doi:10.3969/j.issn.1001-4616.2025.06.007

基于超混沌系统的实景三维模型数据选择性加密算法

王　丹[1],顾进杰[2],任　娜[2],朱长青[2]

(1.江苏省基础地理信息中心,江苏 南京 210013)
(2.南京师范大学虚拟地理环境教育部重点实验室,江苏 南京 210023)

[**摘要**]　实景三维模型数据作为地理空间信息的重要载体,凭借其对真实空间的映射能力在智慧城市建设等领域发挥重要作用,但数据安全防护问题也日益凸显. 本文提出一种基于超混沌系统的实景三维模型数据选择性加密算法,利用超混沌系统初值敏感性与轨迹可复现性,对模型数据进行空间分块和多层级扰动加密;通过密钥参数控制加密区域与强度,实现模型的精细化、可调式加密策略. 实验结果表明,该方法在保证模型可用性的同时,具备较强的抗分析、抗攻击能力,且计算开销低,适用于对三维模型数据安全性和应用灵活性要求较高的场景.

[**关键词**]　实景三维,超混沌系统,选择性加密,动态加解密
[**中图分类号**]P208;TP309.7　[**文献标志码**]A　[**文章编号**]1001-4616(2025)06-0058-12

Selective Encryption Algorithm for Real-Scene 3D Model Data Based on Hyperchaotic System

Wang Dan[1],Gu Jinjie[2],Ren Na[2],Zhu Changqing[2]

(1.Provincial Geomatics Centre of Jiangsu,Nanjing 210013,China)
(2.Key Laboratory of Virtual Geographic Environment of Ministry of Education,Nanjing Normal University,Nanjing 210023,China)

Abstract:As an essential carrier of geospatial information,real-scene 3D model data plays a vital role in smart city construction and related fields due to its capability of mapping real-world environments. However,data security protection has become an increasingly critical issue. A selective encryption algorithm for real-scene 3D model data based on hyperchaotic system is proposed. Leveraging the hyperchaotic system's sensitivity to initial conditions and trajectory reproducibility,the algorithm performs spatial partitioning and multi-level perturbation encryption on the model data. By adjusting key parameters to control the encryption regions and intensity,a refined and tunable encryption strategy is achieved. Experimental results demonstrate that the proposed method maintains model usability while exhibiting strong resistance against analysis and attacks,with low computational overhead. It is suitable for scenarios requiring both high security and flexibility in 3D model data applications.

Key words:real-scene 3D,hyperchaotic system,selective encryption,dynamic encryption and decryption

近年来,实景三维模型数据在数字孪生、智慧城市、低空经济等战略性新兴领域的规模化应用持续推进,其数据安全防护问题日益凸显[1-2]. 考虑到该类数据兼具海量存储需求与复杂拓扑结构的双重特性,探索安全防护与加解密效率的协同优化路径,已成为当前地理空间信息安全领域的关键科学问题[3].

根据加密过程中加密算法直接操作的对象不同,加密算法可以分为两类. 一类是不解析文件内容,采用基于现代密码学体制的方法直接对文件的二进制序列进行整体变换. 常见的算法有国密 SM1 算法、国密 SM4 算法、高级加密标准(advanced encryption standard,AES)、数据签名算法(digital signature algorithm,DSA)、RSA 加密算法(Rivest-Shamir-Adleman)等[4-9]. 这类算法都将文件视为整体的加密对象,对其二进

收稿日期:2025-06-21.
基金项目:江苏省自然资源厅科技项目(JSZRKJ202405).
通讯作者:王丹,硕士,高级工程师,研究方向:智慧城市、新型基础测绘与实景三维. E-mail:554532067@qq.com

制流进行直接操作,但对数据量大的实景三维模型数据进行整体加解密效率慢,且丰富的坐标和属性等特有信息没有被充分利用. 第二类是需要解析文件内容,对文件中存储的坐标值、属性值或纹理等具体内容进行变换处理的置乱加密算法[10-12]. 该方法是指通过混沌系统等方式生成可复现的伪随机序列,然后根据伪随机序列对数据的坐标值、像素值或属性值进行置乱和扩散的加密算法. Gao 等[13]构建了二维混沌系统 2D-LAIC,并对坐标值整数和小数部分分别进行异或加密和 STP 加密,保证了加密后数据被充分置乱,解密后数据完全无损. Jin 等[14]利用一个三维混沌映射产生随机序列,然后将三维模型顶点坐标按照序列重新排序的方式进行加密,有效减少了置乱加密后的文件大小膨胀的问题,但并未顾及算法效率. 为了提高置乱的随机性,有学者提出利用多个混沌系统复合来提高混沌系统安全性. 许佶等[15]提出了一种基于三维自治混沌系统的复混沌系统,进一步提高了置乱的随机性. 但是上述加密算法仍然相对简单,仅利用混沌序列的随机性进行简单的排列置乱和扩散,面对小数据量数据时安全性和效率尚可,面对 OSGB 此类海量数据时,安全性和效率仍有待提高.

将随机序列和其他加密算法结合是一种有效提高安全性的手段[16-19]. Chu 等[16]利用混沌序列动态控制 3D Arnold 的置乱过程,并结合 RNA 编码与变异机制,对三维模型的顶点进行分步置乱和扩散加密,通过两轮不同类型的加密方式保证了算法的安全性. Jolfaei 等[17]利用混沌系统对三维模型进行随机填充,并围绕最小包围圆圆心进行置乱和旋转,在提高了安全性的同时,保证模型加密后仍然处于最小包围圆内,但填充算法难以避免带来了文件大小膨胀的问题. 这些算法在安全性更高的同时往往需要更多的计算时间,效率普遍较低.

