Peer-to-Peer Networking and Applications (2025) 18:0
https://doi.org/10.1007/s12083-024-01850-4

Multimodal data privacy protection and completeness verification method for mobile crowd sensing

Jian Wang[1] · Fanfan Meng[1] · Jia Liu[1] · Guanzhi He[2] · Guosheng Zhao[3]

Received: 16 August 2023 / Accepted: 26 September 2024

Abstract
Most privacy-preserving approaches for mobile crowd sensing systems consider the privacy of single-modal data, while the data that sensing equipment may be sensing is frequently multimodal. Therefore, this paper proposes a multimodal data privacy protection and completeness verification method for mobile crowd sensing. Firstly, a cross-attention mechanism is utilized to fusion multimodal data to facilitate access to data information across various modes, improving the accuracy and reliability of data encryption. Secondly, a scalable superincreasing sequence is applied to store the multimodal data gathered by each sensing user, and the multimodal data is encrypted using an upgraded Paillier algorithm to prevent malicious attackers from obtaining the data information. Then, each ciphertext provides a validator using the Boneh-Lynn-Shacham signature algorithm. The sensing platform can apply a verification code to validate the completeness of the aggregated ciphertext data to ensure that the encrypted multimodal data has not been changed before being decrypted. Finally, experimental results demonstrate that the method proposed in this article not only effectively protects the privacy of multimodal data but also minimizes Communication costs and computational overhead.

Keyword Mobile crowd sensing; multimodal data; Paillier encryption algorithm; data completeness

1 Introduction

Mobile Crowd Sensing (MCS) is an emerging sensing paradigm with significant flexibility, scalability, and application advantages over traditional wireless sensor networks [1]. Due to the prevalence of mobile devices and wireless communication technologies (e.g., 4G/5G), MCS has become widely used in many different kinds of IoT applications, such as environmental monitoring [2], path planning [3], healthcare [4] and social services [5]. The IoT is expanding rapidly with the Internet's transition from IPv4 to IPv6 and WiFi networks. ABI Research predicts that by 2026, more than 3 billion IoT devices will be connected to the Internet worldwide [6]. This trend indicates that MCS systems continue to be a research focus and are utilized in many IoT devices.

Despite the evident benefits of the MCS, as the use of data continues to increase and the volume of data continues to grow, the risks of data leakage and misuse are growing. The issue of privacy preservation has been a primary focus of research in the field of MCS. It has attracted considerable interest from academia and industry. Sensing users' sensing data contains many sensitive data, such as daily life, location, social relationships, and personal health [7], which may result in privacy leakage. When data collection is performed at the sensing layer, it is possible for malicious data to be artificially injected or for the collected data to be tampered with, consequently affecting the reliability and security of subsequent data analysis and processing. The main security threats when transmitting data at the network layer are data leakage, tampering, and channel measurement attacks. Users tend to delegate data storage to third-party cloud servers when storing data at the platform level, which makes it difficult to ensure data confidentiality and can lead to data leakage and misuse. When data processing is performed at the

✉ Jian Wang
wangjianlydia@163.com

✉ Guanzhi He
hit_hgz_1206@163.com

[1] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China

[2] College of Bioinformatics, Science and Technology, Harbin Medical University, Harbin 150081, China

[3] College of Computer Science and Information Engineering, Harbin Normal University, Harbin 150025, China

Published online: 09 December 2024

application layer, users frequently migrate between shared resources, resulting in indistinct private data security boundaries and malevolent tenants traversing virtual boundaries to access personal data unlawfully [8].

Traditional MCS systems primarily protect the privacy of single-modal data, whereas sensing devices' sense data is typically multimodal (images, text, audio, etc.). The research emphasis is shifting from single-mode to multimodal data, as shown in Fig. 1. Single-mode data are of a single type and have poor data correlation. In contrast, multimodal data are complex and diverse, correlate highly, and contain much information. Encryption operations can ensure the completeness of the data. By mining the relationship between various modalities [9], multimodal data can more effectively mine the value of multiple data. Multimodal data can also provide multi-perspective [10] content that is closely related to the task context of the user from various perspectives [11]. Multimodal data privacy can better assist users in discovering global associations, changes, and patterns in data. However, the leakage of multimodal data can have a more significant impact, as the data are represented differently in various modalities, and there is a "semantic gap" between modalities, making it challenging to measure the similarity of multimodal data explicitly [12].

This paper designs a Paillier encryption algorithm to verify the completeness of multimodal data in the MCS system to address the above-mentioned issues. The method fusion of multimodal data using a cross-attention mechanism facilitates the mining of relationships between specific modal data. To protect the privacy of multimodal data, a scalable superincreasing sequence is used to store the multimodal data collected by each sensing user, and an enhanced Paillier algorithm is used to encrypt the multimodal data. The Boneh-Lynn-Shacham (BLS) signature algorithm is also used to generate a verifier for each ciphertext to verify the completeness of the encrypted data and ensure that the multimodal data information that has been encrypted is not compromised. The significant contributions of this paper are summarized as follows:

(1) Leverage multimodal data in MCS for data privacy assessment. Cross-attention is used for feature extraction and fusion of multimodal data, which facilitates the mining of correlation information between various modal data to improve the accuracy and dependability of data encryption.
(2) The scalable superincreasing sequence is used to store the multimodal data collected by each sensing user, and the improved Paillier algorithm encrypts the multimodal data to ensure that attackers cannot eavesdrop on the multimodal data information in the MCS system, thereby enhancing the data's security.
(3) Using the BLS signature algorithm to generate a verifier for each ciphertext, the sensing platform can check the completeness of the aggregated ciphertext data by the verification code, ensuring that any encrypted multimodal data will not be replaced, tampered with, or destroyed before decryption and ensuring the data completeness.

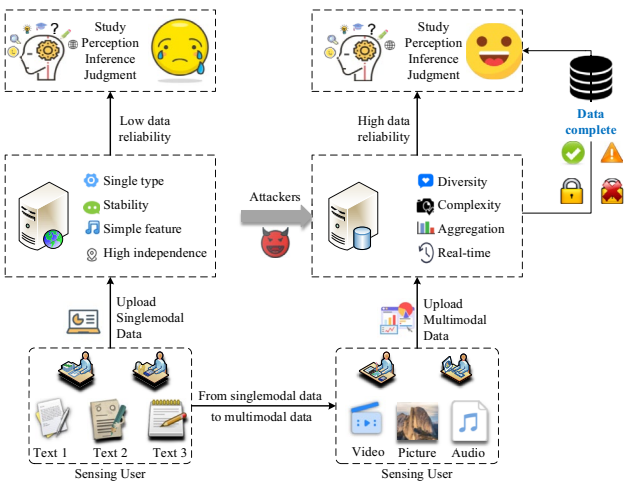


Fig. 1 Comparison of unimodal data and multimodal data

2 Related work

Privacy protection has been the central issue of data mining and extensive data analysis in MCS systems, causing pervasive concern. In this section, privacy protection mechanisms in MCS systems are briefly described in terms of the perceived users' identity privacy, location privacy, and data privacy.

Identity information is sensitive and closely related to the privacy of the user. Qian et al. [13] segmented user information using anonymization techniques to safeguard the identity privacy of sensing users. During participation in the task, the user submits multiple location information using virtual location methods to confound the attacker's estimate of the user's valid location. Nkenyereye et al. [14] employed id-based signatures and pseudonym techniques to assure entity authentication and confidentiality. And prevent unauthenticated data proprietors from leveraging network resources using information center network methods to protect identity information. During the anonymization procedure, the original data may be modified or deleted, reducing the availability of the data. Therefore, some of the studies use encryption techniques. Xiao et al. [15] designed lightweight security protocols employing secret sharing schemes that do not rely on encryption and decryption operations and any trusted third party, rendering the user's private data inaccessible to the platform and other users. Arulprakash et al.

[16] implemented signatures using identity-based encryption, which addresses the certificate administration problem in conventional public key signature schemes. However, the encryption technology's key can readily disclose the signer's identity information. Some researchers have considered spatial and temporal factors in light of the pervasive use of information. Liu et al. [17] proposed a dynamic clustering-based spatiotemporal privacy-preserving approach to anonymize spatiotemporally sensitive data using dynamic *k*-anonymity and *l*-diversity methods to address worker privacy concerns. Zhang et al. [18] proposed a caching and spatial *k*-anonymity-based user privacy protection scheme. The solution mitigates the possibility of information leakage through multilevel caching and Markov models. Existing identity privacy protection methods frequently lack user control over their identity information, making it simple for personal data to be disclosed or exploited. The solution mitigates the possibility of information leakage through multilevel caching and Markov models.

Location-based services are required for the MCS system. Liu et al. [19] proposed a quality-of-service controlled privacy-preserving scheme for location *k*-anonymity, which combines a greedy strategy with a location-full measurement mechanism to generate *k*-anonymity sets that are resistant to background knowledge inference attacks, and then selects a perturbed location to complete the anonymity. Zhang et al. [20] mediated using an edge server between the user and the LBS server. Reduces burden on user devices by employing multilevel caching and protects location privacy through dual anonymity, but is susceptible to sensitive data leakage during caching. Zhang et al. [21] proposed a location privacy protection method based on local differential privacy, which constructs a task mapping based on the Voronoi diagram according to the task location, and maps each task location to an area to hide the task location. Build a local coordinate system within the task area, recalculate and encode the relative position coordinates of all workers in the area, and then perturb the encoding using local differential privacy to ensure worker position privacy. Although this method performs well in the privacy protection of local location information, it may not fully protect other sensitive information related to location. Zhang et al. [22] protected the location privacy of sensing users using homomorphic encryption and circle-based location verification. Only the employee's grid is displayed for multilevel privacy protection for employee transportation locations. It also employs order-preserving encryption and non-interactive zero-knowledge proofs to prevent employees from fraudulently obtaining rewards by misrepresenting their driving locations. Zou et al. [23] proposed a new decentralized crowdsensing system, which adopts a hybrid blockchain architecture and uses smart contracts to achieve location privacy preservation and ensure data quality while improving the system performance. Wang et al. [24] proposed a novel location obfuscation mechanism that solves the location privacy issue of sensing cars by combining ϵ-differential privacy with δ-differential privacy in sparse group-wise sensing systems. Li et al. [25] proposed a method for differential data aggregation based on worker partitioning and location obfuscation. The optimal clustering of workers was accomplished by enhancing the *k*-means algorithm and designating each group of workers with a unique privacy budget. In the meantime, using differential privacy mechanisms to interfere with employees' location information before submitting data to third-party platforms can result in location information errors. Although these studies have made progress in protecting location privacy in MCS systems, there are still issues, such as low data validity and availability and inadequate defense against attacks.

