Multi-level encryption for 3D mesh model based on 3D Lorenz chaotic map and random number generator

Nashwan Alsalam Ali[1], Abdul Monem S. Rahma[2], Shaimaa H. Shaker[3]
[1]Department of Computer Sciences, College of Education for Women, University of Baghdad, Baghdad, Iraq
[2]Computer Sciences Department, Al-Maarif University College, Anbar, Iraq
[3]Department of Computer Sciences, University of Technology, Baghdad, Iraq

Article Info

Article history:

Received Jul 29, 2021
Revised Jun 8, 2022
Accepted Jul 3, 2022

Keywords:

3D Lorenz chaotic map
3D mesh model
Multi-level encryption
Random number generator
Shuffling

ABSTRACT

The increasing 3D model applications in various areas of life and widespread use like industry leads to 3D models being stolen and attacked by hackers; therefore, 3D model protection is a fundamental matter nowadays. In this paper, the proposed scheme will provide stringent security for the 3D models by implementing multiple levels of security with preserving the original dimensionality of the 3D model using the weight factor (w). The first level of security is achieved by applying a shuffling process for the vertices based on a key from random number generator (RNG), which provides good confusion. The second level is implemented by modifying the vertices values based on 3D keys from 3D Lorenz chaotic map, which provides good diffusion. The proposed scheme was applied on different 3D models varying in the vertices and faces number. The results illustrate that the proposed scheme deforms the entire 3D model based on Hausdorff distance (HD) approximately 100 after the encryption process, making it resist statistical attack. The scheme provides high security against brute force attack because it has a large key space equal to 10,105 and high security against deferential attack through secret key sensitivity using number of pixels change rate (NPCR) near to 99:6% and unified average changing intensity (UACI) near to 33:4%.

Corresponding Author:

Nashwan Alsalam Ali
Department of Computer Sciences, College of Education for Women, University of Baghdad
Baghdad, Iraq
Email: nashwan_alsalam60@coeduw.uobaghdad.edu.iq

1. 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 science that implements encoding and encryption processes which are done by a key that defines how the data are to be coded so only the authorized user can access and understand it [2]–[4]. Many algorithms for encryption are suggested to transform data into unidentifiable formats and prevent illegal access by users. Presently, 3D models have become an increasingly significant aspect of multimedia content; as their applications grow in popularity, there is a need for protecting these 3D contents; as a result, the thread problem must be resolved [5]. Many suggested encryption methods have been introduced and standardized around the world; however, they are not appropriate to encrypt a 3D model, such as data encryption standard (DES) and advanced encryption standard (AES), since the difficulty of 3D model encryption is because of the huge data size and application formats which cause a large time complexity [6], to solve that various chaotic cryptography-based encryption methods have been implemented. The researcher is drawn to and uses chaotic systems in cryptosystems because they have desirable qualities [7], [8]. In this paper, the proposed 3D model encryption method will be introduced, provides a multi-level of security with preserving the original dimensionality and spatial stability (usability) of a 3D model, which is considered a big problem in 3D model encryption that are not solved by other researchers. Our contribution is encrypting the 3D model with preserving the dimensionality (original size) as the original.

2. RELATED WORK

Jin *et al.* [9] apply the 3D encryption method using 3D Lu chaotic maps to encrypt 3D models with texture to achieve the main requirements of security for 3D content. They encrypted textures, vertices, and polygons using the same chaotic map, they used a color image for texture and encrypt each band separately (red, green, and blue) using 3D Lu chaotic maps and combined them to get the final encrypted texture after encryption was complete merge the encrypted texture with the encrypted mesh model to obtain the final encrypted content mixing with texture. The proposed encrypt the 3D textured model with good results, good resist brute-force attacks, good key space, and resist to statistical attack by histogram and positions of coordinate distribution according to the testing results.

Wang *et al.* [5] suggested a fast system for encrypting 3D models because the old encryption methods suffer from a long execution time. The authors scheme converts the 3D model to a 2D image then performs the encryption process. The encryption system is divided into two stages: the confused and diffusion phases. The authors inserted random numbers during the confusion phase, and throughout the diffusion phase, they separated the data, which they are of type floating-point, into two parts, including the integer part and the decimal parts. The XOR function is used to encrypt the integer component, whereas the decimal part was just jumbled. According to the security study, the scheme is very safe and resistant to common assaults.

Pham *et al.* [10] introduced a protection methodology to encrypt the 3D printing models using random encryption of the 3D triangle mesh geometry representation. Following geometrical modification, is the suggested approach uses a private key generated by the Hash function to encrypt the vertices of each deformed face at random. Shear transformation, 3D printing model destruction faces, and construction of a matrix of the 3*3 vertices from the destroyed faces vertices that represent all parts of the geometric transformation. To build the encrypted 3D printing model, the coefficients of the matrix are encrypted at random, employing a random integer from a different matrix. The entire 3D triangular mesh is transformed after the encryption procedure. The experimental results reveal that the suggested approach for 3D printing models is very efficient and secure.

