

3D Textured Model Encryption Using 2D Logistic and 3D Lorenz Chaotic Map

Nashwan Alsalam Ali¹, Abdul Monem S. Rahma², Shaimaa H. Shaker³

¹Department of Computer Sciences, College of Education for Women, University of Baghdad, Baghdad, Iraq.

^{2,3}Department of Computer Sciences, University of Technology, Baghdad, Iraq.

¹nashwan_alsalam60@coeduw.uobaghdad.edu.iq, ²110003@uotechnology.edu.iq,

³Shaimaa.h.shaker@uotechnology.edu.iq

Abstract— The widespread of recent multimedia, including various 3D model applications in different domains of areas, may lead to 3D models being stolen and attacked by hackers. Moreover, 3D models must be protected from unauthorized users and when transmitting over the internet. Nowadays the 3D model protection is a very important issue. This paper proposed a scheme that provides high protection for the textured 3D model by implementing multiple levels of security. The first level of security is achieved by encrypting the texture map based on a key generated by a 2D Logistic chaotic map. The second level of security is implemented by modifying the vertices values of the 3D mesh based on keys generated by the 3D Lorenz chaotic map. The proposed scheme was implemented on various 3D textured models varying in the number of vertices and faces. The experimental results show that the proposed scheme has a good encryption and provides high security by completely deforms the whole texture and 3D mesh of the textured 3D model into the two levels. The encryption scheme has a large key space 10^{135} making the scheme resists violent attacks. The Hausdorff Distance (HD) and histogram metrics are adopted to calculate the matching degree between the original and extracted model. The results show that the original and extracted model are identical through the values of HD, which are approximate to zero, and the histogram visually is similar.

Index Terms— 3D textured model, encryption, 2D Logistic map, 3D Lorenz map, chaotic map.

I. INTRODUCTION

Digital media development in the internet, multimedia applications, transmitting digital data over the unsecured channel, and widespread use of personal computers allow users to protect the digital data from threats and attacks [1]. Cryptography is applied for protecting digital data by performing the encryption process, which is a powerful tool that performs encoding operation, encryption process commonly done with an encryption key that defines how the data are to be coded so only the authorized user can access and understand it [2]. Various encryption algorithm types are proposed to convert data into unrecognized forms and prevent unauthorized user access [3]. Nowadays the 3D models are becoming an important part of the multimedia content where its applications are developed and become more popular such as Virtual Reality (VR), Computer-Aided Design (CAD), 3D printing, and digital visualization, so providing security for 3D models which become an important matter and must provide a

Received 15/7/2021; Accepted 21/8/2021

high level of confidentiality, integrity, and protection against unauthorized access for data; therefore, the problem of the thread must be solved [4,5]. There are many proposed encryption schemes, some of them have been adopted and standardized in the world; however, they are not suitable for a 3D model such as Advanced Encryption Standard (AES) and Data Encryption Standard (DES) because the problem of 3D model encryption due to the application requirements and the data structure such as format compliance, content usability, complexity, security level and real-time performance [6], to overcome these problems, various encryption methods based on chaotic cryptography have been implemented. Chaotic systems are attracted and adopted in cryptosystems by the researcher because they have excellent properties like control parameters, sensitivity to the initial condition, ergodicity, randomness, deterministic, and periodicity [7,8]. In this paper, the proposed encryption scheme of the 3D textured model will be introduced based on the keys generated from the Logistic map and 3D Lorenz map.

II. RELATED WORK

Alireza Jolfaei et al. in 2016 [9] proposed a texture encryption scheme where they used fast stream cipher Salsa20/12 to encrypt texture images by using permutation and bit masking operation. The encryption process involves scrambling the lower nibble-image using zigzag permutation and Salsa20/12 to encrypt the upper nibble-image. The proposed scheme has satisfied security requirements, and it is lightweight and protects the texture from partially leaked. Furthermore, the 500 sample texture images are used to implement and test the proposed scheme. The encryption speed of the scheme has a better speed profile than encryption by 128-bit AES.

Xin Jin et al. in 2017 [10] focused on the 3D textured from the 3D content to provide security and privacy for 3D content. The authors encrypted 3D textured models using proposed 3D Lu chaotic maps which are a high-order and strange attractor chaotic map for encrypting textures, vertices, and polygon, then gathering these encrypted content to form the final encrypted 3D texture model. The experimental results show that the proposed method can correctly encrypt the 3D textured model and resist various brute-force attacks.

Xingyuan Wang et al. in 2019 [5] proposed the scheme in which the 3D object is converted into 2D objects as the same as image format to perform encryption on it. The encryption scheme is performed via two phases: the confusion phase and the diffusion phase. During the confusion phase, the authors have introduced random points, while during the diffusion phase, the authors divided the floating-point data into two parts, the integer and decimal parts. The integer part was encrypted using the XOR operation, while the decimal part was scrambled only. The security analysis has shown that the scheme is highly secured and resistant to common attacks.

Chaochuan Jia et al. in 2019 [11], the authors proposed two schemes for encrypting the 3D point cloud using a chaotic cat map. In the first scheme, Permutation using 2D Cat Map(P2DCM) with time complexity $O(3N^2)$. The author's used a 2D cat map to perform permutation for each coordinate (x, y, and z) in every point cloud. In the second scheme, Random Transformation Matrix using 3D Cat Map (RTM3DCM) with time complexity $O(6M)$, the authors encrypted 3D point cloud using a 3D cat map to generate a random

transformation matrix to transform a point in 3D space into a different position using permutation encryption process.