In the MCS system, data information contains a great deal of sensitive user information, and improper use of data information can result in the disclosure of such information. Zhang et al. [26] proposed an efficient and strong privacy-preserving truth discovery scheme, which first exploits the randomizable matrix to express users' tasks and sensory data, then based on the matrix computation properties, designed key derivation and encryption mechanisms to enable truth discovery to be performed in an efficient and privacy-preserving manner. Zheng et al. [27] proposed a top-k query scheme for vertically distributed data that protects privacy. Utilizes homomorphic encryption technology to safeguard sensitive data, such as data from each data source and scoring functions. In most extant investigations, the data store and proprietor are separated. Xiong et al. [28] segmented the sensitive data and proposed a decentralized privacy protection framework for Shamir's secret sharing. Because each sharp data point is divided into shares and only a sufficient number of stakes are aggregated to recover the data, this method accomplishes effective truth discovery while remaining resistant to collusion assaults. Liu et al. [29] proposed a reliability-enhanced, privacy-protecting truth discovery scheme that considers dynamic user changes. A multi-client inner product function encryption is designed to identify outliers in user-submitted encrypted data through a new filtering method, thereby removing the interference of outliers with the reliability of true value discovery. There are also data privacy violations during the process of data aggregation, Li et al. [30] proposed a differential privacy truth discovery mechanism for data aggregation in MCS systems, with the central concept of independently perturbing each user's data and then weighting the user's perturbed data for collection. Even if a significant amount of noise is introduced, the aggregation results do not deviate significantly, ensuring the accuracy and confidentiality of the data. Consider the existence of redundant data and increased communication costs. Lin et al. [31] proposed a secure deduplication scheme for distributed mobile edge nodes that facilitates

secure deduplication with the assistance of other nodes to prevent the collection of undesirable data if the local edge node is unavailable. Although the above methods are beneficial for data privacy protection, the diversity of perceived data can make data privacy protection more complicated.

Table 1 summarizes the approaches to identity privacy, location privacy, and data privacy. This paper focuses primarily on the privacy protection of sensory data and proposes a Paillier encryption algorithm that can verify the authenticity of multimodal data. This method effectively integrates multimodal data and ensures their integrity. It can encrypt multimodal data, prevent multimodal information leakage, and verify the integrity of encrypted data to ensure that encrypted multimodal data will not be altered or replaced before decrypted.

3 Problem descriptions

Sensing users need to request and collect sensing data from various locations, interact with cloud platforms, and share and transmit information. This process involves a large amount of user data, which may contain private data that users do not want to disclose. For example, uploading location information contained in sensing data may reveal users' daily movement trajectories, and sensing tasks related to gender and age may expose users' private information. More and more users are concerned about the leakage of their privacy data, and these security issues will reduce the desire of users to participate. As a solution to this problem, Paillier encryption technology is provided to confirm the integrity of multimodal data, prevent malicious access to user privacy data, and ensure that sensing users can engage in sensing tasks with a certain degree of privacy protection.

The system comprises four entities: the requester, the sensing user, the cloud server, and the sensing platform. Figure 2 depicts the system model. When this occurs, the requester submits a request for the appropriate sensing job according to the system specifications. To guarantee the completeness and dependability of the encrypted data, the sensing user gathers each sensing data following the specifics of the sensing job and then conducts feature

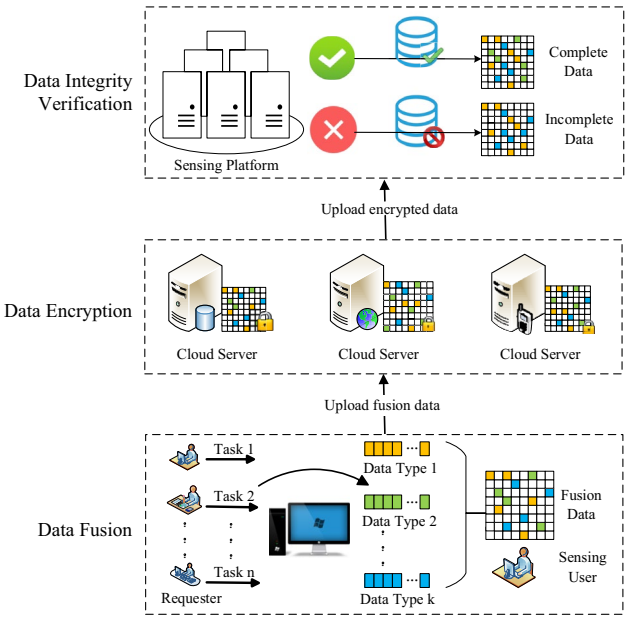


Fig. 2 System model figure

extraction and fusion on the collected multimodal sensing data. The cloud server encrypts the combined multimodal sensor data to avoid data leaking. The cloud server combines the encrypted multimodal sensory input simultaneously to increase data processing efficiency, resulting in an aggregated ciphertext that is more comprehensive and integrated. The sensing platform must verify the completeness of the encrypted data externally after receiving the aggregated encrypted data given by the cloud server. Decryption is carried out if the encrypted data is complete when restoring encrypted data to ensure data availability. Instead, the cloud server must resent the encrypted sensory data information.

This research considers an MCS system with a sensing platform and i sensing users. Multimodal data fusion, initialization, encryption and aggregation, completeness verification, and decryption comprise most of the system's operational procedure. The sensing user primarily employs a bottom-up attention model to extract picture region features and a BERT model to extract text region

Table 1 Classification of privacy protection methods

	Literature	Existence of problems	Threat Model
Identity Privacy	[13–15], [16–18]	Lack of control by the user over their identity information	Witch attack, sensing task association attack, background knowledge attack, etc
Location Privacy	[19–22], [23–25]	Unrestricted access to and utilization of location data	Collusion attacks, location tracking attacks, member inference attacks, etc
Data Privacy	[26–28], [29–31]	Unreliable and inaccurate unimodal data	False data attacks, sensing task association attacks, background knowledge attacks, etc

features during the multimodal data processing stage. By simultaneously modeling the inter-modal and intra-modal interactions of picture areas and sentence words in a single deep model, the cross-attention mechanism is then utilized to accomplish image and sentence matching. A trustworthy third party must choose the public security parameters κ and communicate the critical parameters to the sensing platform and sensing user $User_i(i=1,2,\ldots,n)$ during the data initialization phase to create a secure system. A superincreasing sequence is also implemented to guarantee that users of the sensing platform may upload various forms of sensing data. We primarily encrypt the k kinds of data $m_{i1},m_{i2},\ldots,m_{ik}$ and produce the ciphertext CT_i during the data encryption and aggregation. The BLS signature procedure creates the ciphertext verification code σ_i to guarantee the encrypted data's completeness further. The cloud server may create an aggregated authentication code σ for all ciphertexts and combine all forms of encrypted data into a single ciphertext CT. The BLS signature technique is homomorphic, and the sensing platform is flexible enough to validate the data completeness of encrypted multimodal data throughout the completeness verification and decryption stages. If the validation fails, the data must be resent because it is invalid (has been changed or tampered with). The sensing platform may utilize the private key γ_0 to get the aggregated multi-type data and obtain the decrypted data if the verification formula is valid.

In this part, we first provide an overview of the algorithm's system model before going into more depth on the system model's operation. Table 2 provides an overview of the key symbols used in this essay for your convenience.

Table 2 Parameter descriptions

Parameter	Description
$User_i$	Sensing users
G_1,G_2	Multiplicative cyclic group
ρ	Generating element of G_1
q_1,q_2,p	Three prime numbers
κ	Safety parameters
H	Hash function
prg	Random Number Generator
CT_i	Ciphertext
CT	Aggregate ciphertext
σ_i	Verification Code
σ	Aggregation verification code
ξ	Random blind value
$nonce$	Serial number
$RAID$	Identifiers for specific sensing users

4 Paillier encryption algorithm for verifying the multimodal data completeness

4.1 Multimodal data processing

Numerous sensory data sets, complex and varied in various media forms (such as photos, video, text, etc.), are collected, stored, and sent by ubiquitous sensing devices (such as computers, mobile phones, etc.). Furthermore, in a connected and private society, it is crucial to protect the privacy of personal data since multimodal data includes sensitive personal information about the perceived user. Therefore, multimodal data must be processed to prevent the disclosure of sensitive user information from increasing user trust and preserving platform credibility.

4.1.1 Multimodal data extraction

During multimodal data processing, a given image I is fed into a bottom-up attention model pre-trained on the visual genome [32] to extract image region features and output a set of image region features $O=\{o_1,o_2,\ldots,o_k\}$, which o_i denotes the convolutional parts of the ith region after averaging the image's pooling layer. The image feature extraction pre-trained model is not engaged in parameter updates during model training, and a fully-connected layer must be added to convert the image features to meet the job's demands. The retrieved features are features changed through the fully connected layer, with the transformed features denoted as $R=\{r_1,r_2,\ldots,r_k\}$, where r_i corresponds to the transformed features of o_i.

The Word-Piece of phrase T is utilized as a text fragment and the words in the sentence are separated according to a particular pattern. Each word's final embedding is a mixture of its token, positional, and segmental embeddings, designated as $X=\{x_1,x_2,\ldots x_n\}$. A pre-trained BERT model (bi-directional encoder representation from the Transformer) is fed the pass $X=\{x_1,x_2,\ldots x_n\}$ of the phrase T [33]. The BERT comprises many Transformer units, the output $E=\{e_1,e_2,\ldots e_n\}$ of which contains intra-modal information.

4.1.2 Multimodal data fusion

Due to the vast visual-semantic differences between vision and language, matching images and sentences remains unresolved. As shown in Fig. 3, a multimodal cross-attention mechanism is used to accomplish image and sentence matching by concurrently modeling the inter and intra-modal relationships between image regions and sentence words in a single-depth model. $Y=[R;E]=\{r_1,r_2,\ldots r_k;e_1,e_2,\ldots e_n\}$ is utilized as input to achieve the robust cross-modal matching,

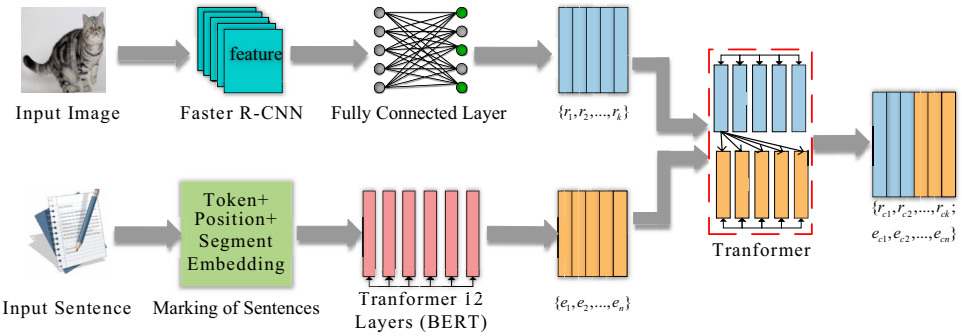


Fig. 3 Multimodal data processing

where $Y\bullet \in R^{(k+n)\times d_x}$. Y is transferred to a different Transformer unit. The attention module can be characterized as a mapping from a query to a set of key-value pairs to the output. The output of the attention function is a weighted sum of values, where the query and its corresponding keys determine the weight matrix. Attention is implemented using the Transformer [34], which consists of two sublayers: a multi-headed self-attention sublayer and a location-based feedforward network. Concentration is computed h times in the multi-headed self-attention sublayer to make it multi-headed. This is accomplished by projecting queries, keys, and values h times using distinct linear projections that can be learned. Given a set of fragments, precisely compute the input query $K_Y = YW^K$, key $Q_Y = YW^Q$, and value $V_Y = YW^V$. The weight matrix can then be obtained by scaling the attention mechanism for the dot product. In addition, the following formula is used to calculate the weighted sum of this value.

$$Attention(Q_Y, K_Y, V_Y) = softmax(\frac{Q_Y K_Y^T}{\sqrt{d}})V_Y \tag{1}$$

To simplify and clarify the derivation, the softmax and scaled dot product attention functions are eliminated from the preceding equation to obtain $Q_Y \bullet K_Y^T \bullet V_Y$. This has no bearing on the central concept of the attention mechanism, nor does it affect the experimental results.

