Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.040729
Article

3D Model Encryption Algorithm by Parallel Bidirectional Diffusion and 1D Map with Sin and Logistic Coupling

Yongsheng Hu*

School of Information Engineering, Binzhou University, Binzhou, 256603, China
*Corresponding Author: Yongsheng Hu. Email: huys1208@bzu.edu.cn
Received: 29 March 2023; Accepted: 09 May 2023; Published: 28 July 2023

Abstract: 3D models are essential in virtual reality, game development, architecture design, engineering drawing, medicine, and more. Compared to digital images, 3D models can provide more realistic visual effects. In recent years, significant progress has been made in the field of digital image encryption, and researchers have developed new algorithms that are more secure and efficient. However, there needs to be more research on 3D model encryption. This paper proposes a new 3D model encryption algorithm, called the 1D map with sin and logistic coupling (1D-MWSLC), because existing digital image encryption algorithms cannot be directly applied to 3D models. Firstly, this paper introduce 1D-MWSLC, which has a wider range of parameters compared to traditional 1D chaotic systems. When the parameter exceeds a specific range, the chaotic phenomenon does not weaken. Additionally, 1D-MWSLC has two control parameters, which increases the cryptosystem's parameter space. Next, 1D-MWSLC generates keystreams for confusion and diffusion. In the confusion stage, this paper use random confusion, and the keystream generates an index matrix that confuses the integer and decimal parts of the 3D model simultaneously. In the diffusion stage, this paper use parallel bidirectional diffusion to simultaneously diffuse the integer parts of the three coordinates of the 3D model. Finally, this paper verify the proposed algorithm through statistical analysis, and experimental results demonstrate that the proposed 3D model encryption algorithm has robust security.

Keywords: Parallel bidirectional diffusion; chaos theory; 1D-MWSLC; image encryption; information security

1 Introduction

The Internet's rapid development and widespread usage have brought convenience to our lives but also created a range of privacy and security issues [1–4]. Various applications and services generate a significant amount of data on the Internet, including sensitive information such as personal health, finance, and shopping records [5–8]. Hackers may attack or leak this data to third parties, posing threats to privacy and data security [9–13].

Digital images have become an integral part of modern society, playing a significant role in various aspects of our daily lives. Digital images can quickly and easily convey information [14–17], illustrate an idea, provide visual examples, and enhance the impact of text-based communication [18–22]. Chaotic image encryption is a technology that uses chaotic systems to encrypt digital images [23–26]. The objective is to create a secure and efficient encryption method to protect sensitive image data from unauthorized access. In recent years, researchers have made significant progress in this field, developing new encryption algorithms that use chaotic maps to generate encryption keys for image encryption [27–32]. These algorithms have proven more secure and efficient than traditional encryption methods [33–35]. For instance, Mansoor et al. [36] utilized two one-dimensional chaotic maps to generate pseudorandom sequences and proposed a unique hybrid adaptive image encryption (HAIE). Pratyusha et al. designed a new encryption method to use a new conservative chaotic system based on memristors, aiming to prevent external attacks [37].

Although digital image encryption algorithms have proven to be efficient and secure, they are not suitable for 3D models [38–42]. The representation of 3D models is different from that of digital images, as they provide more precise information about the size, shape, and location of objects, enabling more accurate measurements and analysis in various fields such as medicine and engineering design [43–46].

To address this issue, this paper proposes a 3D model encryption algorithm to ensure secure transmission of 3D models over the internet. Firstly, a new chaotic system called 1D map with sin and logistic coupling (1D-MWSLC) is introduced. This system has two control parameters, providing a larger parameter space for cryptographic systems. 1D-MWSLC is then utilized to encrypt 3D models, with a secret key generated based on the data information of the 3D model. Four keystreams are generated for scrambling and diffusion according to 1D-MWSLC. The scrambling phase simultaneously scrambles the integer and fractional portions of the 3D model. In the diffusion phase, a parallel bidirectional diffusion strategy is proposed to diffuse the integer parts of the three coordinates of the 3D model simultaneously, starting from the center position of each coordinate axis. This approach increases the security of the algorithm.

This paper presents the following main contributions:

1. A novel image encryption algorithm is proposed for 3D models.
2. Based on the Sin Map and Logistic Map, a 1D-MWSLC is introduced with a wide chaotic range and good randomness, which is highly suitable for cryptography.
3. The Parallel Bidirectional Diffusion strategy is proposed to improve the efficiency of the encryption system.
4. Simulation experiments demonstrate the effectiveness of the proposed algorithm, and comparison with state-of-the-art methods shows superior performance.

The remaining sections of this paper are arranged as follows. Section 2 describes the proposed 1D Map with Sin and Logistic Coupling (1D-MWSLC) and analyzes its dynamic behavior through methods such as chaotic attractor. Section 3 describes the proposed encryption algorithm, including normalization and key generation, generating the keystream of the cryptographic system, scrambling, and parallel bidirectional diffusion. Section 4 presents the simulation experiments of the algorithm and validates its practicality through statistical tests. Section 5 concludes the paper and describes future work.

2 Chaotic System

2.1 Sin Map and Logistic Map

The Sin map is represented by a mathematical expression which is as follows [47]:

$$X(r+1) = \beta \sin(\pi X(r)). \tag{1}$$

where, $X(1)$ is the initial values of Sin map, β is the parameter of Sin map.

The Logistic Map is represented by a mathematical expression which is as follows:

$$X(r+1) = \alpha X(r)(1 - X(r)). \tag{2}$$

where, $X(1)$ is is the initial values of Logistic Map, α is the parameter of Logistic Map.

2.2 1D Map with Sin and Logistic Coupling (1D-MWSLC)

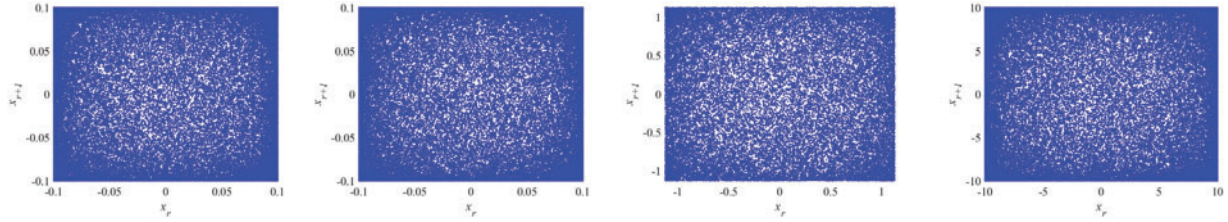
One-dimensional chaotic maps have fewer parameters and smaller ranges, chaos will be reduced or even eliminated once the parameters exceed a certain range. Therefore, this paper proposes the 1D-MWSLC. The 1D-MWSLC is represented by a mathematical expression which is as follows:

$$X(r+1) = \beta \sin(e^{5\pi} \alpha X(r)(1 - X(r))). \tag{3}$$

where, $X(1)$ is is the initial values of 1D-MWSLC, $X \in [-\beta, \beta]$, α ($\alpha \in (0, +\infty)$) and β ($\beta \in (0, +\infty)$) are the parameters of 1D-MWSLC.

2.3 Chaotic Attractor

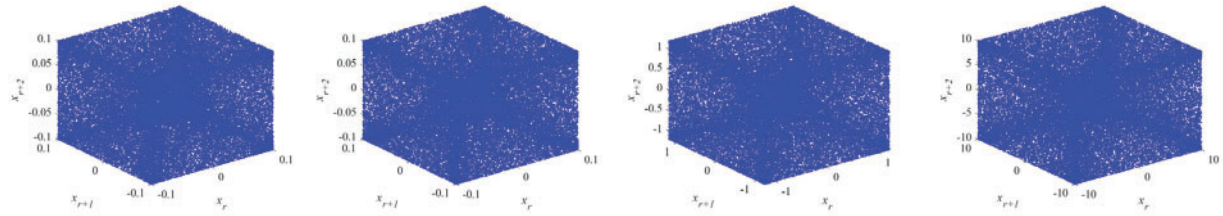
Chaotic attractors describe the changes in the output of a system. Systems with complex, chaotic behaviors have complex attractors occupying a large space. Fig. 1 describes the 2D chaotic attractor of 1D-MWSLC. Fig. 2 describes the 3D chaotic attractor of 1D-MWSLC. Note that x is randomly selected and can be any value. To show the chaotic attractor, this paper randomly select the initial value of 1D-MWSLC is $x(1) = 0.94905165561134$.



