ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
Received: 10 May 2025; Accepted: 30 September 2025

https://doi.org/10.17559/TV-20250510002656
Original scientific paper

A Hybrid MOPNA-SPM Algorithm for Secure Digital Information Embedding in Enterprise Data Protection

Qiu RAN, Hongyan ZHU*

Abstract: With the acceleration of enterprise digital transformation, the risk of sensitive data leakage has significantly increased. Traditional Digital Information Embedding (DIE) techniques suffer from critical limitations: limited embedding capacity, insufficient concealment (prone to visual distortion), weak robustness against attacks (e.g., noise and compression), and high computational complexity, failing to meet enterprises' demands for secure and large-scale sensitive data hiding. To address these issues, this study proposes a novel hybrid algorithm that integrates the Modulus Calculations on Prime Number Algorithm (MOPNA) and Single Pixel Modification (SPM) algorithm, which is an innovative combination that leverages MOPNA's pixel grouping strategy (dividing carriers into dual-pixel groups) and modular operation optimization based on prime weight, along with SPM's "high capacity-low distortion" single-pixel adjustment advantage. Based on this hybrid algorithm, a DIE model integrated with a Security and Authentication Module (SAM) is constructed to enhance data security during transmission and storage. The core contribution of this research lies in developing a DIE model with higher embedding capacity and stronger robustness than traditional methods. The practical application in an architectural design enterprise shows that the model achieves an embedding capacity of 3.51 bpp (36.58% higher than the SPM and 100.57% higher than the INFO algorithm), a Peak Signal-to-Noise Ratio (PSNR) of 48.25 dB, and a Structural Similarity (SSIM) of 0.98, ensuring near-lossless visual concealment. In terms of security, its anti-noise recovery rate reaches 95.50%, the anti-compression attack recovery rate is 90.25%, and the average information extraction time is only 0.45 s per image (62.50% faster than the SPM). This model provides a new technical solution for secure transmission and storage of enterprise sensitive information, with important reference value for data protection in other fields.

Keywords: data protection; digital information; MOPNA; SM4; SPM

1 INTRODUCTION

In the context of global digitization, financial statements and customer information in daily operations, as well as enterprise data related to core competitiveness such as research and development data and strategic planning, have become key assets that drive enterprise development and maintain competitive advantages [1-3]. However, the explosive growth in the storage and transmission of sensitive data has led to an increased risk of leakage, such as frequent financial fraud caused by customer account information breaches, and long-term collapse of enterprise innovation results caused by research and development data breaches [4-5]. This makes enterprise data protection a strategic priority related to survival [6]. Traditional data protection technologies such as encryption and access control can only secure static data but fail to meet the covert requirements of data transmission. Digital Information Embedding (DIE) technology, which hides sensitive data in carrier files without altering file appearance, has emerged as a key means for enterprise data protection [7, 8]. Among DIE techniques, the Single Pixel Modification (SPM) algorithm has gained attention for its "high capacity-low distortion" feature, while traditional algorithms such as Least Significant Bit (LSB), Discrete Cosine Transform (DCT), Singular Value Decomposition (SVD) and even some deep learning-based methods still have critical limitations [9-10].

Existing methods (LSB, DCT, SVD, and deep learning) fail for the following reasons. Although the LSB algorithm is simple, it concentrates the embedded information in the high-frequency region of the image, resulting in low embedding efficiency, significant visual distortion, and difficulty in balancing capacity and concealment. The DCT algorithm has improved the capacity, but it damages the visual quality of the image and lacks robustness. The SVD algorithm requires complete image decomposition and reconstruction, resulting in high computational complexity, low embedding efficiency, and insufficient robustness. Deep learning methods typically require excessive computing resources, are difficult to scale, and often sacrifice robustness or efficiency when optimizing individual performance metrics. Most traditional technologies overlook security risks in data transmission and storage, and cannot meet high-intensity protection requirements. Therefore, this study proposes a hybrid MOPNA-SPM algorithm, which combines the Modulus Calculations on Prime Number Algorithm (MOPNA) and SPM algorithm. It optimizes modular operation parameters through MOPNA's pixel grouping strategy to ensure image quality, cleans redundant data to reduce computational consumption, and introduces encryption algorithms to enhance data security, thereby addressing the aforementioned issues.

This study makes three key contributions: 1) An improved hybrid embedding algorithm (MOPNA-SPM): MOPNA and SPM are fused. The MOPNA expands the embedding space through modular operations based on prime weight, and the SPM's single pixel fine-tuning is utilized to ensure concealment, breaking the bottleneck of traditional algorithms in balancing capacity, distortion, and efficiency, and improving the embedding effect. 2) A DIE model with security and authentication: Based on the MOPNA-SPM algorithm, the model integrates the Security and Authentication Module (SAM), including SM4-Galois/Counter Mode (SM4-GCM) encryption and Galois Message Authentication Code (GMAC) authentication, effectively addressing security vulnerabilities of traditional embedding technologies in data transmission and storage. 3) Extensive evaluation on enterprise data: The model is validated using the Public Enterprise Document Image Dataset (PEDID) covering multiple document types and applied in an architectural

design enterprise, comprehensively verifying its performance in embedding capacity, concealment, anti-attack capability, and practical applicability.

2 LITERATURE REVIEW

The development of DIE technology is a process of constantly seeking breakthroughs in key performance dimensions such as embedding capacity, concealment, and anti-attack. Wang et al. proposed an image watermark encryption method based on an improved Fisher shuffling algorithm to protect information security. The results indicated that this method performed well in time cost and encryption effectiveness [11]. To solve the image art copyright protection, Wang et al. proposed a data embedding scheme called "SDCP-IE". Experiments showed that the Peak Signal-to-Noise Ratio (PSNR) of the image after data embedding exceeded 57 dB, and the performance was excellent under the adversary detection model [12]. Yu et al. proposed a new image Steganography framework called Controllabel, Robust, and Secure Image Steganography (CRoSS) to improve the security and natural robustness of image Steganography tasks. The results indicated that compared with the coverage-based image steganography method, CRoSS had significant advantages in controllability, robustness, and security [13]. In terms of data security protection, Cao et al. proposed an algorithm based on improved decision tree classification to solve the low precision and high noise of data mining methods for blockchain privacy protection. Experiments showed that the mining accuracy was more than 90%, and the data noise was stable [14]. Jiang proposed a new Reversible Data Hiding Algorithm for Encrypted Images (RDHEI) based on adaptive total variation and cross-cyclic shift to reduce prediction error and improve the embedding rate of secret messages. Experimental results indicated that the proposed RDHEI scheme was privacy secure and had high embedding capacity and image fidelity [15]. Singla et al. proposed a secret data detection method based on minimum redundancy and maximum correlation for data transmission leakage in the Internet of things environment. Experiments showed that this method could accurately identify sensitive data and prevent continuous data leakage [16].