$[R_{up};E_{up}] = Q_Y \bullet K_Y^T \bullet V_Y$ is known, so the image and text fragments are updated accordingly.

$$R_{up} = \{r_{up1}; r_{up2}; \ldots; r_{upk}\} = Q_R \bullet K_R^T \bullet V_R \bullet + Q_R \bullet K_E^T \bullet V_E \tag{2}$$

$$E_{up} = \{e_{up1}; e_{up2}; \bullet\bullet\bullet e_{upk}\} = Q_E \bullet K_E^T \bullet V_R + Q_E \bullet K_R^T \bullet V_R \tag{3}$$

This Transformer unit's multi-headed self-attentive sublayer's output considers inter and intra-modal relations. The $[R_{up};E_{up}]$ is then transmitted to the position feedforward sublayer. The result of the Transformer unit is finally acquired in the cross-notice module and represented by the letter $Y_c = [R_c;E_c] = \{r_{c1}, r_{c2}, \ldots, r_{ck}; e_{c1}, e_{c2}, \ldots, e_{cn}\}$. The method enables a compelling fusion of image and text features to reduce the disparities between distinct modal data, allowing heterogeneous data to be fused in a common semantic space while reducing the consumption of unnecessary computational resources.

4.2 Improved paillier encryption algorithm

The processed multimodal data can effectively integrate information and provide more reliable and superior data for data encryption. In the study of data privacy protection, the completeness of the data is just as important as ensuring that the data information of perceived users is not compromised and any encrypted multi-type data can be inadvertently replaced or obliterated. A Paillier encryption algorithm that can verify the completeness of multimodal data is proposed for this purpose.

4.2.1 Data initialization

The data must be initialized to establish a secure MCS system. The trusted third party must identify the public parameters and transmit the key parameters to the sensing platform and user. For the security parameter κ, three prime integers q_1, q_2, and p must be selected to calculate the Paillier encryption regime's public key $N = q_1 q_2$, $g = 1 + N$, and matching private key λ [35]. A trusted third party determines the bilinear pair mapping $e: G_1 \times G_1 \rightarrow G_2$, where G_1 and G_2 are multiplicative cyclic groups of order p and ρ is the generating element of G_1. A trustworthy third party selects v at random from group G_1. Four anti-collision hash functions $H: \{0,1\}^* \rightarrow G_1$, $h_1: \{0,1\}^* \rightarrow Z^* N$, $h_2: Z^* N \rightarrow Z_p$, $h_2: Z_{N^2}^* \times \{0,1\}^* \rightarrow Z_p$ are put up by trusted third parties. A reliable third party chooses $\gamma_1, \gamma_2, \cdots, \gamma_n$ at random from $Z^* N$ and computes γ_o.

$$\gamma_o + \sum_{i=1}^{n} \gamma_i = 0 (mod \lambda) \tag{4}$$

If the attacker acquires data from some sensing users, the attacker cannot gain data from other sensing users.

Because all key parameters are chosen at random by trusted third parties and are independent of one another, obtaining the key parameters of one or a few sensing users does not recover the key parameters of other sensing users. In the extreme case, an attacker may compromise the key parameter of $n-1$ sensing users, such $\gamma_1, \gamma_2, ..., \gamma_n$. Since$\gamma_o + \gamma_n + \sum_{i=1}^{n-1} \gamma_i = 0(mod\lambda)$, without any information from γ_n andλ, the attacker can still not recoverγ_0. As a result, an attacker cannot compromise additional sensing user information that is not compromised, regardless of how much sensing user information is compromised.

Then, a trusted third party computes $\beta_1 = \rho^{h2(\gamma_1)}, \cdots, \beta_n = \rho^{h_2(\gamma_n)}, \gamma' = -\gamma_o(mod\lambda)$ and corrupts the Paillier cryptosystem's private key.

A trusted third party generates a superincreasing sequence $\{\omega_1, \cdots, \omega_k, \omega_{k+1} \cdots, \omega_{2k}\}$ consisting of *2 k* positive integers so that users of the sensing platform can submit multiple data types simultaneously. These coefficients must meet these constraints.

$$\omega_\alpha > \sum_{i=1}^{\alpha-1} (\omega_j n_j n) \tag{5}$$

where $\omega_1 = 1$, $\alpha = 2,3, \cdots, 2k, n_j$ is an upper limit for the jth data type, and $j = 1,2, \cdots, k$ Therefore,$n_{k+1} = n_1^2, n_{k+1} = n_2^2, \cdots n_{2k} = n_k^2$

The trusted third party transmits $(, y')$ to the sensing platform through a secure channel,y_i to the user of the sensing platform, and the system parameter $\Omega = (N, \mathrm{g}, e, G_1, G_2, \rho, H, h_1, h_2 h_3, v, \{\beta_i\}_{1 \le i \le n}, \{\omega_\alpha\}_{1 \le \alpha \le 2k})$ is made public.

The sensing platform sets a pseudo-random number generator *prg* and $SK_{prg} \times I \rightarrow Z_p^{n-1}$ to ensure data completeness verification, where SK_{prg} denotes the set of keys for *prg* and *i* denotes the set of sequence numbers, and then randomly selects a key $sk_{prg} \epsilon SK_{prg}$, which is shared secretly by the sensing platform and will not be disclosed to other users to protect data privacy.

4.2.2 Data encryption and aggregation

Each sensing user possesses *k* variety of sensing data, each containing a substantial quantity of information readily accessible to assailants. Consequently, the current system timestamp *T* is derived, and the k-type data m_{i1},m_{i2},...,m_{ik} is encrypted, as illustrated in Algorithm 1. To generate the ciphertext data, the following steps are taken.

$$CT_i = \mathrm{g}^{\sum_{j=1}^{k} (\omega_j m_{ij} + \omega_{k+j+j} m_{ij}^2)} \bullet h_1(T)^{N_{\gamma i}} (modN^2) \tag{6}$$

Each CT_i is a Paillier ciphertext, whereas each $h_1(T)^{N\gamma i}$ is a random number in the original Paillier algorithm. According to Li et al. [36] analysis, even if the attacker eavesdrops on the communication between the sensing user and the server, only ciphertext information is captured, and semantic security is achieved for the chosen plaintext attack, so the attacker cannot recover any data information from the sensing user.

In addition, the BLS signature algorithm generates the ciphertext verification code to ensure the encrypted data's completeness.

$$\sigma_i = (H(att_i || T) \bullet V^{CT_i})^{h_2(\gamma_i)} \tag{7}$$

where $atti = RAID||i$ is a specific string, and *RAID* is an identifier for a particular user sensing. Zero-knowledge proof can derive the exponential form of σ_i with $H(att_i||T) \bullet V^{CT_i}$ as the base. Since the bottom σ_i varies at different times, $h_2(\gamma_i)$ is unknown, thus severing the correlation between the verifier at other times and preserving the completeness of the encrypted data.

Because the Paillier cryptosystem is homomorphic, the cloud server can combine all forms of encrypted data into a particular ciphertext. First, after receiving $\{CT_i, \sigma_i, T\}, i = 1,2, \cdots, n$, the cloud server combines numerous ciphertexts from all perceiving users to produce the aggregated ciphertext.

$$CT = \prod_{i=1}^{n} \mathrm{CT}_i = \mathrm{g}^{\sum_{j=1}^{k} (\omega_j \sum_{i=1}^{n} m_{ij} + \omega_{k+j} \sum_{i=1}^{n} m_{ij}^2)} h_1(T)^{N \sum_{i=1}^{n} \gamma i} (modN^2) \tag{8}$$

In reality, all ciphertexts are combined into a particular ciphertext with the same format as the ciphertext of the Paillier cryptosystem. If an adversary compromises the perceiving user's data information, he can only obtain the encrypted data and the aggregated ciphertext. Due to the semantic security of the Paillier cryptosystem, the encrypted data does not escape into the plaintext, thereby protecting the user's data.

The cloud server and sensing platform share the pseudo-random number generator $pr\ g : SK_{prg} \times \rightarrow Z_p^{n-1}$ and the key SK_{prg}, allowing the cloud server to generate $(\tau_1, \tau_2, \cdots, \tau_{n-1}) \leftarrow prg(Sk_{prg} nonce)$. In addition to computing $\tau_n = h_3(h_1(T)^{N^2 \gamma'} CT^N \parallel nonce)$, the cloud server generates a random vector $(\tau_1 \tau_2, \cdots, \tau_{n-1}, \tau_n)$ based on the sequence number *nonce*. In conjunction with this random vector, the cloud server combines all ciphertexts of the authentication code into a single value $\sigma = \prod_{i=1}^{n} \sigma_i^{\tau i}$, and generates a random blind value $\xi = \prod_{i=1}^{n} \beta_i^{\tau_i CT_i}$. This key technology ensures that any encrypted multi-type data is not replaced, tampered with, or corrupted before decryption and that the cloud server multiplies all ciphertexts CT_i to obtain the aggregated ciphertext *CT* rather than arbitrarily returning the aggregated value with a random number.

Algorithm 1 Data encryption

Input: Timestamp T, k kind of data $m_{i1}, m_{i2}, \ldots m_{ik}$

Output: Aggregate ciphertext, Aggregation verification code

1. Calculate ciphertext data CT_i
2. Calculate the verification code of the ciphertext σ_i
3. Receive data information $\{CT_i, \sigma_i, T\}, i = 1, 2, \ldots, n$
4. Computing aggregate ciphertexts CT
5. Shared pseudo-random number generator prg and key SK_{prg}
6. Calculation $\tau_n = h_3(h_1(T)^{N^2\gamma'} CT^N \parallel nonce)$,
7. The verification code of the ciphertext is aggregated into $\sigma = \prod_{i=1}^{n} \sigma_i^{\tau_i}$

4.2.3 Data completeness verification and decryption

Using the homomorphic character of the BLS signature algorithm, the sensing platform can verify the completeness of encrypted multimodal data without information loss during the data completeness verification and decryption process. Due to superincreasing sequences, the sensing platform can also correctly decrypt the aggregated ciphertext and obtain the original information for data utilization.