(a) $\alpha = 0.1, \beta = 0.1$ (b) $\alpha = 1.5, \beta = 0.1$ (c) $\alpha = 4.2, \beta = 1.2$ (d) $\alpha = 10, \beta = 10$

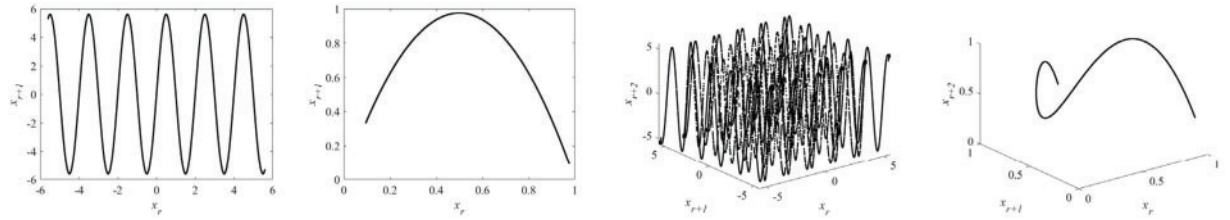
Figure 1: 2D chaotic attractor of 1D-MWSLC

Fig. 3 describes the 2D and 3D chaotic attractors of the Sin map and Logistic map. Set the same parameters and initial values as 1D-MWSLC. The comparison results show that under the same parameters and initial values, 1D-MWSLC has more complex attractors and occupies more space than the Sin and Logistic maps. 1D-MWSLC has good chaotic behavior and generates random sequences.



(a) $\alpha = 0.1, \beta = 0.1$ (b) $\alpha = 1.5, \beta = 0.1$ (c) $\alpha = 4.2, \beta = 1.2$ (d) $\alpha = 10, \beta = 10$

Figure 2: 3D chaotic attractor of 1D-MWSLC

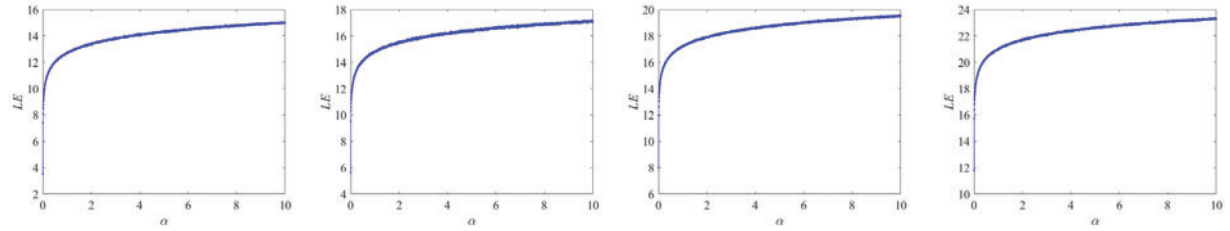


(a) 2D for Sin ($\beta = 5.6$) (b) 2D for Logistic ($\alpha = 3.9$) (c) 3D for Sin ($\beta = 5.6$) (d) 3D for Logistic ($\alpha = 3.9$)

Figure 3: Chaotic attractor of 3D and 2D of Sin map and Logistic map

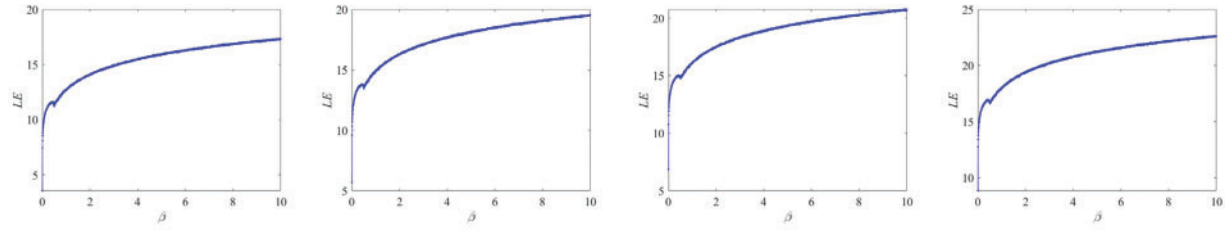
2.4 Lyapunov Exponents

The Lyapunov exponent (LE) is applied to evaluate chaotic behavior of systems. Fig. 4 describes Lyapunov exponent of the 1D-MWSLC with different parameters β. Fig. 5 describes Lyapunov exponent of the 1D-MWSLC with different parameters α.



(a) LE of $\beta = 0.1$ (b) LE of $\beta = 0.9$ (d) LE of $\beta = 3$ (f) LEof $\beta = 20$

Figure 4: Lyapunov exponents of the 1D-MWSLC with different parameters β

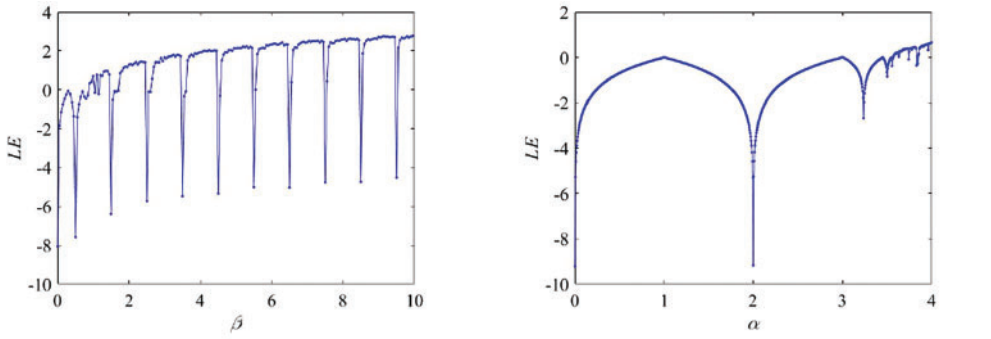


(a) LEof $\alpha = 0.1$ (b) LE of $\alpha = 0.9$ (c) LE of $\alpha = 3$ (d) LE of $\alpha = 20$

Figure 5: Lyapunov exponents of the 1D-MWSLC with different parameters α

LE analysis shows that 1D-MWSLC is globally chaotic, and the value of LE tends to increase as β and α increase, indicating that the larger the values of α and β, the better the chaotic performance of 1D-MWSLC.

Fig. 6 describes the Lyapunov exponents of the Sin map and Logistic map. Because the Sin map and Logistic map have fewer parameters and smaller ranges, chaos will be reduced or even eliminated once the parameters exceed a certain range. Compared to Sin and Logistic maps, 1D-MWSLC has a larger chaotic range.



(a) Lyapunov exponents of Sin Map (b) Lyapunov exponents of Logistic Map

Figure 6: Lyapunov exponents of Sin map and Logistic map

2.5 Approximate Entropy (ApEn)

The ApEn comparison results for 1D-MWSLC, Sin map, and Logistic map are shown in Table 1. The ApEn analysis results show that under the same parameter conditions, the ApEn of 1D-MWSLC has a more stable value and a larger value than the ApEn of the Sin map and Logistic map. This indicates that 1D-MWSLC has a stable performance of generating many random numbers.

Table 1: Approximate entropy

α	β	Sin map	Logistic map	1D-MWSLC
0.5	1	0.6007	-	**1.2281**
2	1	0.6007	0.0000007	**1.2259**
3.9	1	0.6007	0.4578	**1.2442**
3.9	1.2	0.6693	0.4578	**1.2347**
3.9	2	0.9760	0.4578	**1.2349**
3.7	10	**1.2677**	0.3603	1.2319
3.7	19	1.2128	0.3603	**1.2315**
10	9.8	1.1912	-	**1.2260**
20	19	1.2128	-	**1.2300**

3 Application to 3D Symmetric Encryption of 1D-MWSLC

Because 1D-MWSLC exhibits excellent performance and can generate many acyclic key streams, it is very suitable for cryptography. Therefore, this section discusses the application of 1D-MWSLC in 3D model encryption. The encryption process is described below. The encryption process diagram is shown in Fig. 7.