However, the above methods have obvious limitations in enterprise-level data protection. Traditional non-Artificial Intelligence (AI) methods (e.g., SDCP-IE, RDHEI) only optimize single indicators, failing to balance capacity, concealment, and efficiency. AI-based methods (e.g., CRoSS) rely on massive labeled data (hard to obtain for sensitive enterprise data), have complex structures (high resource consumption, CPU usage > 40%), and perform poorly on enterprise-specific data (anti-attack ability drops by 15-20%). In addition, all lack systematic security designs (no encryption/authentication) and support only single-type carriers, unable to adapt to diverse enterprise data. Therefore, the MOPNA-SPM algorithm with an integrated encryption module is proposed. It differs from recent methods mainly in three aspects: 1) Balancing capacity, concealment, and efficiency through MOPNA-SPM fusion without relying on large-scale training data; 2) Building an integrated security system with "embedding encryption authentication" to address security vulnerabilities in existing methods; 3) Supporting multi-type enterprise carriers with low computational complexity, suitable for enterprise deployment. The comparison of related work methods is shown in Tab. 1.

Table 1 Comparison table of related work methods

Research Method	Research Content	Performance	Limitations
Improved Fisher Shuffling [11]	Image watermark encryption through improved Fisher shuffling	Good time cost & encryption effect	No embedding function; lacks transmission concealment & authentication
SDCP-IE [12]	Data embedding for image art copyright protection	PSNR > 57 dB; good anti-detection performance	Low embedding capacity; only supports images; incompatible with enterprise docs
CRoSS [13]	Image steganography framework for security/robustness improvement	Superior controllability/robustness vs coverage-based methods	Needs massive labeled data; high CPU usage; poor adaptability to enterprise data
Decision Tree-Based [14]	Improved decision tree for blockchain privacy protection data mining	Mining accuracy > 90%; stable noise	No embedding function; only for blockchain; unable to meet covert transmission
RDHEI [15]	Reversible data hiding for encrypted images (adaptive total variation)	High privacy/embedding capacity/fidelity	High complexity; only supports encrypted images; unbalanced efficiency/concealment
mRMR-Based [16]	mRMR-based sensitive data leakage detection for IoT	Accurate leakage prevention	No embedding/protection; only for IoT; incompatible with enterprise scenarios
This study (MOPNA-SPM & DIE)	MOPNA-SPM fusion + SAM (SM4-GCM encryption + GMAC authentication)	3.51 bpp (↑ 36.58% vs SPM); PSNR 48.25 dB; anti-noise 95.50%; extraction 0.45 s/image	Poor adaptability to high-dynamic images/videos; limited in ultra-large data

3 RESEARCH METHODOLOGY

3.1 Improved SPM Algorithm Based on MOPNA

In the context of digitalization, the confidentiality and transmission security of enterprise data is crucial. Traditional DIE techniques are prone to the dilemma of limited capacity and easy detection, making it difficult to meet the security needs of enterprises for hiding large amounts of sensitive data [17]. As an advanced information hiding technology, SPM algorithm realizes information embedding through SPM, which can effectively improve the imperceptibility of encrypted image in the high embedding capacity. The structure of SPM algorithm is shown in Fig. 1.

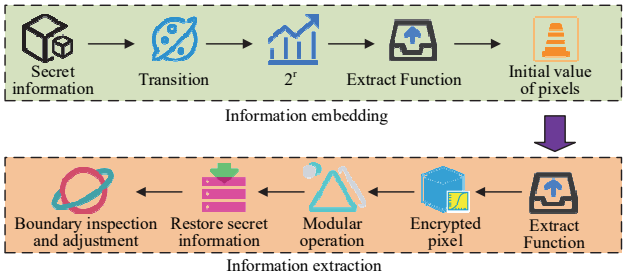

Figure 1 Structure of SPM Algorithm

From Fig. 1, SPM algorithm is an information hiding algorithm based on SPM. Its structure is mainly composed of information embedding and extraction. In the information embedding stage, the secret information is first converted to 2^r. Then, the initial value of the carrier pixel is calculated by the extraction function. Meanwhile, the pixel value offset that meets the conditions is determined, so that the secret information can be successfully embedded into a single pixel. The extraction function is shown in Eq. (1) [18].

$$d = (a + x) \bmod 2^r \tag{1}$$

In Eq. (1), d represents the value of the extraction function. a represents the grayscale value of the carrier pixel. x represents the pixel value offset. r represents the number of bits of the embedded secret information. Specifically, based on the number of bits r of the secret information, the cardinality of the modulo operation is determined as 2^r. By calculating the sum of the carrier pixel value and offset, the modulo of 2^r is taken to obtain the pixel value embedded with the secret information. The extracted values of pixels after embedding secret information have been determined, which can provide a calculation basis for subsequent embedding operations. The pixel value adjustment during the embedding process is shown in Eq. (2) [19].

$$a' = a + x \tag{2}$$

In Eq. (2), a' represents the value of secret carrying pixels after embedding secret information. After specifying the update method for the grayscale values of the carrier pixels after embedding secret information, pixel value adjustment can be directly achieved through offset to carry secret data. To ensure that the embedded pixel value is still within the effective range (0-255), the boundary of the result is checked and adjusted. The adjustment calculation is shown in Eq. (3) [20].

$$a' = \begin{cases} a' - 2^r & \text{if } a' > 255 \\ a' + 2^r & \text{if } a' < 0 \end{cases} \tag{3}$$

This ensures that the adjusted pixel grayscale values remain within the effective range of 0-255, avoiding severe image distortion caused by numerical overflow. In the information extraction stage, the receiver performs modular operation on the encrypted pixels through the same extraction function to recover the embedded secret information [21, 22]. SPM algorithm realizes information hiding through SPM and modular operation, which has high embedding capacity and good imperceptibility. To further improve the embedding efficiency and comprehensive performance, MOPNA algorithm is adopted. The structure of MOPNA algorithm is shown in Fig. 2.

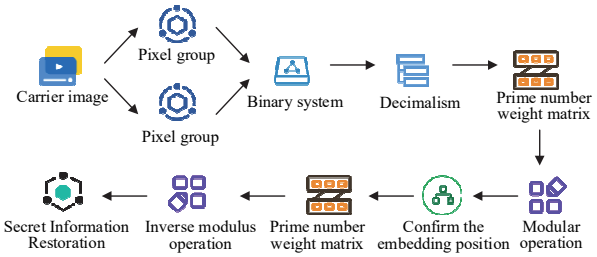

Figure 2 Structure of MOPNA algorithm

From Fig. 2, MOPNA is a high comprehensive performance information hiding method for digital image carriers. The algorithm uses the pixel grouping strategy, divides the carrier image into two pixel groups, and improves the embedding efficiency by optimizing the modular operation parameters. The eigenvalue calculation of pixel group is shown in Eq. (4) [23].