When $\{CT, \sigma, \xi, T\}$ is received, the sensing platform generates the random vector $(\tau_1, \tau_2, \cdots, \tau_{n-1}) \epsilon Z_p^{n-1}$, employs the pseudo-random generator prg, and calculates $\tau_n = h_3(h_1(T)^{N^2\gamma} CT^N \parallel nonce)$. After that, completeness verification and aggregated data decryption are performed as follows:

$$e(\sigma, \rho) =^{?} (\prod_{i=1}^{n} e(H(att_i)^{\tau i}, \beta_i)) e(v, \xi) \tag{9}$$

Suppose the perpetrator transmits data information to the server instead of the sensing user and then generates fabricated aggregated data to bypass the completeness verification process. However, according to Eq. (9), ξ will alter, invalidating the equation. Assuming an adversary can manipulate some ciphertext and the corresponding authenticator, generating a forged aggregated message that will pass the completeness verification process is still possible. However, according to Eq. (9), ξ and σ will alter, invalidating the equation. Suppose an adversary replaces or tampers with the final aggregated ciphertext CT without replacing or tampering with any individual ciphertext CT_i, and can generate forgeries of aggregated messages that pass the completeness checking process. In that case, the completeness checking process is compromised. However, according to Eq. (9), τ_i will alter, rendering the equation invalid. An attacker cannot complete the sensing task by replacing some sensing user data, tampering with the ciphertext and the corresponding authenticator, fiddling with the aggregated ciphertext CT, and forging the aggregated information. If the verification fails, the data is invalid (replaced or tampered with), and the data is resent; if the equation holds, the sensing platform decrypts the data using the private key γ_o. The following are the precise execution steps.

$$W = CT \bullet h_1(T)^{N \bullet \gamma_o} = g^{\sum_{j=1}^{k} (\omega_j \sum_{i=ij}^{n} + \omega_{k+j} \sum_{i-1}^{n} m_{ij}^2)} (mod N^2)$$

$$W = CT \cdot h_1(T)^{N \cdot \gamma_0} = g^{\sum_{j=1}^{k} (\omega_j \sum_{i=1}^{n} m_{ij} + \omega_{k+j} \sum_{i=1}^{n} m_{ij}^2)} (\mathrm{mod} N^2) \tag{10}$$

Let $Q = \omega_1 \sum_{i=1}^{n} m_{i1} + \cdots + \omega_k \sum_{i=1}^{n} m_{ik} + \omega_{k+1} \sum_{i=1}^{n} m_{i1}^2 + \cdots + \omega_{2k} \sum_{i=1}^{n} m_{ik}^2$, then $W = g^Q (\mathrm{mod}\ N^2)$. Specifically.

$$(1+N)^Q = \sum_{l=0}^{Q} \binom{Q}{l} N^l = 1 + NQ + \binom{Q}{2} N^2 + higher\ power\ of\ N \tag{11}$$

This indicates that:

$$(1+N)^Q = (1+NQ)(\mathrm{mod} N^2) \tag{12}$$

When $W = (1+N)^Q (\mathrm{mod} N)^2$, the sensing platform can retrieve aggregated multi-type data based on the following condition.

$$Q = W - 1 / (\mathrm{mod} N^2)$$

The sensing platform then retrieves $\{M_1, M_2, \ldots, M_k\}$ and $\{QM_1, QM_2, \ldots, QM_k\}$ using Algorithm 2.

Algorithm 2 Decryption $\{QM_j\}, \{M_j\}, j = 1,2,...,k$

1. for j=k to 1 do
2. $QM_j = (Q - Q \bmod \omega_{k+j}) / \omega_{k+j}$
3. $Q = Q - (\omega_{k+j} \cdot QM_j)$
4. end for
5. for j=k to1do
6. $M_j = (Q - Q \bmod \omega_j) / \omega_j$
7. $Q = Q - (\omega_j \cdot M_j)$
8. end for
9. return $\{QM_j\}, \{M_j\}, j = 1,2,...,k$

5 Experiments and analysis

Experiments are primarily conducted on the Wikipedia dataset to validate the efficacy of the method presented in this paper. The experiments' pertinent parameters are discussed, and the performance is evaluated regarding computational overhead and communication costs.

5.1 Experimental setup

Experiments were conducted on an Intel Core i74710U CPU with a 2.5 GHz processor, 4 GB RAM, and Ubuntu 18.04 Linux, programmed in C with the GNU Multi-Precision Arithmetic Library (GMP), a library supporting mathematical operations for paired cryptosystems (PBC), and the Secure Sockets Layer cryptographic library (OpenSSL). The bilinear pairing utilized by the PBC library is built on the curve $y^2 = x^3 + x$ of a prime $p = 3(mod\ 4)$ divided by the field F_p. In addition, Table 3 displays the time required to execute the cryptographic operators.

The experiment employs the Wikipedia dataset [37], a multimodal dataset compiled from "featured articles" on the Wikipedia website commonly used in multimodal deep learning experiments. The dataset includes ten semantic categories: "art, biology, geography, history, literature, media, music, kingship, sports, and war." Each of the 2866 image-text combinations in the dataset is tagged with one of the ten semantic categories. The dataset contains 128-dimensional feature vectors for each image and 10-dimensional topic vectors for each text document. Where Table 4 provides a summary of the datasets utilized in the paper.

5.2 Parameter analysis

The image region feature vector extracted using the bottom-up attention mechanism in multimodal data extraction for image regions has 2048 dimensions, and a fully connected layer is added to convert it into a d-dimensional vector, which is then input to the Transformer unit. A BERT model with 12 self-attentive layers, 12 headings, and 768 concealed units per token has been pre-trained for the text data. During the training phase, the weights of the BERT model are specified for simplicity. The Transformer unit is utilized to establish inter and intra-modal connections.

Experiments are conducted on the Wikipedia dataset to examine the impact of spatial dimensionality on the method's efficacy. Figure 4 depicts the results of image sentence retrieval in various dimensions. Figure 4(a) demonstrates the effect of varying spatial dimensions on the return of textual information through images. In contrast, Fig. 4(b) reflects the impact of various spatial dimensions on retrieving image information through text. As shown in Fig. 4, the model's efficacy increases and decreases as the spatial dimension increases. When the concealed space dimension was set to 256, improved experimental results were obtained on the Wikipedia dataset. The findings indicate that larger sizes do not necessarily yield superior performance. This could be because larger concealed space dimensions make it more challenging to train the model. Therefore, it is necessary to select an appropriate medium-sized size for implementing the model presented in this paper to obtain improved results for future applications.

Much sensory data must be processed and analyzed in real-time. Still, due to the limited storage capacity of the sensory devices, the sensory data must be uploaded to the cloud server, where it is analyzed, mined, and applied using the mighty computing power and high-performance storage capacity of the cloud server. The communication cost is also affected by the key size set by the Paillier algorithm utilized in this paper. The communication costs generated by key lengths of 256 bits, 512 bits, 1024 bits, and 2048 bits will be contrasted to assess the effect of various key sizes on the

Table 3 Symbols of execution time

Symbols	Meaning	Running time
T^e	The computation time of bilinear pairing operations	0.0176 *s*
T^h	The computation time of the hash function	0.00061 *s*
T^m	The computation time of mapping to a point hash function	0.0155 *s*
$T^{\wedge}_{Zn}$	The computation time of the power operation in $Z^*_{n^2}$	0.0027 *s*
$T^{\wedge}_{G1}$	The computation time of the power operation in G_1	0.00162 *s*
$T^{\wedge}_{Zp}$	The computation time of the power operation in Z^*_p	0.00041 *s*
$T^{\times}_{G1}$	The computation time of multiplication operation in G_1	0.00062 *s*
$T^{\times}_{Zp}$	The computation time of multiplication operation in Z^*_p	0.00021 *s*
$T^{\times}_{Zn}$	The computation time of multiplication operation in $Z^*_{n^2}$	0.00083 *s*
T^{+}_{G1}	The computation time of the dot-add operation in G_1	0.003 *s*
$T^{s\times}_{G1}$	The computation time of scalar multiplication operation in G_1	0.0039 *s*

Table 4 Experimental dataset

Dataset	Test set size	Training set size	Category	Graphic feature dimension	Text feature dimension
Wikipedia	462	2173	10	4096	300

algorithm presented in this paper. Figure 5 displays, for different key lengths, the communication cost of the algorithm described in this paper. Figure 5 illustrates that as the key length increases, so does the communication cost. Paillier can support larger key sizes, such as 2048 bits and 4096 bits, to enhance security. In this paper, the intermediate data is set to 1024 bits for a more explicit comparison of communication costs, whereas Paillier can support larger key sizes, such as 2048 bits and 4096 bits. Under the premise of assuring security, however, the key length should be abridged as much as possible, and the space required to retain the key should be decreased to reduce the communication cost and save money.

The processes of data encryption, data signature, completeness verification, ciphertext aggregation, and decryption are outlined to assess the effect of varying numbers of perceived users on the algorithms presented in this paper. The astute user first encrypts the k species type data $m_{i1}, m_{i2}, \ldots, m_{ik}$ and generates the ciphertext CT_i. The BLS signature algorithm generates the ciphertext verification code σ_i to ensure the encrypted data's CT_i completeness. Compare the time it takes to construct the ciphertext (T_{m1}), the authentication code ($T_{\sigma 1}$), and the overall computing overhead (Sum_1) on the sensing user side for various numbers of sensing users in the data encryption process, assuming the number of sensing users is $i = 200$, 400, 600, 800, and 1000. Table 5 demonstrates that the time needed to create the ciphertext and the authentication code grows as the number of perceived users increases. Ciphertext and authentication codes are generated to assure the security and authenticity of data transmission and to effectively avoid issues like data tampering, theft, and forging. Therefore, the safety and dependability of data processing may be increased by lengthening the time to create ciphertext and authentication codes, which is crucial for protecting personal information and data security.

Secondly, after receiving $\{CT_i, \sigma_i, T\}, i = 1,2, \ldots, n$ from the sense user, the cloud server aggregates the ciphertext to create the final aggregated message $\{CT, \sigma, \xi, T\}$, where σ is the last aggregated authentication code and CT is the final aggregated ciphertext. Assume the number of sensing users $i = 200$, 400, 600, 800, 1000, and compare the time to create aggregation verification code ($T_\sigma 2$), aggregated ciphertext (T_{m2}), and overall computing overhead (Sum_2) on the server side for each number of sensing users. As shown in Table 6, the time required to aggregate ciphertext and aggregation verification code increases with the number of sensing users,

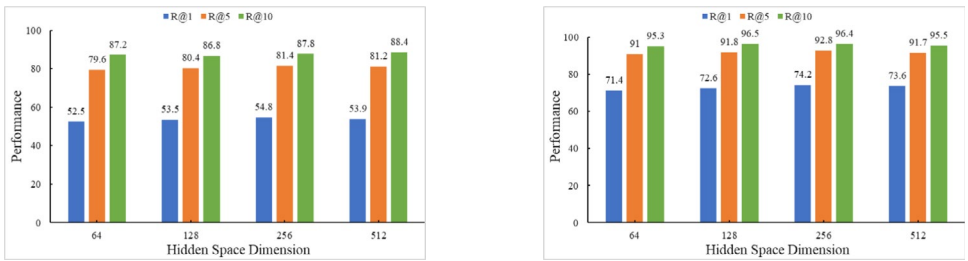


Fig. 4 The influence of hidden space dimension

(a) Image-Text

(b) Text-Image

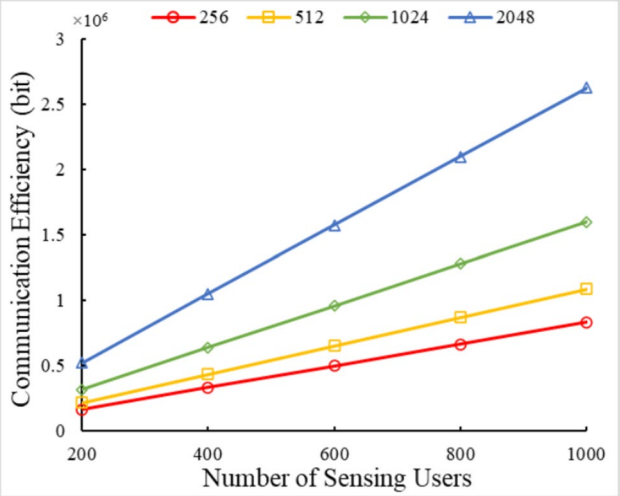


Fig. 5 Effect of key size on communication cost

Table 5 Computational overhead of data encryption process

i	T_{m1}	$T_{\sigma 1}$	Sum_1
200	1.08	0.648	1.728
400	2.16	1.296	3.456
600	3.24	1.944	5.184
800	4.32	2.592	6.912
1000	5.4	3.24	8.64

Table 6 Computational overhead of the data aggregation process

i	$T_{\sigma 2}$	T_{m2}	Sum_2
200	0.082	0.166	0.248
400	0.164	0.332	0.496
600	0.246	0.498	0.744
800	0.328	0.664	0.992
1000	0.41	0.83	1.24

demonstrating that the algorithm in this paper can handle large-scale multimodal data while ensuring data security.

