# Blockchain-Assisted Verifiable and Secure Remote Sensing Image Retrieval in Cloud Environment

Xue Ouyang, Yanyan Xu, Yangsu Mao, Yunqi Liu, Zhiheng Wang, and Yuejing Yan

***Abstract*—Secure retrieval of remote sensing images in an outsourced cloud environment garners considerable attention. Since the cloud service provider (CSP) is considered as a semitrusted third party that may return incorrect retrieval results to save computational resources or defraud retrieval fees for profit, it becomes a critical challenge to achieve secure and verifiable remote sensing image retrieval. This article presents a secure retrieval and blockchain-assisted verifiable scheme for encrypted remote sensing images in the cloud environment. In response to the characteristic that geographical objects in remote sensing images with clear category attributes, we design a remote sensing image retrieval method to facilitate secure and efficient retrieval. In addition, we propose a verifiable method combined with blockchain and Merkle trees for checking the integrity and correctness of the storage and retrieval services provided by CSP, which can replace the traditional third-party auditor. The security analysis and experimental evaluation demonstrate the security, verifiability, and feasibility of the proposed scheme, achieving secure remote sensing image retrieval while preventing malicious behavior of CSP.**

***Index Terms*—Blockchain, cloud computing, Merkle tree, secure remote sensing image retrieval, verification.**

## I. Introduction

THE number of remote sensing images has reached a petabyte level due to the growing volume and variety of satellites [1], placing a significant burden on users' storage and processing resources. At the same time, with the fast development of cloud computing, users with limited resources commonly employ cloud service providers (CSPs), which provide outsourced solutions and content-based remote sensing image retrieval [2], [3], [4].

However, as a semitrusted third party, CSP may illegally gather image information from users while providing retrieval services [5]. Because remote sensing images depict the distribution patterns and evolution patterns of morphological information, they expose highly sensitive data, such as resources and geographic locations. A data leak would cause various image security concerns and huge economic losses for users. For the sake of data protection, remote sensing images are encrypted before outsourcing to prevent CSP from gathering image information [6]. There are already some secure image retrieval schemes in the cloud environment, including randomized feature protection [7], homomorphic encryption [8], robust hash [9], bag-of-encrypted-words (BOEW) model [10], and secure multiparty computing [11], etc. Encryption makes a security paradigm available for outsourced data, but here is another fundamental challenge: *How can we ensure that CSP can provide correct storage and retrieval services in the encrypted environment?*

The primary cause of such an issue is that image owner and retrieval users will lose control of their data if images are outsourced to CSP, resulting in CSP returning incomplete or incorrect retrieval results to save computational resources or defraud retrieval fees for profit. CSP withholds technical proof from users, leaving users unable to verify whether CSP is truly providing services that meet their expectations [12]. In response to the drawbacks of semitrusted CSP, a traditional and straightforward option is to incorporate a third-party auditor (TPA) to verify that the CSP is providing services truthfully. For example, in the public auditable verification schemes [13], [14], it assumes that the TPA is unbiased and delegates it to evaluate the services of CSP. This may seem like an appropriate strategy, but only if the TPA is trusted without doubt. However, the truth is that TPA operates in a black box, with users not knowing its internal working procedures in reality [14], [15]. The TPA serves as a centralized third party, and its compromise may result in the termination of the entire verification service. Moreover, the TPA and CSP may collude to provide a mendacious verification report, i.e., regardless of the correctness of the services provided by the CSP, the TPA will report accurate verification results to the users.

Fortunately, blockchain technology provides a new perspective for the verification of data. As an excellent candidate for a trusted entity, the blockchain is a decentralized, nontampering, and traceable distributed ledger technology. In the blockchain, each participating full node saves replication to jointly maintain the integrity of the data, and the data structure of chained hashing and consensus algorithms ensures that data cannot be arbitrarily deleted or altered [16]. Therefore, blockchain technology ensures the integrity, reliability, and verifiability of the

Manuscript received 9 June 2022; revised 21 November 2022; accepted 21 December 2022. Date of publication 26 December 2022; date of current version 24 January 2023. This work was supported by the National Key Research and Development Program of China under Grant 2021YFB2501103, in part by the National Natural Science Foundation of China under Grant 42271431, and in part by the Hubei Provincial Key Program of the Natural Science Foundation of China under Grant 2020CFA001. *(Corresponding author: Yanyan Xu.)*

Xue Ouyang, Yanyan Xu, Yangsu Mao, Zhiheng Wang, and Yuejing Yan are with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China (e-mail: ouyangxue602@whu.edu.cn; xuyy@whu.edu.cn; maoyangsu@whu.edu.cn; wangzhih@whu.edu.cn; yuejingyan@whu.edu.cn).

Yunqi Liu is with the Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China (e-mail: yunqi1028@whu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3231890

on-chain data, preventing the malicious behaviors of CSP or TPA mentioned above. However, the current technical architecture of the blockchain can hardly meet the existing verification methods [17]. Since each full node of the blockchain backs up the complete ledger, storing a large amount of verification data on the storage-constrained blocks will result in huge data size and high storage overhead. Furthermore, most of the verifiable schemes focus on data integrity, and several works focus on data integrity in cloud storage, such as the works of [3], [18], [19]. Moreover, the well-known distributed cloud storage commercial platforms such as Storj [20], Sia [21], and Filecoin [22], verify the integrity of stored data by filing metadata on blockchain. However, in a real-world image retrieval scenario, it is necessary to consider not only the integrity of the cloud-stored data, but also the correctness of the retrieved results and the correctness of the similarity ranking of the returned images. Therefore, we emphasize the integrity of data and the correctness of retrieval services.

Based on the aforementioned challenges, we propose a blockchain-assisted verifiable and secure remote sensing image retrieval in the cloud environment in this article. Our goal is to share remote sensing images securely and efficiently in a semitrusted cloud environment, as well as check the integrity and correctness of retrieval results without TPA.

The flow of the scheme in this article is mainly as follows: Initially, the image owner extracts and encrypts the image features, sets the access policy based on the geographical objects, and then constructs a Merkle hash tree based on the encrypted features. Next, the image owner outsources the encrypted image data to CSP while uploading the lightweight information about images (e.g., the root hash of the Merkle hash tree and storage indexes) to the blockchain as a copy. If the geographical object in the query image of the retrieval user fulfills the access policy, CSP only retrieves the images with the same object. Then, CSP constructs a Merkle hash tree based on the retrieved distances and sent its root hash to the blockchain to commit this retrieval service. Upon the completion of the retrieval service, if the user has doubts about the returned images, he/she can initiate a challenge request to the blockchain. The blockchain acts as a trusted entity to calculate and compare the root hash of the Merkle tree calculated by CSP to the root hash stored by the image owner (or the root hash calculated by the blockchain) to verify the integrity and correctness of the retrieval results.

The contributions of the proposed scheme are three-folded:

1) *Verifiable Retrieval Results:* Considering the vulnerabilities of TPA, the blockchain is employed for security verification in this article. In particular, since current retrieval verification schemes do not consider the correctness of the retrieved results and the correctness of the similarity ranking of the returned images, we combine blockchain with Merkle hash tree to not only verify the integrity of the data stored by the CSP, but also calculate and verify the correctness of the retrieval service provided by the CSP.
2) *Secure and Efficient Remote Sensing Image Retrieval:* To address the problem of inefficiency caused by retrieving all remote sensing images in an encrypted environment, we use a remote sensing image dataset that is prelabeled with geographic objects [23]. Moreover, we focus on geographical objects in remote sensing images and combine them with the CBIR technique. In particular, the ciphertext policy attribute-based encryption (CP-ABE) [24] is employed to improve the security of the retrieval process, and only the distances between images with the same object as the query image are calculated, thus avoiding the need for CSP to retrieve the whole image dataset.
3) *Lightweight On-Chain Storage:* We design a storage structure to compensate for the limitations of blockchain in terms of storage and facilitate subsequent verification. The lightweight information such as hashes and storage indexes is stored in the blockchain, which prevents malicious tampering or removal of CSP.

The remainder of this article is organized as follows. The related work is reviewed in Section II. The preliminary study of Merkle hash tree is presented in Section III. The system model and adversary model are presented in Section IV. The construction of the proposed scheme is detailed in Section V. Security analysis is provided in Section VI and the simulation results are given in Section VII. Finally, Section VIII concludes this article.

## II. Related Works

### A. Secure CBIR

The framework of secure CBIR is generally composed of two modules: feature protection and feature similarity measurement. The progress of these modules will be reviewed specifically.