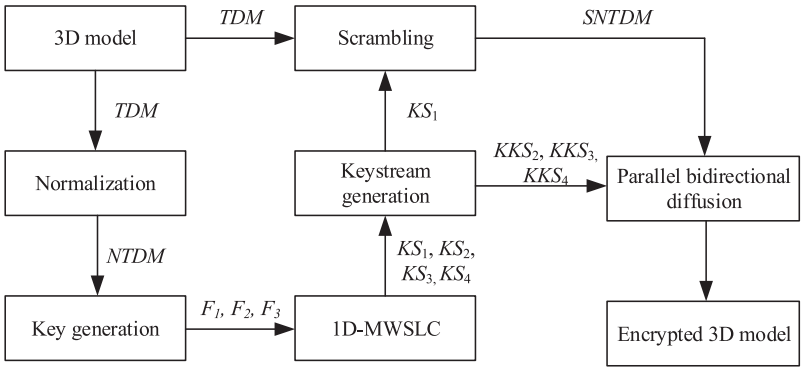


Figure 7: The encryption process diagram

3.1 Normalization and Key Generation

Unlike digital images, the data type of a 3D model contains arbitrary floating point numbers. Therefore, in the designed encryption algorithm, normalization processing is first performed to align the data type of the 3D model with the data type of the digital image.

TDM represents 3D model, $TDM \in (-\infty, +\infty)$, and the size of TDM is $L \times 3$. The normalization process is as follows,

$$NTDM = 255 \times \frac{TDM - \min TDM}{\max TDM - \min TDM}. \tag{4}$$

Now $NTDM \in [0, 255]$.

The secret key of the cryptosystem is generated by the new 3D model ($NTDM$), which are

$$\begin{cases} KM_1 = \sum_{d=1}^{L} NTDM\,(d,1) \times 10^{-10} \\ f_1(c+1) = 3.99997 \times f_1(c) \times (1 - f_1(c)) \\ c = 1, 2, 3, \ldots., 100, f_1(1) = KM_1 \end{cases}, \quad \begin{cases} KM_2 = \sum_{d=1}^{L} NTDM\,(d,2) \times 10^{-10} \\ f_2(c+1) = 3.99998 \times f_2(c) \times (1 - f_2(c)) \\ c = 1, 2, 3, \ldots., 100, f_2(1) = KM_2 \end{cases}. \tag{5}$$

$$\begin{cases} KM_3 = \sum_{d=1}^{L} NTDM\,(d,3) \times 10^{-10} \\ f_3(c+1) = 3.99999 \times f_3(c) \times (1 - f_3(c)) \\ c = 1, 2, 3, \ldots., 100, f_3(1) = KM_3 \end{cases}, \quad \begin{cases} F_1 = f_1(100) \\ F_2 = f_2 \times 100 + 1 \\ F_3 = f_3 \times 100 + 1 \end{cases}. \tag{6}$$

F_1, F_2, F_3 are the secret keys of the cryptographic system. During the stream cipher generation phase, F_1, F_2, F_3 are the initial values and parameters of 1D-MWSLC.

3.2 Generating the Keystream of the Cryptographic System

Generate keystreams based on the secret key (F_1, F_2, F_3) by 1D-MWSLC, which are

$$\begin{cases} KS_1 \colon X_{c+1} = \beta \sin(e^{5\pi}\alpha X_c(1 - X_c)) \\ \beta = F_1, \alpha = F_2, X_1 = F_3 \end{cases}, \begin{cases} KS_2 \colon X_{c+1} = \beta \sin(e^{5\pi}\alpha X_c(1 - X_c)) \\ \beta = F_1, \alpha = F_3, X_1 = F_2 \end{cases}. \tag{7}$$

$$\begin{cases} KS_3 \colon X_{c+1} = \beta \sin(e^{5\pi}\alpha X_c(1 - X_c)) \\ \beta = F_1, \alpha = F_2 + F_3, X_1 = F_3 \end{cases}, \begin{cases} KS_4 \colon X_{c+1} = \beta \sin(e^{5\pi}\alpha X_c(1 - X_c)) \\ \beta = F_1, \alpha = F_2, X_1 = F_3 + F_2 \end{cases}. \tag{8}$$

When 1D-MWSLC is sufficiently chaotic, the key stream is extracted,

$$\begin{cases} KS_1 = KS_1(floor(F_1 \times 50) + 200 : L \times 3 - 1) \\ KS_2 = KS_2(floor(F_2 \times 10) + 200 : L - 1), KKS_2 = floor(KS_2 \times 10^{10}) \bmod 256 \\ KS_3 = KS_3(floor(F_3 \times 10) + 200 : L - 1), KKS_3 = floor(KS_3 \times 10^{10}) \bmod 256 \\ KS_4 = KS_4(floor(F_1 \times 40) + 200 : L - 1), KKS_4 = floor(KS_4 \times 10^{10}) \bmod 256 \end{cases}. \tag{9}$$

3.3 Scrambling and Parallel Bidirectional Diffusion

In the scrambling phase, both the integer and fractional parts of the 3D model undergo simultaneous scrambling. In the diffusion phase, this paper suggest a bidirectional diffusion strategy that operates in parallel. All three coordinates of the 3D model are diffused simultaneously, starting from the center position of each coordinate axis. The diffusion operations are exclusively performed on the integer portions of the 3D model.

Input: $NTMD, KS_1, KS_2, KS_3, KS_4$

Output: $CTDM$

Step1: Convert $NTMD$ $(L \times 3)$ to $NTMD$ $(1 \times 3L)$ by

$$NTDM = reshape(NTDM, 1, L \times 3). \tag{10}$$

Step2: Generate Index Matrix LP,

$$LP = sort(KS_1). \tag{11}$$

where, KS_1 is sorted from small to large and the index of its original matrix is found, recorded as LP.

Step3: The scrambling steps is,

$$SNTDM(c) = NTDM(LP(c)), c = 1, 2, 3, \ldots., L \times 3. \tag{12}$$

Step4: Convert $SNTMD$ $(1 \times 3L)$ to $SNTMD$ $(L \times 3)$ by

$$SNTDMA = reshape(SNTDM, L, 3). \tag{13}$$

Step5: During the diffusion process, only integer operations are performed on 3D models, and integer extraction of 3D models is performed through,

$$\begin{cases} SNTDMA_1 = floor(SNTDMA) \\ SNTDMA_2 = SNTDMA - SNTDMA_1 \end{cases}. \tag{14}$$

Step 6: In the parallel bidirectional diffusion stage, the starting point of bidirectional diffusion is PAW, where

$$PAW = floor(L/2). \tag{15}$$

Step 7: The specific steps for parallel bidirectional diffusion are as follows. Note that these three steps are performed simultaneously,

$$\begin{cases} CAP(PAW) = SNTDMA_1(PAW,1) + KKS2(PAW)\text{mod}256 \\ CAP(r) = SNTDMA_1(r,1) + KKS2(r) + CAP(r-1)\text{mod}256,\ r = PAW+1 : PA \\ CAP(r) = SNTDMA_1(r,1) + KKS2(r) + CAP(r+1) + CAP(r+2)\text{mod}256, r = PAW-1 : -1 : 1 \end{cases}. \tag{16}$$

$$\begin{cases} CBP(PAW) = SNTDMA_1(PAW,2) + KKS3(PAW)\text{mod}256 \\ CBP(g) = SNTDMA_1(g,2) + KKS3(g) + CBP(g-1)\text{mod}256), g = PAW+1 : PA \\ CBP(g) = SNTDMA_1(g,2) + KKS3(g) + CBP(g+1) + CBP(g+2)\text{mod}256,\ g = PAW-1 : -1 : 1 \end{cases}. \tag{17}$$

$$\begin{cases} CBP(PAW) = SNTDMA_1(PAW,3) + KKS3(PAW)\text{mod}256 \\ CBP(b) = SNTDMA_1(b,3) + KKS4(b) + CCP(b-1)\text{mod}256), b = PAW+1 : PA \\ CBP(b) = SNTDMA_1(b,3) + KKS4(b) + CCP(b+1) + CCP(b+2)\text{mod}256,\ b = PAW-1 : -1 : 1 \end{cases}. \tag{18}$$

Step 8: The ciphertext CTMD of the 3D model is

$$\begin{cases} CTDM = [CAP; CBP; CCP] \\ CTDM = CTDM + SNTDMA_2 \end{cases}. \tag{19}$$

3.4 Decryption Algorithm

The proposed 3D model encryption algorithm is a symmetric encryption algorithm. Therefore, the decryption process is the inverse of the encryption process. More specifically,

Input: $CTDM$, $KS_1, KKS_2, KKS_3, KKS_4$.