Finally, after receiving the aggregated information $\{CT, \sigma, \xi, T\}$ from the server, the sensing platform must determine if the encrypted multi-type data is complete. After confirming that any encrypted multi-type data is intact and has not been mistakenly altered or damaged by any attacker, the sensing platform decrypts the aggregated ciphertext CT using the key parameter γ_0. Assume the number of sensing users $i = 200, 400, 600, 800, 1000$ during the verification and decryption process, and compare the data completeness (T_w), aggregated data decryption (T_d), and total computational overhead of the sensing platform (Sum_3) for different numbers of sensing users in the raw sensing platform. As demonstrated in Table 7, the time needed for data completeness verification and decryption grows as the number of sensing users increases. It also indicates that the method described in this research can handle more multimodal data and has a high processing speed while maintaining data completeness.

Figure 6 depicts the effect of perceived user count on communication costs. The greater the number of sensing users who transmit data information to the server, the greater the number of sensing users who provide more information, and the greater the amount of data that must be processed. As shown in Fig. 6, the cost of communication increases as the number of sensing users rises. An increase in the number of sensing users results in an increase in the volume of data information, which compels the server and sensing platform to optimize the efficacy of data processing and transmission perpetually. In addition, from a security standpoint, the algorithm presented in this paper can guarantee that the sensing platform can verify the encrypted data's completeness even if the server aggregates the incorrectly replaced or destroyed encrypted multi-type data during the data aggregation process. Consequently, the scheme accomplishes a reasonable communication cost and provides a firm guarantee for the stable operation of the sensing platform, and the algorithm is ready for practical application.

5.3 Performance analysis

This section analyzes and compares the efficacy of the algorithms discussed in this paper. Before data integration, the data from Wikipedia were visualized so that changes could be observed. Then evaluate the performance of the multimodal data processing method. And compare it with the ACMR algorithm [38], the MCCA algorithm [39], and the DCCA algorithm [40] to verify the effectiveness of the multimodal data processing method used in this paper. The ACMR seeks an effective common subspace based on adversarial learning. The MCCA presents a generalization of the canonical correlation analysis to more than two variables, that can detect similar patterns across multiple domains. The DCCA learns complex nonlinear transformations of two views of data such that the resulting representations are highly linearly correlated. Finally, the algorithm's computational overhead and communication cost are discussed and compared with algorithms such as the EPPA [41], APPA [42], LVPDA [43], HC-SDPM [44], ASAS [45], etc. The EPPA uses a superincreasing sequence to structure multidimensional data and encrypts the structured data by the homomorphic Paillier cryptosystem technique. The APPA is a device-oriented Anonymous Privacy-Preserving scheme with Authentication for data aggregation applications in fog-enhanced IoT systems, which also supports multi-authority to manage smart devices and fog nodes locally. The LVPDA combined the Paillier homomorphic encryption method and an online/

Table 7 Computational overhead of data decryption process

i	T_w	T_d	Sum_3
200	7.162	0.54	7.702
400	14.324	1.08	15.404
600	21.486	1.62	23.106
800	28.648	2.16	30.808
1000	35.81	2.7	38.51

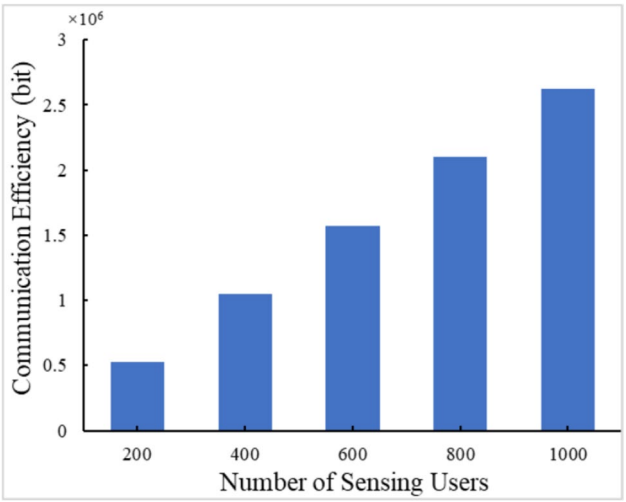


Fig. 6 Impact of the number of perceived users on communication costs

offline signature technique to ensure the privacy preserving and integrity verification during the data aggregation process. The HC-SDPM is the homomorphic cryptosystem-based secure data processing model, enabling secure data collection and aggregation offloading on edge nodes. The ASAS protects the identities of terminal devices by using pseudonyms and guarantees data secrecy via a homomorphic encryption technique.

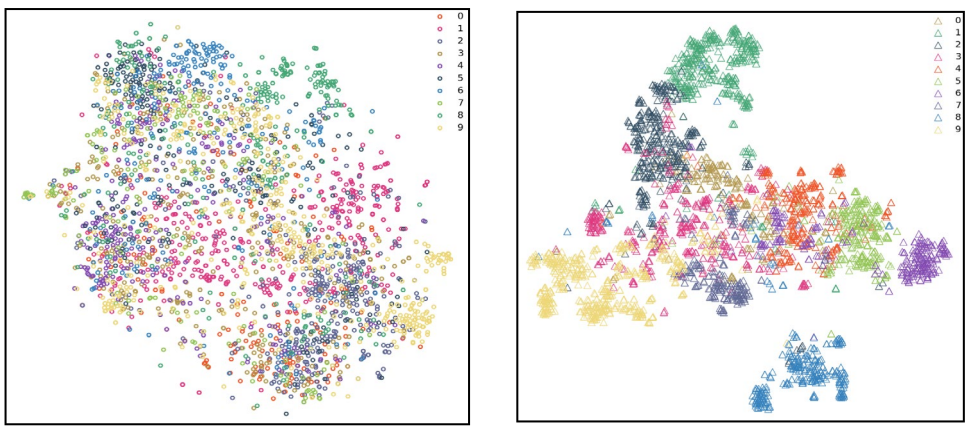
5.3.1 Data visualization

To visualize the efficacy of multimodal data fusion, the t-SNE method embeds the representation of image and text samples in an ordinary space into a two-dimensional visualization plane. Figure 7(a) and (b) depict the outcomes of the original image represented by 4096-dimensional features and the text excerpts characterized by 300-dimensional features, respectively. It is evident that the distribution of images and text in the Wikipedia dataset is quite distinct and that the sample distribution space of each category is relatively chaotic and irregular, making it challenging to classify the samples in the original input space. Figure 8(a) and (b) illustrate the two-dimensional distribution of image and text representations in the joint space. The results demonstrate that illustrations with loss of distinction in public and the label spaces can model the difference between samples from various semantic categories and effectively separate the representations into several semantic distinction clusters. A limited number of pictures from distinct semantic types are also found to be combined, allowing for a clearer understanding of the relationships between the categories. In addition, the image and text modal distributions in Fig. 8(c) are well blended and challenging to distinguish, resulting in a perfect integration of multimodal data. The results of the experiments indicate that the method can effectively reduce cross-modal errors.

5.3.2 Performance analysis of multimodal data processing

Two everyday modal retrieval tasks are employed to evaluate the performance of multimodal data processing methods: querying text information over image information and interrogating image information through text information. The precision-recall curve and the mean average precision (mAP) were used as evaluation metrics. The various performance indicators, as defined and computed below.

Fig. 7 Visualization of Wikipedia dataset



(a) Original image samples (b) Original text sample

The precision rate is the probability that an actual positive sample will occur among all predicted positive examples, as calculated as follows.

$$Prcision = \frac{TP}{TP + FP} \tag{14}$$

Recall is the probability that a positive sample will be predicted among the actual positive examples, and it is calculated using the following formula.

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

where TP represents a positive sample that the model predicts will be positive. FP denotes a false positive or an adverse selection that the model indicates to be positive. FN denotes a positive sample that the model predicts is negative but is falsely negative. TN represents a truly negative sample with a pessimistic model prediction.

Precision-recall curves (P-R curves) are represented in Fig. 9. The P-R curves for image querying text information and text querying image information are depicted in Fig. 9(a) and (b), respectively. Not only is this method superior to other methods for interrogating textual information through image information, but it is also superior for querying image information through textual information. Observing Fig. 9 reveals that the algorithm presented in this paper is substantially more efficient than the MCCA algorithm, indicating that the proposed method has distinct efficiency advantages. The benefit of this model is its ability to develop relationships between images and text using a bottom-up attention mechanism so that more appropriate embeddings can be obtained to measure the relevance between visual and textual data, resulting in improved retrieval tasks and integration of image and textual data. Multimodal data fusion combines data from various modalities, which provides a more dependable means of protecting data security and preserving data security and completeness.

The mAP is one of modal retrieval systems' most frequently employed evaluation metrics. The mAP is computed by locating the mean precision of all query data and then locating the mean value.

$$mAP = \frac{1}{Q}\sum_{i=1}^{Q} AP(q_i) \tag{16}$$

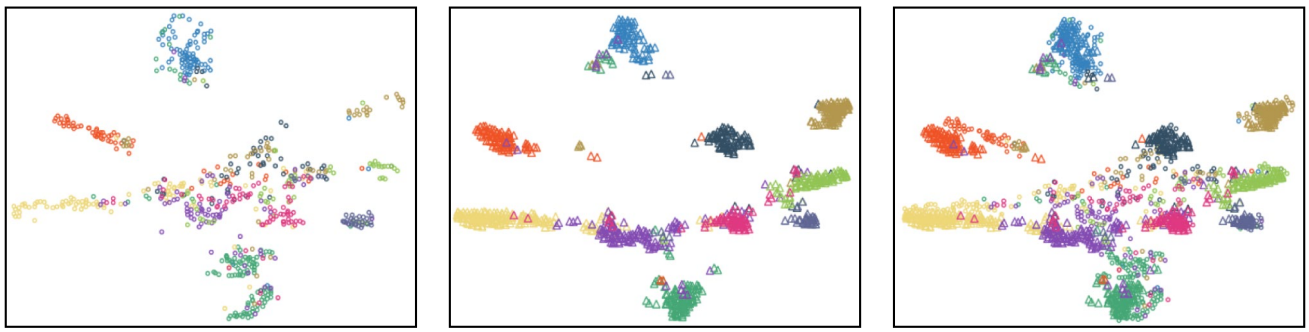
where Q represents the number of inquiry results, and AP represents the precision on average.