The goal of feature protection is to extract the image features and encrypt them with encryption techniques to ensure the encrypted image feature descriptors can be used for retrieval calculation. Generally, the visual content of remote sensing images is expressed by traditional hand-crafted or deep network features [25]. In particular, deep learning networks have been demonstrated to have a strong ability to recognize essential features of images [26]. As a result, it is feasible and acceptable to employ deep network features to retrieve remote sensing images, since they show an overwhelming advantage over hand-crafted features [27], [28], [29].

After extracting image features, the image owner encrypts them and sends them to the CSP for further similarity measurement. In the work of [7], three randomization-based methods for secure image retrieval are introduced, and then using L1 distance and Hamming distance to accurately calculate the distances between encrypted features. Although works [7] have low computational complexity, the encrypted features still remain the original feature information and the retrieval performance is insufficient, implying that the retrieval security and performance of these works are inadequate. To solve the above problems, a secure image retrieval based on homomorphic encryption is proposed in [8], in which image features are protected by homomorphic encryption and the most significant bit is used to achieve distance measurement in the encrypted domain. Therefore, the retrieval performance of this scheme is as accurate as plain image retrieval, but homomorphic encryption requires huge computation, which is a considerable burden for users with

limited resources and makes efficient retrieval of large-scale images difficult. To improve the efficiency of secure image retrieval, a large-scale image retrieval is presented in [9], in which each image generates a robust hash-based fingerprint and then compares the fingerprints of the images by Hamming distance. Although this scheme prevents CSP from inferring the image content, the significant reduction of features leads to its low retrieval performance. In the work of [10], a novel BOEW model for extracting image features is designed to improve retrieval efficiency, but the security of encrypted features should be improved. Moreover, a secure multiparty computation-based image retrieval scheme is provided to enhance retrieval security in [11], but the improved Euclidean distance employed in similarity measurement lacks image information, resulting in the sacrifice of retrieval accuracy. In summary, due to the enormous number and high-dimensional features of remote sensing images, it is challenging to strike a compromise between retrieval efficiency, security, and retrieval performance in these studies. In addition, most secure image retrieval studies assume that CSP will carry out storage and retrieval work honestly, but CSP is a semitrusted third party that may return incorrect retrieval results for the sake of profit.

### B. Data Verification

Due to the inherent risks of CSP, several TPA-based schemes are proposed to verify that the CSP is honestly providing services [13], [19], [30]. Wang et al. [13] presented a method for public audibility in which the TPA is used to verify the integrity of the data stored in the cloud. In the approach of [30], a privacy-aware public audit mechanism is proposed by constructing a homomorphic verifiable group signature. Moreover, the content and timestamp are allowed timestamp verification during data integrity checks in [19]. However, if the TPA is compromised, the verification service may be disrupted.

An alternative technology, blockchain opens new opportunities for solving the issues raised by TPA in the verification process. Thanks to one of the key technologies of blockchain, peer-to-peer distributed technology, all participants in the network gossip with other participants without relying on third parties, avoiding a single point of failure for the third parties. For example, the approach of [31] designed a two-layer blockchain to solve the data integrity problem in the cloud, and the approach of [18] presented a blockchain-based data integrity service framework. In general, most of the current blockchain-based verifiable schemes or platforms focus on data integrity. However, image retrieval also needs to consider the correctness of retrieval results, which is not mentioned in these schemes.

In view of the inadequacies of secure image retrieval and verifiable schemes, a secure and efficient remote sensing image retrieval method is proposed in this article, in which the remote sensing image features are extracted by deep network ResNet [32] and efficiently retrieved based on the geographical objects in remote sensing images as attributes. Additionally, a blockchain-assisted verifiable method is designed for image retrieval applications to ensure the integrity and correctness of retrieval images.

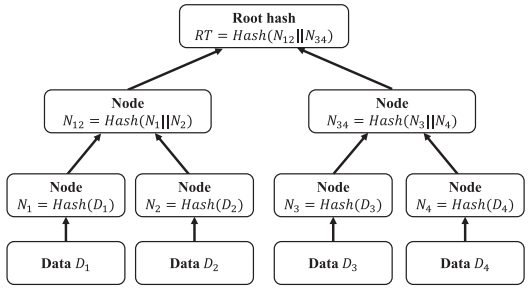


Fig. 1. Construction of Merkle hash tree.

## III. Preliminary

In this section, we provide preliminary knowledge of the Merkle hash tree.

### A. Merkle Hash Tree

Merkle tree is a tree in which every leaf node is labeled with the hash of a data, and the root of the tree can be seen as a commitment of all data stored in the tree. The benefit of the Merkle tree is that it allows rapid verification. It can verify that a data is consistent with the root hash by only checking a small subset of the hashes rather than the entire dataset.

An example of Merkle hash tree construction is shown in Fig. 1, where $D_1 < D_2 < D_3 < D_4$ are four data, $Hash(\cdot)$ is hash operation, and "$\|$" denotes the concatenation operation. The Merkle hash tree is constructed as follows. The first step is to hash the bottom data $D_1$ to $D_4$, and generate leaf nodes $N_1$ to $N_4$. Next, two neighboring leaf nodes are hashed, and the results $N_{12}$ and $N_{34}$ serve as nonleaf nodes. Finally, the hash operation is applied to the two nonleaf nodes, and the result $RT$ is the root hash.

## IV. Problem Statement

In this section, we will introduce the system model and adversary model of the proposed scheme.

### A. System Model

The system model is shown in Fig. 2, and the proposed system involves five entities, which are the image owner, retrieval users, cloud server provider (CSP), blockchain, and key management center (KMC). The functions of each entity are described as follows.

The image owner is in charge of encrypting image information and sending it to CSP and the blockchain; The retrieval user can submit a request to retrieve similar images and can also request the blockchain to verify the correctness of the retrieved result; The CSP is responsible for storing and retrieving remote sensing images; The blockchain is responsible for storing the lightweight information of encrypted images and verifying the retrieval results; The KMC is responsible for generating and distributing keys for retrieval users and image owner.

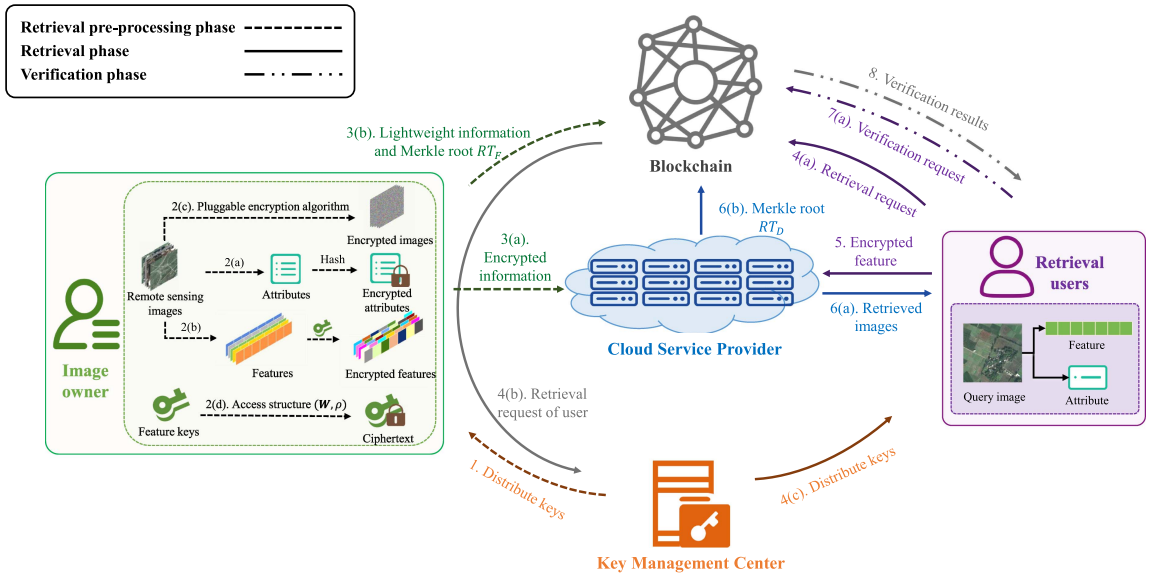


Fig. 2. System model of the proposed scheme.