综上所述,现有三维模型置乱加密方法主要面向小规模单体模型,难以适应 OSGB 等实景三维数据的特殊需求. 一是缺乏局部加密能力,无法支持按需访问;二是处理海量数据时效率不足,混沌加密等算法计算开销大、效率低. 针对 OSGB 数据的大体量、多层次特性,亟需建立一套兼顾安全性、高效性和轻量化的加密方法. 因此,本文提出了一种基于超混沌系统的实景三维模型选择性加密方法,通过多层级空间分块与超混沌扰动机制,实现模型数据的区域化、差异化加密,显著提升加解密过程的灵活性与安全性,在保证三维数据结构完整性的同时,为高敏感地理空间数据的保护提供了技术支撑.

1　算法设计

实现动态加解密的关键是在不破坏数据格式的前提下对模型数据进行扰动,使得非法用户无法提取有效信息. 实景三维模型的核心数据是模型点的空间坐标,通过对这些坐标进行充分扰乱,可以在保证文件格式完整的情况下实现数据加密. 若加密算法能够对部分数据点坐标实现恢复,则可实现动态加解密.

本文采用经典的“置乱-扩散”加密思路(见图 1). 首先将模型数据分块处理,引入混沌系统产生的混

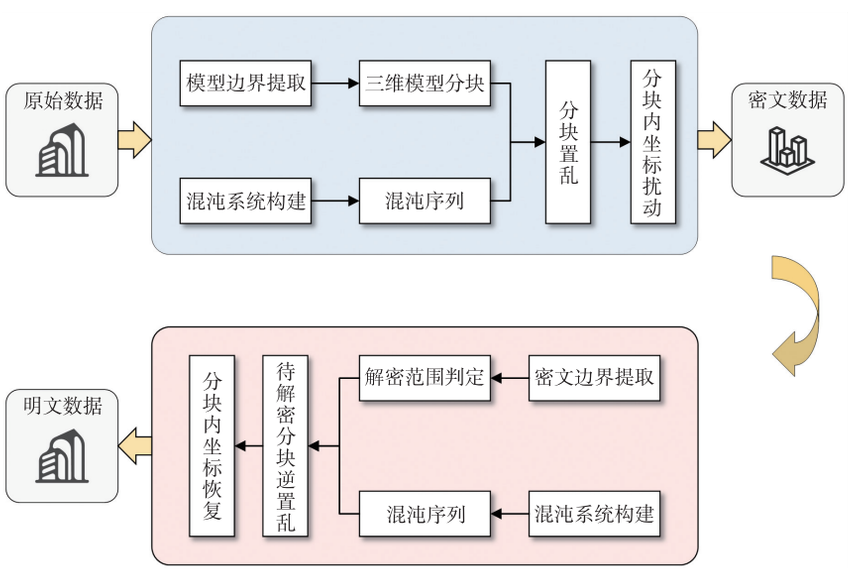


图 1　算法总体思路

Fig. 1　Overall algorithm approach

沌序列对分块后的数据进行置乱排序,然后通过非线性坐标映射对置乱后的坐标进行扰动,充分破坏空间信息. 为提升安全性,算法采用“一次一密”的加密策略,通过分块分步的加密机制,实现对模型数据不同区域和层级的灵活动态加解密.

1.1　分块置乱

为保证算法的效率和置乱的可恢复性,结合数据特性将模型数据进行空间分块,并对分块进行置乱. 分块的思路为:提取模型数据的最小包围盒,将最小包围盒的长、宽、高分别 n 等分,构建出 n^3 个空间分块,并从下到上、从左到右进行编号. 以 $n=3$ 为例,分块结果如图 2 所示.

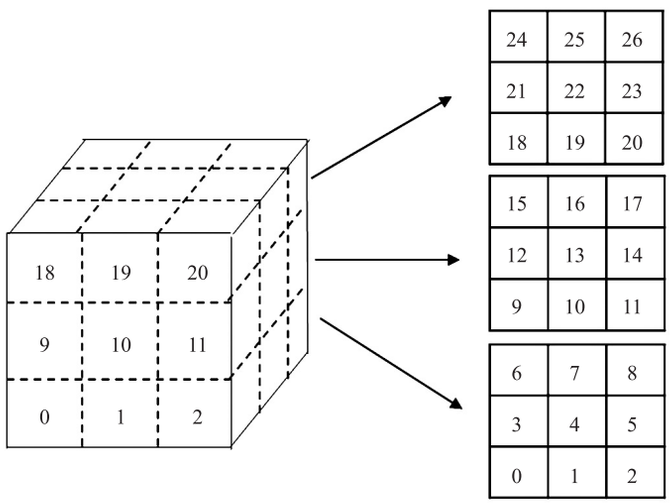


图 2　分块结果示意图

Fig. 2　Schematic diagram of block division

将每一个空间分块中的模型数据视为一个单独的体素,将各体素重新排序后构建新的立方体. 为保证算法的安全性,每个模型文件置乱的序列由混沌系统实时生成,其中一个序列对应的置乱效果如图 3 所示.