$$\Gamma = \left(\sum_{i=1}^{m} v_i \cdot u_i \right) \bmod Q \tag{4}$$

In Eq. (4), Γ represents the eigenvalue of the current pixel group, which is used to map secret data. m represents the number of pixels in the pixel group (default $m = 2$). v_i represents the gray value ($v_i \in [0,255]$) of the i-th pixel in the carrier pixel group. u_i stands for the i-th prime weight coefficient (e.g. $u_1 = 3$ and $u_2 = 5$). Q stands for the modulus base, which is determined by the prime weight and parameters, and $Q = 2k + 1$. Combining prime weight with pixel grayscale value to calculate pixel group feature values can provide a quantization standard for mapping secret data to pixel groups. First, the binary secret information is segmented into decimal, and the prime weight matrix is used to calculate the embedding position of each group of pixels. The secret data encoding rule is shown in Eq. (5) [24].

$$S_{\text{target}} = \text{Decode}(B_{\text{bin}}, Q) \tag{5}$$

In Eq. (5), S_{target} represents the encoded decimal secret value. B_{bin} represents a binary secret information fragment of length $2k + 1$. Decode represents converting binary to Q-based numerical values. Binary secret information fragments are transformed to adapt to the feature value range of pixel groups for embedding. Then, by adjusting the pixel values to match the modular operation results with the secret information, data embedding is achieved. The pixel adjustment function is shown in Eq. (6) [25].

$$\Delta = (S - \Gamma + Q) \bmod Q \tag{6}$$

In Eq. (6), Δ represents the pixel index offset that needs to be adjusted. By calculating the deviation between the secret value and the feature value, the pixel adjustment amount is determined to ensure that small adjustments can complete data embedding and reduce visual artifacts in the image. If $\Delta \le k$, the adjustment strategy is shown in Eq. (7) [26].

$$v'_{\Delta} = v_{\Delta} + 1 \tag{7}$$

In Eq. (7), $v'_{Q-\Delta} = v_{Q-\Delta} - 1$ represents the gray value of the $v'_{Q-\Delta} = v_{Q-\Delta} - 1$ -th pixel after adjustment. $v'_{Q-\Delta} = v_{Q-\Delta} - 1$ represents the gray value of the $v'_{Q-\Delta} = v_{Q-\Delta} - 1$ -th pixel before adjustment. When the adjustment amount is small, fine tuning is achieved by adding 1 to the specified pixel grayscale value to maintain the original visual effect of the image to the maximum extent. If $v'_{Q-\Delta} = v_{Q-\Delta} - 1$, the adjustment strategy is shown in Eq. (8) [27].

$$v'_{Q-\Delta} = v_{Q-\Delta} - 1 \tag{8}$$

In Eq. (8), v'_{Δ} represents the gray value of the $Q-\Delta$-th pixel after adjustment. v_{Δ} represents the gray value of the $Q - \Delta$-th pixel before adjustment. When the adjustment amount is large, the corresponding pixel grayscale value is subtracted by 1 for adjustment, and the pixel change amplitude is controlled while completing data embedding. In the specific implementation, the algorithm selects the coprime as the weight parameter to construct the modular operation system, ensuring that each pixel group can embed $2k$ + 1-bit information. For example, when k = 3, each group can embed 7 bits of data, significantly improving the embedding capacity per pixel. In the extraction phase, the secret information is restored by inverse modular operation with the same prime weight. Compared with the traditional method, MOPNA expands the embeddable value range through the special number theory characteristics of prime weight, and controls the pixel modification range within ±1. To further improve the efficiency and security of enterprise data protection, MOPNA and SPM are combined to form MOPNA-SPM algorithm. The data protection process of the MOPNA-SPM algorithm is shown in Fig. 3.

From Fig. 3, the data protection process of MOPNA-SPM algorithm includes dual-mode cooperation mechanism and dynamic adjustment strategy to achieve safe and efficient information embedding and extraction. The whole process starts from the preprocessing stage of secret data. The original sensitive information is converted into binary streams and coded according to the multi-prime weight matrix. Each group can carry 7 bit of data. Subsequently, the carrier image is divided into two pixel groups, and the eigenvalues of the pixel groups are calculated based on the coprime prime weights. A nested modular operation system (prime weight module and single pixel module) is constructed to expand the embedding space. The core of the embedding phase is the dynamic adjustment strategy. Firstly, the adjustment mode is determined by the modulus deviation (the difference between the prime weight module and the target secret value). The modulus deviation calculation is shown in Eq. (9).

$$P_{\Delta} = \left| S_{\text{target}} - \left(p_1 \cdot x_1 + p_2 \cdot x_2 \right) \bmod N \right| \tag{9}$$

In Eq. (9), P_{Δ} represents the modulus deviation. p_1 and p_2 represent a weight matrix composed of two coprime prime numbers. x_1 and x_2 represent the original dual-pixel grayscale values (range 0-255) in the carrier image. N represents the modulus determined by the weights and parameters of prime numbers. The deviation between the secret target value and the weighted calculation result of the dual pixel group is quantified, providing a basis for selecting single pixel fine-tuning or prime compensation mechanisms. When $P_{\Delta} \le 1$, the single pixel ±1 correction strategy of SPM is adopted to ensure visual concealment. If $P_{\Delta} > 1$, the prime compensation mechanism of MOPNA is activated to achieve large-scale numerical adaptation by adjusting the weight combination of pixel pairs. Each pixel embedded in this process needs to undergo boundary constraint verification, and the overflow value is forcibly corrected using a grayscale truncation function to ensure the legitimacy of the encrypted image. The extraction stage separates information through reverse dual-mode operation. The receiving end performs prime weight modular inverse operation on the encrypted pixel group to obtain the original binary stream.

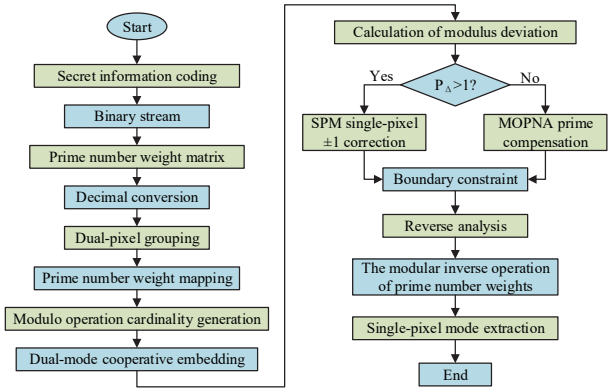

Figure 3 The data protection process of MOPNA-SPM algorithm

3.2 Construction of Digital Information Embedding Model Based on MOPNA-SPM Algorithm

MOPNA-SPM algorithm can realize efficient, secure and imperceptible information hiding through pixel grouping and SPM strategy to optimize modular operation parameters. This helps to improve the efficiency and security of enterprise data protection. To better apply MOPNA-SPM algorithm to the actual protection scenario, and build a flexible and scalable framework to meet the diversified needs of different enterprises, the DIE model is established based on MOPNA-SPM algorithm. To enhance the security of embedded information, the DIE model adds the SAM. SAM includes encryption sub-module and authentication sub-module. The encryption sub-module

uses SM4-GCM algorithm to encrypt the secret information, ensuring that even when the embedded encrypted image is intercepted. It is difficult for unauthorized users to obtain the original information. SM4-GCM combines the high security of the SM4 algorithm that complies with national security standards and has strong anti-attack capabilities with the "encryption authentication integration" feature of the GCM mode. It can synchronously achieve data confidentiality, integrity verification, and identity authentication, greatly improving the efficiency and security of enterprise data protection. In the integration process of SM4-GCM and SAM, a 128 bit SM4 symmetric key is first generated using cryptoFrame.creatSymKeyGenerator and SymKeyGenerator.generateSymKey. Then, a Cipher instance is created for encryption through cryptoFrame.createCipher and passed in 'SM4_128 | GCM | PKCS7' to initialize the encryption process. The structure of SM4-GCM is shown in Fig. 4.