Najlaa Hamza et al. in 2019 [12], the authors proposed a method for encrypting the 3D object using the Transformation, Substitution, Folding, and Shifting (TSFS) algorithm. The encryption method takes the vertices of the 3D model and inputs them to the TSFS algorithm. The four stages in TSFS are based on three keys, wherein in the transformation step, the position of the vertex will be changed, in substitution step, the data matrix component will be altered with a different element, in the folding step, the matrix elements are folded in a diagonal, vertical and horizontal manner and in shifting step which is the last step of TSFS it uses of element 16 in a set of numeric digits for replacing the code with another one. The experimental results show that the proposed method successfully encrypts the 3D model, where the system achieved effective and robust security.

III. CRYPTOGRAPHY AND CHAOTIC SYSTEM

Data protection has many requirements, the most important requirements including confidentiality and security. The proportionate mixture of chaotic mathematical theory and the science of cryptography is called chaotic cryptography. The chaotic system consists of the dynamic equation that varies with time. When the dynamic system satisfies the three conditions below, it will be considered chaotic [7].

- 1- Sensitive to initial conditions.
- 2- Topological mixing.
- 3- The density of periodic orbits.

Cryptosystem and chaotic systems have a relationship between them; however, the main variation between chaotic and cryptographic systems is that chaos is valid in an infinite domain while cryptography operates on a finite domain [13]. The relationship between cryptography and chaotic systems makes cryptography-based chaos a normal candidate for cryptography and secure communication [14]. Similar properties have been shared between chaotic systems and cryptographic such as control parameters, sensitivity to the initial conditions, unstable periodic orbits with long periods, and random behavior [15]. Due to the random behavior, the system output seems random in the attacker's view, whereas it appears as defied in the receiver's view, and decryption is possible [16]. In Table 1, the comparison between chaotic systems and cryptography will be illustrated.

TABLE 1. PROPERTY COMPARISON BETWEEN CHAOTIC SYSTEM AND CRYPTOGRAPHY

Chaotic	Cryptography
Sensitive to initial condition	Diffusion
Ergodicity	Diffusion
Structure complexity	Algorithm complexity
System parameter	Key
Deterministic dynamics	Deterministic Pseudo-Random (PR)

Received 15/7/2021; Accepted 21/8/2021

IV. CHAOTIC MAPS

A. 2D Logistic map

The 2D Logistic map was derived from the 1D Logistic map where 1D Logistic map is simple and exhibited not complicated chaotic behavior since it had a small key space and just one control parameter, making it easy to attack [17]. The 2D Logistic map is more complicated and has large key space than a 1D Logistic map, making it widely used in cryptography. The 2D Logistic map is mathematically represented in equation (1), where r is a control parameter and (x_i, y_i) are represented the pair-wise points at i th iteration [18].

$$x_{i+1} = r(3y_i + 1)x_i(1 - x_i) \quad (1)$$

$$y_{i+1} = r(3x_i + 1)y_i(1 - y_i)$$

The r parameter is a positive value, and it is in the range $[0, 4]$, x_i and y_i in the range $[0, 1]$, the iteration x_{i+1} and y_{i+1} are based on the previous value of x_i and y_i , respectively, which mean it is iterative sequences [19].

B. 3D Lorenz Chaotic Map

The 3D Lorenz is a chaotic map of a three-dimension developed by the scientist Edward Lorenz by a combined differential equation. The Lorenz chaos sequences generated attractors, which is the deck of chaotic solutions for the Lorenz system. The 3D Lorenz chaotic formula is depicted by equations (2), (3), and (4) [20].

$$\frac{dx}{dt} = a(y - x) \quad (2)$$

$$\frac{dy}{dt} = rx - y - xz \quad (3)$$

$$\frac{dz}{dt} = xy - bz \quad (4)$$

The control parameters a , r and b , are the parameters where the system depending on them. When the parameters value $a=10$, $r=28$ and $b=8/3$. The x , y , and z solution curves for these equations circle two equilibrium points and the projections of its phase portrait. The initial values of $(x, y, \text{ and } z)$ are the bases of the 3D Lorenz Chaotic trajectory framework, where they represent the secret key for performing the permutation process [21].

V. THE 3D MODEL REPRESENTATION

The surface of the 3D model is represented by a 3D polygon mesh where the polygons are straight-sided shapes which can be triangles that have three sides or quadrilaterals when having four sides; the typical structure is the 3D triangular mesh [22]. An individual polygon is called a face, and when connecting many faces, they create a network of faces called polygon mesh. The representation of the polygon mesh can be represented as $M = \{G, C\}$, where G represents geometry information and C represents topological information. The geometry information is denoted as $G = \{V, E, F\}$, where V is the vertices of the mesh, E is the straight lines that connect vertices, and F is the faces information. The topological information, including the

Received 15/7/2021; Accepted 21/8/2021

connectivity data between the geometry elements that specify which vertices belong to each polygon [23,24]. Fig. 1 illustrates the triangle and quadrilaterals representation.

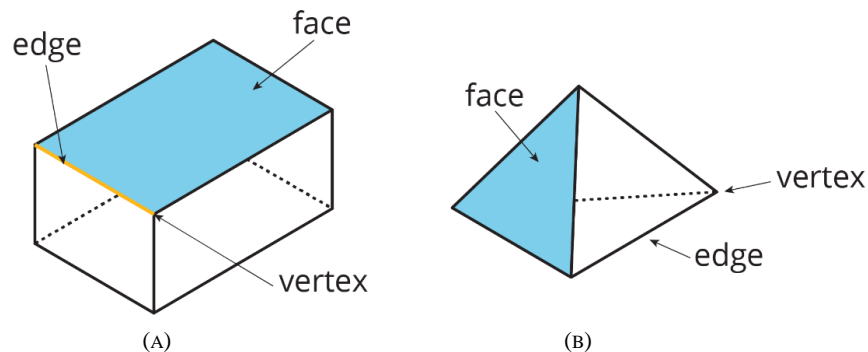


FIG. 1. (A) QUADRILATERALS REPRESENTATION, (B) TRIANGLE REPRESENTATION [25]

The 3D object is visualized through its geometry which represents the 3D surface. The surface must be characterized by material to achieve the vision laid out by the notion artist using a texture map. The texture map is a material that defines the physical characteristics of the 3D object surface [26].