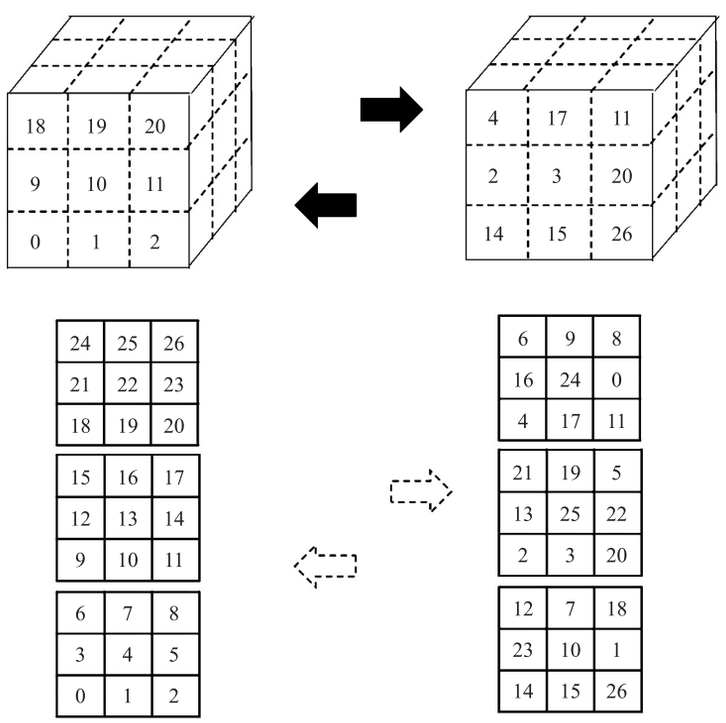


图 3　分块置乱效果图

Fig. 3　Effect diagram of block-based scrambling

首先,根据式(1)计算分组中各分块的长宽高;然后,根据式(2)计算各个分块的坐标范围.

$$\begin{cases} X_{\text{step}}=\dfrac{X_{\max}-X_{\min}}{n} \\ Y_{\text{step}}=\dfrac{Y_{\max}-Y_{\min}}{n} \\ Z_{\text{step}}=\dfrac{Z_{\max}-Z_{\min}}{n} \end{cases}, \tag{1}$$

$$\begin{cases} Xcrd_{\min}=X_{\min}+c\cdot X_{\text{step}} \\ Xcrd_{\max}=X_{\min}+(c+1)\cdot X_{\text{step}} \\ Ycrd_{\min}=Y_{\min}+r\cdot Y_{\text{step}} \\ Ycrd_{\max}=Y_{\min}+(r+1)\cdot Y_{\text{step}} \\ Zcrd_{\min}=Z_{\min}+d\cdot Z_{\text{step}} \\ Zcrd_{\max}=Z_{\min}+(d+1)\cdot Z_{\text{step}} \end{cases}, \tag{2}$$

式(1)中,$X_{\min}$、$X_{\max}$、$Y_{\min}$、$Y_{\max}$、$Z_{\min}$、$Z_{\max}$是模型数据最小包围盒的范围坐标值,X_{step}、Y_{step}、Z_{step}是分块的大小. 式(2)中 $Xcrd_{\min}$、$Xcrd_{\max}$、$Ycrd_{\min}$、$Ycrd_{\max}$、$Zcrd_{\min}$、$Zcrd_{\max}$是每个分组的六至坐标,c、r、d 分别为分组的行号、列号和深度号,与分组序号的映射关系可以根据式(3)计算:

$$id=n^3d+nc+r, \tag{3}$$

式中,id 为分组后的序号. 将分组序号 id 按照混沌序列的顺序进行排序,排序后获得序列 id',根据式(4)可以解得置乱后的行号、列号和深度号,根据式(5)可以计算出分块置乱后空间点的坐标值.

$$\begin{cases} c'=id' \bmod n \\ r'=(id'/n) \bmod n \\ d'=id'/n^2 \end{cases}, \tag{4}$$

$$\begin{cases} x'=x-c\cdot X_{\text{step}}+c'\cdot X_{\text{step}} \\ y'=y-r\cdot Y_{\text{step}}+r'\cdot Y_{\text{step}} \\ z'=z-d\cdot Z_{\text{step}}+d'\cdot Z_{\text{step}} \end{cases}, \tag{5}$$

式(4)中,c'、r'、d'分别为置乱后的行号、列号和深度号. 式(5)中,x'、y'、z'为置乱后空间点的坐标. 解密过程为加密过程的逆过程.

1.2　混沌密钥生成

为增强算法的安全鲁棒性,本文将分块置乱的序列和空间坐标的扩散的非线性映射分别采用不同的序列,故需要至少 4 个混沌序列,本文选取了一种基于传统三维洛伦兹混沌系统拓展的、具有多涡旋共存吸引子的超混沌系统(hyperchaotic system with multi-scroll coexistence attractors,HSMCA),用于产生伪随机序列[20]. 该混沌系统的参数方程如式(6)所示:

$$\begin{cases} \dot{x}_1=\alpha(2x_2-x_1)+8x_4 \\ \dot{x}_2=\beta x_1/2-x_2-2\rho x_1x_3 \\ \dot{x}_3=-\chi x_3+\rho x_1x_2/2 \\ \dot{x}_4=-x_1/8-\delta x_2/4 \end{cases}, \tag{6}$$

式中,$\dot{x}_1$、$\dot{x}_2$、$\dot{x}_3$、$\dot{x}_4$ 为系统的状态变量,α、β、χ、ρ、δ 为混沌系统的系统参数.

为生成多个滚动吸引子,显著提高混沌吸引子的复杂性和分布范围,增强系统的非线性程度、灵活程度和混沌特性,将混沌系统取值根据式(7)分段.

$$\theta_i(x_i)=x_i+\sum_{m=1}^{g}(-1)^{g-m}(|x_i-(2m-1)|-|x_i+(2m-1)|), \tag{7}$$

式中,参数 g 控制分段数量,决定滚动吸引子的数量,参数 m 控制滚动吸引子的幅度和范围. 为了保证伪随机序列的安全性,该系统的参数取值如表 1 所示.