Table 8 compares the mAP of this method to other image query text and text query image methods. The data in Table 8 demonstrate that the method presented in this paper outperforms the MCCA algorithm in both image query text and text query image, with a mean accuracy rate of 0.3 higher, indicating that the algorithm presented in this paper has clear advantages. Compared to the DCCA and ACMR algorithms, the algorithm presented in this paper has a superior ability to fuse image and text data precisely. In conclusion, the algorithm presented in this paper can effectively combine diverse categories of data information, promote the complementarity between data information, and encrypt and process the data to assure the data's security and confidentiality.

5.3.3 Performance analysis of communication costs

To compare this scheme with the EPPA algorithm, APPA algorithm, LVPDA algorithm, HC-SDPM algorithm, and ASAS algorithm in terms of computational overhead and communication cost, the same level of security must be provided. Therefore, the size of the Paillier cryptosystem's security parameters is set to 1024 bits, the bit length of N is 1024 bits, the dimensions of the elements in G_1 and G_2 are set to 512 bits and 1024 bits, respectively, and the sizes of the identifier, timestamp, and authorization information are all set to 64 bits, and the size of the sequence number is set to 128 bits.

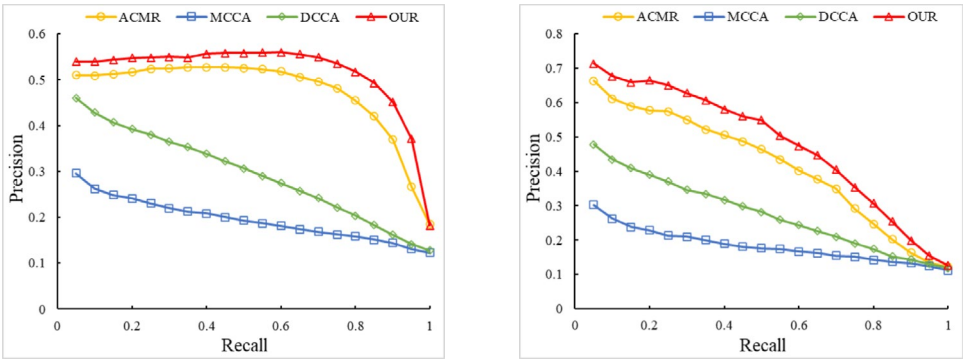
Due to sensing users' limited storage and computing capacity, one server can receive encrypted multimodal data and authenticators uploaded by most sensing users



(a) Image representation (b) Text representation (c) Image and text

Fig. 8 Visualization of Wikipedia dataset after fusion

Fig. 9 P-R curve



(a) Image-Text (b) Text-Image

simultaneously. Hence, the communication cost of each sensing user is a critical factor affecting the MCS system's feasibility. The communication cost will be evaluated based on two processes: user-to-server and server-to-platform. Firstly, each sensing user sends $\{CT_i, \sigma_i, T\}$ to the server, the communication cost from n sensing users to the server is $(2048+512+64)*n=2624n$ bits. Secondly, when each server sends $\{CT, \sigma, \xi, T\}$ to the sensing platform, the communication cost between the server and the sensing platform is $(2048+512+64+64)*n=2688n$ bits。

The EPPA method transmits $\{CT_i, RA, U_i, TS, \sigma_i\}$ to the gateway, where TS stands for timestamp, RA and U_i are identifiers, and then the communication cost from n users to the gateway is $(2048+64+64+64+512)*n=2752n$ bits. Secondly, each gateway transmits $\{C, RA, GW, TS, \sigma_g\}$ to the operator, where RA and GW are identifiers, and the gateway-to-operator communication cost is $(2048+64+64+64+512)*n=2752n$ bits.

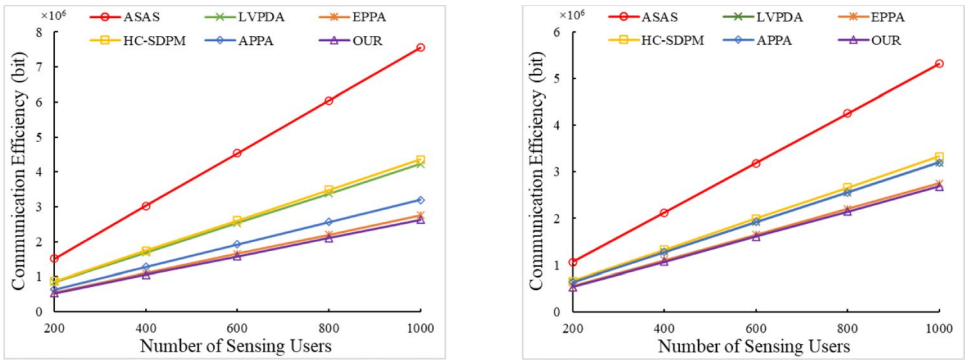
The APPA algorithm first sends $\{C_i, \sigma_i, Cerp_{SD_i}, TS\}$ to the fog node, where $Cerp_{SD_i}$ is the identification and then the communication cost from n intelligent devices to the fog node is $(2048+1024+64+64)*\ n=3200n$ bits. Secondly, each fog node transmits $\{C_a, \sigma_{c_a}, Cert_{FN_k}, TS\}$ to the public cloud server, where $Cert_{FN_k}$ is the identification and the communication cost $(2048+1024+64+64)*\ n=3200n$ bits.

The LVPDA method sends $\{ID_i, C_i, TS_i, \sum_i^{on}\}$ to the edge server, where $\sum_i^{on}=(s\prime_i, u\prime_i)$ displays the system's security settings, and n intelligent IoT devices to edge server communication cost is $(64+2048+64+1024*2)*n=4224n$ bits. Each edge server sends $\{ID_j, C, TS_t, \Sigma_{Agg}\}$ to the control center, where Σ_{Agg} represents the system's security settings, and the communication cost from the edge server to the control center is $(64+2048+64+1024)*n=2656n$ bits. According to the description above, in the HC-SDPM algorithm, the communication cost from n end devices to the fog node is $(64+2048+1024*2+64+128)*n=4352n$ bits, while the communication cost from the fog node to cloud servers is $(64+2048+1024+64+128)*n=3328n$ bits. The communication cost from each edge node to cloud servers in the ASAS method is $(1024+64+64+64+2048*2)*n=5312n$ bits and the communication cost from n smart devices to edge nodes is $(1024+64+64+64+64+1024+64+1024+64+2048*2)*n=7552n$ bits.

Table 8 Comparison of mean accuracy rates

Approach	Image-Text	Text-Image	Average
ACMR	0.48	0.43	0.46
MCCA	0.21	0.19	0.20
DCCA	0.30	0.29	0.30
OUR	0.53	0.49	0.51

Figure 10(a) illustrates the cost of communication between n sensing users and the server, while Fig. 10(b) depicts the cost of communication between the server and the sensing platform. It can be observed that the algorithm presented in this paper has a lower communication cost than the other five methods. The ASAS algorithm incurs higher communication costs in these two phases than all other compared methods. Compared to the ASAS algorithm, the method presented in this paper reduces communication costs by about 65% in the user-to-server process and by about 49% in the server-to-sensing platform process. The communication cost of the algorithm presented in this paper and the communication cost of the EPPA algorithm are relatively close in these two processes. However, the gap is still more significant in the overall data encryption process, particularly as the number of sensing users who need to collect and process more data increases, and the communication cost rises. In conclusion, the algorithm presented in this paper can minimize communication costs while maintaining data accuracy and dependability, and it can be implemented in various complex sensor network environments.



(a) User-to-server-side (b) Server-to-sensing platform-side

Fig. 10 Comparison of communication costs

5.3.4 Performance analysis of computational overhead

The computational overhead varies depending on the stage. Sensing users must perform two power operations in $Z_{n^2}^*$ to create the ciphertext CT_i and two power operations in G_1 to generate the authentication code σ_i of the ciphertext CT_i during the uploading process for a total computational overhead of $2T_{Zn}^{\wedge} + 2T_{G1}^{\wedge}$. After receiving $\{CT_i, \sigma_i, T\}, i = 1,2,...,n$ from the sense user, the cloud server must do a times multiplication operation in $Z_{n^2}^*$ to get the aggregated ciphertext CT, and compute σ must perform a power operation in Z_p^*. As a result, the server's overall calculation overhead is $a \times T_{Zn}^{\times} + T_{Zp}^{\wedge}$. To verify data completeness, one hash operation and $(a+1)$ bilinear pairing operations are required, and the sensing platform needs one power operation in $Z_{n^2}^*$ to decode the encrypted data. Hence the server's computing overhead is $(a+1)T^e + T^h + T_{Zn}^{\wedge}$.

During user data upload in the EPPA algorithm, each user requires two power operations in $Z_{n^2}^*$ to create ciphertext C_i. Two power actions in $Z_{n^2}^*$ are needed to create a signature σ_i. As a result, the user's overall computing overhead is $4T_{Zn}^{\wedge}$. The data aggregation method involves $(a+2)$ multiplication operations, two hash function operations in $Z_{n^2}^*$, one power operation in Z_p^*, and two power operations in $Z_{n^2}^*$ to construct the aggregated ciphertext and the new signature on the aggregated ciphertext. As a result, the server's overall computing overhead is $2T_{Zn}^{\wedge} + (a+2) \times T_{Zn}^{\times} + 2T^h + T_{Zp}^{\wedge}$. The public cloud server must conduct two power operations in $Z_{n^2}^*$ to decode the encrypted data, resulting in a computational overhead is $2T_{Zn}^{\wedge}$.

During user data upload in the APPA algorithm, each user requires two power operations in $Z_{n^2}^*$ to construct a ciphertext C_i. A signature requires six multiplication operations in G_1 and two power operations in G_1. As a result, the user's overall computing overhead is $2T_{Zn}^{\wedge} + 6T_{G1}^{\times} + 2T_{G1}^{\wedge}$. The data aggregation procedure, $Z_{n^2}^*$ must execute a multiplication operation to produce the aggregated ciphertext, and G_1 must do a power operation and a hash operation to generate the new signature on the aggregated ciphertext. As a result, the edge server's overall computing overhead is $a \times T_{Zn}^{\times} + T^h + T_{G1}^{\wedge}$. The edge server must do $(a+1)$ bilinear pairing operations and one hash operation to validate the signature. The control center must then do two power operations in $Z_{n^2}^*$ to decode the encrypted data, resulting in a computational overhead of $2T_{Zn}^{\wedge} + (a+1)T^e + T^h$.

By the procedure described above, the total computational overhead of the user side of the LVPDA algorithm can be calculated to be $2T_{Zn}^{\wedge} + T^m + T_{G1}^{\times}$, the server's total computational overhead is $T^m + T_{G1}^{\times} + T_{Zn}^{\times}$, and the computational overhead of the control center is $(a+3)T^e + T^h$. The total computational overhead of the user side of the HC-SDPM algorithm is $2T_{Zn}^{\wedge} + m \times 2T_{G1}^{\wedge}$, the total computational overhead of the edge server is $mT^h + T_{G1}^{\wedge} + a \times T_{Zn}^{\times}$, and the cloud service center's computational overhead is $2T_{Zn}^{\wedge} + (a+1)T^e + T^h + T_{Zp}^{\wedge}$. The total computational overhead for the user side of the ASAS algorithm is $3T_{G1}^{\wedge} + 2T_{G1}^{\times} + T^m$, the total computational overhead for the fog node server is $T^m + T_{G1}^{\times} + a \times (2T^e + T_{G1}^{\times} + T_{G1}^{s\times}) + 2(a-1)T_{G1}^{+}$, and the total computational overhead for the public cloud server is $(a+3)T^e + 2T^h + T_{G1}^{\wedge} + T_{G1}^{\times}$. Details are provided in Table 9. Where k represents the quantity of data uploaded, m represents the number of vectors, a represents the total number of encrypted messages in a single operation, and c represents the number of vectors that have been combined.

