCT to change all DCT coefficients one more time in order to generate the encrypted vertices $\mathbf{EM}^{g}_{3,3\times N_g}$ as shown in Eq. (5).

$$\mathbf{FM}^{g}_{3,3\times N_g} = DCT(\mathbf{M}^{g}_{3,3\times N_g})$$

$$= \begin{bmatrix} fx^g_{i1} & fx^g_{i2} & fx^g_{i3} & \dots & fx^g_{N_g1} & fx^g_{N_g2} & fx^g_{N_g1} \\ fy^g_{i1} & fy^g_{i2} & fy^g_{i3} & \dots & fy^g_{N_g1} & fy^g_{N_g2} & fy^g_{N_g1} \\ fz^g_{i1} & fz^g_{i2} & fz^g_{i3} & \dots & fz^g_{N_g1} & fz^g_{N_g2} & fz^g_{N_g1} \end{bmatrix} \quad (3)$$

$$fx'^{g}_{i1} = EC(\mathbf{K}, fx^g_{i1}) = \frac{\mathbf{K}}{(i+1)\times g} \cdot fx^g_{i1} \quad (4)$$

$$\mathbf{EM}^{g}_{3,3\times N_g} = IDCT(\mathbf{M}^{g}_{3,3\times N_g})$$

$$= IDCT\left(\begin{bmatrix} fx'^{g}_{i1} & fx^g_{i2} & fx^g_{i3} & \dots & fx^g_{N_g1} & fx^g_{N_g2} & fx^g_{N_g1} \\ fy^g_{i1} & fy^g_{i2} & fy^g_{i3} & \dots & fy^g_{N_g1} & fy^g_{N_g2} & fy^g_{N_g1} \\ fz^g_{i1} & fz^g_{i2} & fz^g_{i3} & \dots & fz^g_{N_g1} & fz^g_{N_g2} & fz^g_{N_g1} \end{bmatrix}\right)$$

$$= \begin{bmatrix} ex^g_{i1} & ex^g_{i2} & ex^g_{i3} & \dots & ex^g_{N_g1} & ex^g_{N_g2} & ex^g_{N_g1} \\ ey^g_{i1} & ey^g_{i2} & ey^g_{i3} & \dots & ey^g_{N_g1} & ey^g_{N_g2} & ey^g_{N_g1} \\ ez^g_{i1} & ez^g_{i2} & ez^g_{i3} & \dots & ez^g_{N_g1} & ez^g_{N_g2} & ez^g_{N_g1} \end{bmatrix} \quad (5)$$

Due to the fact that DC coefficients are changed, after the inverse DCT process, all DCT coefficients will be changed one more time. Finally, the encrypted vertices are re-arranged in order to obtain the encrypted 3D triangle mesh. The encrypted 3D triangle mesh $\mathbf{E}_M = \{\mathbf{E}_{\mathbf{F}i}(e_{i1}, e_{i2}, e_{i3}) | i \in [1, |\mathbf{M}|\}$ is a set of the encrypted facets where $\mathbf{E}_{\mathbf{F}i}(e_{i1}, e_{i2}, e_{i3})$ is the encrypted facet of the facet $\mathbf{F}_i = \{v_{i1}, v_{i2}, v_{i3}\}$. Fig. 3 shows the selective encryption process in DCT domain for |**M**| facets of a 3D triangle mesh.

### C. Decryption Process

The decryption process is an inverse process with the selective encryption process. The encrypted facets are also extracted from the encrypted 3D triangle mesh in order to divide into groups based on normal vectors. Next, all vertices of facets in each category are used to construct a 2D matrix before transforming it to DCT domain. In the DCT domain, DC coefficients are decrypted by a secret key, and then all DCT coefficient will be performed inverse DCT in order to restore the decrypted facets. Finally, the decrypted facets are re-arranged to generate the decrypted 3D triangle mesh.