TABLE I
NOTATIONS USED IN THE PROPOSED SCHEME

| Notation | Description |
|---|---|
| $\mathcal{K}$, $\mathcal{D}$, $\mathcal{C}$, and $\mathcal{U}$ | The key management center, the image owner, the cloud server provider, and the retrieval user |
| $\boldsymbol{Q}$ | The query image of retrieval user |
| $\boldsymbol{F}$, $\overline{\boldsymbol{F}}$, and $E(\overline{\boldsymbol{F}})$ | The image features, reshaped features, and encrypted features of $\boldsymbol{Img}$ |
| $\boldsymbol{F'}$, $\overline{\boldsymbol{F'}}$, and $E(\overline{\boldsymbol{F'}})$ | The image feature, reshaped feature, and encrypted feature of $\boldsymbol{Q}$ |
| $\boldsymbol{Idx_{Dis}}$ | The similar index collection |
| $\boldsymbol{\Omega}$, $\rho$ | The image attribute set and access policy |
| $Att_\omega$ | The image attribute of retrieval user |
| $\boldsymbol{Dis}$ | The distance collection between $E(\overline{\boldsymbol{F}})$ and $E(\overline{\boldsymbol{F'}})$ |
| $\boldsymbol{X}$ and $\boldsymbol{Y}$ | The chaotic sequences of LTM and TSM |
| $\boldsymbol{T}$ | The challenge set |
| $MT$ and $RT$ | Merkle hash tree and its Merkle root hash |

### B. Adversary Model

There are two threats considered in the proposed scheme, including CSP and eavesdroppers.

We assume that CSP is a semitrusted third party, which implies that it provides services while potentially collecting or analyzing outsourced data. In addition, CSP may perform storage and computation services dishonestly to save computational resources for profit.

Additionally, eavesdroppers may eavesdrop on information about entities, and they can be defended by encryption. Moreover, we assume that the eavesdroppers can be active or passive in the Yao–Dolev attack model [33], in which passive eavesdroppers are able to eavesdrop on communication.

## V. DESIGN OF THE PROPOSED SCHEME

The notations used in the proposed scheme are described in Table I. Moreover, the description of the proposed scheme is given as follows, where the smart contract functions on the blockchain are shown in Table II. In particular, the image owner and retrieval users utilize ResNet-34 [32] to extract the features of remote sensing images, which can address the problem that the hand-crafted features are unable to robustly represent the various contents of images.

TABLE II
MAIN FUNCTIONS OF THE SMART CONTRACT

| Function | Caller | Description of the function |
|---|---|---|
| *RegisterImage*() | $\mathcal{D}$ | Upload remote sensing image information. |
| *Verify*() | $\mathcal{U}$ | Verify the correctness of the retrieval results. |

Assume that a KMC $\mathcal{K}$, an image owner $\mathcal{D}$, a CSP $\mathcal{C}$, a retrieval user $\mathcal{U}$, and a blockchain are involved in the proposed scheme. For the sake of simplicity, we divide the proposed scheme into three phases: retrieval preprocessing, retrieval, and verification. The system overview is given as follows.

In the retrieval preprocessing phase, $\mathcal{D}$ is responsible for calculating the encrypted features, setting the geographical objects in the remote sensing image as attributes. Next, $\mathcal{D}$ outsources information about encrypted images to $\mathcal{C}$ and then constructs

a Merkle hash tree $MT_F$ based on the encrypted features to generate a root hash $RT_F$. Finally, $\mathcal{D}$ deploys the lightweight information about remote sensing images and the root hash $RT_F$ on the blockchain.

In the retrieval phase, $\mathcal{U}$ submits the retrieval request to $\mathcal{D}$. If the request from $\mathcal{U}$ is valid, $\mathcal{D}$ notifies $\mathcal{K}$ to distribute the keys for feature encryption. Next, $\mathcal{C}$ measures the similarity distances and finds similar encrypted images to send to $\mathcal{U}$. Finally, $\mathcal{C}$ constructs a Merkle hash tree $MT_D$ based on the distance results to generate a root hash $RT_D$ and uploads $RT_D$ on the blockchain to commit this retrieval service.

Upon the completion of the retrieval service, if $\mathcal{U}$ has doubts about the similar images returned, he/she can submit a challenge-based request to the blockchain to check the challenge set in the verification phase. The smart contract of the blockchain verifies the retrieval results of the challenge set and responds with a verification result to prove the correctness of the retrieval service provided by $\mathcal{C}$.

### A. Retrieval Preprocessing Phase

There are three steps involved in the retrieval preprocessing phase.

*1) Setup:* $\mathcal{D}$ and $\mathcal{C}$ register the blockchain account addresses to join the blockchain network.

Given random keys $a, \alpha, t, z \in \mathbb{Z}_p$, a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ based on two certain multiplicative cyclic groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order $p$, a generator $g$, and hash function $H : \{0,1\}^* \to \mathbb{G}$.

The public key $PK = \{g, H, Y_1, Y_2, K, R, K_\omega\}$ and the master key $MSK = \{Y_3\}$ are generated by $\mathcal{K}$, in which $Y_1 = g^a$, $Y_2 = e(g,g)^\alpha$, $Y_3 = g^\alpha$, $K = Y_1^{\frac{t'}{z}} Y_3^{\frac{1}{z}}$, $R = g^{\frac{t'}{z}}$, and $K_\omega = (H(x))^{\frac{t'}{z}}$.

Then, $\mathcal{K}$ asks $\mathcal{D}$ for the image size, denoted as $M$, and a pair of $2M$ invertible matrices $\boldsymbol{P}$ and $\boldsymbol{P^{-1}}$ are generated. Note that $\boldsymbol{P}$ is sent to $\mathcal{D}$ for encryption, while $\boldsymbol{P^{-1}}$ is sent to the $\mathcal{U}$ in the retrieval phase.

*2) Encrypt Image Information of Image Owner:* First, remote sensing images are assigned different attributes based on their geographical objects. For example, suppose the attribute set is $\boldsymbol{\Omega} = \{Att_k\}_{k=1}^3$, where attribute $Att_1$ is "Airplane," $Att_2$ is "Buildings," and $Att_3$ is "Court." The attribute of $Img_1$ is $Att_1$ if the geographical object in $Img_1$ is airplane, and the attribute of $Img_2$ is $Att_3$ if the geographical object in $Img_2$ is court, etc.

After assigning the attributes of remote sensing images, $\mathcal{D}$ randomly selects $r_x, r_y \in \mathbb{Z}_p$ and calculates $r = H(r_x)$. Then, two pseudorandom sequences $\boldsymbol{X} = \{x_j\}_{j=1}^{M-1}$ and $\boldsymbol{Y} = \{y_j\}_{j=1}^{2M}$ are, respectively, generated by the coupled Logistic-Tent map (LTM) and the coupled Tent-Sine map (TSM) chaotic systems [34]. Next, $\overline{\boldsymbol{F}}$ of size $2M$ can be expanded based on $\boldsymbol{F}$, i.e.,

$$\overline{\boldsymbol{F}} = [f_1, f_2, \ldots, f_M, -\tfrac{1}{2}\textstyle\sum_{i=1}^{M} f_j^2, x_1, x_2, \ldots, x_{M-1}].$$

Then the encrypted feature $E(\overline{\boldsymbol{F}}) = \{E(\overline{F_j})\}_{j=1}^{2M}$ can be given by

$$E(\overline{F_j}) = \left\{(\xi \overline{F_j} + y_j) \times \boldsymbol{P}\right\}_{j=1}^{2M} \tag{1}$$

where $\xi \in \mathbb{Z}_p$ is a public parameter.

After $\mathcal{D}$ has encrypted the image features and stored the related information to the CSP, $\mathcal{D}$ encrypts the feature keys $r_x, r_y, s$ as follows. $\mathcal{D}$ constructs a random vector $\boldsymbol{v} = [s, \gamma_2, \ldots, \gamma_l]^\top$ of length $l$ to share the secret $s$. Moreover, in order to generate the secret sharing matrix $\boldsymbol{W}$ of size $n \times l$, $\mathcal{D}$ sets an access policy $\rho$ based on the attribute set $\boldsymbol{\Omega} = \{Att_k\}_{k=1}^n$. Consequently, the encrypted secret vector $\boldsymbol{\lambda} = \{\lambda_k\}_{k=1}^n$ can be formulated as

$$\lambda_k = \{\boldsymbol{W}_k \cdot \boldsymbol{v}\}_{k=1}^n. \tag{2}$$

Then, $\mathcal{D}$ calculates $C_1 = r_x \cdot (Y_2)^s$, $C_2 = g^s$, $C_3 = r_y \oplus r$, $C_4 = g^{r_y}$, and $C_k = \{(Y_1)^{\lambda_k} \cdot H(\rho(k))^{-r_y}\}_{k=1}^n$. As a result, the ciphertext can be published as $KG = \{C_1, C_2, C_3, C_4, \{C_k\}_{k=1}^n\}$.