Figure 11 depicts the computational burden at each stage of the Paillier encryption algorithm's execution. Specifically, Fig. 11(a) illustrates the computational overhead of n sensing users during the phase of generating digital signatures, Fig. 11(b) depicts the computational overhead of n sensing users during the step of generating aggregated signatures, and Fig. 11(c) depicts the computational overhead of n sensing users during the phase of decrypting data. It can be observed that the algorithm's computational latency in these three phases is low compared to other methods, which enhances the algorithm's computational efficiency. Even though the computational overhead of this paper's

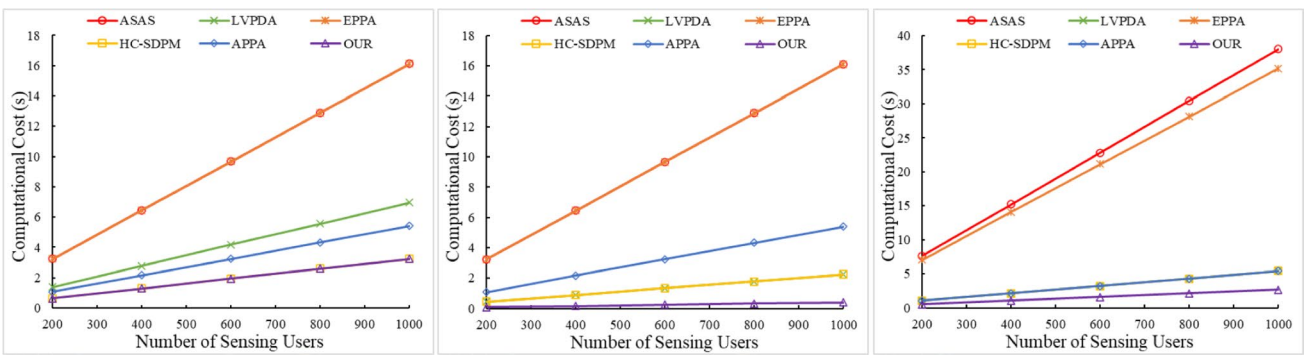
Table 9 Comparison of computational overhead by phase

	Encrypt	Data signature	Completeness verification	Aggregate signature	Aggregate encrypted data	Decrypt
EPPA	$(k+1)T^{\wedge}_{Zn}$	$T^{m}+T^{\wedge}_{Zn}$	$(a+1)T^{e}+T^{h}$	$T^{m}+T^{\times}_{G1}$	$T^{s\times}_{G1}$	$2T^{e}$
APPA	$2T^{\wedge}_{Zn}$	$2T^{\wedge}_{Zn}$	None	$2T^{\wedge}_{Zn}$	$(a+2)T^{\times}_{Zn}+2T^{h}+T^{\wedge}_{Zp}$	$2T^{\wedge}_{Zn}$
LVPDA	$2T^{\wedge}_{Zn}$	$6T^{\times}_{G1}+2T^{\wedge}_{G1}$	$(a+1)T^{e}+T^{h}$	$m\times T^{h}+T^{\wedge}_{G1}$	$a\times T^{s\times}_{G1}$	$2T^{\wedge}_{Zn}$
HC-SDPM	$2T^{\wedge}_{Zn}$	$m\times 2T^{\wedge}_{G1}$	$(a+1)T^{e}+T^{h}$	$m\times T^{h}+T^{\wedge}_{G1}$	$a\times T^{s\times}_{G1}$	$2T^{\wedge}_{Zn}$
ASAS	$3T^{\wedge}_{G1}+T^{\times}_{G1}$	$T^{m}+T^{\times}_{G1}$	$(a+1)T^{e}+T^{h}$	$T^{m}+T^{\times}_{G1}$	$a\times(2T^{e}+T^{\times}_{G1}+T^{s\times}_{G1})+2(a-1)T^{e}$	$2T^{e}+T^{h}+T^{\wedge}_{G1}+T^{\times}_{G1}$
OUR	$2T^{\wedge}_{Zn}$	$2T^{\wedge}_{G1}$	$(a+1)T^{e}+T^{h}$	$T^{\wedge}_{Zp}$	$a\times T^{s\times}_{G1}$	$T^{\wedge}_{Zn}$

algorithm in the data signature phase of Fig. 11(a) is identical to that of the HC-SDPM algorithm, the computational overhead of this paper's algorithm is lower than that of the HC-SDPM algorithm in both the aggregated signature phase of Fig. 11(b) and the decryption phase of Fig. 11(c). In the decryption phase, the EPPA algorithm has a computational overhead compared to the algorithm presented in this paper. However, its computational overhead is significantly higher in the digital and aggregated signature phases. The computational overhead of this algorithm is reduced by approximately 80% compared to the EPPA algorithm in the data signature phase, by around 97% compared to the EPPA algorithm in the aggregated signature phase, and by about 92% compared to the ASAS algorithm in the data decryption phase when the number of perceived users is 1000. In conclusion, the scheme presented in this paper reduces the computational overhead of communication devices, reduces the deterioration of sensing devices, satisfies the requirements of practical applications, and is deployable in MCS.

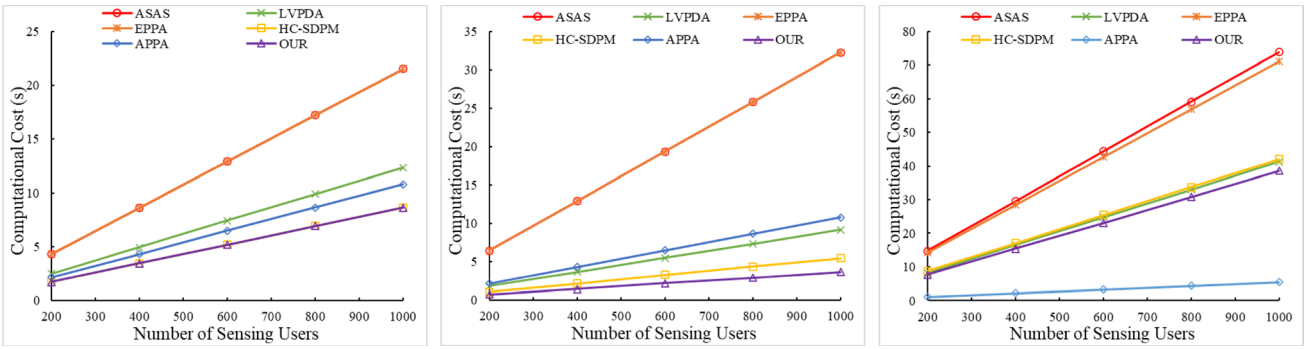
Figure 12 summarizes the total compute overhead for each platform. Figure 12(a) depicts the total computational overhead incurred by the sensing user, Fig. 12 (b) shows the total computational overhead incurred by the server, and Fig. 12(c) depicts the total computational overhead incurred by the sensing platform. Because the APPA algorithm does not perform data completeness checking, which is vulnerable to data loss, omission, and tampering and causes data crises, the computational overhead of the sensing platform shown in Fig. 12(c) is lower than that of the algorithm in this paper. The HC-SDPM algorithm has the same computational overhead for sensing users as the algorithm presented in this paper. Still, it is slightly higher for the total server and sensing platform. When there are 1000 sensing users, the perceived user computation overhead of the text algorithm is approximately 60% less than that of the EEPA and ASAS algorithms. Compared to the ASAS algorithm, the server computational burden of the text algorithm is reduced by approximately 98%. The text algorithm reduces the computational burden of the sensing platform by 48% compared to the ASAS algorithm. In conclusion, the algorithm presented in this paper still depends on other algorithms in the overall data encryption process, mainly when there are many sensing users in the encryption process. The collected data information will be stored on a cloud server, which can reduce the storage overhead and the computational overhead in the overall encryption process and device expenditures.

Figure 13(a) depicts the total communication cost of the algorithm described in this paper, while Fig. 13(b) depicts



(a) Digital Signature (b) Aggregate Signature (c) Decryption

Fig. 11 Computational overhead by phase



(a) Sensing user's total computing overhead (b) Server total computing overhead (c) Platform total computational overhead

Fig. 12 Total computational overhead by platform

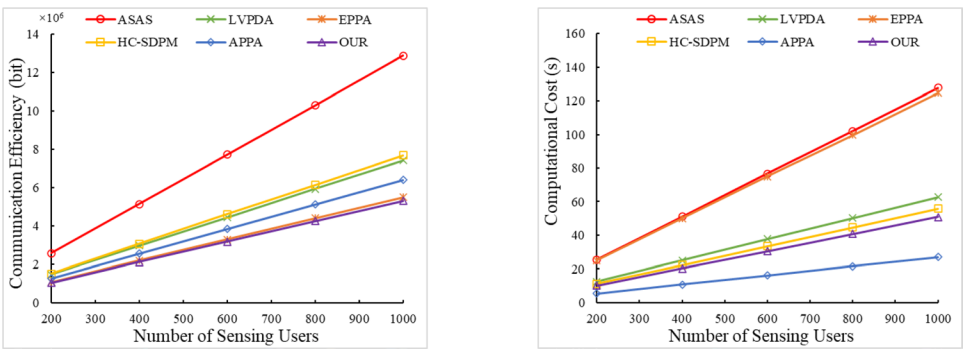
the total computational overhead. The computational overhead of the APPA algorithm is lower than that of the algorithm presented in this paper because the APPA algorithm does not conduct data completeness verification, which is susceptible to the risk of data leakage. In Fig. 13, the total communication cost and computational overhead of ASAS are significantly higher than those of the algorithm presented in this paper. The algorithm presented in this paper reduces the total communication cost by approximately 58% and the whole computational overhead by around 69% compared to ASAS. The total computational cost of the EPPA algorithm in Fig. 13(a) is comparable to that of the algorithm in this paper. Still, the total computational overhead of EPPA in Fig. 13(b) is greater than that of the algorithm in this paper. The total computational overhead of the EPPA algorithm is approximately 60% greater than that of the algorithm in this paper when the number of sensing users reaches 1000. In conclusion, the algorithm presented in this paper can minimize communication costs and computational overhead while maintaining data completeness and dependability.

In conclusion, the encryption algorithm presented in this paper significantly increases computational efficiency, decreases communication costs, and ensures data completeness. The method described in this paper provides more effective data privacy protection for sensing users and promotes the development of privacy protection overall.

6 Conclusion

To address the privacy protection problem in MCS, the data privacy protection method is investigated, the existing related literature is analyzed, and a multimodal data privacy protection and completeness verification method for MCS is proposed. The method fusion of multimodal data using a cross-attention mechanism facilitates the acquisition of data information between different modalities. Scalable super-increasing sequences are used to store the multimodal data collected by each sensing user, and the multimodal data is encrypted with the improved Paillier algorithm so that data privacy information is protected even if the decryption key is leaked to an attacker. A verifier is generated for each ciphertext using the BLS signature algorithm to ensure data completeness. The sensing platform provides the ability to verify the completeness of encrypted data, ensuring that encrypted multimodal data is never supplanted, altered, or destroyed inadvertently before decryption. The experimental results demonstrate that the scheme verifies data completeness,

Fig. 13 Total Communication Cost and Total Computational Overhead



(a) Total Communication Cost (b) Total Computational Overhead

protects data privacy, and has relative computational efficiency and communication overhead advantages. Data privacy protection faces challenges in data management, storage, and analysis; therefore, in the following work, we will consider inter and intra-semantic association modeling and data security of cross-modal data and attempt to design lightweight and secure privacy protection models for a larger environment.