The texture map is a 2D image that overlaid upon the geometry of a 3D model that improves surface details to add realism to computer graphics. It consists of an array of elements representing texture space where each element in the array is called texel, texture element, or texture pixel. Each texel is assigned to a vertex of the 3D object; it maps pixels from a texture to a 3D surface. (wrapping the image around the 3D object) [26]. Fig. 2. depicts the concept of texture mapping.

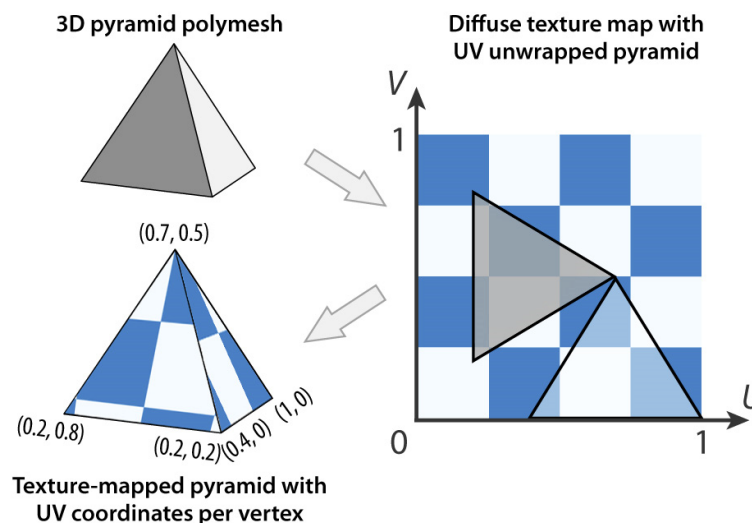


FIG.2. CONCEPT OF TEXTURE MAPPING [27]

Received 15/7/2021; Accepted 21/8/2021

VI. PROPOSED 3D MODEL ENCRYPTION SCHEME

This section will describe the main steps of the encryption and decryption process for the proposed scheme. The textured 3D model surface contains texture and a 3D mesh; the texture and 3D mesh model will be encrypted. The 3D mesh contains the vertices and faces, the vertices composing together faces by connecting every three vertices (face elements); each vertex consists of three coordinates (v_x , v_y , and v_z). The proposed encryption scheme has two main stages: firstly, encrypting texture stage, and secondly, encrypting 3D mesh stage.

A. Encrypting texture stage

The first stage of the encryption process is implemented by encrypting the texture of the 3D model based on keys generated by a 2D Logistic map with a length equal to the number of texels in the texture map of the 3D model.

B. Encrypting 3D mesh stage

The second stage of the encryption process will encrypt the mesh of the 3D model, which modifies the vertices values based on keys generated by a 3D Lorenz Chaotic map. When 3D Lorenz is implemented, it produces three keys at each iteration (key_1 , key_2 , and key_3) representing the 3D key, where key_1 is responsible for changing the v_x value, key_2 for changing v_y value, and key_3 for changing v_z value, such that

$$K = \{(K_{x1}, K_{y1}, K_{z1}), \dots, (K_{xn}, K_{yn}, K_{zn})\}$$

The vertices in the 3D model are listed as an array V .

$$V = \{(V_{x1}, V_{y1}, V_{z1}), \dots, (V_{xn}, V_{yn}, V_{zn})\}$$

Where n is the number of vertices in the model, Fig. 3 shows the 3D textured model Face Man and a close view of its triangle mesh.

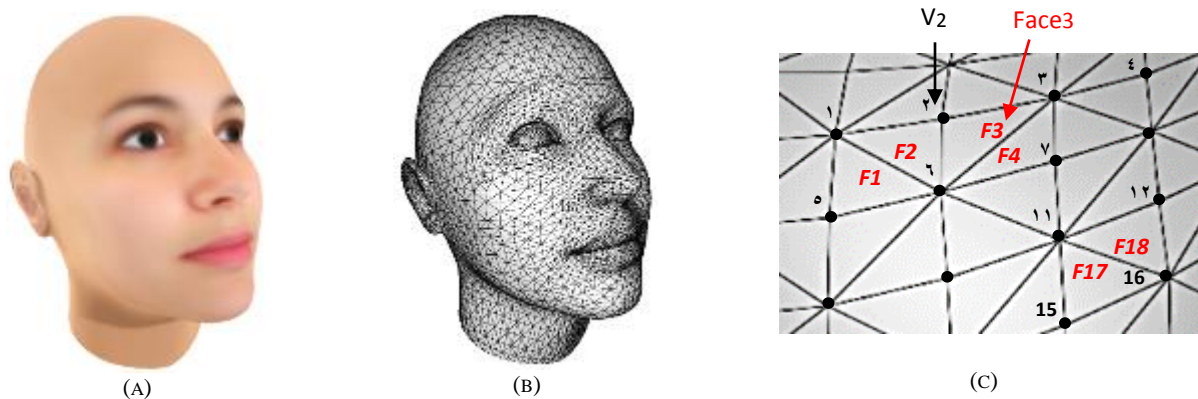


FIG. 3. (A) ORIGINAL 3D MODEL WITH TEXTURE, (B): 3D MESH MODEL, (C): CLOSE VIEW OF 3D MESH MODEL

The “.obj” file is used in the proposed scheme, which is a text-based, open file format used to exchange and store 3D data. The vertices and faces structures in the “.obj” file are shown in

Table 2, where the vertices and the faces are represented by a list of indices to reduce the size needed in memory.