Hamza *et al.* [11] recommended employing an encryption algorithm to encrypt a 3D mesh model without texture based on transformation, substitution, folding, and shifting (TSFS) for 3D object encrypting. The 3D model's vertices are fed into the TSFS algorithm through the encryption method. Three keys are used in the four stages of TSFS, firstly the vertices position in the transformation step are changed; secondly, the data matrix component is replaced with a different component; thirdly, the components of the matrix are folded diagonally, vertically, and horizontally; and in the last step, to replace the code with another in the final step of TSFS using the element 16. Good results are achieved for the encryption and decryption system based on the values of the peak signal to noise ratio (PSNR) and mean square error (MSE).

3. CRYPTOGRAPHY AND CHAOTIC SYSTEM

Data protection has many requirements, the most important requirements including confidentiality and security. Confidentiality and security are the most crucial for data protection. Cryptographic is defined as a proportionate mixing of chaotic mathematical theory and cryptography science. The chaotic system is made up of a dynamic equation that evolves over time. Chaotic is defined as a dynamic system that meets all three of the characteristics (topological mixing, periodic orbits density, and initial conditions sensitivity).

Between the chaotic system and a cryptographic system, there is an existence of a relationship; however, the main distinction has been that chaos operates on an infinite domain, whereas cryptography is confined on a finite domain [7], [12]. Cryptography-based chaos is a natural choice for cryptography and safe interaction because of the link between cryptography and chaotic systems. Chaotic systems and cryptography share initial conditions sensitivity, control parameters, unstable periodic orbits with enormous periods, and unpredictable behavior. The attacker sees the system output as random due to the unexpected behavior. But it appears predictable to the receiver, allowing decryption [13]–[16].

4. 3D LORENZ CHAOTIC MAP

Edward Lorenz developed mixed differential equations and produced a three-dimensional chaotic map called the 3D Lorenz. The array containing chaotic solutions for a Lorenz system is called attractors, which were formed by the Lorenz chaos sequences. In (1)-(3) illustrate a 3D Lorenz formula. The initial values for parameters are a=10, r=28, and b=8/3. The solution of curves for (1)-(3) are x, y, and z. The chaotic trajectory of the 3D Lorenz is based on the initial values of x, y, and z, where the permutation processes are based on them, and they constitute a secret key for them [17]–[19].

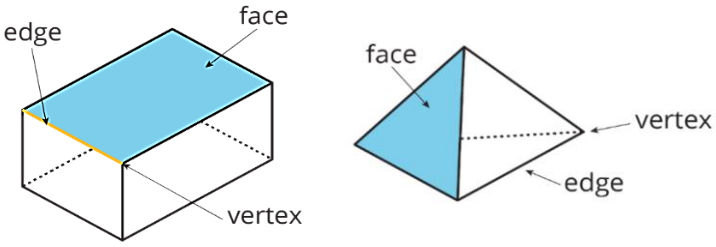
$$dx/dt = a(y - x) \tag{1}$$

$$dy/dt = rx - y - xz \tag{2}$$

$$dz/dt = xy - bz \tag{3}$$

5. THE 3D MODEL REPRESENTATION

The 3D model is expressed by a 3D polygon mesh; each polygon can be either triangle or quadrilaterals polygon depending on the sides, can be triangles if there are three sides are available, or quadrilaterals if four sides are there. The mathematical structure of the polygon mesh is labelled as *M={G, C}*, such that *G* denotes the data of geometry and *C* denotes the topological data. The geometry data is marked as *G={V, E, F}*, such that *V* is the vertices list, *E* is the edge connected between vertices, and *F* is the data for faces. The topological data includes the geometry elements connection information [20]–[23]. Figure 1 cited in the manuscript [24] demonstrates the representation 3D polygon structure with quadrilaterals in Figure 1(a) and triangle in Figure 1(b).



(a) (b)

Figure 1. 3D polygon structure with (a) quadrilaterals structure and (b) triangle structure

6. PROPOSED 3D MODEL ENCRYPTION SCHEME

The major phases of the encryption and decryption process for the proposed method will be described in this section. The 3D model surface contains texture and a 3D mesh; we will be encrypting the 3D mesh model without texture, which contains the vertices and faces. The vertices compose 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, shuffling the vertices stage, and secondly, modifying the vertices value stage.

Shuffling the vertices stage: the first stage of the encryption process is implemented by scrambling (permuting) the position of all vertices in the 3D mesh model based on a key generated by random number generator (RNG) with a length equal to the number of vertices in the 3D model so that the faces elements will be replaced with each other. This process represents the confusion stage, and it will be deforming the general shape of the 3D model. The sequence of numbers produced by RNG is determined by the uniform random number generator settings that underlie RNG, the seed number based on the clock that changes at every moment.