Output: $NTMD$

Step1: Decompose $CTDM$ into CAP, CBP, CCP and $SNTDMA_2$.

Step 2: The inverse process of diffusion is

$$\begin{cases} SNTDMA_1(PAW,1) = CAP(PAW) - KKS2(PAW)\text{mod}256 \\ SNTDMA_1(r,1) = CAP(r) - KKS2(r) - CAP(r-1)\text{mod}256,\ r = PAW+1 : PA \\ SNTDMA_1(r,1) = CAP(r) - KKS2(r) - CAP(r+1) - CAP(r+2)\text{mod}256,\ r = PAW-1 : -1 : 1 \end{cases}. \tag{20}$$

$$\begin{cases} SNTDMA_1(PAW,2) = CBP(PAW) - KKS3(PAW)\text{mod}256 \\ SNTDMA_1(g,2) = CBP(g) - KKS3(g) + CBP(g-1)\text{mod}256), g = PAW+1 : PA \\ SNTDMA_1(g,2) = CBP(g) - KKS3(g) + CBP(g+1) + CBP(g+2)\text{mod}256,\ g = PAW-1 : -1 : 1 \end{cases} . \tag{21}$$

$$\begin{cases} SNTDMA_1(PAW,3) = CBP(PAW) - KKS3(PAW)\text{mod}256 \\ SNTDMA_1(b,3) = CBP(b) - KKS4(b) - CCP(b-1)\text{mod}256), b = PAW+1 : PA \\ SNTDMA_1(b,3) = CBP(b) - KKS4(b) - CCP(b+1) - CCP(b+2)\text{mod}256,\ b = PAW-1 : -1 : 1 \end{cases} \tag{22}$$

Step 3: The reverse process of scrambling is

$NTDM(LP(c)) = SNTDM(c), c = 1, 2, 3, \ldots., L \times 3.$

where $SNTDM = SNTDMA_1 + SNTDMA_2$, and LP is generated by KS_1.

3.5 An Example of Encryption

Due to its two control parameters, the 1D-MWSLC offers a larger parameter space for the cryptographic system. The algorithm generates a key stream for both confusion and diffusion. The 3D model is regularized ($TDM \rightarrow NTMD$) and mapped to $[0, 255]$. In the confusing stage, random confusion is applied. The key stream generates an index matrix that confuses both the integer and decimal parts of the 3D model, as illustrated in Fig. 8. In the diffusion stage, parallel bidirectional diffusion is employed simultaneously to diffuse the integer parts of the three coordinates of the 3D model, as depicted in Fig. 9.

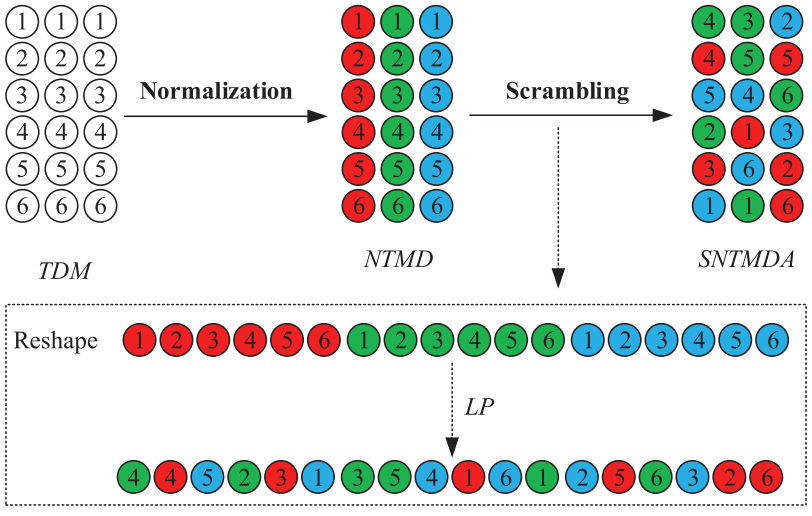


Figure 8: Normalization and scrambling

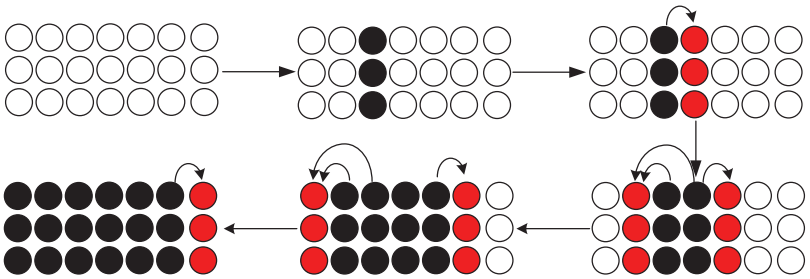
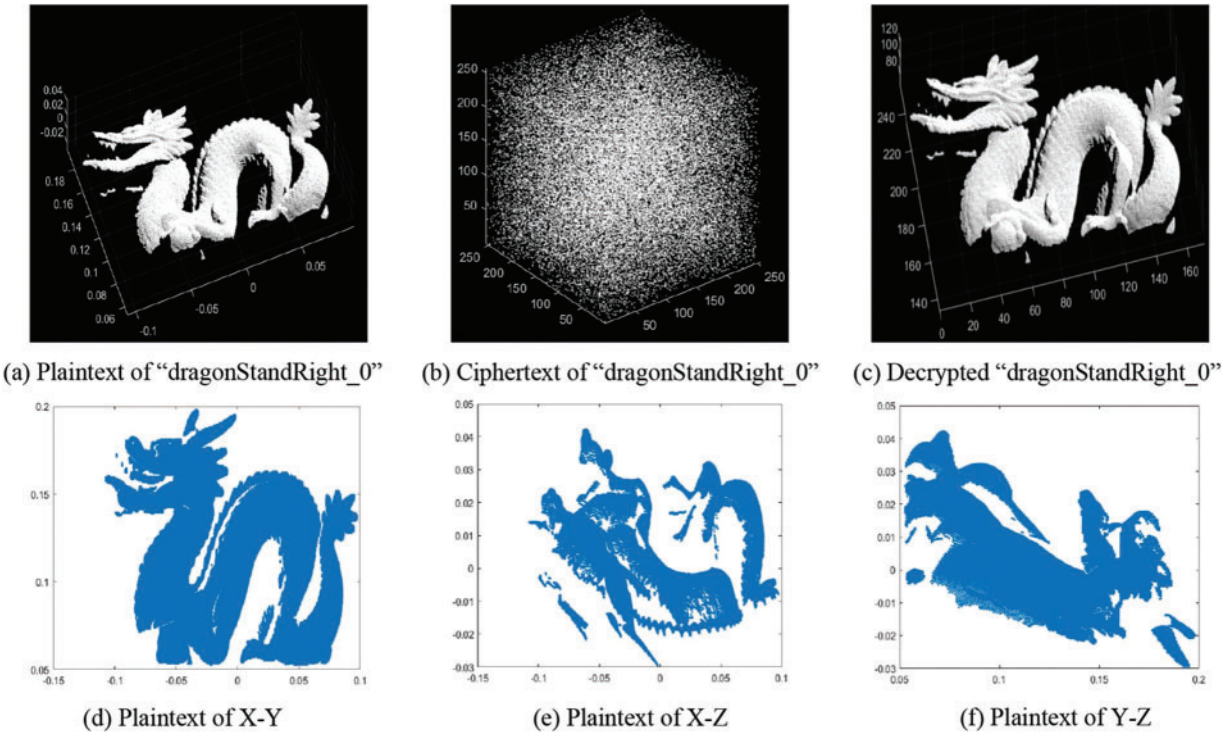