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

We experimented using the proposed algorithm with 3D triangle meshes as shown in Table 1. The format of 3D triangle meshes is STL file, and VRML file [9, 10]. The

**Table 1.** Experimental results

| | # Facets | # Groups | Entropy (dB) | | | Computation time (ms) |
|---|---|---|---|---|---|---|
| | | | Proposed method | Eluard et al. [5] | Cai et al. [8] | |
| Knife | 746 | 31 | 7,272 | 2,918 | 3,215 | 75 |
| Conduit mount | 1,020 | 31 | 10,347 | 3,812 | 4,615 | 136 |
| Gun | 1,912 | 59 | 21,189 | 5,665 | 9,493 | 208 |
| End cap | 3,616 | 113 | 37,194 | 12,461 | 19,588 | 445 |
| Church | 5,696 | 178 | 72,392 | 20,645 | 32,711 | 724 |
| Airplane | 9,052 | 282 | 121,275 | 35,328 | 54,992 | 1,309 |
| Batman | 13,566 | 339 | 189,079 | 55,507 | 86,360 | 2,378 |
| Cup | 28,630 | 715 | 430,653 | 112,838 | 197,650 | 7,501 |
| Yoda | 49,844 | 1,246 | 790,634 | 145,359 | 364,017 | 17,400 |
| Lion | 79,162 | 1,979 | 1,309,836 | 432,525 | 604,530 | 41,122 |

detailed information of models is shown in Table 1. We used the K-mean algorithm [12] to cluster facets in groups. The number of groups can be decided by users. The number of groups is always smaller than a half of the facets. To satisfy the above condition, we defined the number of groups $G$ according to the number of facets $|\mathbf{M}|$ as in Eq. (6).

$$G = Integer\ part\left(\frac{|\mathbf{M}|}{2^3 \times N_D}\right) \quad (6)$$

where $N_D$ is the number of digits of $|\mathbf{M}|$. For example, if $|\mathbf{M}| = 2146$ then $N_D = 4$. In order to evaluate the proposed algorithm, we evaluate the visualization experiments, security and computation time of the proposed algorithm. Section IV-A shows visulization experiments. Section IV-B shows the security evaluation and the computation time of the proposed algorithm is shown in Section IV-C.

## A. Visualization Experiments

We would like to show some experimental results of visualization in Fig. 4. The number of facets of 3D triangle meshes is different, and the number of groups of each 3D triangle mesh is also different. After the encryption process, facets are distorted into small facets (see "Encrypted Knife" and "Encrypted Gun") or big facets (see "Encrypted Conduit Mount"), changed location, positioned disorderly and not connected together (see "Encrypted Yoda"). This leads to the shape of 3D triangle meshes to change. Consequently, the content of 3D triangle meshes is completely altered after the selective encryption process. Pirates or un-authorized users cannot extract or view the content of 3D triangle meshes.

In Cai et al.'s method [8], the encrypted CAD model is changed a little. Anybody can see the content of the encrypted CAD model. Comparing with Cai et al.'s method, the perceptual results of the proposed algorithm is better.

## B. Security Evaluation

To decrypt the encrypted 3D triangle mesh, any pirate has to decrypt all the encrypted facets of 3D triangle mesh without knowledge of the keys. In the proposed algorithm, we used the SHA-512 algorithm with a 128 bits salt to generate random keys. The length of the bits salt can be altered to 128, 256 or 512. Thus, if a user uses English words of length $L_k$ as his password, an attacker has to calculate $L_k \times 2^{128}$ keys to access the encrypted 3D triangle mesh. To evaluate the security of the proposed algorithm, we will analyze the entropy of the encrypted 3D triangle mesh. If the entropy is high, the security will be high.