表 1　超混沌系统参数

Table 1　Hyperchaotic system parameters

参数	α	β	χ	δ	ρ	g
取值	10	28	8/3	2	15	6

在此系统参数下,系统初始值 $x_i \in [-10,10]$ 时,系统呈现超混沌状态. 本文将系统参数固定,初始值作为混沌密钥,构建混沌系统,并事先迭代 1 000 次,使得系统进入混沌状态,输出混沌序列作为伪随机序列用以加密算法.

算法将混沌系统初始值作为密钥,若只使用一个混沌系统,虽然也能保证“一钥一密”的加密策略,但是由于模型数据坐标点数量巨大,在明文攻击模型下仍可能存在安全风险,故本文为每一个文件设置不同的混沌系统初始值. 为快速生成大量在混沌系统超混沌行为取值范围内的初始值,本文设计了一种多级密钥派生算法. 该算法利用密码学中常见的 PBKDF2(password-based key derivation function 2)算法,算法步骤如下:

步骤 1　盐值计算. 如式(8)所示,将文件名输入到 SHA256 哈希函数中,生成一个固定长度的哈希值作为盐值. 哈希处理的作用是将初始密钥转换为一个不可逆且长度固定的值.

$$salt = \mathrm{HASH256}(fileName), \tag{8}$$

式中,$salt$ 为每个初始密钥生成的盐值,$fileName$ 为文件名.

步骤 2　使用 PBKDF2 算法派生密钥. 如式(9)所示,将主密钥、盐值输入 PBKDF2 算法进行密钥派生. PBKDF2 算法通过多次迭代计算生成一个字节数组,字节数组的长度由所需的密钥长度决定.

$$K = \mathrm{PBKDF2}(PK, salt_i), k_i \in K, \tag{9}$$

式中,K 为生成的字节数组,k_i 为 K 中第 i 个字节的取值,PK 为用户密钥.

步骤 3　从字节数组生成密钥. 从 PBKDF2 算法返回的字节数组中提取每个字节值. 每个字节值的范围是 0~255,这代表了一个 8 比特的数字. 对每个字节值进行映射,将其转换到-10~10 范围内的浮点数.

$$CSK = \left\{ csk_i \mid csk_i = 10.0 + \frac{20.0k_i}{256.0} \right\}, \tag{10}$$

式中,CSK 为转换后的浮点数组,称为混沌密钥,csk_i 为第 i 个浮点数. 将上述从初始密钥 IK 到混沌密钥 CSK 的过程称为混沌密钥生成,并将此过程记录为式(11).

$$CSK = \mathrm{GCSKP}(PK). \tag{11}$$

1.3　算法流程

加密算法的流程如图 4 所示.

加密算法的具体步骤为:

步骤 1　边界提取. 逐次遍历各文件($file_i$),提取各文件边界,计算整个模型的边界,记为 $BBox_{model}$.

$$\begin{cases} BBox_i = (Ci_{\min}, Ci_{\max}) \\ Ci_{\min} = (Xi_{\min}, Yi_{\min}, Zi_{\min}) \\ Ci_{\max} = (Xi_{\max}, Yi_{\max}, Zi_{\max}) \\ BBox_{model} = (\min(Ci_{\min}), \max(Ci_{\max})) \end{cases}, \tag{12}$$

式中,$BBox_i$ 为第 i 个 OSGB 文件 $file_i$ 的最小包围盒,$Xi_{\min}$、$Yi_{\min}$、$Zi_{\min}$、$Xi_{\max}$、$Yi_{\max}$、$Zi_{\max}$ 为最小包围盒的六至坐标,$\min(\cdot)$ 和 $\max(\cdot)$ 分别为求最小值和最大值的过程,最后整个模型的包围盒为 $BBox_{model}$.

步骤 2　生成空间分块置乱序列. 利用主密钥生成混沌密钥,构建混沌序列,根据混沌序列获取置乱后的序列 ξ_m.

$$\begin{cases} CSK_m = \mathrm{GCSKP}(PK) \\ \xi_m = \mathrm{HSMCA}(CSK_m) = [S_{m1}, S_{m2}, S_{m3}, S_{m4}] \end{cases}, \tag{13}$$

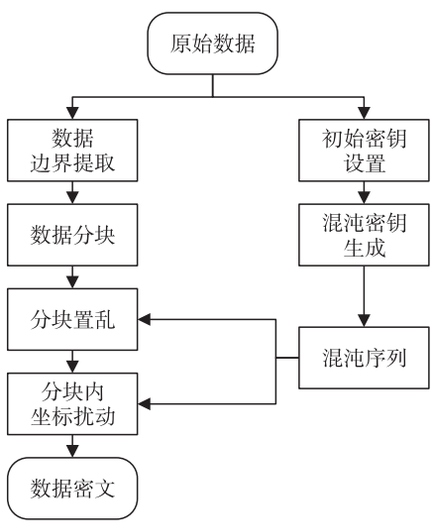


图 4　加密算法流程

Fig. 4　Encryption algorithm flow

式中,CSK_m 为使用主密钥生成的混沌密钥,ξ_m 对应混沌密钥和生成的混沌序列,HSMCA(·)为利用混沌系统生成混沌序列的过程,S_{m1}、S_{m2}、S_{m3}、S_{m4}为混沌系统的 4 个状态量构成的序列.

步骤 3　空间分块置乱. 对整个模型进行分块,分块个数为 n^3,将分块后的模型数据按照置乱序列 ξ_m 进行置乱.

$$model_m=\mathrm{M}_{\mathrm{block}}(model,S_{m1}),\tag{14}$$

式中,$model_m$ 为分块置乱后的模型数据,$\mathrm{M}_{\mathrm{block}}(\cdot)$是分块置乱的过程,利用 S_{m1} 前 n^3 个值的排序对模型数据进行置乱,$model$ 是被分块置乱的模型数据.