TABLE 2. VERTICES AND FACES STRUCTURE REPRESENTATION

Vertices list information				Faces list information	
Index of vertex	x-coordinate	y-coordinate	z-coordinate	Index of face	Vertices index in each face
1	$V_{1,x}$	$V_{1,y}$	$V_{1,z}$	1	(5,6,1)
2	$V_{2,x}$	$V_{2,y}$	$V_{2,z}$	2	(1,6,2)
3	$V_{3,x}$	$V_{3,y}$	$V_{3,z}$	3	(6,3,2)
.....	4	(6,7,3)
7	$V_{7,x}$	$V_{7,y}$	$V_{7,z}$
.....
15	$V_{15,x}$	$V_{15,y}$	$V_{15,z}$	17	(15,16,11)
16	$V_{16,x}$	$V_{16,y}$	$V_{16,z}$	18	(11,16,12)
.....

The main steps of the proposed encryption scheme are listed in the encryption algorithm below.

Encryption Algorithm:

Input: 3D textured model in '. Obj' format.

Output: Encrypted 3D textured model.

Step1: Read 3D textured model, separate texture map and 3D mesh model from 3D textured model, store vertices of 3D mesh model in an array V and texture map in array T .

Step2: For each texel in the texture map:

apply 2D Logistic map to generate 2D key and perform encryption process using equation (5).

$$T^* = T \text{ xor } k(x, y) \quad (5)$$

End for.

Step3: For each vertex in the 3D mesh model, do the following.

Step4: Apply 3D Lorenz map to generate 3D key, key for x, key for y and key for z coordinates, store in K .

Step5: Apply the encryption process using equation (6) to modifying the values of vertices V .

$$V^*(x, y, z) = V(x, y, z) * W + K(x, y, z) \quad (6)$$

Where V^* is the encrypted vertex, V is the vertex in 3D mesh model, K is the 3D key generated by 3D Lorenz map, W is the weight factor to preserve the dimensionality and spatial stability, the value is chosen by trial and test, the best value is 0.5 .

Step6: End for

Step7: Combine encrypted textured map from step2 with encrypted 3D mesh model from step3-step6.

Step8: Save the encrypted 3D textured model as a new file.

Received 15/7/2021; Accepted 21/8/2021

The block diagram in Fig. 4 illustrates the encryption process of the 3D textured model. 3D Lorenz generate 3D key for each vertex $k(x,y,z)$ using parameters value as $a=16$, $r=45$, $b=4$, $x=0.1$, $y=0.1$ and $z=0.1$. The V_i is multiplied by the W factor then added with the 3D key.

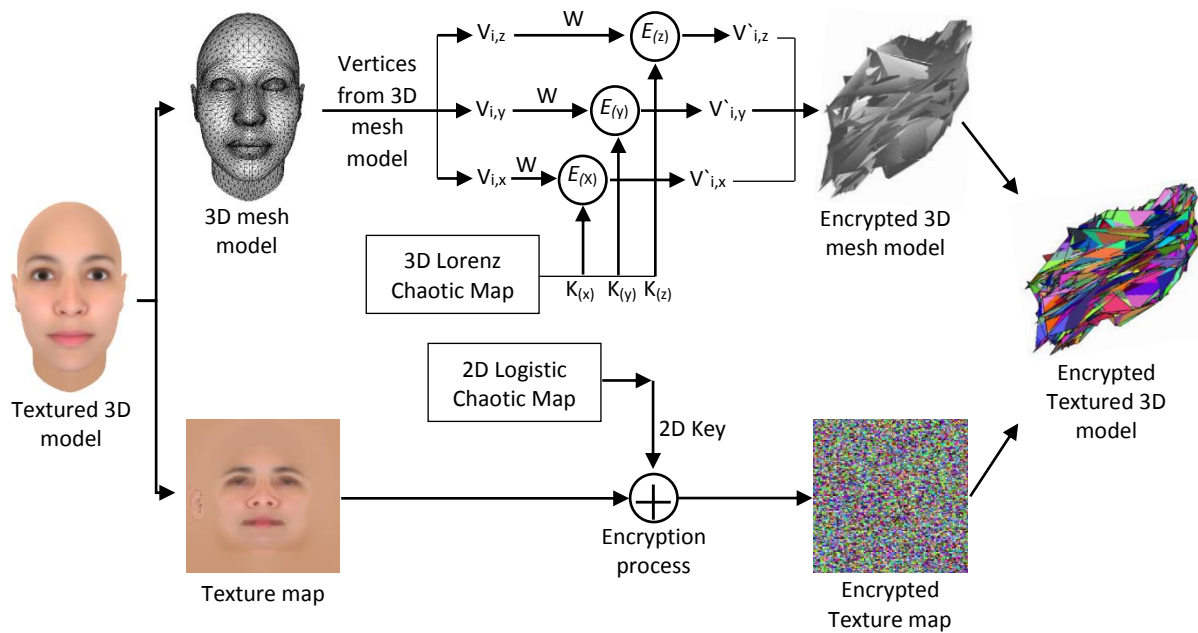


FIG. 4. BLOCK DIAGRAM FOR TEXTURED 3D MODEL ENCRYPTION PROCESS

The decryption process steps are illustrated in Fig. 5; they are similar to the encryption process steps but in reverse order, except dividing by the factor W instead of multiplication.