Modifying the vertices values stage: The second stage of the encryption process will modify the vertices values. It represents the diffusion stage in which the vertices values will be modified using a key generated by a 3D Lorenz chaotic map. When 3D Lorenz is used, three keys are generated for 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, and key_3 for changing v_z value. The two main encryption stages are illustrated in Figure 2, which shows the confusion and diffusion stages. The 3D model's vertices are provided in an array *V*, $V=\{(v_{x1}, v_{y1},$

v_{z1}), ……, (v_{xn}, v_{yn}, v_{zn})}, where n indicates the vertices number in the model. Figure 3 displays the face man details with its textured 3D model in Figure 3(a), 3D polygon mesh in Figure 3(b), and a see in detail of its triangular mesh in Figure 3(c).

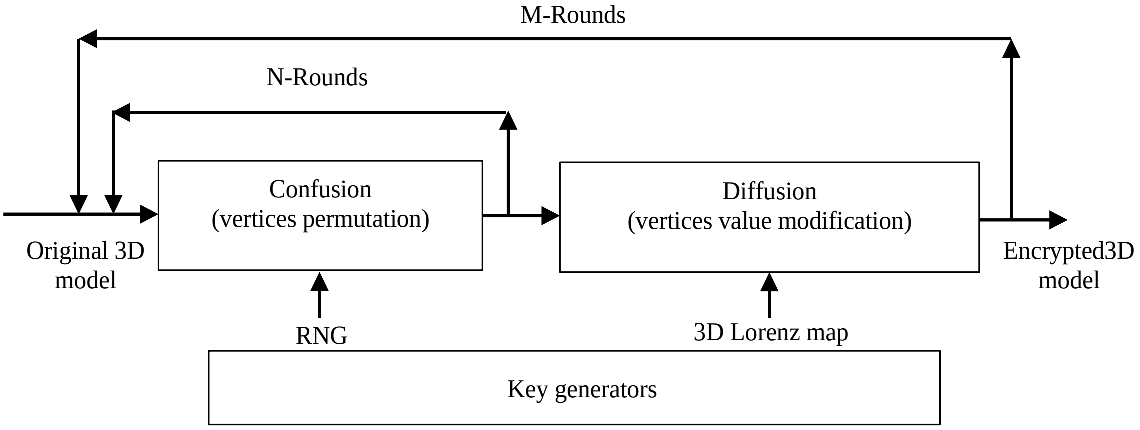


Figure 2. 3D model encryption stages

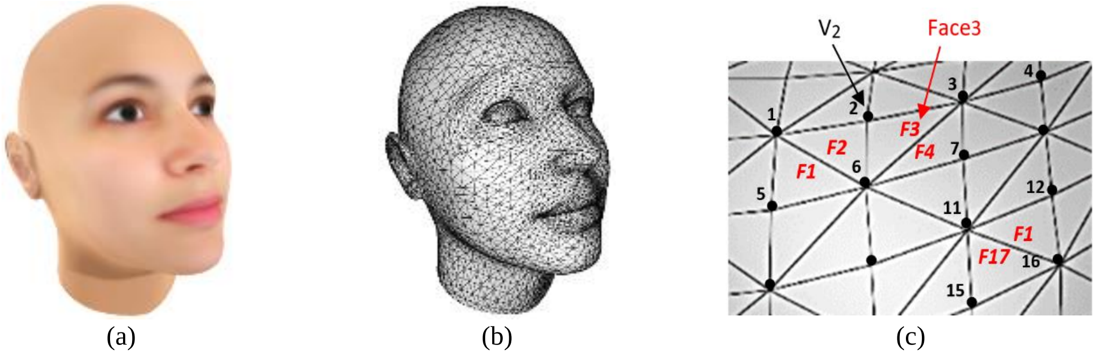


Figure 3. Face man with (a) 3D textured model, (b) 3D mesh model, and (c) a close view for 3D mesh

Table 1 shows the vertices and faces structures in the ".obj" file, where an array of indexes expresses the vertices and faces to decrease memory usage. In the right-hand side (face list information) in the Table 1, the indices of the vertices are used to represents the faces. The individual index refers to vertex coordinates, where each three indices are grouped to generates individual face.

Table 1. 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)

In the proposed scheme, the random number generator (RNG) is used to generate a random key for the shuffling process. The 3D Lorenz map is used for generating a random 3D key for each vertex in the model, such that, *K={(K_{x1}, K_{y1}, K_{z1}), …… (K_{xn}, K_{yn}, K_{zn})}*, where K is the 3D keys produced for all vertices in the model. The essential phases of the encryption process are explained in the algorithm 1.