Acknowledgements All the authors listed have approved the manuscript that is enclosed.

Authors' contributions Jian Wang, Fanfan Meng, Jia Liu, Guanzhi He and Guosheng Zhao wrote and revised the manuscript together.

Funding This present research work was supported by the National Natural Science Foundation of China (61403109, 61202458), the Specialized Research Fund for the Doctoral Program of Higher Education of China (20112303120007) and the Heilongjiang Natural Science Foundation (LH2020F034).

Data availability The datasets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Ethical approval and consent to participate I would like to declare on behalf of my co-authors that the work described was original research that has not been published previously, and not under consideration for publication elsewhere, in whole or in part.

Human and animal ethics Not applicable.

Consent for publication We would like to submit the manuscript entitled "Multimodal Data Privacy Protection and Completeness Verification Method for Mobile Crowd Sensing", which we wish to be considered for publication in "Peer-to-Peer Networking and Applications".

Competing interests The authors declare no competing interests.

References

1. Wang Y, Yan Z, Feng W et al (2020) Privacy protection in mobile crowd sensing: a survey. World Wide Web 23(1):421–452
2. Liu J, Cao H, Li Q et al (2018) A large-scale concurrent data anonymous batch verification scheme for mobile healthcare crowd sensing. IEEE Internet Things J 6(2):1321–1330
3. Abdelrahman A, El-Wakeel AS, Noureldin A et al (2020) Crowdsensing-based personalized dynamic route planning for smart vehicles. IEEE Network 34(3):216–223
4. Cecilia JM, Cano JC, Hernández-Orallo E et al (2020) Mobile crowdsensing approaches to address the COVID-19 pandemic in Spain. IET Smart Cities 2(2):58–63
5. Khorshidi S, Carter J, Mohler G et al (2021) Explaining crime diversity with google street view. J Quant Criminol 37:361–391
6. Gupta S, Tanwar S, Gupta N (2022) A systematic review on internet of things (IoT): applications & challenges. In: Proceedings of the 10th international conference on reliability, infocom technologies and optimization (Trends and Future Directions) (ICRITO). IEEE, Noida, India, pp 1–7
7. Sciancalepore S, Alhazbi S, Di Pietro R (2021) Receivers location privacy in avionic crowdsourced networks: issues and countermeasures. J Netw Comput Appl 174(1):102892.1-102892.17
8. Lirong M, Xiaoli G, Xiaoqiong Z (2022) Research on password-based data security protection system. Inf Secur Commun Secrecy 346(09):48–56
9. Wang Z, Qin J, Xiang X et al (2023) A privacy-preserving cross-media retrieval on encrypted data in cloud computing. J Inf Secur Appl 73:103440
10. Wang D, Wang Q, An Y et al (2020) Online collective matrix factorization hashing for large-scale cross-media retrieval. In: Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval (SIGIR'20). Association for Computing Machinery, New York, NY, USA, pp 1409–1418
11. Peng Y, Huang X, Zhao Y (2017) An overview of cross-media retrieval: concepts, methodologies, benchmarks, and challenges. IEEE Trans Circuits Syst Video Technol 28(9):2372–2385
12. Song Y, Soleymani M (2019) Polysemous visual-semantic embedding for cross-modal retrieval. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR). IEEE Computer Society, Long Beach, CA, USA, pp 1979–1988
13. Qian Y, Ma Y, Chen J et al (2021) Optimal location privacy preserving and service quality guaranteed task allocation in vehicle-based crowdsensing networks. IEEE Trans Intell Transp Syst 22(7):4367–4375
14. Nkenyereye L, Islam SMR, Bilal M et al (2021) Secure crowdsensing protocol for fog-based vehicular cloud. Futur Gener Comput Syst 120:61–75
15. Xiao M, Gao G, Wu J et al (2020) Privacy-preserving user recruitment protocol for mobile crowdsensing. IEEE/ACM Trans Networking 28(2):519–532
16. Arulprakash M, Jebakumar R (2021) People-centric collective intelligence: decentralized and enhanced privacy mobile crowd sensing based on blockchain. J Supercomput 77(11):1–27
17. Liu T, Wang Y, Cai Z et al (2020) A dynamic privacy protection mechanism for spatiotemporal crowdsourcing. Secur Commun Netw 2020:1–13
18. Zhang S, Li X, Tan Z et al (2019) A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services. Futur Gener Comput Syst 94:40–50
19. Liu T, Yan G, Cai G et al (2020) User personalized location k anonymity privacy protection scheme with controllable service quality. In: Proceedings of the machine learning for cyber security (ML4CS). Guangzhou, China. SpringerInternational Publishing, pp 484–499
20. Zhang S, Hu B, Liang W et al (2023) A caching-based dual k-anonymous location privacy-preserving scheme for edge computing. IEEE Internet Things J 10(11):9768–9781
21. Zhang Q, Wang T, Tao Y et al (2024) Location privacy protection method based on differential privacy in crowdsensing task allocation. Ad Hoc Netw 158:103464
22. Zhang J, Yang F, Ma Z et al (2020) A decentralized location privacy-preserving spatial crowdsourcing for Internet of vehicles. IEEE Trans Intell Transp Syst 22(4):2299–2313
23. Zou S, Xi J, Xu G et al (2021) CrowdHB: A decentralized location privacy-preserving crowdsensing system based on a hybrid blockchain network. IEEE Internet Things J 9(16):14803–14817
24. Wang L, Zhang D, Yang D et al (2020) Sparse mobile crowdsensing with differential and distortion location privacy. IEEE Trans Inf Forensics Secur 15:2735–2749
25. Li S, Zhang G (2020) A differentially private data aggregation method based on worker partition and location obfuscation for mobile crowdsensing. Comput Mater Continua 63(1):223–241

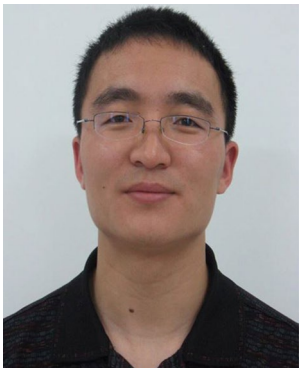
26. Zhang C, Zhao M, Zhu L et al (2022) Enabling efficient and strong privacy-preserving truth discovery in mobile crowdsensing. IEEE Trans Inf Forensics Secur 17:3569–3581
27. Zheng Y, Lu R, Yang X et al (2019) Achieving efficient and privacy-preserving top-k query over vertically distributed data sources. In: Proceedings of the 2019 IEEE international conference on communications (ICC). IEEE, Shanghai, China, pp 1–6
28. Xiong P, Li G, Liu H et al (2023) Decentralized privacy-preserving truth discovery for crowd sensing. Inf Sci 632:730–741
29 Liu Y, Liu F, Wu HT et al (2022) RPTD: Reliability-enhanced Privacy-preserving Truth Discovery for Mobile Crowdsensing. J Netw Comput Appl 207:68–78
30. Li Y, Xiao H, Qin Z et al (2020) Towards differentially private truth discovery for crowd sensing systems. In: Proceedings of the IEEE 40th international conference on distributed computing systems (ICDCS). IEEE, Singapore, Singapore, pp 1156–1166
31. Lin Y, Mao Y, Zhang Y et al (2022) Secure deduplication schemes for content delivery in mobile edge computing. Comput Secur 114:102602
32. Anderson P, He X, Buehler C et al (2018) Bottom-up and top-down attention for image captioning and visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR). IEEE Computer Society, Salt Lake City, UT, USA, pp 6077–6086
33. Devlin J, Chang M W, Lee K et al (2019) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 4171–4186.
34. Vaswani A, Shazeer N, Parmar N et al (2017) Attention is all you need. Advances in neural information processing systems, p 30
35. Fang W, Zamani M, Chen Z (2021) Secure and privacy preserving consensus for second-order systems based on paillier encryption[J]. Syst Control Lett 148:104869
36. Li S, Xue K, Yang Q et al (2017) PPMA: privacy-preserving multisubset data aggregation in smart grid[J]. IEEE Trans Industr Inf 14(2):462–471
37. Rasiwasia N, Costa Pereira J, Coviello E et al (2010) A new approach to cross-modal multimedia retrieval.In: Proceedings of the 18th ACM international conference on multimedia (MM'10). Association for Computing Machinery, New York, NY, USA, pp 251–260
38. Wang B, Yang Y, Xu X et al (2017) Adversarial cross-modal retrieval. In: Proceedings of the 25th ACM international conference on multimedia (MM'17). Association for Computing Machinery, New York, NY, USA, pp 154–162
39. Rupnik J, Shawe-Taylor J (2010) Multi-view canonical correlation analysis. In: Proceedings of the conference on data mining and data warehouses (SiKDD 2010). Slovenian KDD Conference on Data Mining and Data Warehouses, Ljubljana, Slovenia, pp 1–4
40. Andrew G, Arora R, Bilmes J et al (2013) Deep canonical correlation analysis. In: Proceedings of the 30th international conference on machine learning (ICML). Atlanta, GA, USA. PMLR, pp 1247–1255
41. Lu R, Liang X, Li X et al (2012) EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications[J]. IEEE Trans Parallel Distrib Syst 23(9):1621–1631
42. Guan Z, Zhang Y, Wu L et al (2019) APPA: an anonymous and privacy preserving data aggregation scheme for fog-enhanced IoT[J]. J Netw Comput Appl 125(1):82–92
43. Zhang J, Zhao Y, Wu J et al (2020) LVPDA: a lightweight and verifiable privacy-preserving data aggregation scheme for edge-enabled IoT[J]. IEEE Internet Things J 7(5):4016–4027
44. Trivedi HS, Patel SJ (2023) Homomorphic cryptosystem-based secure data processing model for edge-assisted IoT healthcare systems[J]. Internet of Things 22:100693
45. Wang H, Wang Z, Domingo-Ferrer J (2018) Anonymous and secure aggregation scheme in fog-based public cloud computing[J]. Futur Gener Comput Syst 78:712–719

Jian Wang received the PhD degree in computer applications in 2009 from Harbin Engineering University, China. She is currently a professor and doctoral supervisor in school of computer science and technologyHarbin University of Science and Technology. Her research interests include cognitive network, trusted computing and crowd sensing.

Fanfan Meng is currently a postgraduate in school of computer science and technologyHarbin University of Science and Technology. Her main research interests include crowd sensing and information security.

Jia Liu is currently a PHD student in school of computer science and technologyHarbin University of Science and Technology. Her main research interests include crowd sensing and IoT security.

Guanzhi He is currently a postgraduate in College of Bioinformatics, Science and Technology, Harbin Medical University. His main research interests include artificial intelligence and applications.



Guosheng Zhao received the PhD degree in computer applications in 2009 from Harbin Engineering University, China. He is currently a professor and master supervisor in school of computer science and information engineeringHarbin Normal University. His research interests include trusted computing and IoT security.