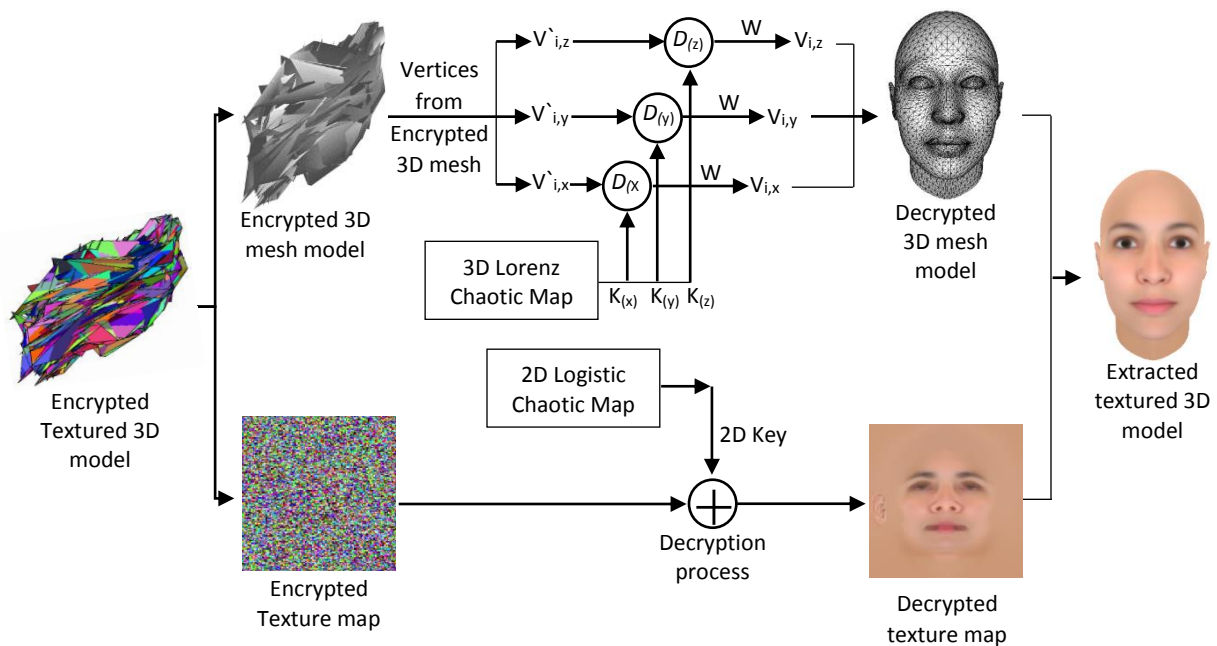


FIG. 5. BLOCK DIAGRAM FOR TEXTURED 3D MODEL DECRYPTION PROCESS

Received 15/7/2021; Accepted 21/8/2021

The 3D textured models may contain a single part of the 3D mesh model with a single texture map or more than one part of the 3D mesh. Each part includes a different group of vertices that have its own individual texture map; *Fig. 6* illustrates the Girl model, which has nine individual groups of vertices combined together to generate the 3D model, including feet, legs, hands, T-shirt, neck wrap, lips, face, eyes, and hair, for each vertices group assigned individual texture map which wrapper the vertices group to add the realism to the 3D model.