Figure 9: Parallel bidirectional diffusion

4 Performance Analysis

4.1 Visualization

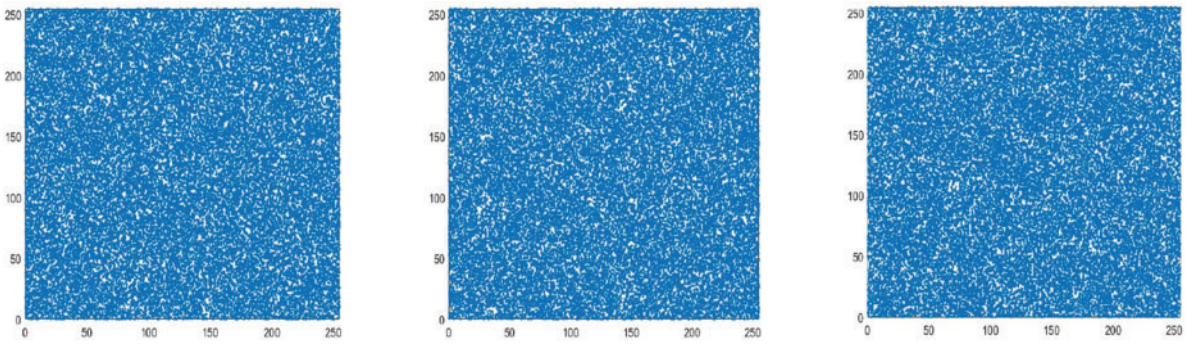
In order to assess the security and viability of the suggested algorithm, an algorithmic test was performed on a 3D model obtained from the Stanford 3D scanning repository database (https://graphics.stanford.edu/data/3Dscanrep/). The evaluation findings, presented in Fig. 10, demonstrate that the plaintext information in the encrypted image cannot be identified. This validates that the proposed encryption technique is highly effective in preventing the leakage of plaintext information content.



(a) Plaintext of "dragonStandRight_0" (b) Ciphertext of "dragonStandRight_0" (c) Decrypted "dragonStandRight_0"

(d) Plaintext of X-Y (e) Plaintext of X-Z (f) Plaintext of Y-Z

Figure 10: (Continued)



(g) Ciphertext of X-Y (h) Ciphertext of X-Z (i) Ciphertext of Y-Z

Figure 10: Visualization of 3D model for dragonStandRight_0 (41841 × 3)

4.2 Key Space and Key Sensitivity Analysis

The secret key of the algorithm includes $F_1(F_1 \in (0,1))$, $F_2(F_2 \in (0,100))$, and $F_3(F_3 \in (0,100))$. If the calculation accuracy of the computer is 10^{-15}. The key space of the algorithm is

$$KeySpace = 10^{15} \times 10^{17} \times 10^{17} = 10^{47} \approx 2^{156}.$$

When the secret key space of the algorithm is greater than 2^{100}, it is considered that the algorithm can resist violent attacks. Therefore, the proposed algorithm has good resistance to violent attacks.

For a secure cryptographic system, the more sensitive the encryption algorithm is to the secret key, the more difficult it is to decipher it with various opportunities for plaintext analysis. In dragonStandRight_0, the initial key is

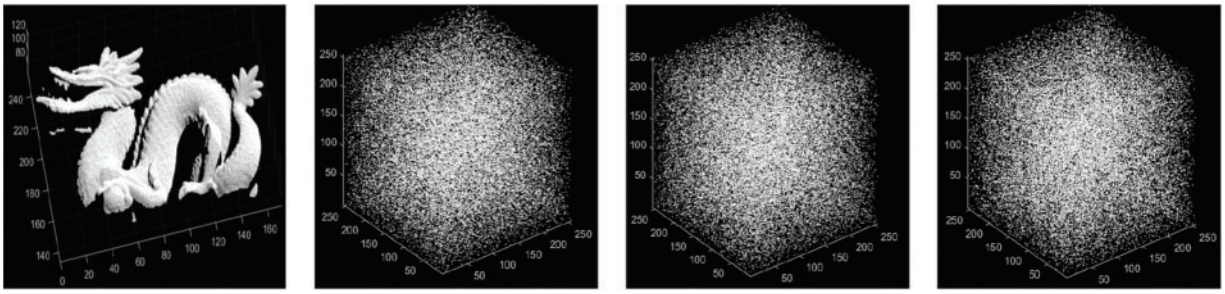
$$KY_1 = \begin{cases} F_1 = 0.7629504319360213 \\ F_2 = 0.3561951297073947 \\ F_3 = 0.5143438570224240 \end{cases}.$$

Use the new secret key to decrypt the encryption algorithm, which are

$$KY_2 = \begin{cases} F_1 = 0.7629504319360213 + 10^{-15} \\ F_2 = 0.3561951297073947 \\ F_3 = 0.5143438570224240 \end{cases}, KY_3 = \begin{cases} F_1 = 0.7629504319360213 \\ F_2 = 0.3561951297073947 + 10^{-15} \\ F_3 = 0.5143438570224240 \end{cases},$$

$$KY_4 = \begin{cases} F_1 = 0.7629504319360213 \\ F_2 = 0.3561951297073947 \\ F_3 = 0.5143438570224240 + 10^{-15} \end{cases}.$$

The decryption results are shown in Fig. 11. The secret key analysis shows that the proposed encryption algorithm is sensitive to the secret key.



(a) Decrypted by KY_1 (b) Decrypted by KY_2 (c) Decrypted by KY_3 (d) Decrypted by KY_4

Figure 11: Key sensitivity analysis

4.3 Information Entropy Analysis

Information entropy can indicate the level of pseudorandomness in an image. A higher information entropy value indicates good pseudorandomness. Table 2 presents the analysis results of information entropy. Furthermore, Table 3 compares the results of our proposed algorithm with algorithms in Ref. [48] and Ref. [49].

Table 2: Information entropy analysis

3D models	Plaintext	Ciphertext
bun_zipper	7.72174	7.99856
dragonStandRight_0	7.59763	7.99884
drill_1.6 mm_0_cyb	6.78711	7.99301
xyzrgb_manuscript	6.61928	7.99998
xyzrgb_dragon	7.46354	7.99998
#Average	7.18144	7.99807

Table 3: Information entropy comparison

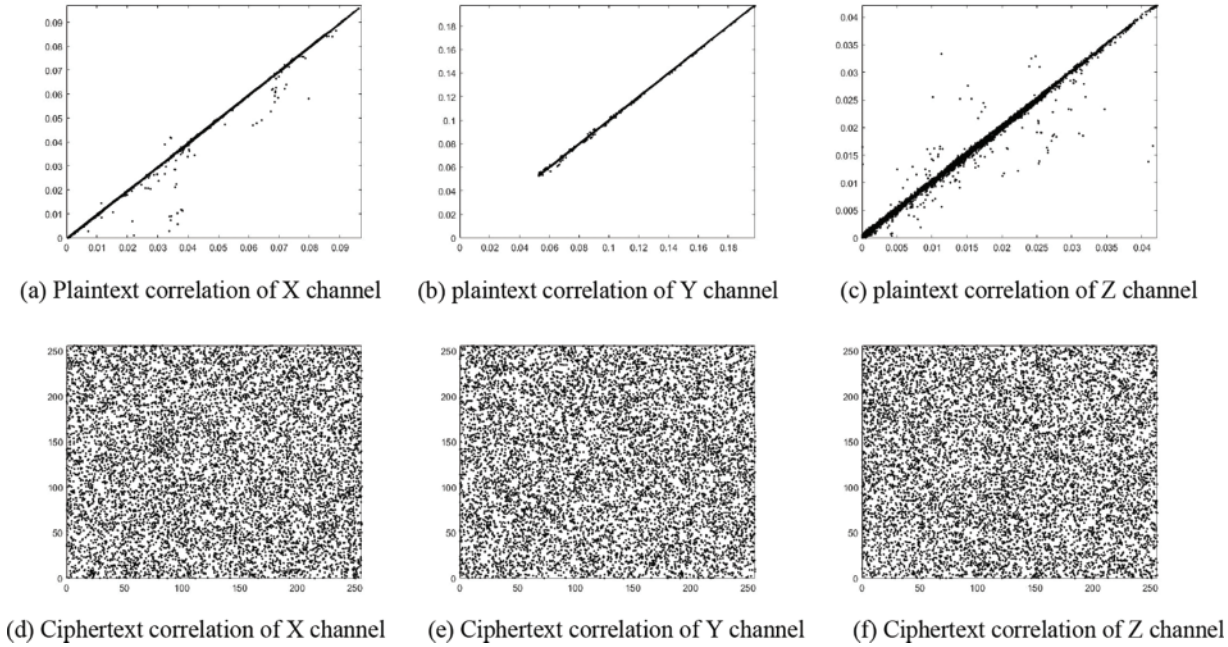
Algorithms	Proposed	Ref. [48]	Ref. [49]
Information entropy	7.9981	7.9980	7.9959

The comparison shows that our proposed algorithm achieves similar information entropy as the theoretical value, and the information entropy is closer to the theoretical value than algorithms in Ref. [48] and in Ref. [49]. The coordinate distribution in the generated ciphertext image demonstrates good randomness.