*3) Upload Encrypted Image Information:* $\mathcal{D}$ generates a proof by creating a Merkle hash tree $MT_F$ with the encrypted feature $E(\overline{\boldsymbol{F}}) = \{E(\overline{F_j})\}_{j=1}^{2M}$ for further verification, where the hashes of encrypted features $Hash(E(\overline{\boldsymbol{F}}))$ are available as leaf nodes and the root hash $RT_F$ is the proof of construction.

Then, $\mathcal{D}$ sends the information about remote sensing images to $\mathcal{C}$, including storage indexes of encrypted images, identifiers, attribute hashes, encrypted images, encrypted features, and the Merkle hash tree $MT_F$.

In addition, to prevent $\mathcal{C}$ from tampering with the remote sensing images, $\mathcal{D}$ uploads lightweight data on the blockchain by using the smart contract function *RegisterImage*(), including identifiers, hashes of plain images, attribute hashes, storage indexes of encrypted images stored in $\mathcal{C}$, and the root hash $RT_F$.

### B. Retrieval Phase

There are two steps involved in the retrieval phase.

*1) Submit Retrieval Request:* Suppose that $\mathcal{U}$ has a query image $\boldsymbol{Q}$, where the feature is $\boldsymbol{F'} = \{f'_j\}_{j=1}^M$ and the corresponding attribute is $Att_\omega$.

Before retrieving the image, $\mathcal{U}$ submits an image retrieval request along with $Att_\omega$ to $\mathcal{D}$. Then, $\mathcal{D}$ verifies the validity of the $\mathcal{U}$'s request. In particular, if the attribute hash of the query image does not satisfy the access policy, i.e., $Hash(Att_\omega) \not\subseteq Hash(\boldsymbol{\Omega})$, the request is deemed invalid, and $\mathcal{D}$ will return an error message to terminate the retrieval task. Otherwise, $\mathcal{D}$ notifies $\mathcal{K}$ to send the invertible matrix $\boldsymbol{P^{-1}}$ and private key $z$ to $\mathcal{U}$.

*2) Encrypt Information of Query Image:* Let $\{\varepsilon_k\}_{k=1}^n$ be a set of constants. If $\{\lambda_k\}_{k=1}^n$ is a secret sharing based on the secret sharing matrix $\boldsymbol{W}$, $\sum_{k=1}^n \varepsilon_k \lambda_k = s$ is satisfied. Then, $\mathcal{U}$ decrypts the ciphertext to get the feature keys, i.e.,

$$\begin{aligned} r_x &= \frac{C_1}{DE} = \frac{r_x \cdot (Y_2)^s}{e(g,g)^{\alpha s}} = \frac{r_x \cdot e(g,g)^{\alpha s}}{e(g,g)^{\alpha s}} = r_x \\ r_y &= C_3 \oplus H(r_x) = r_y \oplus r \oplus r = r_y \\ s &= r_x r_y \end{aligned}$$

where

$$\begin{aligned} DE &= \frac{e(C_2, K)}{e(\prod_{k=1}^n (C_k)^{\varepsilon_k}, R) \cdot \prod_{k=1}^n e((C_4)^{\varepsilon_k}, K_{\rho(k)})} \\ &= e(g,g)^{\frac{\alpha s}{z}}. \end{aligned} \tag{3}$$

Next, $\mathcal{U}$ generates two pseudorandom sequences $\boldsymbol{X} = \{x_j\}_{j=1}^{M-1}$ and $\boldsymbol{Y} = \{y_j\}_{j=1}^{2M}$ based on $r_x$ and $r_y$ in the same way as $\mathcal{D}$. Then, the feature $\boldsymbol{F'}$ of query image $\boldsymbol{Q}$ is reshaped to $\overline{\boldsymbol{F'}}$ with the size of $2M$ by

$$\overline{\boldsymbol{F'}} = [f'_1, f'_2, \ldots, f'_M, 1, x_1, x_2, \ldots, x_{M-1}].$$

Finally, the encrypted query feature $E(\overline{\boldsymbol{F'}}) = \{E(\overline{F'_j})\}_{j=1}^{2M}$ is sent to $\mathcal{C}$ for similarity measurement, i.e.,

$$E(\overline{F'_j}) = \left\{ \boldsymbol{P^{-1}} \times (\xi \overline{F'_j} + y_j)^{\top} \right\}_{j=1}^{2M}. \tag{4}$$

*3) Similarity Measurement:* Based on the request of $\mathcal{U}$, $\mathcal{C}$ identifies the storage index $\boldsymbol{Idx} = \{Idx_i\}_{i=1}^{L_n}$ and the encrypted image features $\{E(\overline{F_i})\}_{i=1}^{L_n}$ with the same hash of attribute as the query image, where $L_n$ represents the number of encrypted images with the same hash of attribute as the query image. Then, $\mathcal{C}$ calculates the distance collection $\boldsymbol{Dis} = \{\mathrm{Dis}_i\}_{i=1}^{L_n}$, i.e.,

$$\mathrm{Dis}_i = \left\{ E(\overline{\boldsymbol{F'}}) E(\overline{F_i}) \right\}_{i=1}^{L_n}. \tag{5}$$

Next, the similar index collection $\boldsymbol{Idx_{Dis}} = \{Idx_{\mathrm{Dis}_i}\}_{i=1}^{L_n}$ can be retrieved, which is sorted in descending order. Finally, according to the $\mathcal{U}$'s requirement for the number of retrieved images, $\mathcal{C}$ returns the top-$\kappa$ encrypted images to $\mathcal{U}$ based on $\{Idx_{\mathrm{Dis}_i}\}_{i=1}^{\kappa}$.

After the top-$\kappa$ encrypted images have been sent to $\mathcal{U}$, $\mathcal{C}$ creates a Merkle hash tree $MT_D$ based on the sorted distance collection $\boldsymbol{Dis}$. Specifically, the hash results of the sorted $\{Dis_i\}_{i=1}^{L_n}$ are available as the leaf nodes, and the root $RT_D$ is the proof of construction. Finally, the root hash $RT_D$ is sent to the blockchain.

### C. Verification Phase

After the retrieval service is complete, if $\mathcal{U}$ finds the retrieved images are considerably different from the query image, or if $\mathcal{U}$ intuitively finds the similarity ranking of $\kappa$ retrieved images is incorrect (e.g., the retrieved images that are intuitively more similar to the query image are ranked lower, while less similar retrieved images are ranked higher), he/she has reasonable suspicion that $\mathcal{C}$ cheated on the retrieval computation. As a result, $\mathcal{U}$ can check the correctness of the retrieval results in the verification phase.

Specifically, $\mathcal{U}$ registers as a node on the blockchain network. Next, $\mathcal{U}$ randomly selects $\tau$ numbers in the range of $[1, \kappa]$, and denoted as a challenge set $\boldsymbol{T}$ to submit the blockchain along with the encrypted query features. Then, the blockchain automatically executes the smart contract function *Verify*(), which is given as follows. 1) The blockchain checks the root hash of the encrypted image features based on the challenge set is equal to $RT_F$. If the check result is different, it is evidence that the CSP has dishonest storage behavior, and the blockchain returns the storage error message to the user. Otherwise, it confirms that the image data stored by the CSP is correct, and processing moves on to the next step. 2) The blockchain calculates the similarity distances of the challenge set based on (5) and checks the root hash of the calculated distance results based on the challenge set is equal to $RT_D$. Similarly, if the check result is different, it is evidence that the CSP has dishonest retrieval behavior, and the blockchain returns the retrieval error message. Otherwise, it confirms that the image data retrieved by the CSP is accurate, and returns the verification correct message to the user.

Fig. 3 shows an example of the verification process. Assume that $L_n = 4$ and $\tau = 2$. First, $\mathcal{U}$ selects 2 and 6 as the challenge set $\boldsymbol{T} = \{2, 6\}$ and submits a request via the smart contract *Verify*(). Next, the blockchain notifies $\mathcal{C}$ to submit verification information associated with the challenge set $\boldsymbol{T}$. Then, the blockchain checks the encrypted query features $\mathcal{U}$ and the encrypted features of $\mathcal{D}$, and calculates the distance $\mathcal{V}_{D_T}$ of $\boldsymbol{T}$ if the checks are correct. Moreover, the blockchain verifies whether $Hash(\mathcal{V}_{D_T})$ exists in the Merkle hash tree $MT_D$. Finally, the blockchain compares the distance $\mathcal{V}_{D_T}$. If the comparison results satisfy the order of $\boldsymbol{T}$, it indicates that the retrieval service of $\mathcal{C}$ is honest, otherwise, it is evidence that the CSP has dishonest storage behavior. Note that the correct order of retrieval results depends probabilistically on the number of challenge set selected.