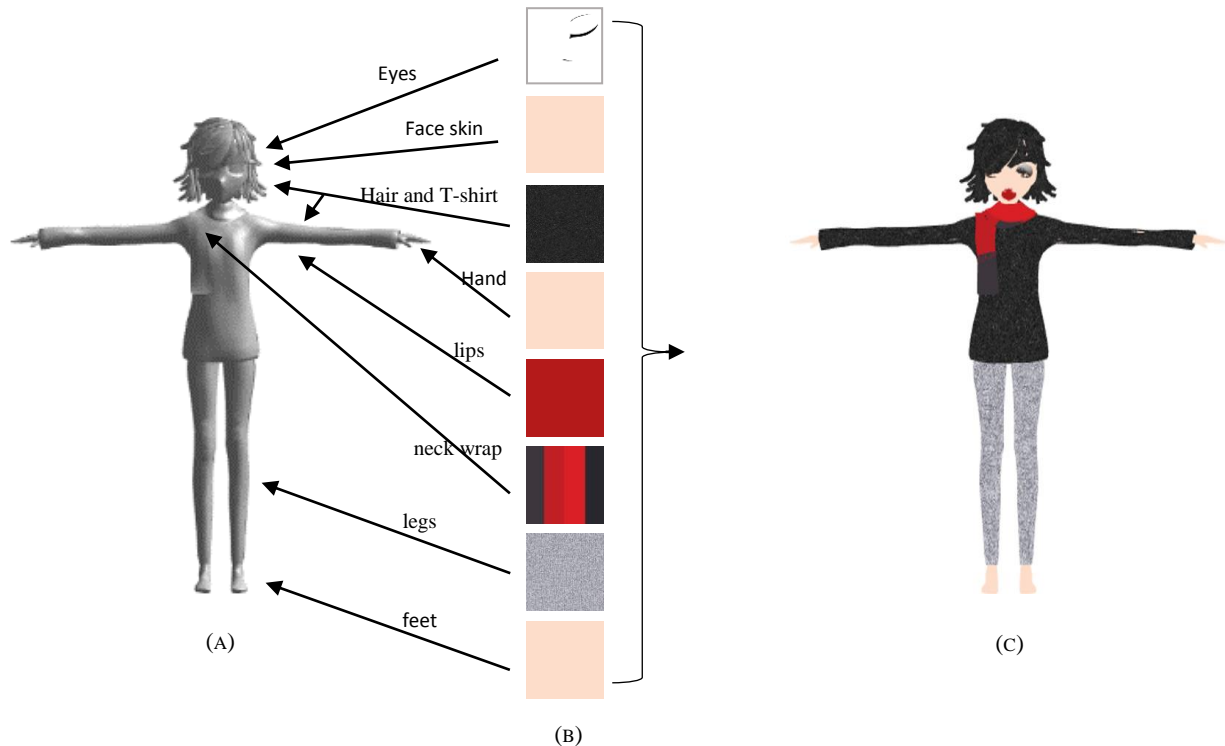


FIG. 6. (A) 3D MODEL WITHOUT TEXTURED, (B) EIGHT DIFFERENT TEXTURES MAP, (C) 3D TEXTURED MODEL


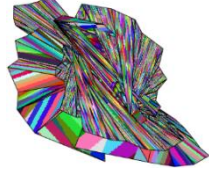

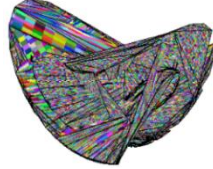

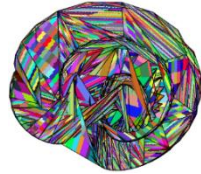

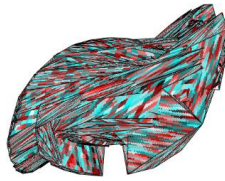


VII. SIMULATION RESULTS

The obj file structure is a text-based, open file format that contains the vertices position, faces as a list of vertices, position of texture coordinate vertex, and material file (mtl).

Various 3D textured models are used of type .obj file format to evaluate the efficiency and performance of the proposed encryption scheme (Vase, Hammer, Knife, Heart, and Girl), which are available at the Free3D website [28]. Free3D is a repository with more than nine thousand 3D models presented under many categories like plants, vehicles, sports, animals, and electronics. These models have a different number of vertices and faces. Table 3 shows the original 3D textured model and the encrypted textured 3D model. It also shows that the encryption time required (speed) is based on the number of vertices and texels, where it increases with the increasing number of vertices and texels, and vice versa.

Received 15/7/2021; Accepted 21/8/2021

TABLE 3. INFORMATION OF THE 3D TEXTURED MODEL ENCRYPTION PROCESS

Model name	No. of vertices	No. of faces	Elapsed time in sec.	Original textured 3D model	Encrypted textured 3D model
Vase	1716	572	1.1868		
Hammer	1818	606	1.2344		
Knife	3555	1185	2.4133		
Heart	33792	11264	17.5150		
Girl	54570	18190	27.0465		

The experimental results in Table 3 show that the encrypted models are entirely different from the original model due to firstly encrypting the texture of the 3D model and secondly encrypting the mesh of the 3D model by changing its vertices values (diffusion).

Received 15/7/2021; Accepted 21/8/2021

VIII. STATISTICAL TESTS

The proposed encryption scheme quality is evaluated by implementing the statistical tests Hausdorff Distance (HD) and histograms. The HD is an important measurement tool used to compute the degree of similarity between two points in two sets represented as HD (X, Y). The HD is used in various application domains like medical, 3D comparison, and pattern matching domains; for such applications, the HD can indicate the error between two-point sets. For two-point sets $X = \{x_1, x_2, x_3, \dots, x_{nx}\}$ and $Y = \{y_1, y_2, y_3, \dots, y_{ny}\}$, the HD takes the spatial position of each point where HD can be defined for two-point sets as [29]:

$$HD(Y, X) = \max(hd(X, Y), hd(Y, X)) \quad (7)$$

where

$$hd(X, Y) = \max_{x \in X} \min_{y \in Y} \|x - y\| \quad (8)$$

$$hd(Y, X) = \max_{y \in Y} \min_{x \in X} \|y - x\| \quad (9)$$

The directed Hausdorff Distance hd in Equations (8) and (9) between two point sets X and Y is the largest distance between each point $x \in X$ to its nearest neighbor $y \in Y$; it takes the maximum distance. The symbol $\| \ \|$ in equations (8) and (9) is the Euclidean distance between point x and point y, where equations (8) and (9) are known as directed Hausdorff distance. Equation (7) is the basic form of HD, which is known as undirected HD. The HD measures the large dis-similarity degree between a two-point set. The greater the HD such as closer to 100, the greater the difference (less similarity) between the two-point set [30, 31] is obtained.

The X and Y represent the two meshes, and $\|x - y\|$ is the Euclidean distance between x and y in the 3D space. If the HD nears zero, this means there is no difference between them and vice versa.

Table 4 illustrates the values of HD between the original, encrypted, and decrypted 3D textured models.

TABLE 4. THE RESULTS OF HAUSDORFF

Model name	Hausdorff after encryption	Hausdorff after decryption
Vase	77.7203	0.000000
Hammer	106.1833	0.000000
Knife	106.5557	0.000000
Heart	84.6480	0.000000
Girl	106.2288	0.000000

Table 4 shows that the HD values for encrypted 3D models are very high, which means they are completely different from the original models. In contrast, HD values for decrypted 3D models are zero, which means they are identical to the original model.

Table 5 shows a complete example of encryption and decryption steps for the heart 3D textured model.