4.4 Correlation Analysis

In the original 3D model, adjacent coordinates have a strong correlation, meaning accessing a small amount of plaintext information can reveal the entire message. Fig. 12 shows a correlation analysis of plaintext and ciphertext images (dragonStandRight_0). Obtaining a correlation image of

the ciphertext shows a scattered pattern, indicating that the correlation between adjacent pixel values is minimal.



(a) Plaintext correlation of X channel (b) plaintext correlation of Y channel (c) plaintext correlation of Z channel

(d) Ciphertext correlation of X channel (e) Ciphertext correlation of Y channel (f) Ciphertext correlation of Z channel

Figure 12: Correlation analysis

Reducing complexity and enhancing security: Table 4 presents the correlation analysis results obtained using the proposed algorithm. The analysis indicates that the pixel correlation in the ciphertext is weak, with a correlation coefficient close to zero. Therefore, it is unlikely for an attacker to extract meaningful information from the ciphertext pixel values.

Table 4: Correlation coefficients

3D models	Plaintext			Ciphertext		
	X	*Y*	Z	X	*Y*	Z
bun_zipper	0.69663	0.79876	0.76797	0.01002	0.00224	-0.01078
dragonStandRight_0	0.97925	0.99998	0.98294	0.00186	0.01132	0.00165
drill_1.6 mm_0_cyb	0.92989	0.99998	0.97733	-0.00543	0.00784	-0.02057
xyzrgb_manuscript	0.99999	0.99931	0.99709	-0.00094	0.00020	-0.00103
xyzrgb_dragon	0.99865	0.99773	0.99907	-0.00045	-0.00009	-0.00005
#Average	0.92088	0.95915	0.94488	0.00101	0.00430	-0.00615

In addition, Table 5 provides a comparison between the correlation results obtained using the proposed algorithm and those from other algorithms, such as those reported in the cited references [48,49]. The comparison reveals that the proposed algorithm produces lower correlation values, indicating that it offers good security.

Table 5: Correlation coefficients comparison

Algorithms	Proposed	Ref. [48]	Ref. [49]
X-direction	0.0010	−0.0055	−0.0254
Y-direction	0.0043	0.0081	−0.0097
Z-direction	–0.0061	0.0115	0.0049

4.5 NIST Statistical Test Suite

The NIST Statistical Test Suite includes various tests, such as the Frequency (Monobit) Test, Block Frequency Test, Runs Test, Longest Run of Ones in a Block Test, and Random Excursions Test. These tests can be applied to both plaintexts and ciphertexts to evaluate the strength and security of a cryptographic algorithm. Table 6 presents the NIST results for plaintexts and ciphertexts. All of the tests resulted in a value of 0 for the plaintexts, indicating that they failed to meet the statistical randomness requirements of the NIST tests. However, the ciphertexts passed all of the tests, indicating that the proposed algorithm is highly resistant to statistical attacks on the ciphertext, which is a desirable property for a secure encryption algorithm.

Table 6: NIST For plaintexts and ciphertexts

Statistical test	Plaintexts	Pass (Y/N)	Ciphertexts	Pass (Y/N)
Non-overlapping template matching	0	N	0.949	Y
Serial	0	N	0.253	Y
Block frequency	0	N	0.739	Y
Frequency	0	N	0.299	Y
Approximate entropy	0	N	0.468	Y
Longest run of ones	0	N	0.148	Y
Cumulative sums	0	N	0.671	Y
Random excursions	0	N	0.350	Y
Random excursions variant	0	N	0.350	Y
Linear complexity	0	N	0.066	Y
Overlapping template matching	0	N	0.739	Y
Spectral	0	N	0.407	Y
Runs	0	N	0.534	Y
Universal	0	N	0.739	Y
Rank	0	N	0.671	Y

4.6 Histogram Analysis

Histogram analysis provides a deeper understanding of the central tendency, variability, and skewness of data. The histogram analysis is shown in Fig. 13. The uniform distribution of the ciphertext's histogram indicates that the encryption algorithm's ciphertext has a high degree of

randomness and unpredictability. This can enhance the security of the encryption algorithm because an attacker cannot gain any information about the plaintext or key by analyzing the histogram of the ciphertext.

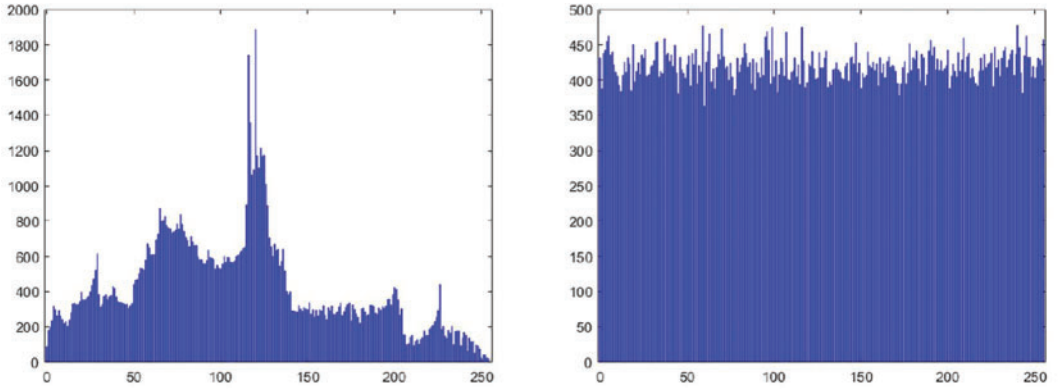


Figure 13: Histogram analysis

The chi squared test of histograms is a statistical method used to evaluate the difference between observed and expected values. The chi squared test is shown in Table 7.

Table 7: Chi square test

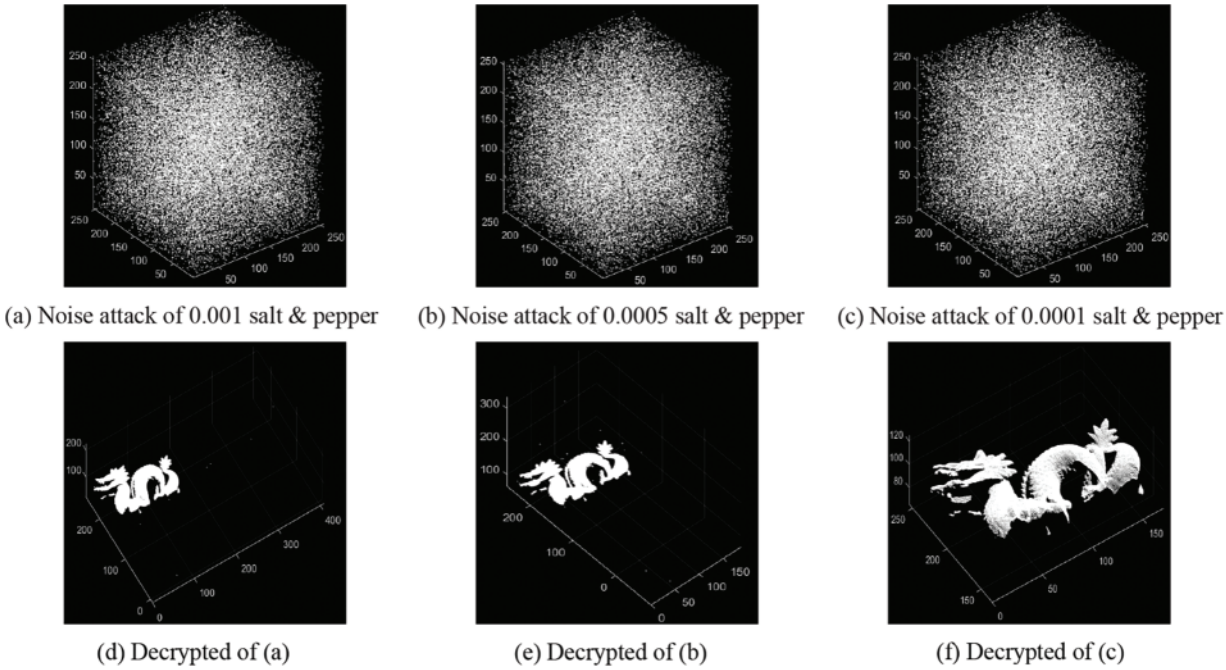
3D models	Plaintext	Ciphertext
bun_zipper	46108.86	254.14
dragonStandRight_0	80485.98	219.78
drill_1.6 mm_0_cyb	25647.97	226.80
xyzrgb_manuscript	64901635.34	239.66
xyzrgb_dragon	7517109.52	244.30