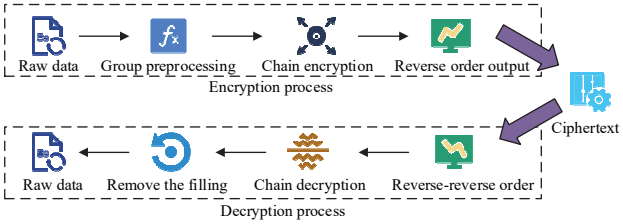

Figure 4 Structure of SM4-GCM

From Fig. 4, SM4-GCM uses the Cipher-Block Chaining (CBC) mode to divide the plaintext into 128 bit packets, and each packet is associated with the previous ciphertext block through XOR operation. The chain encryption is shown in Eq. (10).

$$C_i = \mathrm{SM4}_{\mathrm{enc}}\left(P_i \otimes C_{i-1}\right) \tag{10}$$

In Eq. (10), C_i represents the i-th ciphertext group. P_i stands for the i-th plaintext packet. $\otimes$ stands for exclusive or operation. $\mathrm{SM4}_{\mathrm{enc}}$ stands for the grouping operation. The first block is XOR with the initialization vector, and the subsequent blocks are XOR with the previous ciphertext block in turn to form a chain encryption sequence. The current plaintext group is XORed with the previous ciphertext group and encrypted, constructing a chain encryption structure to improve data encryption security. After 32 rounds of SM4 encryption, reverse order transformation (R transformation) is performed to arrange the bytes of the output ciphertext block in reverse order. The reverse order arrangement is shown in Eq. (11).

$$Y_N = \mathrm{ReverseBytes}\left(X_{32} \| X_{33} \| X_{34} \| X_{35}\right) \tag{11}$$

In Eq. (11), Y_N represents the bytes arranged in reverse order. X_{32}-X_{35} represent the 32-bit word state variable after encryption iteration. $\|$ stands for byte connection operation. ReverseBytes represents the reverse order transformation function (for example, input 0x12345678 and output 0x78563412). This design makes the encryption and decryption process only need to adjust the round key sequence, without the need to develop a separate reverse algorithm, so as to improve the implementation efficiency. The round key generation is shown in Eq. (12).

$$rk_j = K_{j+4} = \left(K_j \otimes FK_j\right) \boxplus L'\left(K_{j+1} \otimes K_{j+2} \otimes K_{j+3} \otimes CK_j\right) \tag{12}$$

In Eq. (12), rk_j represents the round key. K_j stands for intermediate key word, which is the intermediate state value generated iteratively after the XOR between the encryption key and the system parameters. FK_j stands for system parameters. CK_j stands for fixed parameter. $\boxplus$ stands for composite permutation operation. L' represents the linear transformation of key expansion. j stands for natural number. Based on the initial key and system parameters, round keys are iteratively generated to provide secure and efficient key support for 32 rounds of SM4 encryption. After the initial key and system parameters are XOR, the round key is generated through 32 rounds of iteration combined with fixed parameters. Each round of key expansion adopts a simplified linear transformation (shifting 13/23 bits to the left), which can ensure the security and reduce the computational complexity. The authentication sub-module is based on GMAC in SM4-GCM mode, which mainly realizes data integrity verification, identity authentication, and tamper resistance protection. The structure of the authentication sub-module is shown in Fig. 5.

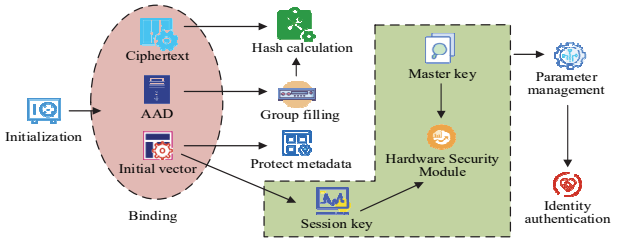

Figure 5 Structure of authentication sub-module

From Fig. 5, the authentication sub-module binds the ciphertext and Additional Authentication Data (AAD) with the initial vector to ensure that the data is not tampered with. For example, in financial transactions, AAD can include transaction serial number and time stamp to prevent replay attacks. AAD is used to protect metadata without encryption (such as protocol header and device ID). The authentication sub-module fills the AAD in 128 bit groups and participates in GMAC hash calculation with the ciphertext synchronously to ensure the strong correlation between metadata and encrypted content. The key is a hierarchical key. The master key is stored in the Hardware Security Module (HSM). The session key is dynamically derived and bound to the initial vector. It is destroyed after a single session. Parameter management adopts anti-side channel protection, introduces random mask in key expansion and label generation, and resists timing attack through constant time algorithm. SAM supports parallel processing of encryption and authentication, and can effectively improve throughput through multi-core CPU hardware acceleration. SAM effectively improves data security through encryption,

decryption, and authentication. To reduce data redundancy and ensure the availability of decrypted data, a data cleaning module and a data post-processing module are also added to SAM. The process of SAM protecting enterprise data is shown in Fig. 6.

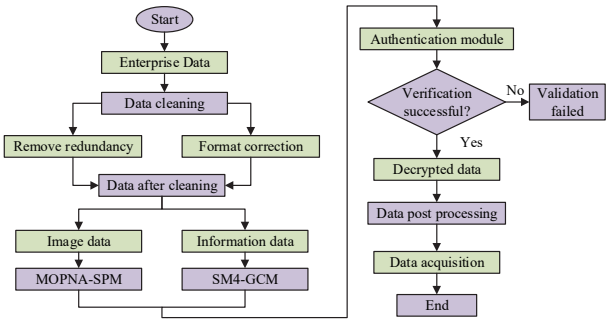

Figure 6 Process of SAM for protecting enterprise data

From Fig. 6, SAM first receives the original data to be protected by the enterprise, which may be various sensitive information, such as financial data, customer information, etc. Then, the data is sent to the data cleaning module. The data cleaning module processes the original data to remove redundancy, errors, or invalid parts. The missing values are filled with mean and median values. MOPNA-SPM algorithm is used for information embedding and data protection. The information embedding is shown in Eq. (13).