Algorithm 1. Encryption algorithm

```
Input: Original 3D model.
Output: Encrypted 3D model.
Step 1: Input the 3D mesh model,
Step 2: List faces in an array $F$ and vertices in an array $V$.
Step 3: Apply the following for every all vertices in $V$.
Step 4: Scrambling the vertices in $V$ (Shuffling) based on a key generated by RNG and
        producing $V`$.
Step 5: Generate a 3D key from a 3D Lorenz chaotic map stored as $k=\{x, y, z\}$.
Step 6: Use (4) to change the values of $V`$ from step 4.

        $V^{``}(x,y.z) = \left(V^{`}(x,y,z) + K(x,y,z)\right) * W$          (4)

where $V^{``}$ is the encrypted vertex, $V`$ is the shuffled vertex in the 3D model, K is the 3D
key created by the 3D Lorenz map, and W is the weight factor for maintaining the original
dimensionality and spatial stability, the optimal value is 0.05.
Step 7: End for
Step 8: Store the result $V``$ as a new 3D model.
```

This algorithm contributes by preserving the original dimensionality of the encrypted 3D model. The proposed encryption method for the 3D model is depicted in Figure 4.

The decryption process steps are illustrated in Figure 5. The decryption process is similar to the encryption process steps but in reverse order. In the decryption process we will divide by the factor W instead of multiplication. The W factor is used to preserve the dimensionality and spatial stability.

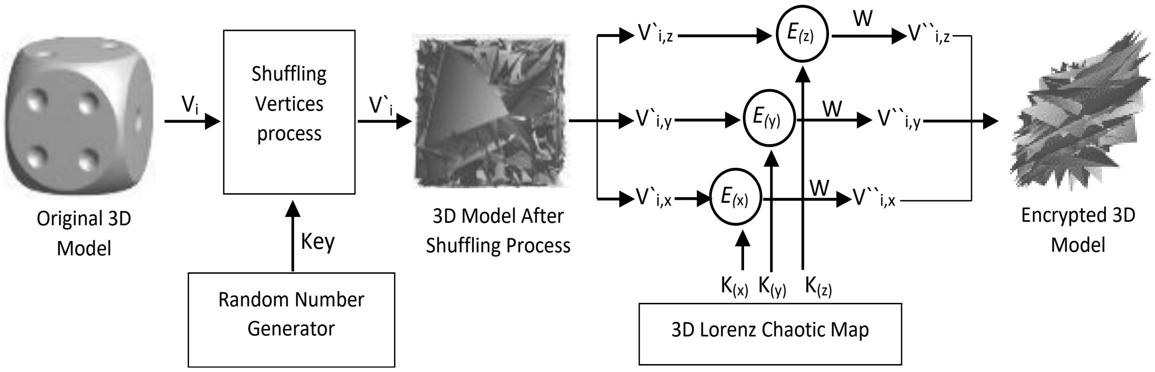


Figure 4. The 3D model encryption process

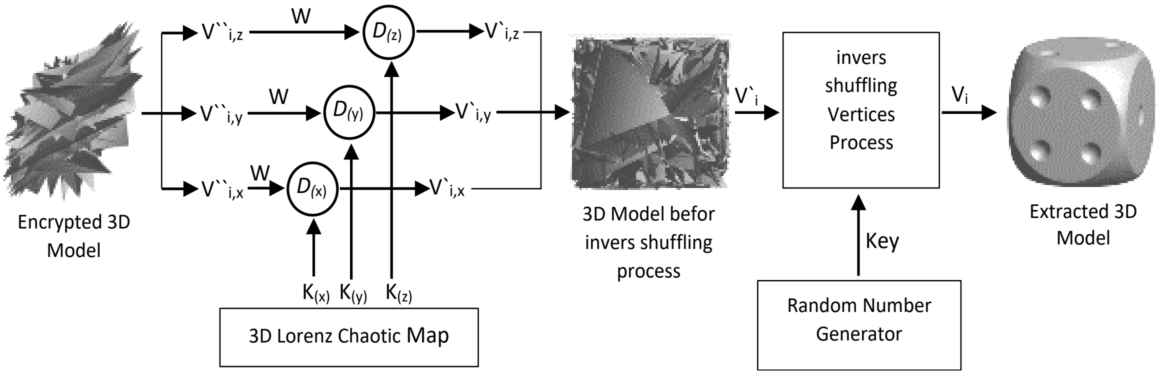


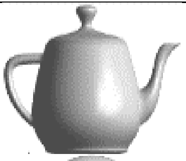
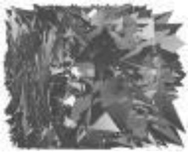
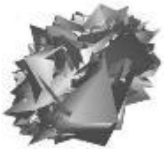
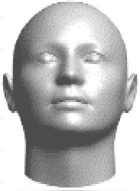
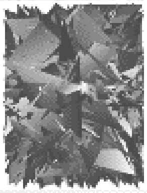
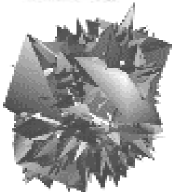
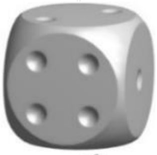
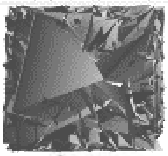
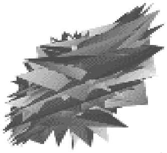
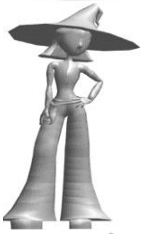
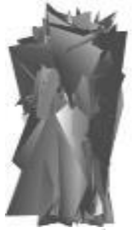
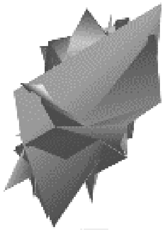
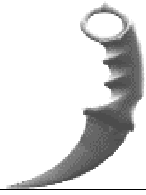
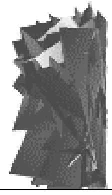
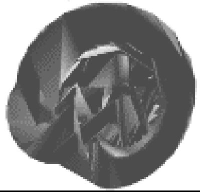
Figure 5. The 3D model decryption process