步骤 4　生成扰动参数. 分别根据文件名、序列号和主密钥生成初始密钥,利用盐值和主密钥通过密钥派生算法生成混沌系统初始值,构建混沌系统,并将混沌系统迭代 1 000 次,使得系统进入混沌状态.

$$\begin{cases}CSK_i=\mathrm{GCSKP}(PK_i)\\ \xi_i=\mathrm{HSMCA}(CSK_i)=[S_{i1},S_{i2},S_{i3},S_{i4}]\end{cases},\tag{15}$$

式中,CSK_i 是第 i 个模型文件对应的混沌密钥,ξ_i 是第 i 个模型文件对应的混沌序列的集合,S_{i1}、S_{i2}、S_{i3}、S_{i4}为 4 个混沌序列.

步骤 5　分块内坐标扰动. 遍历各个坐标点,对分块内数据进行扰动,获得密文数据.

$$model_{md}=\mathrm{Sigmod}(model_m,\xi_i),\tag{16}$$

扰动的过程使用 Sigmod 函数进行,分别对空间坐标的三维坐标进行 Sigmod 映射. 以 x 坐标为例,映射过程如式(17)所示.

$$x_e=c'\cdot X_{\mathrm{step}}+\frac{X_{\mathrm{step}}}{1+\mathrm{e}^{-code\cdot\left(\frac{x-c'\cdot X_{\mathrm{step}}}{X_{\mathrm{step}}}-0.5\right)}},\tag{17}$$

式中,x_e 为分块置乱后的 x 坐标,c'为分块置乱后 x 坐标所在分块的行号,$code$ 为该点对应混沌序列 S_{i2}中的取值. y 坐标和 z 坐标根据式(17)分别利用 S_{i3}和 S_{i4}序列对应的 $code$ 取值进行置乱.

解密算法的流程如图 5 所示. 解密算法的过程可以视为加密算法的逆过程,但完全解密和部分解密的流程略有区别.

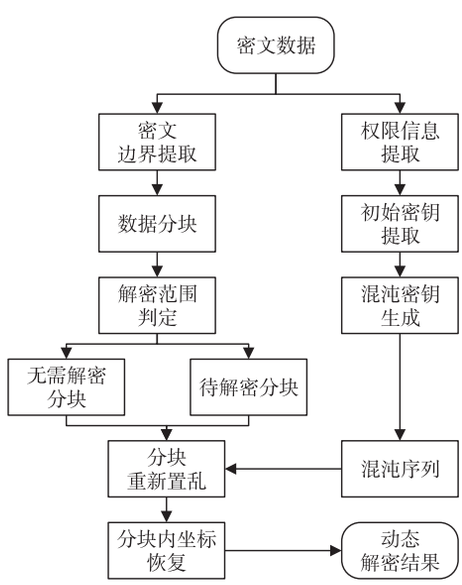


图 5　解密算法流程

Fig. 5　Decryption algorithm flow

步骤 1、步骤 2　与加密算法相同,获取 $BBox_{model}$和分块置乱序列 ξ_m.

步骤 3　空间分块置乱恢复. 对整个模型进行分块,分块数量为 n^3.

若需要完全解密,则将分块后的模型按照置乱序列 ξ_m 进行逆映射,过程如式(18)所示:

$$model'_d=\mathrm{M}^{-1}_{\mathrm{block}}(model_{md},S_{m1}),\tag{18}$$

式中,$\mathrm{M}^{-1}_{\mathrm{block}}$为 $\mathrm{M}_{\mathrm{block}}$的逆过程.

若需要选择性解密数据,则置乱序列需要更新为 ξ'_m. 更新规则为:先判断在解密范围内的分块序号,然后将解密范围内分块序号进行恢复,剩余的序号按照混沌序列进行重新置乱,如图 6 所示,得到 ξ'_m,此时按照此序列进行逆置乱:

$$model'_d=\mathrm{M}^{-1}_{\mathrm{block}}(model_{md},S'_{m1}),\tag{19}$$

式中,S'_{m1}是 S_{m1}更新后的序列.

步骤 4　和加密算法步骤 4 相同.

步骤 5　对解密部分的分块进行逆映射,得到部分解密模型:

$$model'=\mathrm{Sigmod}^{-1}(model'_d,\xi_i),\tag{20}$$

式中,Sigmod^{-1}是 Sigmod 映射的逆过程,计算公式如式(21)所示:

$$x=c'\cdot X_{\mathrm{step}}+X_{\mathrm{step}}\cdot\left(0.5-\ln\left(\frac{X_{\mathrm{step}}}{x_e-c'\cdot X_{\mathrm{step}}}-1\right)/code\right).\tag{21}$$

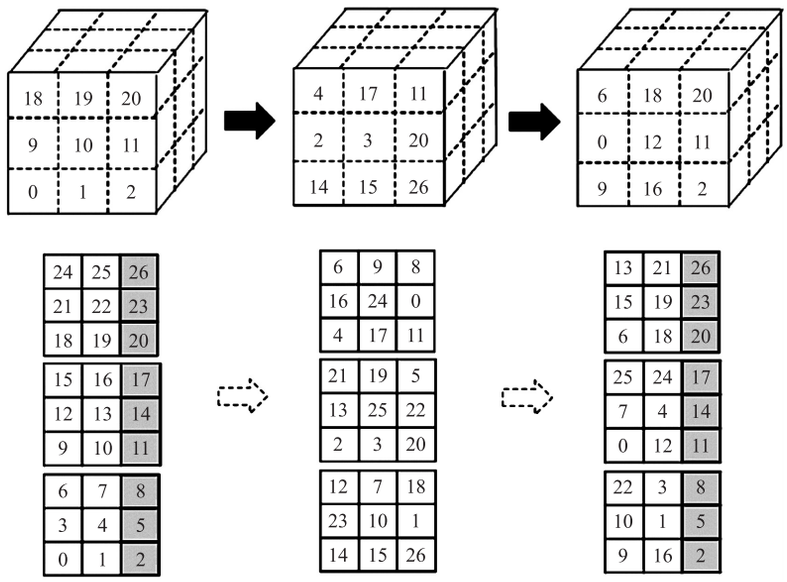


图 6　部分解密示意图

Fig. 6　Schematic diagram of partial decryption

2　实验设计与分析

2.1　实验数据

为验证算法的安全性、效率和可动态加密的能力,本文选取了表 2 中 3 组数据进行实验分析.