## VI. Security Analysis

The security of the proposed scheme is analyzed from the following aspects.

### A. Malicious CSP

We will demonstrate that malicious CSP cannot learn anything about plain image features from encrypted image features and similarity measurement results, and analyze that CSP can only perform storage and retrieval services correctly.

First, we demonstrate that malicious CSP can not infer plain features from the encrypted features of the image owner or users. Formally, we follow the security concept of the Learning with Error (LWE) problem [35]. In other words, if the LWE problem is hard, it is computationally infeasible for an adversary $\mathcal{A}$ to restore plain features from encrypted features.

In addition, the malicious CSP can not get any useful information from the retrieval results in (5). Since CSP cannot restore plain features from encrypted features, which means that even if CSP can gather retrieval results during similarity measurement, it does not learn the plain features of the image owner or users. As a result, the malicious CSP is unable to obtain plain images from the retrieval results.

Finally, the proposed blockchain-assisted verifiable method can prevent CSP from maliciously tampering or removing the data of image owner and users. On the one hand, the image owner publishes the lightweight information to the blockchain for backup when outsourcing the encrypted image information to the blockchain, so that any malicious behavior of the CSP can be identified. On the other hand, upon the similarity measurement, CSP broadcasts the distance results to the blockchain by generating a Merkle hash tree with the root hash, thereby committing the outcome of this measurement. Since the hash function employed in the Merkle hash tree is collision resistant, if CSP gives wrong retrieval results for profit, the dishonest behavior of the CSP can be easily found in the verification phase by the root hash. Therefore, if the verification result is found to be false during the verification phase, it can prove that the CSP has malicious behavior.

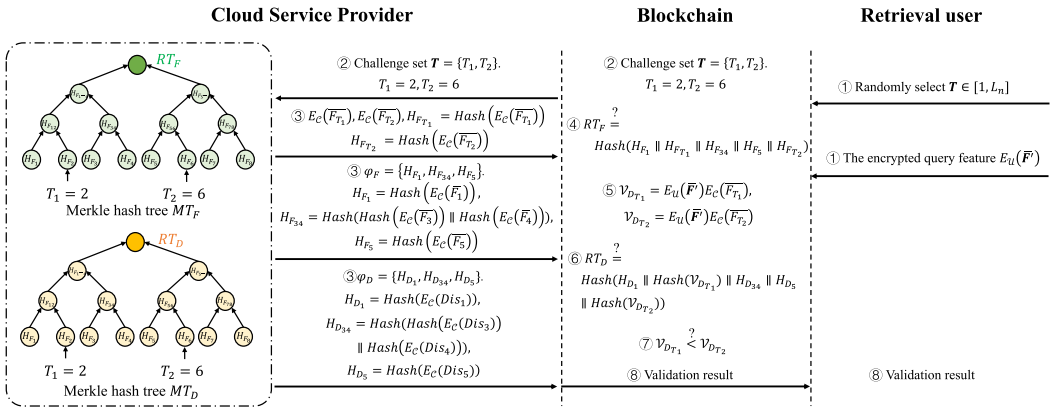


Fig. 3. Blockchain-assisted verification phase.

### B. Eavesdroppers

Generally, eavesdroppers can eavesdrop on entities, and they are guarded against in the same way as CSP in Section VI-A. In addition, the proposed scheme should be able to withstand the following well-known attacks during the verification phase.

*1) Man-in-The-Middle Attacks:* Eavesdroppers may tamper with or forge data on image owner and users, leading both sides to believe that their data is correct, but the entire communication is under the complete control of the eavesdroppers. In the proposed scheme, the lightweight encrypted image storage and verification is based on blockchain technology. As a result, the data of both the image owner and users on the blockchain can be protected from tampering and forgery by eavesdroppers due to the nontampering characteristic of the blockchain.

*2) Impersonation Attacks:* Eavesdroppers can impersonate query users and initiate retrieval verification on their behalf. However, before submitting requests, each query user needs to register on the blockchain as a node to get a unique identity in the proposed scheme. Then, the smart contract verifies the identities of users, rendering impersonation attacks infeasible.

*3) Replay Attacks:* Eavesdroppers may send verify requests that were sent by users to deceive the system. However, the timestamp in the blockchain involves the data for each verification, so that replay attacks can be avoided.

### C. Security in the Verification Phase

Considering the vulnerabilities of TPA, the blockchain is employed for security verification in the proposed scheme. The technological design of the blockchain ensures that the data stored and calculated on the chain is complete and correct, and its security analysis is given as follows:

Firstly, the Merkle tree is used for retrieval verification, which is constructed by hashing encrypted features and retrieval distances. The security of the hashing algorithm ensures the authenticity and integrity of the verified data. Moreover, the blockchain stores and calculates encrypted data to prevent the image data from being compromised, and its security proof is the same as the analysis in Section VI-A. Additionally, the data structure of chained hashing and consensus algorithms in blockchain ensures that all participating nodes can work honestly. Finally, each participating full node in the blockchain saves replication to maintain the integrity of the data jointly, avoiding the risk of data deleted or altered.

However, if the malicious attackers gather more than half of the computational power in the blockchain network, they can launch a 51% attack to forge the longest chain. The attackers can cheat or make chaos by the 51% attack in the verification process. Fortunately, the 51% attack is difficult to succeed from either the mathematical or financial perspectives [16].

As a result, blockchain technology is impenetrable to security attacks and prevents vulnerabilities that might endanger the entire verification phase. In addition, since consensus algorithms are included in the blockchain, all nodes in the blockchain network are responsible for verifying data to ensure it is correct.

## VII. Experimental Evaluation

To validate the effectiveness and generality of the proposed scheme, the simulation environment in this article is conducted on Intel(R) Xeon(R) E-2124 CPU, 3.30 GHz Core processor, 16.0 GB RAM. The smart contract is written using the Remix IDE platform and deployed on the Ethereum virtual machine, and the implementation of the blockchain is based on the official Ethereum test network Rinkby.

We evaluate the proposed scheme on two multilabel remote sensing image datasets: the Dense Labeling Remote Sensing Dataset (DLRSD) and the Wuhan Dense Labeling Dataset (WHDLD) [23]. In particular, we define each class of the datasets as an attribute in this experiment. In this section, the evaluation metrics mainly include image retrieval performance (accuracy, precision, and recall), time and communication costs of the verifiable method, computational overhead, and time consumption.

1) *DLRSD:* The DLRSD is a labeled remote sensing image dataset based on the UC Merced (UCM) archive [36], which is classified into 17 classes. Moreover, each image of DLRSD has a size of 256×256 pixels with a spatial resolution of 0.3 m, and each pixel is measured in the RGB spectral space.
2) *WHDLD:* WHDLD is a dataset cropped from a large remote sensing image of Wuhan city, which is classified into six classes. Moreover, each image of WHDLD has a size of 256×256 pixels with a spatial resolution of 2 m, and each pixel is measured in the RGB spectral space.

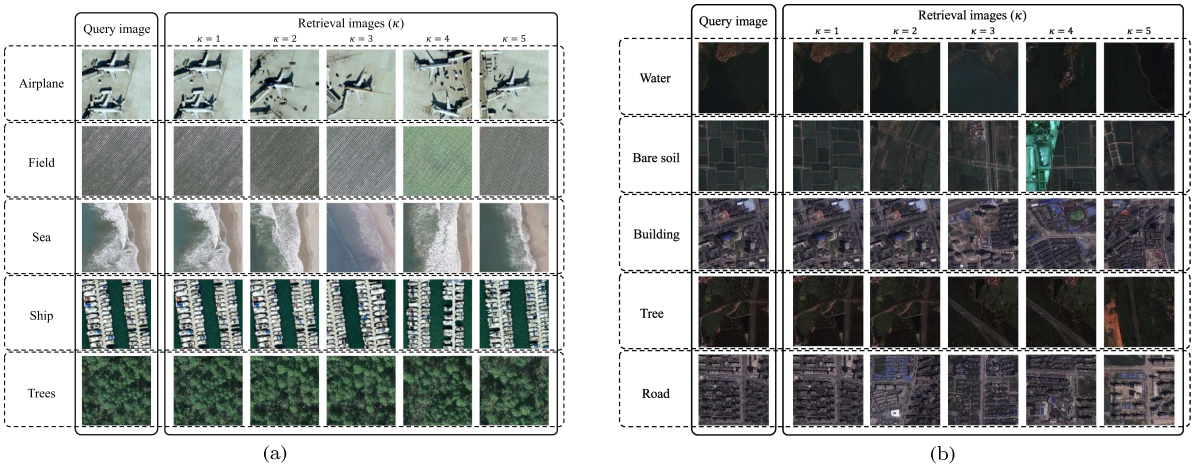


Fig. 4. Image retrieval results. (a) DLRSD dataset. (b) WHDLD dataset.