7. SIMULATION RESULTS

To test the effectiveness and performance of the proposed system, various 3D models of the type obj format are employed. (teapot, face man, dice, cowboy woman, and knife), which are available at the Free3D website [25]. 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 various number of vertices and faces. Table 2 shows the encryption process phases, including the original 3D model, shuffling process, modifying vertices values process, vertices number, and faces number for each model. It also demonstrates that the required encryption time is relatively proportional to the number of vertices and faces they contain and, with the number of vertices and faces growing as the number of vertices and faces rises.

Table 2 shows that the encrypted models change significantly from the original model. The significant changes are related to two factors. The first factor will be altering the vertices location (confusion). The second factor will be changing vertices value (diffusion). As a result all the statistical information will be eliminated, consequently, the proposed scheme resist statistical attack.

Table 2. 3D model encryption phases

Model name	No. of vertices	No. of faces	Elapsed time in sec.	Original 3D model	3D model after Shuffling process	3D model after modifying vertices values process
Teapot	47112	15704	12.364			
Face man	36522	12174	8.815			
Dice	23.088	7.696	6.350			
Cowboy woman	12.951	4.317	3.950			
Knife	3555	1185	1.445			

8. 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 many application fields like medical, pattern matching, and 3D comparison; 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:

$$HD\ (Y,X) = max\ (hd\ (X,Y), hd\ (Y,X)) \tag{5}$$

$$hd\ (X,Y) = max_{x\epsilon X}\ min_{y\epsilon Y}\ ||x - y|| \tag{6}$$

$$hd\ (Y,X) = max_{y\epsilon Y}\ min_{x\epsilon X}\ ||y - x|| \tag{7}$$

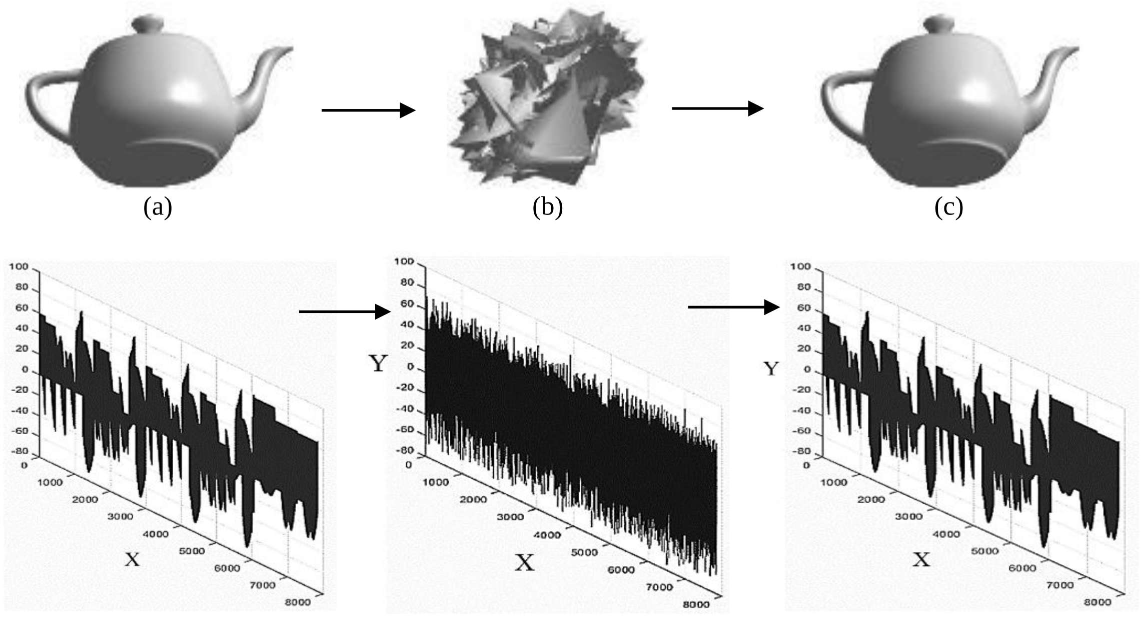
The directed Hausdorff distance hd in (6) and (7) between two-point sets X and Y is the largest distance between each point $x\epsilon X$ to its nearest neighbour $y\epsilon Y$; it takes the maximum distance. The symbol || || in (6) and (7) is the Euclidean distance between point x and point y, where (6) and (7) are known as directed Hausdorff distance. In (5) 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 is less similarity between the two-point set [26]–[28].