表 2　置乱加解密算法实验数据

Table 2　Experimental data for scrambling-based encryption and decryption algorithm

数据编号	数据大小	数据文件数量	模型坐标点数	数据缩略图
M01	3.27 GB	21 823	196 933 533	
M02	8.93 GB	49 083	434 725 383	
M03	10.1 GB	48 538	618 741 048	

2.2　加密效果

对 3 组实验数据实施加密后,视觉分析表明密文数据已呈现完全随机化特征,而解密数据可精确恢复至原始状态,验证了算法的可逆性与有效性,如图 7 所示.

加密后数据已经完全不可用,同时由于加密算法特性,数据均匀分散到最小包围盒内,保证了加密后数据的安全性. 数据解密恢复后,可以发现数据和原数据视觉上相同,不影响数据的正常使用.

为验证置乱加密的安全性和解密的无损性,本文利用均方根误差(root mean square error,RMSE)来进行定量评价,均方根误差越大,代表密文和原文的差异越大,反之则越小. 本文计算了密文与原数据的均方根误差以及密文解密后与原数据的均方根误差,结果如表 3 所示.

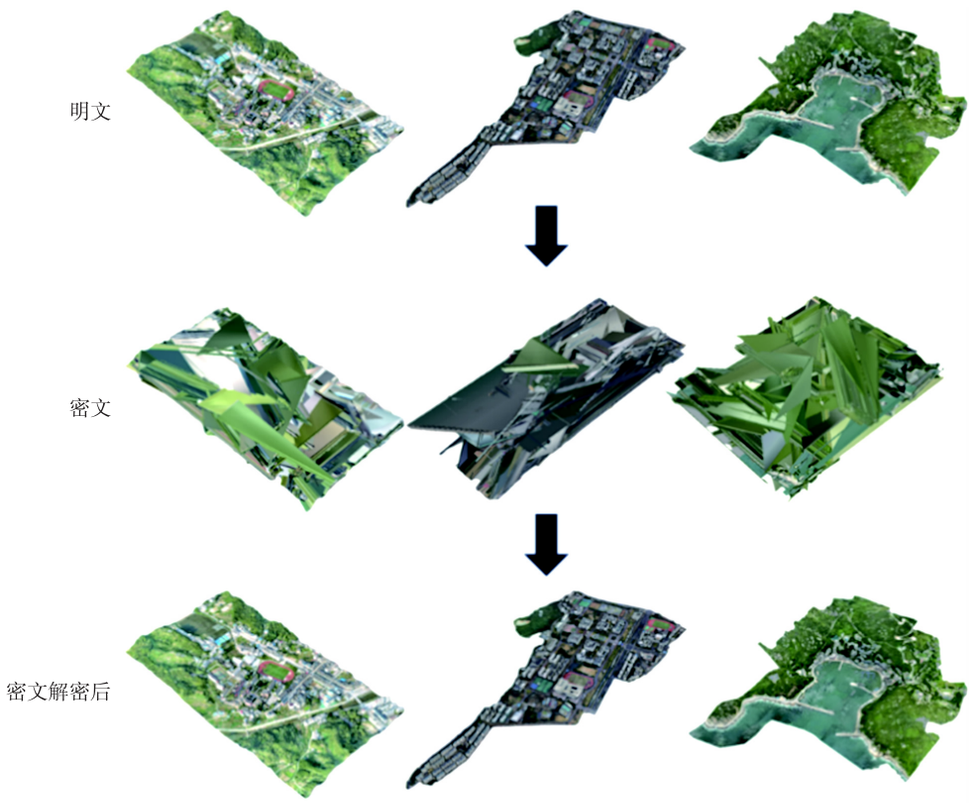


图 7　置乱加密结果

Fig. 7　Scrambling encryption result

从表 3 可以看出,密文与原数据均方根误差较大,说明数据和原数据差异很大,数据进入不可用状态;密文解密后与原数据均方根误差为 0,说明解密后数据在空间坐标上与原数据没有差异,数据已经完全恢复.

表 3　加解密前后坐标的均方根误差

Table 3　RMSE of coordinates before/after encryption and decryption

数据编号	RMSE	
	密文与原数据	密文解密后与原数据
M01	494.251	0
M02	822.896	0
M03	623.215	0

2.3　效率实验

为验证算法的效率,本文从每秒处理的数据大小和节点个数进行评价,对 3 份数据分别进行加解密实验,结果如表 4 所示.

表 4　加解密效率实验结果

Table 4　Experimental results of encryption and decryption efficiency

数据编号	加密速度/(M/s)	解密速度/(M/s)	节点加密速度/(N/s)	节点解密速度/(N/s)
M01	48.795	46.567	2 869 789	2 738 725
M02	49.260	47.134	2 341 853	2 240 794
M03	51.137	47.810	3 059 333	2 860 238

本文选取 2 个三维模型数据的加解密算法进行对比实验. 由于 OSGB 格式的模型数据置乱加解密算法鲜有研究,故本文选取了针对 STL 格式的三维模型数据空间坐标置乱加密算法,并对算法进行适用性优化,使其可以对 OSGB 格式数据进行加解密,分别记为算法 A[14] 和算法 B[21]. 在相同环境下,计算了加解密的平均速度,结果如表 5 所示.