From the equations in Section III, we can see that the entropy of the encrypted 3D triangle mesh is dependent on both the secret key $\mathbf{K}$, the number of the selected DC coefficients for encrypting and the number of facets $|\mathbf{M}|$. The number of the selected DC coefficients is equal to the number of groups in the clustering process. This means that the entropy of the encrypted 3D triangle mesh is dependent on $\mathbf{K}$, $G$ and $|\mathbf{M}|$. But $\mathbf{K}$, $G$ and $|\mathbf{M}|$ are random independent variables. So the entropy of the encrypted 3D triangle mesh $H_{\mathbf{M}}$ is the sum of the entropies of variables $\mathbf{K}$ and $G$, and determined by Eq. (7).

$$H_{\mathbf{M}} = H(\mathbf{K}) + H(G) + H(|\mathbf{M}|) \quad (7)$$

Because $\mathbf{K}$ and $G$ are discrete random variables, the entropy of the encrypted 3D triangle mesh in Eq. (7) will become:

$$H_{\mathbf{M}} = |G|.\log_2|G| + |\mathbf{K}|.\log_2|\mathbf{K}| + |\mathbf{M}|.\log_2|\mathbf{M}| \quad (8)$$

Assume that all 3D triangle meshes are encrypted by

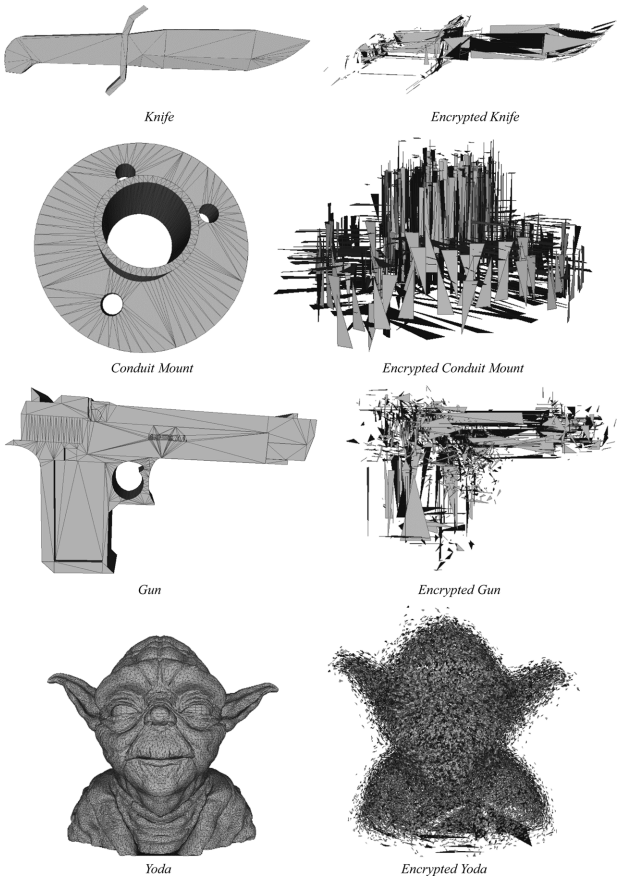


**Fig. 4.** Experimental results with test models.

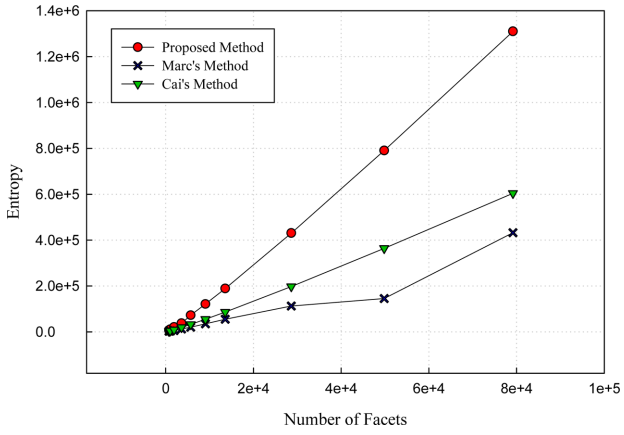


**Fig. 5.** The entropy of the proposed method according to the number of facets.

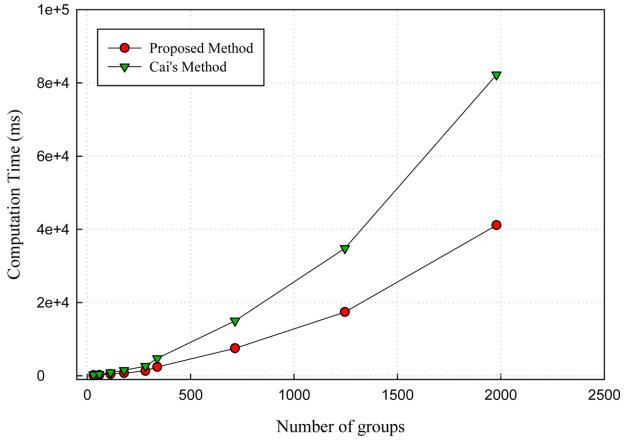


**Fig. 6.** Computation time according to the number of groups.

the same secret key, we can calculate the entropy of the encrypted 3D triangle mesh according to the number of groups and the number of facets as shown in Table 1. The entropy of the encrypted 3D triangle mesh is formed from 7,272 dB to $1.31\times10^6$ dB with $G \in [31,1979]$ and $|\mathbf{M}| \in [746,79162]$. From Eq. (8) and Table 1 we can see that if $G$ and $|\mathbf{M}|$ are high, the entropy will be high.

In Eluard et al.'s method [5], they used the secret key **K** to encrypt and change the location of the vertices of a 3D triangle mesh in OXYZ space. Simply, we can understand that Marc's method encrypted the vertices of 3D triangle mesh by a secret key **K**. But the number of vertices in a 3D triangle mesh is always smaller than the number of facets. Thus the entropy of this method is always lower than the proposed algorithm. With test models in Table 1, the entropy of Eluard et al.'s method is formed from 2,918 dB to 432,525 dB (see Table 1). In Cai et al.'s method [8], they encrypt the features of 3D CAD model by a random 3×3 matrix that is generated from the secret key. Thus, we can consider that Cai et al.'s method is encrypted 3D CAD models based on features and a