The table shows the chi-square test results performed on plaintext and ciphertext data for different 3D models. The values in the table represent the chi-square statistic for each model. For all models, the chi-square statistic for the plaintext is much higher than that of the ciphertext, indicating that the distribution of the plaintext data is not uniform. Overall, the results suggest that the encryption algorithm used in the study produces ciphertext with a more uniform distribution than plaintext, which is desirable for ensuring the security of the data.

4.7 Robustness Analysis

Robustness analysis can help determine the applicability of encryption algorithms in practical applications, as encrypted images may be subject to various unexpected and intentional attacks in practical scenarios. By subjecting the encrypted image to various attacks, such as noise attacks, the robustness of the encryption algorithm against these attacks can be evaluated. This paper evaluates the robustness of the algorithm using dragonStandRight_0, as shown in Fig. 14. If the encryption algorithm can still protect the confidentiality of the image in the face of various attacks, it can be considered to have good robustness. The results of the robustness analysis indicate that the tested

encryption algorithm can protect the confidentiality of the image in the face of attacks and has strong security.



(a) Noise attack of 0.001 salt & pepper (b) Noise attack of 0.0005 salt & pepper (c) Noise attack of 0.0001 salt & pepper

(d) Decrypted of (a) (e) Decrypted of (b) (f) Decrypted of (c)

Figure 14: Robustness analysis

4.8 Differential Attack Analysis

Differential attacks are one of the most common methods used to break encryption algorithms. In image encryption, a differential attack involves identifying the changes in the pixel values between two similar images. The differential attack then uses this information to derive the encryption key or recover the plaintext. As suggested by Wu in Ref. [50], if the NPCR value is greater than 99.5893% and the UACI value falls between 33.3730% and 33.5541%, then the algorithm is considered resistant to differential attacks. The differential attack analysis is shown in Table 8.

Table 8: Differential attack analysis

3D models	NPCR (%)	Pass (Y/N)	UACI (%)	Pass (Y/N)
bun_zipper	99.60	Y	33.46	Y
dragonStandRight_0	99.59	Y	33.51	Y
drill_1.6 mm_0_cyb	99.60	Y	33.41	Y
xyzrgb_manuscript	99.61	Y	33.45	Y
xyzrgb_dragon	99.60	Y	33.48	Y

The analysis of differential attacks shows that the NPCR values of all models are above 99.59%, while the UACI values are between 33.41% and 33.54%, indicating that these algorithms can resist

differential attacks. Therefore, all models have passed the test, demonstrating the algorithm's excellent ability to resist differential attacks.

5 Conclusion

In recent years, significant progress has been made in digital image encryption, resulting in the development of new, more secure, and efficient algorithms. However, there is still a pressing need for research on 3D model encryption, as existing digital image encryption algorithms are not directly applicable to 3D models. This paper proposes a new 3D model encryption algorithm based on a 1D map with sin and logistic coupling (1D-MWSLC). The 1D-MWSLC algorithm has a broader range of parameters than traditional 1D chaotic systems, and its chaotic phenomenon remains robust even when the parameter exceeds a specific range.

Furthermore, the 1D-MWSLC algorithm has two control parameters, resulting in a larger parameter space for the cryptosystem. The algorithm generates keystreams using 1D-MWSLC for both confusion and diffusion. During the confusion stage, random confusion is applied, and the keystream generates an index matrix that simultaneously confuses the integer and decimal parts of the 3D model. In the diffusion stage, parallel bidirectional diffusion is used to diffuse the integer parts of the three coordinates of the 3D model. The proposed algorithm has been verified through statistical analysis, with experimental results demonstrating its robust security. Overall, this new 3D model encryption algorithm based on 1D-MWSLC represents a significant advancement in 3D model encryption.

Despite the proposed algorithm's significant advancement in the field of 3D model encryption, there is still scope for further research. Future work could investigate the algorithm's performance on different types of 3D models, including larger and more complex models, and compare its performance with other existing encryption algorithms. Additionally, it would be useful to explore the algorithm's applicability to color and binary images as a dataset in other domains.

Funding Statement: Funds for New Generation Information Technology of the Industry-University-Research Innovation Foundation of China University (No. 2020ITA03022).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. Guan, X. Yang and W. Hu, "Chaotic image encryption algorithm using frequency-domain DNA encoding," *IET Image Processing*, vol. 13, no. 9, pp. 1535–1539, 2019.

[2] T. M. Hoang, "A novel design of multiple image encryption using perturbed chaotic map," *Multimedia Tools and Applications*, vol. 81, no. 18, pp. 26535–26589, 2022.

[3] P. Singh, "A novel chaotic umbrella map and its application to image encryption," *Optical and Quantum Electronics*, vol. 54, no. 5, pp. 266, 2022.

[4] P. Singh, "Asymmetric cryptosystem based on biological mutation operation in chirp-Z domain," *Multimedia Tools and Applications*, 2023. https://doi.org/10.1007/s11042-023-15190-7

[5] Z. Hua, Y. Zhou and H. Huang, "Cosine-transform-based chaotic system for image encryption," *Information Sciences*, vol. 480, pp. 403–419, 2019.

[6] X. Wang, S. Gao, L. Yu, Y. Sun and H. Sun, "Chaotic image encryption algorithm based on bit-combination scrambling in decimal system and dynamic diffusion," *IEEE Access*, vol. 7, pp. 103662–103677, 2019.

[7] R. Anitha and B. Vijayalakshmi, "Image encryption using multi-scroll attractor and chaotic logistic map," *CMC-Computers, Materials and Continua*, vol. 72, no. 2, pp. 3447–3463, 2022.

[8] R. Kumar and P. Singh, "Modified plaintext attacks in a session for an optical cryptosystem based on DRPE with PFS," *Applied Optics*, vol. 61, no. 2, pp. 623–628, 2022.

[9] M. Lawnik, L. Moysis and C. Volos, "Chaos-based cryptography: Text encryption using image algorithms," *Electronics*, vol. 11, no. 19, pp. 3156, 2022.

[10] H. Lin, C. Wang, L. Cui, Y. Sun, X. Zhang *et al.,* "Hyperchaotic memristive ring neural network and application in medical image encryption," *Nonlinear Dynamics*, vol. 110, no. 1, pp. 841–855, 2022.

[11] H. Xiang and L. Liu, "A random irregular blocking image encryption algorithm based on improved digital chaotic maps at bit level," *International Journal of Bifurcation and Chaos*, vol. 32, no. 4, pp. 2250054, 2022.

[12] X. Wang and S. Gao, "Application of matrix semi-tensor product in chaotic image encryption," *Journal of the Franklin Institute*, vol. 356, no. 18, pp. 11638–11667, 2019.

[13] X. Wang, S. Gao, X. Ye, S. Zhou and M. Wang, "A new image encryption algorithm with cantor diagonal scrambling based on the PUMCML system," *International Journal of Bifurcation and Chaos*, vol. 31, no. 1, pp. 2150003, 2021.

[14] U. Erkan, A. Toktas and Q. Lai, "2D hyperchaotic system based on schaffer function for image encryption," *Expert Systems with Applications*, vol. 213, pp. 119076, 2023.