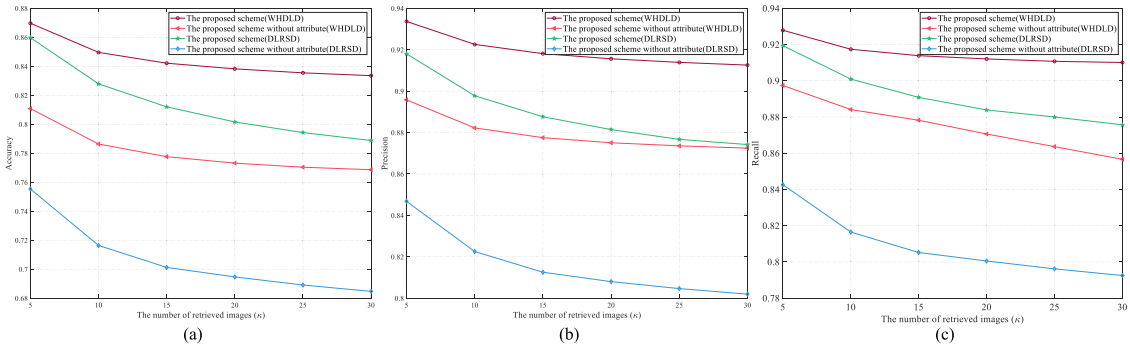


Fig. 5. Test results of DLRSD and WHDLD datasets.

### A. *Evaluation of Image Retrieval Performance*

Fig. 4 shows the remote sensing image retrieval results by utilizing the DLRSD and WHDLD datasets, where top $\kappa = 5$ retrieved images are provided. According to the retrieval results, the proposed scheme provides superior retrieval outcomes, i.e., the image with the highest similarity to the query image appears first in the retrieval results.

Moreover, we employ the measurement approach of [37] to quantitatively assess the retrieval performance of the proposed scheme, where accuracy, precision, and recall are selected as performance evaluation indicators. Specifically, let $\mathcal{L}_{R_i}$ be the label of the $i$th retrieved image $R_i$ and let $\mathcal{L}_Q$ be the label of the query image $Q$. These indicators can be defined as

$$\text{Accuracy} = \frac{1}{\kappa}\sum_{i=1}^{\kappa}\frac{|\mathcal{L}_Q \cap \mathcal{L}_{R_i}|}{|\mathcal{L}_Q \cup \mathcal{L}_{R_i}|}$$

$$\text{Precision} = \frac{1}{\kappa}\sum_{i=1}^{\kappa}\frac{|\mathcal{L}_Q \cap \mathcal{L}_{R_i}|}{|\mathcal{L}_{R_i}|}$$

$$\text{Recall} = \frac{1}{\kappa}\sum_{i=1}^{\kappa}\frac{|\mathcal{L}_Q \cap \mathcal{L}_{R_i}|}{|\mathcal{L}_Q|}$$

where $\kappa$ is the number of retrieved images, $|\cdot|$ is the number of nonzeros, $\cap$ and $\cup$ are the operations of logical AND and logical OR, respectively.

TABLE III
COMPARISON OF RETRIEVAL PERFORMANCE

| | Accuracy | Precision | Recall |
|---|---|---|---|
| SMSC [11] | 0.6167 | 0.8533 | 0.7067 |
| BOEW [10] | 0.7044 | 0.8826 | 0.7794 |
| PLCIR [9] | 0.5844 | 0.7806 | 0.6600 |
| **Ours** | **0.8594** | **0.9249** | **0.9266** |

In this test, to compare the difference between the proposed scheme and the proposed scheme without attribute, we first test DLRSD and WHDLD for the different numbers of retrieved images, and the average test results are shown in Fig. 5. In particular, the proposed scheme without attribute refers to the similarity measurement between the query image and all images in the database. The test results show that the proposed scheme has decent retrieval performance in all kinds of retrieval tests compared with the proposed scheme without attribute.

In addition, we compared the proposed scheme with other methods, including SMSC [11], BOEW [10], and PLCIR [9]. Table III shows each "bare soil" image retrieved from the WHDLD dataset and returns the retrieved image $\kappa = 30$ of the average results. For two primary reasons, the retrieval

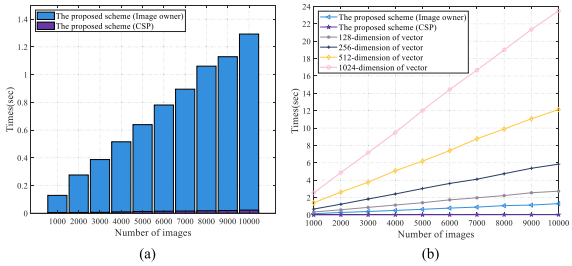


Fig. 6. Cost of constructing Merkle hash trees. (a) Image owner and CSP costs. (b) Comparison of the construction time of the proposed scheme with different dimensional vectors.

performance of the proposed scheme is better than other methods. On the one hand, we employ the deep learning network ResNet to extract the features of remote sensing images. Since the ResNet network can capture the essential features of the image, it exhibits stronger representation capability compared to the traditional manual feature extraction methods, which is conducive to improving the retrieval performance. On the other hand, the proposed scheme retrieves images of the same geographical objects based on attributes in the retrieval phase, which significantly increases retrieval performance. Consequently, it can be demonstrated that the proposed scheme in this article produces satisfactory retrieval results for remote sensing image datasets.

### B. Evaluation of Verifiable Method

In the proposed scheme, two Merkle hash trees, $MT_F$ and $MT_D$ are constructed to verify the retrieval results. Specifically, $MT_F$ is constructed by the image owner, who hashes the encrypted features of the remote sensing image as leaf nodes in the retrieval preprocessing phase. $MT_D$ is constructed by the CSP, which uses the distance results of the similarity measurement as leaf nodes of the Merkle hash tree in the retrieval phase.

In this test, we measure the cost of Merkle hash trees constructed by image owners and CSP with different numbers of images and then compared them with vectors of different dimensions for constructing Merkle hash trees, as shown in Fig. 6. It is clear that the proposed scheme performs ideally in terms of the cost of constructing the Merkle hash tree as the number of images increases. Moreover, in the preprocessing phase, the leaf nodes of Merkle hash tree $MT_F$ are calculated by using the keccak256 hash (the hash algorithm used in Ethereum), which causes the features of each encrypted image of the image owner to be compressed into a 64-b hexadecimal number. In the retrieval phase, the distance results calculated by CSP are employed as the leaf nodes of Merkle hash tree $MT_D$, where the size of each leaf node does not exceed 18 bytes. Therefore, compared with the Merkle hash tree constructed by other dimensions, the proposed scheme can substantially reduce its time cost.

### C. Evaluation of Efficiency

To more systematically verify the efficiency of the proposed scheme, the computational and storage costs are first analyzed, and then the time consumption is experimentally tested.

*1) Computational Overhead:* To analyze the impact of attributes on computational and storage costs, we compare the difference between the proposed scheme and the proposed scheme without attributes, where the proposed scheme without attributes refers to the similarity measurement between the query image and all images in the database. Assume that the image owner encrypts a dataset of $N$ remote sensing images that contains $L_n$ images with the same attributes ($N > L_n$).

We define $m$ as the dimension of the feature, $t$ and $k$ as the number of attributes associated with the ciphertext and the private key. Moreover, $\iota$, $\Gamma$, and $\Lambda$ are the size of a keccak256 hash, an encrypted feature, and an encrypted image, respectively. In addition, $DOT_{2\,m}$ is $2\,m$-dimensional dot product operation, $Pair_{G_T}$ is a pairing operation, $G_1$ and $G_T$ are the calculation in the group $G_1$ and $G_T$, $Hash_G$ and $Hash_K$ are hash operation in the group $G$ and the Ethereum platform, $Build$ is the operation of construct Merkle hash tree, and $Sear$ and $Comp$ are search and comparison operations. The proposed scheme is compared with the proposed scheme without attributes in terms of computational and storage costs, as shown given in Tables IV and V.