$$I_{\text{steg}} = I_{\text{orig}} \oplus S \tag{13}$$

In Eq. (13), I_{steg} represents the carrier data after embedding the secret information. I_{orig} stands for the original carrier data. S represents the secret information to be embedded, usually in the form of binary stream. The original carrier data and binary secret information are XORed to achieve hidden embedding of secret information in the carrier. The cleaned document data is encrypted with SM4-GCM. The encrypted data enters the authentication module for authentication and decryption to ensure data security. The encrypted and authenticated data enters the data post-processing module for format conversion, data verification, and other operations to ensure data integrity and availability. Finally, SAM sends the processed data to the receiver, so as to protect the data security of the enterprise and avoid data leakage. In practice, the SAM module first encrypts the secret information to be embedded using the SM4-GCM encryption algorithm to ensure the confidentiality of the information. Subsequently, the encrypted ciphertext is embedded into the carrier data through specific rules. In the identity verification stage, after obtaining the carrier with embedded information, the receiver first extracts encrypted information according to preset rules, decrypts and restores the original information using SM4-GCM, and verifies the source and integrity of the information using GMAC authentication mechanism. If the verification is successful, the sender's identity is confirmed to be legitimate.

4 RESULTS AND DISCUSSION

4.1 Performance Analysis of MOPNA-SPM Algorithm

To comprehensively analyze the performance of MOPNA-SPM algorithm, a high-performance experimental platform is built. The CPU is Intel Core i5-12400. The memory is 16 GB DDR4 3200 MHz. The GPU is NVIDIA GeForce RTX 3060 12 GB GDDR6. H610 chipset is selected as the main board. The hard disk is 1TB NVMe. The operating system is Windows 10 and the development environment is Visual Studio 2019. The parameter configuration of MOPNA-SPM algorithm is shown in Tab. 2.

Table 2 Parameter configuration of MOPNA-SPM algorithm

Parameter Name	Parameter Value
Prime Weight Matrix	$p_1 = 2, p_2 = 3$
Modulus Base	7
Embedded Bits	3
Pixel Group Size	2
Dynamic Adjustment Threshold	3
Boundary Constraint Check	Grayscale Truncation Function
Prime Compensation Mechanism	Activation Condition: $d > 3$
Preprocessing Group Length	7 bits/group
Grayscale Adjustment Range	±1
Dual-Module Nesting Level	2-level modulus operations

The experimental data set uses the PEDID, which contains about 1000 different types of enterprise document images. The resolution of each image covers the common 600 × 800 pixels to 2400 × 3000 pixels, to cover document types with different resolution requirements, from ordinary office documents to high-resolution design drawings. The storage size of the whole dataset is about 1 GB, and the capacity is moderate, which is convenient for loading, storage and processing in the general experimental environment, and meets the needs of algorithm performance evaluation. The comparison algorithm uses the traditional SPM algorithm and the Weighted Mean of Vectors (INFO) algorithm [28]. The calculation resource consumption and embedding time of each algorithm are shown in Fig. 7.

From Fig. 7a, the CPU usage of MOPNA-SPM algorithm was significantly lower than that of INFO and SPM algorithms, and its fluctuation range was stable at around 28.21%, while the CPU usage of INFO and SPM was around 33.87% and 47.91%, respectively. This shows that MOPNA-SPM algorithm optimizes the allocation of computing resources through the dual-mode cooperative mechanism, thus reducing redundant operations. From Fig. 7b, MOPNA-SPM algorithm had the shortest embedding time, and the average time of embedding information into each image was 0.86s, which was significantly lower than that of INFO (1.94 s) and SPM (2.63 s). This is because MOPNA-SPM algorithm reduces the number of pixel adjustments by segmenting the binary stream into a prime weighted modular space. Meanwhile, the boundary truncation function avoids the iteration time of overflow

check. To more intuitively show the effect of secret information embedding, the secret information is embedded in different types of images. The gray image after embedding the secret information is shown in Fig. 8.

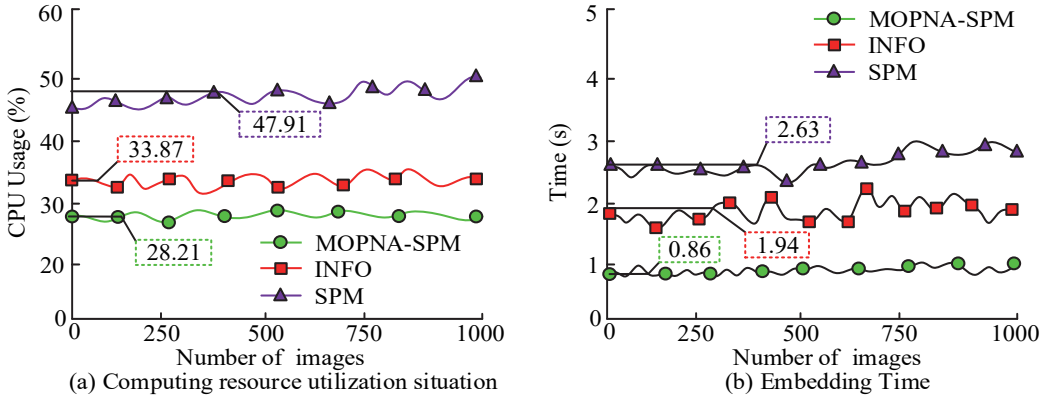

Figure 7 The calculation resource consumption and embedding time of each algorithm

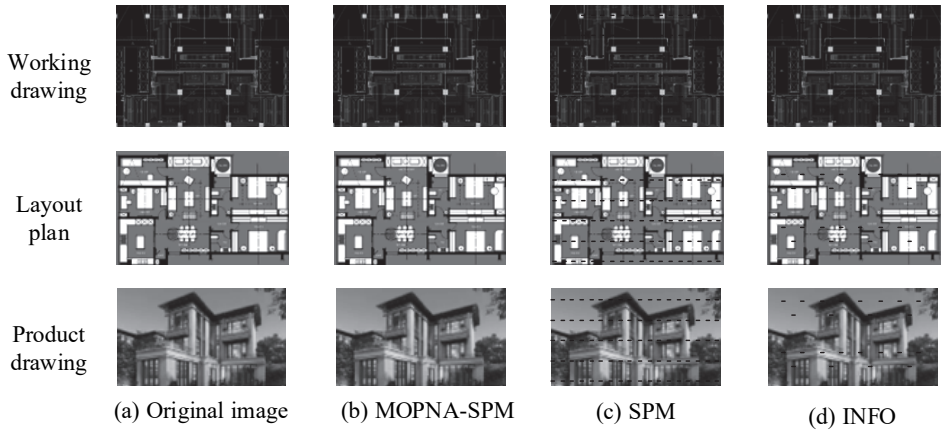

Figure 8 Gray scale image after embedding secret information

From Fig. 8, SPM algorithm and INFO algorithm had stains on the plane layout and product effect drawing, and it was obvious that the image was processed. Since the construction drawing has a black background, the stain is not obvious. The image embedded with secret information through MOPNA-SPM algorithm is basically the same as the original image, and the obvious difference cannot be seen with the naked eye. This is because MOPNA-SPM algorithm uses the pixel grouping strategy to divide the carrier image into two pixel groups. MOPNA-SPM algorithm can make full use of the correlation between pixels, disperse the secret information and reduce the embedding intensity of a single pixel. MOPNA-SPM algorithm can avoid image distortion and stains caused by a large change of a single pixel.