TABLE 5. COMPLETE EXAMPLE OF HEART ENCRYPTION AND DECRYPTION WITH HISTOGRAM

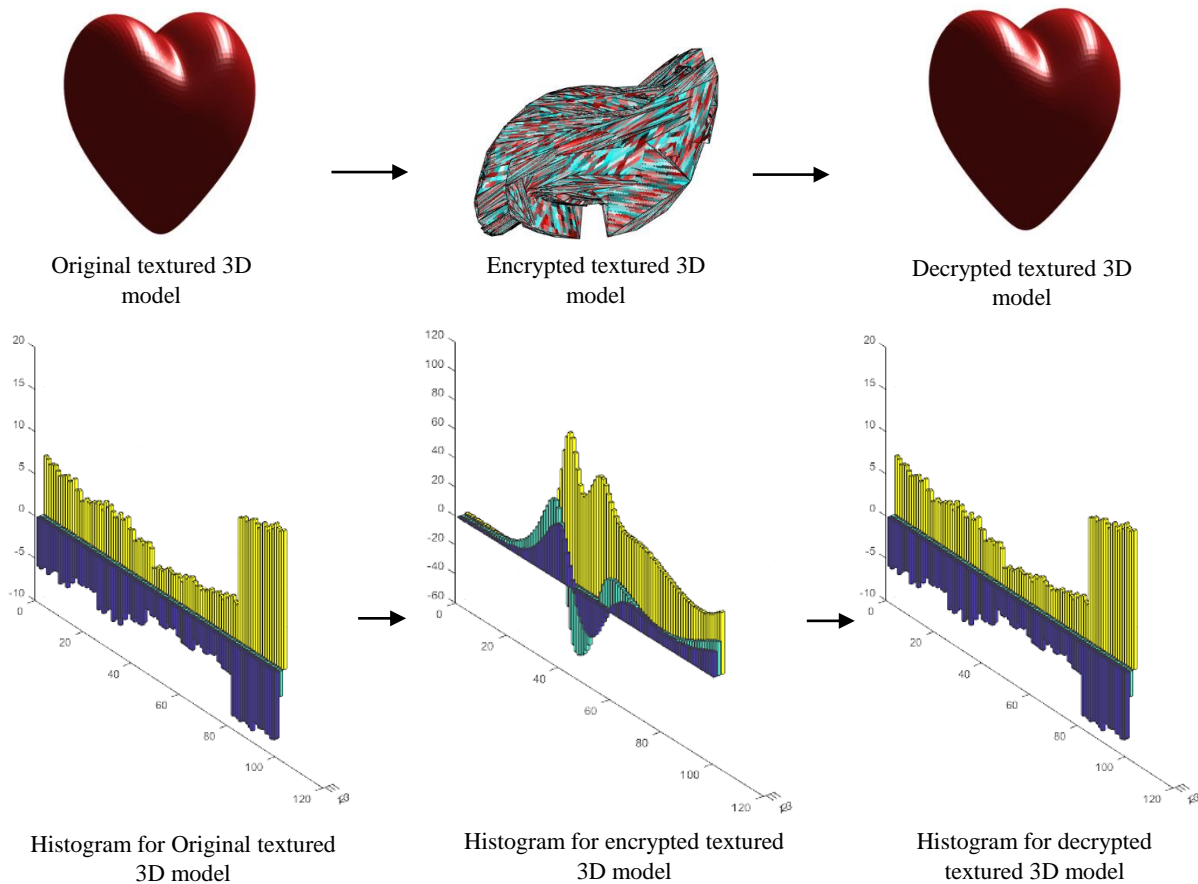


Table 5 illustrates that the histogram for the original and encrypted model is a clear difference to the eyes; this means the algorithm is resistant to statistical attack and completely identical between the histogram of the original and decryption model.

IX. SECURITY ANALYSIS

Security analysis can be used in encryption system by:

A. Key space

The key space must be large enough to resist the various attacks, such as brute-force attacks. When the key space is small, it will lead to break the cipher text by exhaustive search. The precision of 64-bit double data is 10^{-15} , in addition to six initial conditions parameters $a=16$, $r=45$, $b=4$, $x=0.1$, $y=0.1$, and $z=0.1$ by 3D Lorenz and three parameters from 2D Logistic map, resulting in the key space of size $(10^{15})^9 = 10^{135}$. Hence, our proposed scheme resists violent attacks because it has a large key space that makes it resist brute-force attacks.

Received 15/7/2021; Accepted 21/8/2021

B. Secret key sensitivity

A small change (one bit) in the initial condition value of the chaotic map will lead to the wrong decryption process and diffused the error almost to all vertices, so unable to extract the original 3D model.

X. TIME COMPLEXITY ANALYSIS

The encryption algorithm was implemented by Matlab 2018 on a laptop computer with a Core i7 CPU, Ram 16 G DDR4, and graphic card 4G, where the time needed for encryption and decryption are varying according to the number of vertices in the mesh of a 3D model and the size of the texture map.

XI. CONCLUSION

In this paper, different 3D textured models are encrypted based on two encryption stages, the texture encryption stage and the 3D mesh encryption stage (diffusion). From the results explained in the simulation results section, we conclude that the proposed scheme achieved the following:

- 1) Obtained good security by using different levels of encryption stages; each stage was implemented using different chaotic maps, which increased the complexity of the overall encryption scheme.
- 2) The security of the model was increased by breaking the correlation of the texture map using a 2D Logistic map.
- 3) Resist to brute-force attacks because it has large key space.
- 4) Maintained the dimensionality and spatial stability of the encrypted 3D model due to using the weight factor (w).
- 5) The decrypted model is identical to the original model according to the results of HD and histogram metrics.

REFERENCES

- [1] B. Raj, L. Jani Anbarasi, M. Narendra, and V. J. Subashini, "A New Transformation of 3D Models Using Chaotic Encryption Based on Arnold Cat Map," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 29, pp. 322–332, 2019.
- [2] M. A. A. J. A. Mizher, R. Sulaiman, A. M. A. Abdalla, and M. A. A. Mizher, "A simple flexible cryptosystem for meshed 3D objects and images," *J. King Saud Univ. - Comput. Inf. Sci.*, 2019.
- [3] S. M. Kareem and A. M. S. Rahma, "A Modification on Key Stream Generator for RC4 Algorithm," *Eng. Technol. J.*, vol. 38, no. 2B, pp. 54–60, 2020.
- [4] X. Jin et al., "Multi-Level Chaotic Maps for 3D Textured Model Encryption," in *2nd EAI International Conference on Robotic Sensor Networks*, pp. 107–117, 2020.
- [5] X. Wang, M. Xu, and Y. Li, "Fast encryption scheme for 3D models based on chaos system," *Multimed. Tools Appl.*, vol. 78, no. 23, pp. 33865–33884, 2019.
- [6] A. Jolfaei, X. W. Wu, and V. Muthukumarasamy, "A secure lightweight texture encryption scheme," in *Lecture Notes in Computer Science*, vol. 9555, pp. 344–356, 2016.
- [7] J. G. Sekar and C. Arun, "Comparative performance analysis of chaos based image encryption techniques," *J. Crit. Rev.*, vol. 7, no. 9, pp. 1138–1143, 2020.