表 5　加解密效率对比实验结果

Table 5　Comparative experimental results of encryption and decryption efficiency

算法	平均加解密速度/(M/s)
本文算法	48.451
算法 A	11.503
算法 B	12.970

算法 A 对模型数据的空间坐标执行保留格式加密后,进一步对顶点序号进行重排序. 该算法在 STL 等小数据量格式中能够实现效率与安全性的平衡,但

在 OSGB 等大型文件场景下,排序操作的效率难以满足本文的性能需求. 算法 B 通过分别置乱空间坐标的整数部分与小数部分实现无损加解密,然而,由于实景三维模型数据需存储高精度浮点数值,故其计算开销显著增加. 综合来看,本文所提算法在海量数据处理场景中表现出更优的效率特性.

2.4　密钥安全性

2.4.1　密钥空间

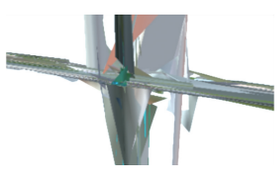
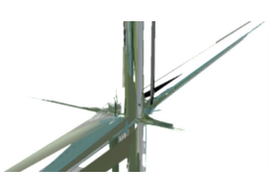
本文的密钥空间主要取决于主密钥的长度和每位可能的取值[22]. 本文主密钥每位有 39 种取值,将密钥长度设定为 30 位,则本文的密钥空间为 3 930,大于阈值 2 100,表明本文方法具有好的密钥安全性.

2.4.2　密钥敏感性

为保证算法的安全性,算法的解密结果必须对密钥的初始值极度敏感. 为了验证本文算法对密钥的安全性,修改密钥的其中 1 位后对数据执行解密操作,虽然本文涉及多个密钥的中间值,但是用户能直接接触的只有主密钥,故将主密钥修改 1 位后,解密效果如表 6 所示.

表 6　密钥敏感性实验结果

Table 6　Experimental results of key sensitivity

数据编号	正确密钥	输入密钥	解密效果
M01	0DOXCH76J5N49H7 54K09H76321UBGS	0DOXCH76J5N49H7 54K09H76321UBGA	
M02	9KJGDW857208NVG DH5920OKVXH689V	9KJGDW857208NVG DH5920OKVXH689A	
M03	6GKIS90OLNBVCD7 RFGYUJ89IKNBV45	6GKIS90OLNBVCD7 RFGYUJ89IKNBV4A	

从实验结果可知,当密钥的某一位发生错误时,解密后的数据不仅完全混乱,其空间分布也远超原始模型的包围盒范围,导致数据彻底失去可用性.

2.5　动态加解密实验

2.5.1　数据范围动态解密

为验证本文算法的动态解密能力,对密文数据按包围盒的不同范围执行部分解密操作. 其中,数据 M01 的解密范围为 50%宽度、长度从 10%增至 60%;数据 M02 的解密范围为 100%宽度、长度从 10%增至 60%;数据 M03 的解密范围为 100%长度、宽度从 10%增至 60%. 通过上述实验设计,验证本文算法对不同权限区域的动态解密能力,实验结果如表 7 所示.

从实验结果可知,本文算法可以根据包围盒的范围对密文数据进行解密,解密部分可以正常使用,其余部分维持密文状态,说明本文算法具备动态解密的能力.

2.5.2　数据范围动态加密

为验证本文算法的动态加密能力,对密文数据按包围盒的不同范围执行部分加密操作. 其中,数据 M01 的加密范围为 50%宽度、长度从 10%增至 60%;数据 M02 的加密范围为 100%宽度、长度从 10%增至 60%;数据 M03 的加密范围为 100%长度、宽度从 10%增至 60%. 实验结果如表 8 所示.

表 7　数据范围动态解密实验结果

Table 7　Experimental results of data-range-driven dynamic decryption

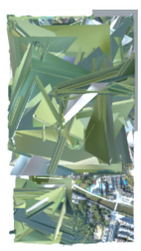
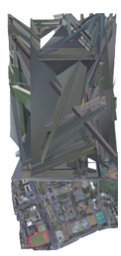
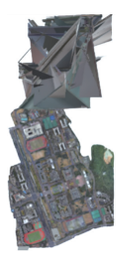
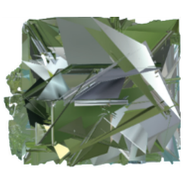
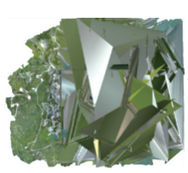
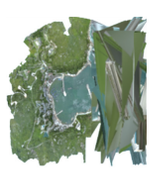
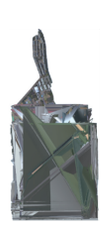
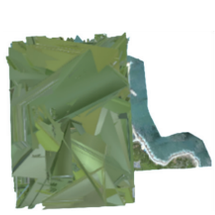
数据编号	10%	30%	60%
M01			
M02			
M03			

表 8　数据范围动态加密实验结果

Table 8　Experimental results of data-range-driven dynamic encryption

数据编号	10%	30%	60%
M01			
M02			
M03			

从实验结果可知,本文算法能够实现数据的部分加密,加密部分无法正常使用,其余部分则保持原始数据精度.