random matrix by a secret key **K**. So, the entropy of this technique is dependent on both the number of features and the 3×3 matrix. For the experimental results, around 50% of facets are selected as the feature of 3D CAD model. With test models in Table 1, the entropy of Cai et al.'s method is formed from 3,215 dB to 604,530 dB. Fig. 5 shows the entropy of the proposed method with the entropy of previous methods according to the number of facets. The entropy of the proposed method is always higher than the entropy of previous methods. Consequently, the proposed method is better and more secure than previous methods.

## C. Performance 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. The computation time of the proposed method is dependent on the number of facets, the number of groups and the clustering algorithm. With test models in Table 1, the computation time is formed from 75 ms to 41122 ms with $G \in [31,1979]$ and $|\mathbf{M}| \in [746,79162]$. From Table 1 we can conclude that if the number of groups and the number of facets are small, the computation time

is small and otherwise. In Eluard et al.'s method, they did not show the computation time, so we could not compare Eluard et al.'s methods with our method. In Cai et al.'s method, they only analyzed the complexity time. The computation time of Cai et al.'s method is dependent on the time of valid check CAD model, time of feature encryption and time of CAD model encryption. They concluded that is enough to meet user's requirements. With the three processes in Cai et al.'s method, we consider and evaluate that the computation time of Cai et al.'s method is at least twice the computation time of our method. Comparing to Cai et al.' method, our method is faster. Fig. 6 show the computation time of the proposed method and Cai et al.'s method according to the number of facets.

## V. CONCLUSION

In this paper, we proposed a selective encryption algorithm for 3D printing models in the frequency domain of discrete cosine transformation. It is based on clustering 3D triangles into groups and then encrypting selectively DC coefficients in DCT domain by the secret key to generate the encrypted 3D printing model. The proposed algorithm is more effective than previous methods. It is also responsive to the various formats of 3D printing model. The performance of the proposed method is better than the previous proposed methods. It provides a better solution and is more secure than the previous methods. In the future, we will improve the proposed algorithm with storage and transmission systems.