4.2 Analysis of the Practical Application Effect of DIE Model

To verify the practical application effect, the DIE model is applied in an architectural design enterprise. Compared with other methods, the information embedding accuracy and file size growth rate of the DIE model are shown in Fig. 9.

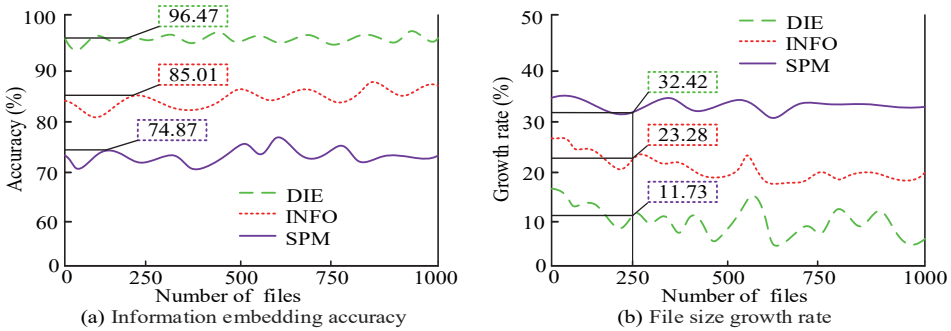

Figure 9 Information embedding accuracy and file size growth rate

According to Fig. 9a, the information embedding accuracy of DIE model was 96.47% on average, which was significantly higher than that of INFO (85.01%) and SPM (74.87%). This shows that DIE model can accurately embed secret information and ensure the integrity of information. The DIE model has no significant performance fluctuation due to the increase in the number of files. From Fig. 9b, the size of different files is different, and the change of file size before and after information embedding is also different. Taking the information embedding of file No. 250 as an example, the file size growth rate of DIE model was the lowest, only at 11.73%, while that of INFO was 23.28%, and that of SPM was 32.42%. This is mainly due to the efficient compression and coding technology of DIE model, which can reduce redundant data while embedding information, thus reducing the file size growth rate. The DIE model performs well in terms of information embedding accuracy and file size growth rate. The DIE model can meet the needs of information hiding and file management in practical applications. To reflect the information security, the DIE model also adds the encryption module. The encryption speed, decryption speed, verification false positive rate, and CPU usage of the DIE model for different types of data are shown in Fig. 10.

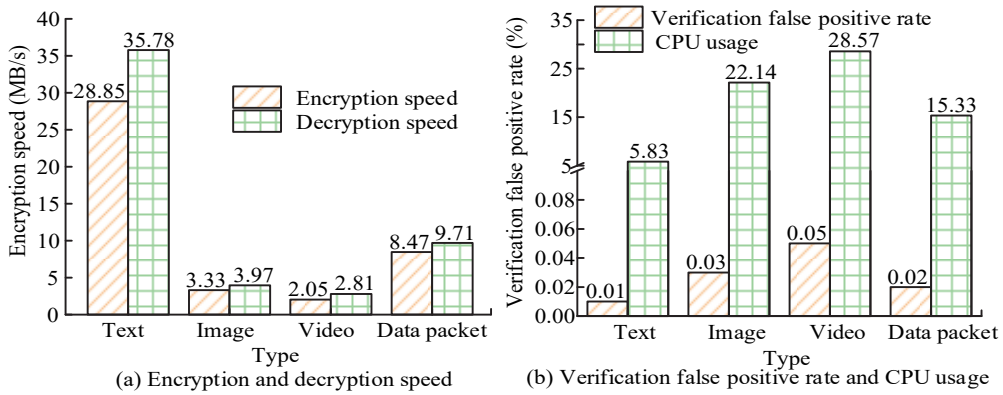

Figure 10 The performance of DIE model on different types of data

From Fig. 10a, the DIE model performed differently in the encryption and decryption speed of different types of data. The encryption speed of text data was 28.85 MB/s, and the decryption speed was 35.78 MB/s, which was the best performance. This is because the text data structure is simple and regular, and the DIE model can process it efficiently. The encryption speed of image data was 3.33 MB/s, and the decryption speed was 3.97 MB/s, which was relatively low. The encryption speed of video stream data was 2.05 MB/s, and the decryption speed was 2.81 MB/s. Due to its continuity and real-time requirements, it is difficult to process. The encryption speed of data packet was 8.47 MB/s, and the decryption speed was 9.71 MB/s. From Fig. 10b, the verification false positive rate and CPU usage of the DIE model differed significantly on different types of data. The verification false positive rate of text data validation was 0.01%, and the CPU usage was 5.83%. The verification false positive rate of image data was 0.03%, and the CPU usage was 22.14%. The false positive rate of video stream data was 0.05%, and the CPU usage rate was 28.57%. The verification false positive rate of data packet was 0.02%, and the CPU usage was 15.33%. The overall performance of DIE model is excellent, thanks to its efficient verification mechanism and resource management strategy. The other performances (mean ± error) of the DIE model in the actual application of the architectural design enterprise are shown in Tab. 3.

Table 3 Performance (mean ± error) of DIE model in architectural design enterprise applications

Metric	DIE (MOPNA-SPM)	SPM	INFO	GAN-based Steganography	Transformer-based Embedding
Embedding Capacity / bpp	3.51 ± 0.12	2.57 ± 0.15	1.75 ± 0.11	2.89 ± 0.18	3.12 ± 0.14
PSNR / dB	48.25 ± 0.85	42.30 ± 1.12	39.80 ± 1.05	45.12 ± 0.98	46.33 ± 0.92
SSIM	0.98 ± 0.01	0.92 ± 0.02	0.89 ± 0.02	0.95 ± 0.01	0.96 ± 0.01
Noise Resistance / %	95.50 ± 1.20	80.20 ± 1.50	75.30 ± 1.35	88.65 ± 1.42	90.18 ± 1.30
Compression Resistance / %	90.25 ± 1.15	70.45 ± 1.60	65.60 ± 1.48	82.33 ± 1.55	84.75 ± 1.40
Extraction Time / s/image	0.45 ± 0.05	1.20 ± 0.10	0.98 ± 0.08	1.55 ± 0.12	1.10 ± 0.09