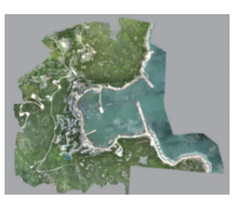
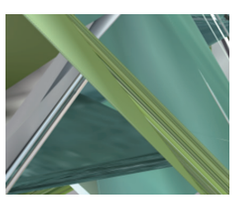
2.5.3　*数据层级动态解密*

为验证本文算法的动态解密能力,实验从文件最低细节等级开始依次对数据进行解密. 表 9 展示了

解密至 L20 及以下层级时的数据显示效果.

表 9　分层解密效果图

Table 9　Effect diagram of layered decryption

数据编号	小于 L20 视角	大于 L20 视角	局部细节
M01			
M02			
M03			

如表 9 所示,在 L20 及以下细节层级下,数据可正常显示;但当数据缩放到需渲染 L20 以上层级的程度时,该部分数据仍保持加密状态,即使进一步缩放到局部细节,未解密数据仍保持混乱状态,无法正常使用. 本质上,本文加密算法对不同细节层级的文件采用分层加解密设计,在解密某一层级数据时,不会对其他层级数据产生影响. 因此,已解密层级在渲染过程中可正常加载显示,而未解密层级在加载时仍维持置乱状态,无法正常显示.

3　结论

本文提出了一种基于超混沌系统的实景三维模型选择性动态加密算法,在多层级空间分块结构基础上,引入混沌序列对“置乱-扩散”过程进行控制,实现了对模型空间坐标的高强度扰动加密. 该算法在保证模型数据结构完整性的基础上,支持对不同区域和层级数据进行灵活的动态加密与解密,提升了加密的精细化与可控性. 实验结果表明,该方法具有较强的抗攻击能力和较低的计算开销,适用于对三维模型数据安全性和灵活性要求较高的实际应用场景. 未来工作可进一步优化加密效率并扩展算法在多模态三维数据保护中的应用,为实景三维模型的数据安全提供更全面的技术保障.

[参考文献]

[1] 陈军,刘建军,田海波. 实景三维中国建设的基本定位与技术路径[J]. 武汉大学学报(信息科学版),2022,47(10):1568-1575.

[2] 陈军,田海波,高鉴,等. 实景三维中国的总体架构与主体技术[J]. 测绘学报,2025,54(4):636-649.

[3] 朱长青,任娜,徐鼎捷. 地理信息安全技术研究进展与展望[J]. 测绘学报,2022,51(6):1017-1028.

[4] LI S, ZHAO R, GUAN Q, et al. A 3D model encryption method supporting adaptive visual effects after decryption[J]. Advanced engineering informatics,2024,59:102319.

[5] GONG H,JU T. Distributed power analysis attack on SM4 encryption chip[J]. Scientific reports,2024,14(1):1007.

[6] FATIMA S, REHMAN T, FATIMA M, et al. Comparative analysis of AES and RSA algorithms for data security in cloud computing[J]. Engineering proceedings,2022,20(1):14.

[7] ALGHAMDI Y,MUNIR A. Image encryption algorithms:a survey of design and evaluation metrics[J]. Journal of cybersecurity and privacy,2024,4(1):126-152.

[8] CHEN J,YOU F. An image encryption algorithm based on SM4 and Base64[J]. Journal of physics:conference series,2021,1812(1):012041.

[9] KAUR M,KUMAR V. A comprehensive review on image encryption techniques[J]. Archives of computational methods in engineering,2020,27(1):15-43.

[10] ZHOU N R,HU L L,HUANG Z W,et al. Novel multiple color images encryption and decryption scheme based on a bit-level extension algorithm[J]. Expert systems with applications,2024,238:122052.

[11] WEN H,LIN Y. Cryptanalysis of an image encryption algorithm using quantum chaotic map and DNA coding[J]. Expert systems with applications,2024,237:121514.

[12] TOKTAS A,ERKAN U,GAO S,et al. A robust bit-level image encryption based on Bessel map[J]. Applied mathematics and computation,2024,462:128340.

[13] GAO S,WU R,WANG X,et al. A 3D model encryption scheme based on a cascaded chaotic system[J]. Signal processing,2023,202:108745.

[14] JIN X,ZHU S,XIAO C,et al. 3D textured model encryption via 3D Lu chaotic mapping[J]. Science China information sciences,2017,60(12):122107.

[15] 许佶,孙博,杨飞飞,等. 基于混沌系统的商标图像加密算法[J]. 大连工业大学学报,2019,38(3):221-228.

[16] CHU R,ZHANG S,GAO X. A novel 3D image encryption based on the chaotic system and RNA crossover and mutation[J]. Frontiers in physics,2022,10:844966.

[17] JOLFAEI A,WU X W,MUTHUKKUMARASAMY V. A 3D object encryption scheme which maintains dimensional and spatial stability[J]. IEEE transactions on information forensics and security,2015,10(2):409-422.

[18] XU J,ZHAO C,MOU J. A 3D image encryption algorithm based on the chaotic system and the image segmentation[J]. IEEE access,2020,8:145995-146005.

[19] QU C,DU J,XI X,et al. A hybrid domain-based watermarking for vector maps utilizing a complementary advantage of discrete fourier transform and singular value decomposition[J]. Computers & geosciences,2024,183:105515.

[20] YAN M,XU H. The multi-scroll hyper-chaotic coexistence attractors and its application[J]. Signal processing:image communication,2021,95:116210.

[21] LU Y,GONG M,GAN Z,et al. Exploiting one-dimensional improved Chebyshev chaotic system and partitioned diffusion based on the divide-and-conquer principle for 3D medical model encryption[J]. Chaos,solitons & fractals,2023,171:113449.

[22] 熊颖. 基于敏感特征的矢量地理数据选择性加密算法研究[D]. 南京:南京师范大学,2023.

[责任编辑:丁　蓉]