The X and Y represent the two meshes in the 3D space, and $||x - y||$ represent the Euclidean distance between two points x and y in the two meshes. If the HD value equals zero, this means there is no difference between them and vice versa. Table 3 depicts the HD values of the encrypted and decrypted 3D models.

Table 3. The results of Hausdorff

Model name	HD after encryption	HD after decryption
Face Man	124.5836	0.0000
Teapot	71.0325	0.0000
Dice	106.2076	0.0000
Cowboy Women	106.5517	0.0000
Knife	106.5528	0.0000

From Table 3, we note that the HD values for encrypted 3D models are very high, which means they are entirely different from the original models. Whereas HD values for decrypted 3D models are zero, which means they are identical to the original model. Figure 6 depicts Teapot 3D model with original 3D model in Figure 6(a), encrypted 3D model in Figure 6(b), decrypted 3D model in Figure 6(c), histogram for original 3D model in Figure 6(d), histogram for encrypted 3D model in Figure 6(e), and histogram for decrypted 3D model in Figure 6(f).



(a) (b) (c)

(d) (e) (f)

Figure 6. Teapot 3D model encryption and decryption with histogram (a) original 3D model, (b) encrypted 3D model, (c) decrypted 3D model, (d) histogram for Original 3D model, (e) histogram for encrypted 3D model, and (f) histogram for decrypted 3D model

From Figure 6, we can see that the histogram is identical to the original and decrypted 3D model. The decrypted 3D model is identical to the original. In contrast, the histogram for the original and encrypted 3D model is a clear difference which means the encrypted 3D model is entirely deformed. As a result, the proposed method is resistant to statistical attack.

9. SECURITY ANALYSIS

The two important analysis factors for security analysis include:

a. Key space: The key space must be large enough to withstand attacks like brute-force attacks. When the key space is small, it will lead to breaking the cipher text by exhaustive search. The precision of 64-bit data is 10^{15}, in addition to sex parameters from the 3D Lorenz map and initial seed of RNG resulting in the key space of size $(10^{15})^7=10^{105}$. As a result, our suggested method is resistant to brute-force attacks due to its large key space, so the attacker needs a very long time to break it by exhaustive search.

b. Secret key sensitivity: for providing high security to the encryption algorithm, the encryption algorithm must be sensitive to the initial secret key and the input data. A slight change (one bit) in the entered data or key will result in a massive change and significant effect on the encrypted model. The number of pixels change rate (NPCR) and unified average changing intensity (UACI) are used to achieve the key sensitivity and the effect of changing one pixel on the full cipher image where the randomization test may be used to assess the image robustness against various types of hackers [29], [30]. In (8) and (9) evaluate the NPCR and UACI by (8) and (9):

$$NPCR = (\Sigma D_{(i,j)} / M * N) * 100\% \tag{8}$$

$$UACI = 1/M * (\Sigma D_{i,j} (\mid c1_{(i,j)} - c1_{(i,j)} \mid /255) * 100\% \tag{9}$$

where, $D_{(i,j)=}0$ when $c_{1(i,j)} \neq c_{2(i,j)}$, else $D_{(i,j)}=1$.

The two metrics are used in our proposed method, where M and N are the dimensions of the plain and encrypted 3D model, $c1$ and $c2$ are the encrypted 3D model before and after modifying one vertex in the original 3D model, respectively. NPCR and UACI have precise values of 99:6% and 33:4%, respectively. By applying the proposed algorithm, Table 4 shows the values of the NPCR and UACI of the tested 3D models by implementing the proposed method. The values 99:63 and 33:52 of the NPCR and UACI are the most similar to the theoretical values. The results of Table 4 show that the proposed method is too sensitive to changes in the plain 3D model and that even small changes in the plain 3D model result in an entirely different cipher 3D model. As a result, the proposed method can withstand differential attacks. We concluded that if one bit of the key is wrong, that lead to cannot being recovered the plain 3D model. All selected random keys have the same sensitivity. Several secret key sensitivity experiments have been conducted; if the initial condition of the chaotic map is changed, it will cause an erroneous decryption procedure to occur, spreading the error to practically all vertices, making it impossible to retrieve the original 3D model. Table 4 shows the results of NPCR and UACI.

Table 4. The NPCR and UACI applied on the 3D model

Model	NPCR	UACI
Teapot	0.99991	0.33345
Face man	0.99984	0.33339
Dice	0.99843	0.33388
Cowboy Women	0.99786	0.33449
Knife	0.99862	0.33391

10. TIME COMPLEXITY ANALYSIS

MATLAB 2018 was used to implement the encryption algorithm; it was applied using a computer that has a processor Ryzen 9, RTX graphics card, and 16 G DDR4 RAM. The amount of time it takes to encrypt and decode a 3D model is based on the number of vertices in the 3D mesh model. As seen in Table 2, the Teapot 3D model having more vertices results in greater time spent, and the one which has less number of vertices will spend less time.