Received 15/7/2021; Accepted 21/8/2021

- [8] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, pp. 17–25, 2016.
- [9] A. Jolfaei, X. W. Wu, and V. Muthukkumarasamy, "A secure lightweight texture encryption scheme," in *Lecture Notes in Computer Science*, vol. 9555, pp. 344–356, 2016.
- [10] X. Jin et al., "3D textured model encryption via 3D Lu chaotic mapping," *Sci. China Inf. Sci.*, vol. 60, no. 12, pp. 1–9, 2017.
- [11] C. Jia, T. Yang, C. Wang, B. Fan, and F. He, "Encryption of 3D Point Cloud Using Chaotic Cat Mapping," *3D Res.*, vol. 10, no. 1, 2019.
- [12] N. A. Hamza, S. H. Jafeer, and A. E. Ali, "Encrypt 3D Model Using Transposition, Substitution, Folding, and Shifting (TSFS)," in *SCCS 2nd Scientific Conference of Computer Sciences*, pp. 126–131, 2019.
- [13] H. A. Abdullah and H. N. Abdullah, "Secure Image Transmission Based on a Proposed Chaotic Maps," in *Multimedia Security Using Chaotic Maps: Principles and Methodologies*, K. M. Hosny, Ed. Springer Nature, pp. 81–109, 2020.
- [14] P. R. Sankpal and P. A. Vijaya, "Image encryption using chaotic maps: A survey," in *Proceedings - 2014 5th International Conference on Signal and Image Processing, ICSIP*, pp. 102–107, 2014.
- [15] Y. H. Ail and Z. A. H. Alobaidy, "Images Encryption Using Chaos and Random Generation," *Eng. Technol. J.*, vol. 34, no. 1 Part (B) Scientific, pp. 172–179, 2016.
- [16] A. S. Hamad and A. K. Farhan, "Image Encryption Algorithm Based on Substitution Principle and Shuffling Scheme," *Eng. Technol. J.*, vol. 38, no. 3B, pp. 98–103, 2020.
- [17] X. Huang, L. Liu, X. Li, M. Yu, and Z. Wu, "A new two-dimensional mutual coupled logistic map and its application for pseudorandom number generator," *Math. Probl. Eng.*, vol. 2019, pp. 1–10, 2019.
- [18] S. Harshitha, D. Ranjan, J. Madhushree, B. and L. Ashwini, "A Systematic Approach for Image Encryption Using Chaotic 2D Logistic Map Using MATLAB," *Int. J. Eng. Res. Technol.*, vol. 6, no. 13, pp. 1–4, 2018.
- [19] M. A. AlZain, "Efficient image cipher using 2D logistic mapping and singular value decomposition," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 11, pp. 196–200, 2018.
- [20] F. Masood, J. Ahmad, S. A. Shah, S. S. Jamal, and I. Hussain, "A novel hybrid secure image encryption based on Julia set of fractals and 3D lorenz chaotic map," *Entropy*, vol. 22, no. 3, 2020.
- [21] P. Rakheja, R. Vig, and P. Singh, "Double image encryption using 3D Lorenz chaotic system, 2D non-separable linear canonical transform and QR decomposition," *Opt. Quantum Electron.*, vol. 52, no. 2, 2020.
- [22] Z. N. Al-Qudsy, S. H. Shaker, and N. S. Abdulrazzque, "Robust Blind Digital 3D Model Watermarking Algorithm Using Mean Curvature," in *Third International Conference, New Trends in Information and Communications Technology Applications*, pp. 110–125, 2018.
- [23] S. Borah and B. Borah, "Three-Dimensional (3D) Polygon Mesh Authentication Using Sequential Bit Substitution Strategy," in *Advances in Intelligent Systems and Computing*, vol. 990, pp. 617–627, 2020.
- [24] R. Jiang, H. Zhou, W. Zhang, and N. Yu, "Reversible data hiding in encrypted three-dimensional mesh models," *IEEE Trans. Multimed.*, vol. 20, no. 1, pp. 55–67, 2018.
- [25] https://favpng.com/png_view/three-dimensional-prism-triangle-polyhedron-face-vertex-line-segment-png/f2kevVAA, Accessed Jun. 2021.
- [26] J. Alireza, "Robust Encryption Schemes for 3D Content Protection," Thesis (Ph.D. Doctorate), Griffith University, 2016.
- [27] G. J. Verhoeven and S. J. Missinne, "UNFOLDING LEONARDO da VINCI'S GLOBE (AD 1504) to REVEAL ITS HISTORICAL WORLD MAP," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. 4, no. 2W2, pp. 303–310, 2017.
- [28] "Free3D," <https://free3d.com/3d-models/obj>, Accessed Feb. 2021.
- [29] A. A. Taha and A. Hanbury, "An Efficient Algorithm for Calculating the Exact Hausdorff Distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 11, pp. 2153–2163, 2015.
- [30] D. Karimi and S. E. Salcudean, "Reducing the Hausdorff Distance in Medical Image Segmentation with Convolutional Neural Networks," *IEEE Trans. Med. Imaging*, vol. 39, no. 2, pp. 499–513, 2020.
- [31] X. Li, Y. Jia, F. Wang, and Y. Chen, "Image Matching Algorithm Based on an Improved Hausdorff Distance," *Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation*, vol. 68, 2013.

Received 15/7/2021; Accepted 21/8/2021