The above tables show that the retrieval preprocessing computational cost is the same since both the image owner and the user encrypt their owned images. Moreover, the computational costs in the retrieval and verification phases of the proposed scheme are less than that of the scheme without attributes. Therefore, the proposed scheme can effectively reduce the computational cost of the blockchain and CSP, significantly improving the retrieval and verification efficiency. In addition, since all the stored data size is the same, the resulting storage cost is the same as the scheme without attributes.

*2) Time Consumption:* In the proposed scheme, the retrieval preprocessing and verification phases are both offline operations, whereas the retrieval phase is an online operation with strict requirements for real-time performance. In this text, we first test the execution time of the attribute "bare soil" in the DLRSD database for returning 30 retrieved images, as shown in Table VI. The results indicate that the time consumption of encryption takes less time in the proposed scheme. Moreover, the steps involving smart contract functions and extracting image features have the greatest impact on the overall time, which can be finished offline. In addition, the actual operation by the user takes less than 1.5 s (1.4212 s), while the user spends less than 0.1 s in the online retrieval phase. Therefore, the retrieval user can achieve a lightweight operation and efficient image retrieval in the proposed scheme.

In addition, since the retrieval phase is operated online, we compare the time consumption of different schemes in the retrieval phase. In this test, we employ different numbers of DLRSD dataset to retrieve an image, and the results are given in Table VII. We observe that the retrieval time of other methods grows gradually as the dataset increases. However, the proposed scheme is based on attribute encryption of geographical objects in remote sensing images, which means that CSP retrieves only those images with the same attributes. Therefore, it can be concluded that employing the proposed scheme can significantly reduce the amount of time required for retrieving massive remote sensing images, which facilitates practical applications.

TABLE IV
COMPUTATIONAL COST

| | Retrieval preprocessing phase | Retrieval phase | | | Verification phase |
|---|---|---|---|---|---|
| | Image owner | Blockchain | Retrieval user | CSP | Blockchain |
| The proposed scheme | $2mNDOT_{2m} + Pair_{G_T}$ $+(t+1)G_1 + 3G_T$ $+tHash_G$ | $L_nSear$ | $(t+1)Pair_{G_T}$ $+2tG_1 + (2+t)G_T$ $+2mDOT_{2m}$ | $L_nDOT_{2m}$ $+Build_{L_n}$ | $(\log_2 N + \log_2 L_n)Hash_K$ $+\tau DOT_{2m} + 2Comp$ |
| The proposed scheme without attributes | | $NSear$ | | $NDOT_{2m}$ $+Build_N$ | $2\log_2 N \cdot Hash_K$ $+\tau DOT_{2m} + 2Comp$ |

TABLE V
STORAGE COST

| | Retrieval pre-processing phase | Retrieval phase | |
|---|---|---|---|
| | Blockchain | CSP | Blockchain |
| The proposed scheme | $2N(1+\iota)+\iota$ | $N(2+\iota+\Gamma+\Lambda)+\iota$ | $\iota$ |
| The proposed scheme without attributes | | | |

TABLE VI
AVERAGE TEST TIME RESULTS

| Phase | Step | Executor | Time (sec) |
|---|---|---|---|
| Extract features from database images | | Image owner | 304.0614 |
| **Retrieval prepocessing** | Setup | KMC | 0.04584 |
| | Encryt image information | Image owner | 0.4073 |
| | Upload encrypted image information | Image owner | 15.3633 |
| | **Total** | | **15.8164** |
| Extract feature of query image | | Retrieval user | 1.4013 |
| **Retrieval** | Encrypt image information | Retrieval user | 0.0187 |
| | Similarity measurement | CSP | 0.0012 |
| | **Total** | | **0.0199** |
| **Verification** | | Blockchain | **15.3332** |

TABLE VII
TIME CONSUMPTION COMPARED WITH OTHER METHODS IN THE RETRIEVAL PHASE (SEC)

| Number of images | 200 | 400 | 800 | 1600 | 2000 |
|---|---|---|---|---|---|
| SMSC [11] | 0.0549 | 0.0588 | 0.0642 | 0.0792 | 0.0858 |
| BOEW [10] | 0.0556 | 0.0619 | 0.0783 | 0.1003 | 0.1163 |
| PLCIR [9] | 6.3595 | 13.1960 | 26.4264 | 55.3397 | 69.5177 |
| **Ours** | **0.011** | **0.0114** | **0.0199** | **0.0199** | **0.0199** |

### *D. Implementation in Different Object-Based Retrieval Schemes*

In this test, the proposed secure retrieval method (see Sections V-A and V-B) is employed to encrypt different object-based retrieval schemes (ReSW [38], RSFCN [23], and DHCNN [39]), and the UCM image dataset [36] is used for evaluation. Then, the retrieval performance is measured in both plain and cipher environments, as given in Fig. 7. The results show that the proposed secure retrieval method can maintain the same retrieval performance results as plain retrieval. Thus, it can be demonstrated that the proposed method serves as an effective security means to protect any object-based model against malicious attacks in a semitrusted cloud environment.

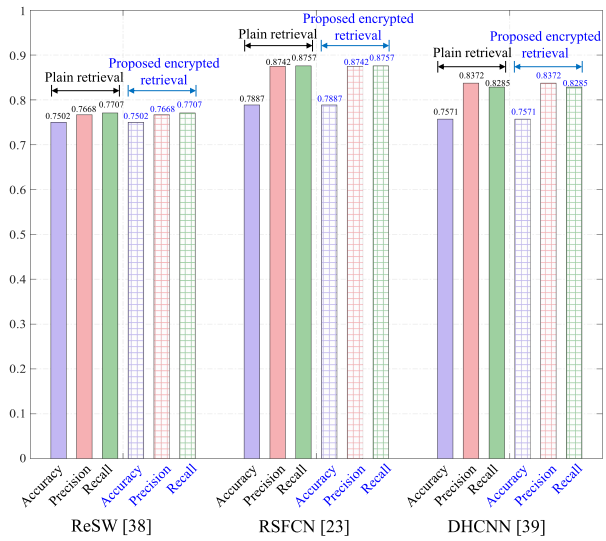


Fig. 7. Comparison of retrieval performance in the plain environment and cipher environment encrypted by the proposed retrieval method.

In addition, since the retrieval phase is operated online, we compare the time consumption of different schemes in the retrieval phase, including the plain retrieval and the encrypted retrieval based on the proposed secure retrieval method, and the results are given in Fig. 8. In particular, plain retrieval involves only the similarity measurement, while the proposed secure retrieval method involves the decryption feature keys, the encryption query image features, and the similarity measurement. Although encryption brings additional time consumption, we observe from the results that the time consumption of the

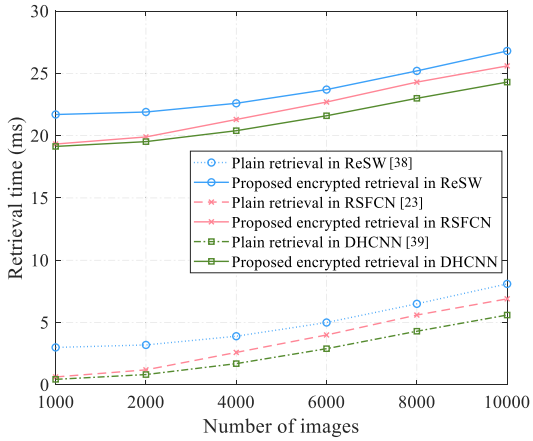


Fig. 8. Comparison of retrieval time in the plain environment and cipher environment encrypted by the proposed retrieval method (Millisecond).

proposed secure retrieval method is also acceptable. Therefore, it can be concluded that the proposed retrieval method is beneficial for practical applications as it maintains the same encryption performance as plain retrieval while consuming less time.

## VIII. Conclusion

In this article, we propose a blockchain-assisted verifiable and secure remote sensing image retrieval scheme in the cloud environment that achieves secure and efficient image retrieval while ensuring the correctness of retrieval results. In particular, we assign the geographical objects in remote sensing images as attributes due to the unique characteristics of remote sensing images. In the retrieval phase, the CSP only measures images with the same attributes as the query image, which can significantly improve the efficiency of image retrieval. In addition, we design a blockchain-assisted verifiable method to enable users to effectively verify the correctness and integrity of retrieval results, while dishonest behavior of CSP in the process of retrieval services can be recorded by the blockchain. We have proved the security and feasibility of the proposed scheme and conducted the experiment evaluation.