11. CONCLUSION

In this paper, different 3D mesh models are encrypted based on two encryption stages shuffling vertices stage (confusion) and modifying vertices value stage (diffusion). The 3D mesh was encrypted using confusion and diffusion process compared with [9]; the encryption algorithm in [9] only has the diffusion process and no confusion process, which means the cipher model can recover easily by rebuilding the 3D model. From the explained results section, we conclude that the proposed scheme achieved the following: i) it obtained good security by using two different encryption stages that increased the complexity of the overall encryption scheme; ii) resist brute-force attacks because it has large key space $(10^{15})^7$ and resist statistical attack through the HD and histogram results; iii) maintained the dimensionality of the encrypted 3D model due to using the weight factor (w); and iv) the decrypted model is identical to the original model according to HD and histogram metrics.

ACKNOWLEDGEMENTS

The authors would like to thank the University of Technology, Baghdad-Iraq, for its support in the present work.

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, Springer International Publishing, 2019, pp. 322–332, doi: 10.1007/978-3-030-12839-5_29.

[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,” *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 6, pp. 629–646, Jul. 2021, doi: 10.1016/j.jksuci.2019.03.008.

[3] S. Kareem and A. M. Rahma, “A modification on key stream generator for RC4 algorithm,” *Engineering and Technology Journal*, vol. 38, no. 2, pp. 54–60, Jul. 2020, doi: 10.30684/etj.v38i2b.404.

[4] O. M. Al-hazaimeh, “A new speech encryption algorithm based on dual shuffling Hénon chaotic map,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 3, pp. 2203–2210, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2203-2210.

[5] X. Wang, M. Xu, and Y. Li, “Fast encryption scheme for 3D models based on chaos system,” *Multimedia Tools and Applications*, vol. 78, no. 23, pp. 33865–33884, Dec. 2019, doi: 10.1007/s11042-019-08171-2.

[6] A. Jolfaei, X.-W. Wu, and V. Muthukkumarasamy, “A secure lightweight texture encryption scheme,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9555, Springer International Publishing, 2016, pp. 344–356, doi: 10.1007/978-3-319-30285-0_28.

[7] J. G. Sekar and C. Arun, “Comparative performance analysis of chaos based image encryption techniques,” *Journal of critical reviews*, vol. 7, no. 09, pp. 1138–1143, Jun. 2020, doi: 10.31838/jcr.07.09.209.

[8] L. Xu, Z. Li, J. Li, and W. Hua, “A novel bit-level image encryption algorithm based on chaotic maps,” *Optics and Lasers in Engineering*, vol. 78, pp. 17–25, Mar. 2016, doi: 10.1016/j.optlaseng.2015.09.007.

[9] X. Jin *et al.*, “3D textured model encryption via 3D Lu chaotic mapping,” *Science China Information Sciences*, vol. 60, no. 12, Dec. 2017, doi: 10.1007/s11432-017-9266-1.

[10] N.-G. Pham, S.-H. Lee, O.-H. Kwon, and K.-R. Kwon, “3D printing model random encryption based on geometric transformation,” *International Journal of Machine Learning and Computing*, vol. 8, no. 2, pp. 186–190, Apr. 2018, doi: 10.18178/ijmlc.2018.8.2.685.

[11] N. A. Hamza, S. H. Jafeer, and A. E. Ali, “Encrypt 3D model using transposition, substitution, folding, and shifting (TSFS),” in *2nd Scientific Conference of Computer Sciences (SCCS)*, Mar. 2019, pp. 126–131, doi: 10.1109/SCCS.2019.8852600.

[12] H. A. Abdullah and H. N. Abdullah, “Secure image transmission based on a proposed chaotic maps,” in *Studies in Computational Intelligence*, vol. 884, Springer International Publishing, 2020, pp. 81–109, doi: 10.1007/978-3-030-38700-6_4.

[13] P. R. Sankpal and P. A. Vijaya, “Image encryption using chaotic maps: a survey,” in *Fifth International Conference on Signal and Image Processing*, Jan. 2014, pp. 102–107, doi: 10.1109/ICSIP.2014.80.

[14] Y. H. Ail and Z. A. H. Alobaidy, “Images encryption using chaos and random generation,” *Engineering and Technology Journal*, vol. 34, no. 1 Part (B) Scientific, pp. 172–179, 2016

[15] A. Hamad and A. Farhan, “Image encryption algorithm based on substitution principle and shuffling scheme,” *Engineering and Technology Journal*, vol. 38, no. 3B, pp. 98–103, Dec. 2020, doi: 10.30684/etj.v38i3b.433.

[16] A. F. Shimal, B. H. Helal, and A. T. Hashim, “Extended of TEA (ETEA): a 256 bits block cipher algorithm for image encryption,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 3996–4007, Oct. 2021, doi: 10.11591/ijece.v11i5.pp3996-4007.