## ACKNOWLEDGMENTS

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 (No. NRF-2016R1D1A3B03931003 & NRF-2017R1A2B2012456) and Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2015-0-00225), and also supported by Brain Busan (BB21) project.

## REFERENCES

1. United States Government Accountability Office, "3D Printing Opportunities, Challenges, and Policy Implications of Additive Manufacturing," 2015; https://www.finnegan.com/en/firm/news/3d-printing-opportunities-challenges-and-policy-implications-of.html.
2. 3D Systems Inc., "How 3D printing works: the vision, innovation and technologies behind Inkjet 3D printing," 2012; http://www.officeproductnews.net/sites/default/files/3dWP_0.pdf.
3. V. Srinivasan and B. Jarrod, "3D Printing and the Future of Manufacturing," CSC Leading Edge Forum, 2012; https://assets1.csc.com/innovation/downloads/LEF_20123DPrinting.pdf.
4. G. Strang, "The discrete cosine transform," *Society for Industrial and Applied Mathematics*, vol. 41, no. 1, pp. 135-147, 1999.
5. M. Eluard, Y. Maetz, and G. Doerr, "Geometry-preserving encryption for 3D meshes," in *Proceedings of International Conference of Compression at Representation Signal Audio (CORESA)*, Le Creusot, France, 2013, pp. 7-12.
6. X. T. Cai, F. Z. He, W. D. Li, X. X. Li, and Y. Q. Wu, "Encryption based partial sharing of CAD models," *Integrated Computer Aided Engineering*, vol. 22, no. 3, pp. 243-260, 2015.
7. X. T. Cai, W. D. Li, F. Z. He, and X. X. Li, "Customized encryption of computer aided design models for collaboration in cloud manufacturing environment," *Journal of Manufacturing Science and Engineering*, vol. 137, no. 4, article no. 040905, 2015.
8. X. T. Cai, F. Z. He, W. D. Li, X. X. Li, and Y. Q. Wu, "Parametric and adaptive encryption of feature-based computer-aided design models for cloud-based collaboration," *Integrated Computer-Aided Engineering*, vol. 24, no. 2, pp. 129-142, 2017.
9. STL format in 3D printing, 2017; https://all3dp.com/what-is-stl-file-format-extension-3d-printing.
10. The Virtual Reality Modeling Language (ISO/IEC 14772-1:1997), http://www.cacr.caltech.edu/~slombey/asci/vrml/.
11. RSA Lab., "Password-Based Cryptography Standard," 2006.
12. J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, CA, 1967, pp. 281-297.

## Giao N. Pham

Giao N. Pham received a B.E. Degree in School of Electronic & Telecommunication from Hanoi University of Science & Technology (HUST) in 2011, and an M.S. degree from Pukyong National University (PKNU), Busan, South Korea in 2014. Currently, he is a Ph.D candidate in Pukyong National University. His research interests include digital image processing & application, GIS visualization, multimedia data security, smart systems and IoT.

## Suk-Hwan Lee

Suk-Hwan Lee received 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.

## Eung-Joo Lee

Eung-Joo Lee received his B.S., M.S., and Ph.D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has been with the Department of Information & Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2000 to July 2002, he was a president of DigitalNetBank Inc. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, China. His main research interests include biometrics, image processing, and computer vision.

## Ki-Ryong Kwon

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 at the University of Minnesota in USA during 2000-2002 as Post-Doc, and Colorado State University during 2011-2012 as visiting professor. He was the General President of Korea Multimedia Society from 2015-2016. He is also a director of IEEE R10 Changwon section. His research interests are in the areas of digital image processing, multimedia security and watermarking, bioinformatics, and weather radar information processing.