According to Tab. 3, embedding capacity, PSNR, Structural Similarity (SSIM), anti-interference ability, and extraction time indicators are selected based on the DIE technology, which needs to meet the core requirements of enterprises for "carrying sensitive data with large capacity, avoiding detection with high concealment, ensuring transmission security with strong resistance to attacks, and adapting to business scenarios with high efficiency". Each indicator corresponds to the technical core performance dimension, providing a comprehensive and suitable evaluation basis for the practical value of the model in enterprise scenarios. In terms of embedding capacity, the DIE model reached 3.51 bpp, significantly higher than that of SPM (2.00 bpp) and INFO (1.75 bpp). Its advantage lies in the dual-pixel group design combined with prime weight matrix, which expands the secret information mapping

range of each pixel group through modular operation. The single pixel ±1 adjustment strategy avoids embedding conflicts, thereby improving the carrying capacity of the unit pixel. In terms of concealment, the PSNR of the DIE model was 48.25 dB, and the SSIM was 0.98, which was close to the lossless level. In terms of anti-attack performance, the DIE model had a higher recovery rate in noise (95.50%) and compression (90.25%) scenarios. The extraction time of the DIE model was only 0.45 s/image, which had a fast information extraction speed. The DIE fusion based on MOPNA pixel grouping and prime weight modular operation expands the embedding space, and SPM single pixel ±1 fine-tuning ensures concealment. The integration of SM4-GCM encryption and GMAC authentication optimizes anti-attack capability, and reduces computational consumption by removing redundant data, achieving collaborative improvement in multiple performance dimensions. The DIE model is superior to traditional methods in embedding capacity, concealment, resistance to attacks, and information extraction speed through number theory optimization and structural innovation. The DIE model can meet the needs of enterprise level data protection for efficiency, security, and low perception.

4 DISCUSSION

The practical application results of the DIE model in architectural design enterprises reveal its unique advantages in enterprise data protection. Its embedding capacity of 3.51bpp (±0.12) far exceeds that of traditional SPM (2.57 ± 0.15 bpp) and INFO (1.75 ± 0.11 bpp) algorithms, and even outperforms AI-based methods such as GAN-based steganography (2.89 ± 0.18 bpp) and transformer-based embedding (3.12 ± 0.14 bpp). This is attributed to the dual-pixel grouping design of the MOPNA-SPM algorithm and the prime weight matrix. By optimizing the modular operation parameters, the mapping range of secret information in each pixel group has been expanded. The single pixel ±1 adjustment strategy avoids embedding conflicts and breaks the bottleneck of limited capacity in traditional algorithms.

Compared with existing research, this study has two core breakthroughs. Firstly, it abandons the "single-index optimization" of traditional methods (such as SDCP-IE focusing only on concealment) and the "data-dependent" defect of AI-based methods (such as CRoSS requiring massive labeled data). By integrating MOPNA and SPM algorithms, a balanced improvement in multiple performance indicators has been achieved without relying on large-scale training data, which is more in line with the actual needs of enterprises for "efficient and secure" data protection. Secondly, most existing studies (such as the RDHEI algorithm) only focus on information embedding technology and ignore the security design for enterprise scenarios. This study integrates the SM4-GCM national encryption algorithm and the GMAC authentication mechanism to form a systematic security protection system, making up for the shortcomings of existing methods such as lack of identity authentication and weak anti-attack capabilities. For example, in the financial data transmission scenario of architectural design enterprises, the DIE model can not only hide sensitive data such as project budgets in design drawings, but also verify the identity of the data sender through GMAC to prevent data tampering and leakage.

However, this study still has certain limitations. The used experimental dataset is relatively small in scale, and transmission testing for large-scale real-time data streams has not been conducted. In addition, there are few baseline coverage methods, and more emerging DIE technologies have not been included, which may affect the universality and comprehensiveness of the conclusions. Future research will focus on three aspects. 1) Developing lightweight algorithm versions suitable for mobile devices, and reducing resource consumption by optimizing modular operation parameters and streamlining module structures. 2) Exploring deep integration with blockchain audit logs, and utilizing the immutable nature of blockchain to strengthen the traceability and supervision of data embedding throughout the entire process. 3) Conducting specialized testing on high-resolution video streams to enhance the real-time embedding and anti-interference performance of the model in dynamic data scenarios.

5 CONCLUSION

This study proposed a DIE model based on MOPNA-SPM algorithm to address the covert transmission needs of enterprise data. By constructing a dual-mode collaborative mechanism and dynamically adjusting strategies, the DIE model achieved secure and efficient information embedding and extraction. The experimental results showed that the DIE model outperformed traditional methods in key indicators such as embedding capacity, concealment, resistance to attacks, and information extraction speed. Its embedding capacity reached 3.51 bpp, which was significantly improved compared with SPM algorithm (2.57 bpp) and INFO algorithm (1.75 bpp). The PSNR of the DIE model was 48.25 dB, and the SSIM was 0.98, demonstrating excellent information concealment. In the actual measurement of architectural design enterprises, the file size growth rate of DIE model after information embedding was controlled at about 11.73%, and the document encryption speed was 28.85 MB/s. To sum up, the DIE model can embed a large amount of secret information in different files, and it is not easy to see the embedding trace, which can meet the needs of most enterprises. The research provides efficient and feasible technical solutions for the secure and covert transmission and storage of sensitive data in enterprises, such as design drawings and financial information. It also provides important practical references for the application of DIE technology in other fields that require high-intensity data protection, such as finance and technology.

6 REFERENCES

[1] Shen, Y. (2024). The impact of investor interest protection on corporate innovation efficiency. *Finance Research Letters*, *62*(1), 105106-105107. https://doi.org/10.1016/j.frl.2024.105106

[2] Wang, Y., Yu, M., & Gao, S. (2022). Gender diversity and financial statement fraud. *Journal of Accounting and Public Policy*, *41*(2), 106903-106904. https://doi.org/10.1016/j.jaccpubpol.2021.106903

[3] Ahn, B. (2024). Implementation of multimedia search and management system based on remote education. *Computer Science and Information Systems*, *21*(2), 419-436. https://doi.org/10.2298/CSIS220509007A

[4] Moon, J., Son, M., Oh, B., Jin, J., & Shin, Y. (2024). Automatic voltage stabilization system for substation using deep learning. *Computer Science and Information Systems*, *21*(2), 437-452. https://doi.org/10.1007/978-981-19-4132-0_14

[5] Damjanović, S., Katanić, P., Zavadskas, E. K., Stević, Ž., Krsmanović, B., & Djalić, N. (2024). Novel fuzzy MCDM model for comparison of programming languages. *Studies in Informatics and Control*, *33*(4), 5-14. https://doi.org/10.24846/v33i4y202401

[6] Ribeiro, J., Santos, R., Analide, C., & Silva, F. (2024). Implementing federated learning and explainability techniques in regression models to increase transparency and reliability. *Studies in Informatics and Control*, *33*(4), 15-24. https://doi.org/10.24846/v33i4y202402

[7] Bernett, J., Blumenthal, D. B., Grimm, D. G., Haselbeck, F., Joeres, R., Kalinina, O. V., & List, M. (2024). Guiding questions to avoid data leakage in biological machine learning applications. *Nature Methods*, *21*(8), 1444-1453. https://doi.org/10.1038/s41592-024-02362-y