[17] 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, Feb. 2020, doi: 10.3390/e22030274.

[18] P. Rakheja, R. Vig, and P. Singh, “Double image encryption using 3D Lorenz chaotic system, 2D non-separable linear canonical transform and QR decomposition,” *Optical and Quantum Electronics*, vol. 52, no. 2, Feb. 2020, doi: 10.1007/s11082-020-2219-8.

[19] O. M. Al-hazaimeh, “A new dynamic speech encryption algorithm based on Lorenz chaotic map over internet protocol,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 5, pp. 4824–4834, Oct. 2020, doi: 10.11591/ijece.v10i5.pp4824-4834.

[20] J. Alireza, “Robust encryption schemes for 3D content protection,” Ph.D. dissertation, School of Information and Communication Technology, Griffith Sciences, Griffith University, 2016.

[21] R. Jiang, H. Zhou, W. Zhang, and N. Yu, “Reversible data hiding in encrypted three-dimensional mesh models,” *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 55–67, Jan. 2018, doi: 10.1109/TMM.2017.2723244.

[22] 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, Springer Singapore, 2020, pp. 617–627, doi: 10.1007/978-981-13-8676-3_52.

[23] Z. N. Al-Qudsy, S. H. Shaker, and N. S. Abdulrazzque, “Robust blind digital 3D model watermarking algorithm using mean curvature,” in *Communications in Computer and Information Science*, vol. 938, Springer International Publishing, 2018, pp. 110–125, doi: 10.1007/978-3-030-01653-1_7.

[24] Claudiouhl, “Three-dimensional prism-triangle polyhedron face vertex line segment PNG PNG image,” *Favpng*, https://favpng.com/png_view/three-dimensional-prism-triangle-polyhedron-face-vertex-line-segment-png/f2kevvaa, (accessed Jul. 1, 2021).

[25] “Free 3D Models,” *free3D*, https://free3d.com/3d-models/obj (accessed Feb. 2, 2021).

[26] A. A. Taha and A. Hanbury, “An efficient algorithm for calculating the exact hausdorff distance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2153–2163, Nov. 2015, doi: 10.1109/TPAMI.2015.2408351.

[27] D. Karimi and S. E. Salcudean, “Reducing the Hausdorff distance in medical image segmentation with convolutional neural networks,” *IEEE Transactions on Medical Imaging*, vol. 39, no. 2, pp. 499–513, Feb. 2020, doi: 10.1109/TMI.2019.2930068.

[28] X. Li, Y. Jia, F. Wang, and Y. Chen, “Image matching algorithm based on an improved Hausdorff distance,” in *Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation*, 2013, vol. 68, doi: 10.2991/3ca-13.2013.61.

[29] B. Yousif, F. Khalifa, A. Makram, and A. Takieldeen, “A novel image encryption/decryption scheme based on integrating multiple chaotic maps,” *AIP Advances*, vol. 10, no. 7, Jul. 2020, doi: 10.1063/5.0009225.

[30] A. Elghandour, A. Salah, and A. Karawia, “A new cryptographic algorithm via a two-dimensional chaotic map,” *Ain Shams Engineering Journal*, vol. 13, no. 1, Jan. 2022, doi: 10.1016/j.asej.2021.05.004.

BIOGRAPHIES OF AUTHORS

Nashwan Alsalam Ali is presently one of the college faculty of education for women, computer science department, University of Baghdad, Iraq. He received his B.Sc. degree in Computer Science in 2003 from Technology University in Baghdad, Iraq, his M.Sc. degree in Computer Science focusing on Multimedia Security from Iraqi Commission for Computers and Informatics from Baghdad, Iraq. He is currently a Ph.D. student in Computer Science, Technology University in Baghdad, Iraq. He can be contacted at email: nashwan_alsalam60@coeduw.uobaghdad.edu.iq.

Abdul Monem S. Rahma received his B.Sc. degree in Mathematics, Al- Mustansiriya University, Baghdad, Iraq, in 1977. His M.Sc. in Numerical Analysis, Brunel University, the United Kingdom, in 1982. His Ph.D. in Computer Science, Loughborough University, the United Kingdom, in 1984. He is presently one of the Faculty Computer Science Department, Technology University, Baghdad, Iraq. His research interests focus on image and video processing, pattern recognition, and information security. He can be contacted at email: 110003@uotechnology.edu.iq.

Shaimaa H. Shaker earned her bachelor’s and master’s degree in Computer Science from the Department of Computer Science at the University of Technology-Baghdad-Iraq. Her Ph.D. in Computer science from the Department of Computer Science at the University of Technology-Baghdad-Iraq since 2006. She presently is the head of the networks management branch since 2017-till until now. Her research interests focus on image processing, pattern recognition, and security visual cryptography systems. She can be contacted at email: Shaimaa.h.shaker@uotechnology.edu.iq.