## References

[1] H. Tamiminia, B. Salehi, M. Mahdianpari, L. Quackenbush, S. Adeli, and B. Brisco, "Google earth engine for geo-big data applications: A meta-analysis and systematic review," *ISPRS J. Photogrammetry Remote Sens.*, vol. 164, pp. 152–170, 2020.

[2] P. Gao et al., "Secure cloud-aided object recognition on hyperspectral remote sensing images," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3287–3299, Mar. 2021.

[3] Q. Zhao, S. Chen, Z. Liu, T. Baker, and Y. Zhang, "Blockchain-based privacy-preserving remote data integrity checking scheme for iot information systems," *Inf. Process. Manage.*, vol. 57, no. 6, 2020, Art. no. 102355.

[4] K. N. Sukhia, M. M. Riaz, A. Ghafoor, and S. S. Ali, "Content-based remote sensing image retrieval using multi-scale local ternary pattern," *Digit. Signal Process.*, vol. 104, 2020, Art. no. 102765.

[5] J. Zhu, Q. Li, C. Wang, X. Yuan, Q. Wang, and K. Ren, "Enabling generic, verifiable, and secure data search in cloud services," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 8, pp. 1721–1735, Aug. 2018.

[6] Q. Tong et al., "VFIRM: Verifiable fine-grained encrypted image retrieval in multi-owner multi-user settings," *IEEE Trans. Serv. Comput.*, vol. 15, no. 6, pp. 3606–3619, Nov./Dec. 2022.

[7] W. Lu, A. L. Varna, A. Swaminathan, and M. Wu, "Secure image retrieval through feature protection," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2009, pp. 1533–1536.

[8] Y. Zhang, L. Zhuo, Y. Peng, and J. Zhang, "A secure image retrieval method based on homomorphic encryption for cloud computing," in *Proc. Int. Conf. Digit. Signal Process.*, 2014, pp. 269–274.

[9] L. Weng, L. Amsaleg, A. Morton, and S. Marchand-Maillet, "A privacy-preserving framework for large-scale content-based information retrieval," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 1, pp. 152–167, Jan. 2015.

[10] Z. Xia, L. Jiang, D. Liu, L. Lu, and B. Jeon, "BOEW: A content-based image retrieval scheme using bag-of-encrypted-words in cloud computing," *IEEE Trans. Serv. Comput.*, vol. 15, no. 1, pp. 202–214, Jan./Feb. 2022.

[11] M. Shen, G. Cheng, L. Zhu, X. Du, and J. Hu, "Content-based multi-source encrypted image retrieval in clouds with privacy preservation," *Future Gener. Comput. Syst.*, vol. 109, pp. 621–632, 2020.

[12] W. Huang, A. Ganjali, B. H. Kim, S. Oh, and D. Lie, "The state of public infrastructure-as-a-service cloud security," *ACM Comput. Surv.*, vol. 47, no. 4, pp. 1–31, 2015.

[13] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 5, pp. 847–859, May 2011.

[14] Y. Zhang, C. Xu, X. Lin, and X. Shen, "Blockchain-based public integrity verification for cloud storage against procrastinating auditors," *IEEE Trans. Cloud Comput.*, vol. 9, no. 3, pp. 923–937, Sep. 2021.

[15] J. Li, J. Wu, G. Jiang, and T. Srikanthan, "Blockchain-based public auditing for Big Data in cloud storage," *Inf. Process. Manage.*, vol. 57, no. 6, 2020, Art. no. 102382.

[16] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[17] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues and challenges," *IEEE Commun. Surv. Tut.*, vol. 21, no. 2, pp. 1508–1532, Secondquarter 2019.

[18] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu, "Blockchain based data integrity service framework for IoT data," in *Proc. IEEE Int. Conf. Web Serv.*, 2017, pp. 468–475.

[19] T. Wu, G. Yang, Y. Mu, R. Chen, and S. Xu, "Privacy-enhanced remote data integrity checking with updatable timestamp," *Inf. Sci.*, vol. 527, pp. 210–226, 2020.

[20] Storj Labs, "Storj: A decentralized cloud storage network framework," 2018. [Online]. Available: https://www.storj.io/storj.pdf

[21] D. Vorick and L. Champine, "Sia: Simple decentralized storage," 2014. [Online]. Available: https://sia.tech/sia.pdf

[22] Protocol Labs, "Filecoin: A decentralized storage network," 2017. [Online]. Available: https://filecoin.io/filecoin.pdf

[23] Z. Shao, W. Zhou, X. Deng, M. Zhang, and Q. Cheng, "Multilabel remote sensing image retrieval based on fully convolutional network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 318–328, Jan. 2020.

[24] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of abe ciphertexts," in *Proc. 20th USENIX Conf. Secur.*, 2011, Paper no. 34.

[25] Y. Li, J. Ma, and Y. Zhang, "Image retrieval from remote sensing Big Data: A survey," *Inf. Fusion*, vol. 67, pp. 94–115, 2021.

[26] Y. Liu, L. Ding, C. Chen, and Y. Liu, "Similarity-based unsupervised deep transfer learning for remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 7872–7889, Nov. 2020.

[27] Y. Liu, Y. Liu, C. Chen, and L. Ding, "Remote-sensing image retrieval with tree-triplet-classification networks," *Neurocomputing*, vol. 405, pp. 48–61, 2020.

[28] Y. Liu, C. Chen, Z. Han, L. Ding, and Y. Liu, "High-resolution remote sensing image retrieval based on classification-similarity networks and double fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1119–1133, Mar. 2020.

[29] W. Zhou, S. Newsam, C. Li, and Z. Shao, "Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval," *ISPRS J. Photogrammetry Remote Sens.*, vol. 145, pp. 197–209, 2018.

[30] A. Fu, S. Yu, Y. Zhang, H. Wang, and C. Huang, "NPP: A new privacy-aware public auditing scheme for cloud data sharing with group users," *IEEE Trans. Big Data*, vol. 8, no. 1, pp. 14–24, Feb. 2022.

[31] B. Ravishankar, P. Kulkarni, and M. Vishnudas, "Blockchain-based database to ensure data integrity in cloud computing environments," in *Proc. Int. Conf. Mainstreaming Block Chain Implementation*, 2020, pp. 1–4.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[33] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.
[34] Y. Zhou, L. Bao, and C. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, pp. 172–182, 2014.
[35] Z. Brakerski, C. Gentry, and S. Halevi, "Packed ciphertexts in LWE-based homomorphic encryption," in *Proc. Int. Workshop Public Key Cryptogr.*, 2013, pp. 1–13.
[36] B. Zhao, Y. Zhong, G. Xia, and L. Zhang, "Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2108–2123, Apr. 2016.
[37] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone, "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 1144–1158, Feb. 2018.
[38] S. Pang, J. Zhu, J. Wang, V. Ordonez, and J. Xue, "Building discriminative CNN image representations for object retrieval using the replicator equation," *Pattern Recognit.*, vol. 83, pp. 150–160, 2018.
[39] W. Song, S. Li, and J. A. Benediktsson, "Deep hashing learning for visual and semantic retrieval of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9661–9672, Nov. 2021.

**Xue Ouyang** received the M.S. degree in electronics and communications engineering from Guangxi Normal University, Guilin, China, in 2020. She is currently working toward the Ph.D. degree in communication and information systems from the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China.

Her research interests include blockchain and clouding computing security.

**Yanyan Xu** received the Ph.D. degree in communication and information system from Wuhan University, Wuhan, China, in 2007.

She is currently a Professor with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University. She has published more than 50 research papers and one book. Her research interests include information security, multimedia communication systems.

**Yangsu Mao** received the master's degree in electronic information in 2022 from the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China, where he is currently working toward the Ph.D. degree in communication and information systems.

His research interests include multimedia information processing, security, and blockchain applications.

**Yunqi Liu** received the master's degree in electronic and communication engineering from Guangxi Normal University, Guilin, China, in 2018. He is currently working toward the Ph.D. degree in cyberspace security with the School of Cyber Science and Engineering, Wuhan University, Wuhan, China.

His research interests include deep learning and AI security.

**Zhiheng Wang** received the B.S. degree in electronic and information engineering from Zhengzhou University, Zhengzhou, China, and the M.S. degree in communication and information systems from Wuhan University, Wuhan, China, where he is currently working toward the Ph.D. degree in photogrammetry and remote sensing.

His research interests include indoor positioning security, security authentication, and biometrics.

**Yuejing Yan** received the B.S. and M.S. degrees in information security and computer technique in 2016 and 2019, respectively, from Wuhan University, Wuhan, China, where she is currently working toward the Ph.D. degree with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing.

Her research interest focuses on information security and privacy issues in cloud computing.