[8] Li, M. G., Zhang, Z., Lu, M., Jia, X., Liu, R., Zhou, X., & Zhang, Y. (2023). Internet financial credit risk assessment with sliding window and attention mechanism LSTM model. *Tehnicki Vjesnik-Technical Gazette*, *30*(1), 1-7. https://doi.org/10.17559/TV-20221110173532

[9] Avci, S. & Yildirim, M. (2023). Solving weapon-target assignment problem with salp swarm algorithm. *Tehnicki Vjesnik-Technical Gazette*, *30*(1), 17-23. https://doi.org/10.17559/TV-20220113192727

[10] Chen, S., Li, G., Chang, K., Hu, X., Li, P., & Wang, Y. (2024). Ultra-short-term load forecasting based on XGBoost-BiGRU. *International Journal of Computers Communications & Control*, *19*(5), 6631-6632. https://doi.org/10.15837/ijccc.2024.5.6631

[11] Wang, L., Banerjee, S., Cao, Y., Mou, J., & Sun, B. (2024). A new self-embedding digital watermarking encryption scheme. *Nonlinear Dynamics*, *112*(10), 8637-8652. https://doi.org/10.1007/s11071-024-09521-y

[12] Wang, F., Fu, Z., & Zhang, X. (2024). A self-defense copyright protection scheme for NFT image art based on information embedding. *ACM Transactions on Multimedia Computing, Communications and Applications*, *21*(2), 1-23. https://doi.org/10.1145/3652519

[13] Yu, J., Zhang, X., Xu, Y., & Zhang, J. (2023). Cross: Diffusion model makes controllable, robust and secure image steganography. *Advances in Neural Information Processing Systems*, *36*, 80730-80743. https://doi.org/10.52202/075280-3539

[14] Cao, Y., Wei, W., & Zhou, J. (2022). Privacy protection data mining algorithm in blockchain based on decision tree classification. *SAGE Publications*, *20*(2), 103-112. https://doi.org/10.3233/WEB-210485

[15] Jiang, M. (2023). Reversible data hiding algorithm in encrypted images using adaptive total variation and cross-cyclic shift. *International Journal of Autonomous and Adaptive Communications Systems*, *16*(6), 611-631. https://doi.org/10.1504/IJAACS.2023.134851

[16] Singla, P., Garg, H., Pathak, A., & Singh, S. P. (2024). Privacy enhancement in Internet of Things (IoT) via mRMR for prevention and avoidance of data leakage. *Computers and Electrical Engineering*, *116*, 109151-109152. https://doi.org/10.1016/j.compeleceng.2024.109151

[17] Gupta, I., Singh, A. K., Lee, C. N., & Buyya, R. (2022). Secure data storage and sharing techniques for data protection in cloud environments: A systematic review, analysis, and future directions. *IEEE Access*, *10*, 71247-71277. https://doi.org/10.1109/ACCESS.2022.3188110

[18] Liu, Z., Gao, Y., Du, Q., Chen, M., & Lv, W. (2023). YOLO-extract: Improved YOLOv5 for aircraft object detection in remote sensing images. *IEEE Access*, *11*, 1742-1751. https://doi.org/10.1109/ACCESS.2023.3233964

[19] Zhang, M. J., Wang, D., Wu, H., Li, Y., & Xiang, Z. (2023). Multi-view contrastive learning for multilayer network embedding. *Journal of Computational Science*, *67*, 101975-101976. https://doi.org/10.1016/j.jocs.2023.101975

[20] Ren, S., Zhang, Y., Hang, J., & Lin, X. (2023). Hand-object information embedded dexterous grasping generation. *Pattern Recognition Letters*, *174*, 130-136. https://doi.org/10.1016/j.patrec.2023.09.006

[21] Li, S., Guo, H., Tang, X., Tang, R., Hou, L., Li, R., & Zhang, R. (2024). Embedding compression in recommender systems: A survey. *ACM Computing Surveys*, *56*(5), 1-21. https://doi.org/10.1145/3637841

[22] Mihalca, V. O., Moldovan, O., Țarcă, I., Anton, D., & Noje, D. (2024). Integrating deep learning in target tracking applications, as enabler of control systems. *International Journal of Computers Communications & Control*, *19*(6), 6854-6855. https://doi.org/10.15837/ijccc.2024.6.6854

[23] Zhang, X. (2024). A more secure framework for open government data sharing based on federated learning. *Government Information Quarterly*, *41*(4), 101981. https://doi.org/10.1016/j.giq.2024.101981

[24] Zhao, W., Liu, Y., Zhang, J., Shao, Y., & Shu, J. (2022). Automatic pixel-level crack detection and evaluation of concrete structures using deep learning. *Structural Control and Health Monitoring*, *29*(8), e2981-e2982. https://doi.org/10.1002/stc.2981

[25] Buttow, C. V. & Weerts, S. (2024). Managing public sector data: National challenges in the context of the European Union's new data governance models. *Information Polity*, *29*(3), 261-276. https://doi.org/10.3233/IP-230003

[26] Ye, W., Ren, J., Zhang, A. A., & Lu, C. (2023). Automatic pixel-level crack detection with multi-scale feature fusion for slab tracks. *Computer-Aided Civil and Infrastructure Engineering*, *38*(18), 2648-2665. https://doi.org/10.1111/mice.12984

[27] Fan, Q., Bi, Y., Xue, B., & Zhang, M. (2022). Genetic programming for feature extraction and construction in image classification. *Applied Soft Computing*, 118, 108509-108510. https://doi.org/10.1016/j.asoc.2022.108509

[28] Ahmadianfar, I., Heidari, A. A., Noshadian, S., Chen, H., & Gandomi, A. H. (2022). INFO: An efficient optimization algorithm based on weighted mean of vectors. *Expert Systems with Applications*, *195*, 116516-116517. https://doi.org/10.1016/j.eswa.2022.116516

Contact information:

Qiu RAN
Sichuan Vocational and Technical College,
Suining, Sichuan, 629000, China
E-mail: sweety3655@126.com

Hongyan ZHU
(Corresponding author)
Sichuan Vocational and Technical College,
Suining, Sichuan, 629000, China
E-mail: Zxiaoxiao068@126.com

APPENDIX

The mean filling calculation is shown in Eq. (14).

$$\mu = \frac{1}{A}\sum_{a=1}^{A} D_a \tag{14}$$

In Eq. (14), μ represents the mean value of the dataset. A stands for the total amount of data. D_a stands for the a-th data record. The mean filling calculation is shown in Eq. (15).

$$\text{med}(U) = \begin{cases} U_{(A+1)/2} & A\text{=odd} \\ \dfrac{U_{A/2} + U_{A/2+1}}{2} & A\text{=even} \end{cases} \tag{15}$$

In Eq. (15), med(U) represents the median of the dataset. U stands for the dataset. This step aims to improve the quality of data, ensure that only accurate and effective data enter the subsequent encryption link, and reduce unnecessary encryption and transmission burden.