[15] L. Ding and Q. Ding, "A novel image encryption scheme based on 2D fractional chaotic map, DWT and 4D hyper-chaos," *Electronics*, vol. 9, no. 8, pp. 1280, 2020.

[16] Y. Zhang, Y. He, J. Zhang and X. Liu, "Multiple digital image encryption algorithm based on chaos algorithm," *Mobile Networks and Applications*, vol. 27, no. 4, pp. 1349–1358, 2022.

[17] H. Lin, C. Wang, J. Sun, X. Zhang, Y. Sun *et al.,* "Memristor-coupled asymmetric neural networks: Bionic modeling, chaotic dynamics analysis and encryption application," *Chaos, Solitons and Fractals*, vol. 166, pp. 112905, 2023.

[18] A. Manzoor, A. H. Zahid and M. T. Hassan, "A new dynamic substitution box for data security using an innovative chaotic map," *IEEE Access*, vol. 10, pp. 74164–74174, 2022.

[19] X. Zhang and L. Zhang, "Multiple-image encryption algorithm based on chaos and gene fusion," *Multimedia Tools and Applications*, vol. 81, no. 14, pp. 20021–20042, 2022.

[20] K. Jain, A. Aji and P. Krishnan, "Medical image encryption scheme using multiple chaotic maps," *Pattern Recognition Letters*, vol. 152, pp. 356–364, 2021.

[21] G. Atali and E. Sonmez, "Efficient chaos-based image encryption approach for secure communication," *Journal of Electronic Imaging*, vol. 30, no. 2, pp. 023026, 2021.

[22] Z. Wu, P. Pan, C. Sun and B. Zhao, "Plaintext-related dynamic key chaotic image encryption algorithm," *Entropy*, vol. 23, no. 9, pp. 1159, 2021.

[23] H. Z. Amjad, J. A. Muhammad, A. Musheer, F. S. Naglaa and E. Walid, "Dynamic S-box generation using novel chaotic map with nonlinearity tweaking," *Computers, Materials & Continua*, vol. 75, no. 2, pp. 3011–3026, 2023.

[24] W. Song, C. Fu, M. Tie, C. W. Sham, J. Liu *et al.,* "A fast parallel batch image encryption algorithm using intrinsic properties of chaos," *Signal Processing: Image Communication*, vol. 102, pp. 116628, 2022.

[25] Y. Pourasad, R. Ranjbarzadeh and A. Mardani, "A new algorithm for digital image encryption based on chaos theory," *Entropy*, vol. 23, no. 3, pp. 341, 2021.

[26] X. Li, J. Mou, S. Banerjee, Z. Wang and Y. Cao, "Design and DSP implementation of a fractional-order detuned laser hyperchaotic circuit with applications in image encryption," *Chaos, Solitons and Fractals*, vol. 159, pp. 112133, 2022.

[27] Z. Huang and N. Zhou, "Image encryption scheme based on discrete cosine stockwell transform and DNA-level modulus diffusion," *Optics and Laser Technology*, vol. 149, pp. 107879, 2022.

[28] X. Gao, J. Mou, L. Xiong, Y. Sha, H. Yan *et al.,* "A fast and efficient multiple images encryption based on single-channel encryption and chaotic system," *Nonlinear Dynamics*, vol. 108, no. 1, pp. 613–636, 2022.

[29] M. Alawida, A. Samsudin, J. S. Teh and R. S. Alkhawaldeh, "A new hybrid digital chaotic system with applications in image encryption," *Signal Processing*, vol. 160, pp. 45–58, 2019.

[30] G. Ye, M. Liu and M. Wu, "Double image encryption algorithm based on compressive sensing and elliptic curve," *Alexandria Engineering Journal*, vol. 61, no. 9, pp. 6785–6795, 2022.

[31] S. Liu, C. Li and Q. Hu, "Cryptanalyzing two image encryption algorithms based on a first-order time-delay system," *IEEE Multimedia*, vol. 29, no. 1, pp. 74–84, 2021.

[32] S. Zhu, C. Zhu and H. Yan, "Cryptanalyzing and improving an image encryption algorithm based on chaotic dual scrambling of pixel position and bit," *Entropy*, vol. 25, no. 3, pp. 400, 2023.

[33] X. Wang and S. Gao, "A chaotic image encryption algorithm based on a counting system and the semi-tensor product," *Multimedia Tools and Applications*, vol. 80, no. 7, pp. 10301–10322, 2021.

[34] S. Gao, R. Wu, X. Wang, J. Liu, Q. Li *et al.,* "Encryption for face recognition based on the chaos and semi-tensor product theory," *Information Sciences*, vol. 621, pp. 766–781, 2023.

[35] H. Lin, C. Wang, L. Cui, Y. Sun, C. Xu *et al.,* "Brain-like initial-boosted hyperchaos and application in biomedical image encryption," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 8839–8850, 2022.

[36] S. Mansoor and S. A. Parah, "HAIE: A hybrid adaptive image encryption algorithm using chaos and DNA computing," *Multimedia Tools and Applications*, 2023. https://doi.org/10.1007/s11042-023-14542-7

[37] N. Pratyusha and S. Mandal, "Design and implementation of a novel circuit-based memristive non-autonomous hyperchaotic system with conservative and offset boosting for applications to image encryption," *Circuits, Systems, and Signal Processing*, 2023. https://doi.org/10.1007/s00034-023-02322-5

[38] A. Kanso, M. Ghebleh and M. Bou Khuzam, "A probabilistic chaotic image encryption scheme," *Mathematics*, vol. 10, no. 11, pp. 1910, 2022.

[39] W. Dong, Q. Li, Y. Tang, M. Hu and R. Zeng, "A robust and multi chaotic DNA image encryption with pixel-value pseudorandom substitution scheme," *Optics Communications*, vol. 499, pp. 127211, 2021.

[40] Z. Hua, Z. Zhu, Y. Chen and Y. Li, "Color image encryption using orthogonal Latin squares and a new 2D chaotic system," *Nonlinear Dynamics*, vol. 104, pp. 4505–4522, 2021.

[41] L. Liu and J. Wang, "A cluster of 1D quadratic chaotic map and its applications in image encryption," *Mathematics and Computers in Simulation*, vol. 204, pp. 89–114, 2023.

[42] J. Xu and B. Zhao, "Designing an image encryption algorithm based on hyperchaotic system and DCT," *International Journal of Bifurcation and Chaos*, vol. 33, no. 2, pp. 2350021, 2023.

[43] B. Liu, Y. Liu, Y. Xie, X. Jiang, Y. Ye *et al.,* "Privacy protection for 3D point cloud classification based on an optical chaotic encryption scheme," *Optics Express*, vol. 31, no. 5, pp. 8820–8843, 2023.

[44] K. Priyadarsini, A. K. Sivaraman, A. Q. Md and A. Malibari, "Securing 3D point and mesh fog data using novel chaotic cat map," *Computers, Materials & Continua*, vol. 74, no. 3, pp. 6703–6717, 2023.

[45] P. Liu, S. Zhou and W. Q. Yan, "A 3D cuboid image encryption algorithm based on controlled alternate quantum walk of message coding," *Mathematics*, vol. 10, no. 23, pp. 4441, 2022.

[46] S. Gao, R. Wu, X. Wang, J. Wang, Q. Li *et al.,* "A 3D model encryption scheme based on a cascaded chaotic system," *Signal Processing*, vol. 202, pp. 108745, 2023.

[47] Z. Zhang, J. Tang, F. Zhang, H. Ni, J. Chen *et al.,* "Color image encryption using 2D sine-cosine coupling map," *IEEE Access*, vol. 10, pp. 67669–67685, 2022.

[48] X. Wang, M. Xu and Y. Li, "Fast encryption scheme for 3D models based on chaos system," *Multimedia Tools and Applications*, vol. 78, pp. 33865–33884, 2019.

[49] A. B. Joshi, D. Kumar, A. Gaffar and D. C. Mishra, "Triple color image encryption based on 2D multiple parameter fractional discrete Fourier transform and 3D Arnold transform," *Optics and Lasers in Engineering*, vol. 133, pp. 106139, 2020.

[50] Y. Wu, J. Noonan and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Journal of Selected Areas in Telecommunications*, vol. 1, no. 2, pp. 31–38, 2011.