Journal of Intelligent Systems and Internet of Things
Vol. 17, No. 01, PP. 129-144, 2025
DOI: https://doi.org/10.54216/JISIoT.170110

SecureRS-CBIR: A Privacy-Preserving Deep Learning Framework for Content-Based Remote Sensing Image Retrieval

Ahmed Sabah Ahmed AL-Jumaili[1,*], Huda Kadhim Tayyeh[2]

[1]Department of Business Information Technology (BIT), College of Business Informatics, University of Information Technology and Communications, Iraq

[2]Department of Informatics Systems Management (ISM), College of Business Informatics, University of Information Technology and Communications, Iraq

Emails: asabahj@uoitc.edu.iq; haljobori@uoitc.edu.iq

Abstract

Recent advancements in Remote Sensing (RS) have created challenges in data storage, retrieval, and privacy. Existing Content-Based Image Retrieval (CBIR) systems are useful but often face limitations related to hypersensitivity towards remote sensing data in the cloud, scalability, and security. This article presents SecureRS-CBIR, a privacy-preserving framework for remote sensing image retrieval combining deep learning with multi-level encryption. The system uses three CNN models (VGG16, ResNet50, and DenseNet121) for feature extraction and implements encryption through image division, texture extraction, subblock shuffling, and color encryption. Experiments on the Aerial Image Dataset show VGG16 achieving 96% validation accuracy, with ResNet50 and DenseNet121 at 95% and 94% respectively. DenseNet121 excelled at DenseResidential classification (41/42 correct) with minor confusion between Beach and Desert categories. The framework successfully balances security with retrieval efficiency, maintaining privacy through robust encryption while enabling accurate content-based searches, providing a scalable solution for secure image retrieval in cloud environments. This work offers a new approach for remote sensing image retrieval by enabling efficient searching in large-scale datasets while addressing privacy concerns in cloud environments, thereby contributing to the relevant literature.

Received: January 21, 2025 Revised: February 19, 2025 Accepted: March 21, 2025

Keywords: Content-Based Image Retrieval; Remote Sensing; Privacy Preservation; Artificial Intelligence; Deep Learning; Multi-level Encryption; Convolutional Neural Networks; Image Security; Cloud Computing; Feature Extraction; Aerial Image Dataset

1. Introduction

In the context of imaging and data management, as well as for high-level applications like social media, cloud-based image retrieval systems have become critical components [1]. In the past, these systems depended exclusively on metadata or image tagging to enable image search and retrieval. While these methods often work for simpler and smaller datasets, like those emerging from remote sensing technologies, these datasets are extraordinarily complex and multi-faceted [2]. The limitation is more pronounced in terms of scalability, where manual tagging is infeasible, while search based on structured metadata lacks precision in capturing the content of the images [3]. Recently developed privacy-preserving techniques have further enhanced the security of image retrieval from the cloud, including approaches like the privacy-preserving thumbnail-based JPEG image retrieval put forward by Ma et al. [4] Which sought to balance security and efficiency. The introduction of Artificial Intelligence (AI) marked the beginning of a different epoch concerning image retrieval, as new methods that can directly comprehend and analyze the content of images have emerged. AI image retrieval systems utilize image content features through machine learning algorithms, which provide the ability to search for images and retrieve them using their content. This certainly increases the accuracy of retrievals, but most importantly improves the experience of users who are enabled to formulate search queries in more natural ways [5].

More importantly, in remote sensing, there is an acute need for advanced retrieval systems. Remote sensing technologies have progressed remarkably, with the scale and quality of remote sensing images increasing

exponentially [6]. These images are a cornerstone in addressing emerging problems of great importance in several fields like weather forecasting, climate change studies, urban studies, geology, and monitoring of disasters, etc. Content-Based Remote Sensing Image Retrieval (CBRSIR) is fast becoming one of the important areas of research that looks into such large datasets taking into consideration the specific information requirements of users to allow efficient retrieval of relevant images from huge databases [7]. Zhang et al. [8] further advanced this area by proposing deep attention hashing with distance-adaptive ranking, which improves the efficiency of remote sensing image retrieval in large-scale datasets.

Nonetheless, the extreme volume of data generated through remote sensing activities poses a serious problem to CBRSIR frameworks. The 'big data' issue in remote sensing image retrieval includes but is not limited to, data volume, variety, and velocity [9]. Most traditional and some AI-based retrieval algorithms do not perform well at scale due to the bounds on memory and computational efficiency. The already large amount of computing, memory, and energy required to process remote sensing images is complicated by their data abundance, complexity, and high dimensionality [10]. Privacy issues in distributed systems as discussed by Zhou et al. [11] complicate the design of effective retrieval systems even further. Deep learning techniques have become one of the best solutions to the problems posed because they are remarkable at feature extraction and analyzing images. For example, the model can learn remote sensing image features automatically; hence it can capture the subtle details of such images and has much retrieval accuracy [6]. As Chen et al. [12] showed with their work that the MERGING of deep learning and traditional techniques also proves useful; they combined Sobel operator techniques and other advanced algorithms for radar remote sensing image retrieval.

The advancements in the domain of deep learning and image recognition have been enormous as various frameworks were developed. For instance, the image recognition domain received great attention with the introduction of the Very Deep Convolutional Networks (VGG), which showed that the network depth significantly impacts network performance on image recognition tasks, as noted by Simonyan and Zisserman [13]. Building upon this, He et al. [14] proposed Deep Residual Networks (ResNet), which effectively addressed the issue of degradation in deep networks by suggesting residual connections that facilitate the training of much deeper networks and achieving excellent performance on various benchmarks.

To promote feature reuse and maintain high performance in the presence of a reduced number of parameters, Huang et al. [15] presented the Densely Connected Convolutional Networks (DenseNet), which introduced additional architectural innovations by creating direct connections between any layer and all subsequent layers. More recently, Tan and Le [16] introduced EfficientNet, which proposed a principled method for scaling up networks in terms of depth, width, and resolution, achieving better performance with fewer parameters than previous models.

Different deep learning (DL) models have been tested for performance in feature extraction and recognition and found to rely highly on the input data's spatial resolution, spectral bands, and scene complexity. This variation in performance needs a solution in the form of a framework combining multiple deep-learning models for dependable and vigorous image retrieval in heterogeneous remote sensing datasets. A complete framework can be designed using an ensemble of complementary CNNs with different architectural styles and depths so that each model's strengths are utilized while alleviating its weaknesses. This research introduces SecureRS-CBIR, a framework using multiple deep learning models to enable effective retrieval of remote sensing images while ensuring privacy protection in cloud environments.

2. Related Work

The advances in technology have driven rapid development in the area of CBIR. Several approaches have emerged across domains like cloud computing and remote sensing in view of digital data proliferation and the need for sophisticated retrieval mechanisms. The current literature survey categorizes the approaches into privacy-preserving techniques, efficient feature extraction techniques, and optimization methods aimed at improving retrieval efficiency; each approach is evaluated individually to understand its role in addressing the complex problems associated with CBIR systems.

The importance of data privacy in CBIR has been noted in a good number of studies and various methods have been proposed to secure data in the cloud. Many scholars have described these methods, which include preserving privacy with respect to maintaining utility in CBIR systems [11], [17], [18]. These methods use encryption, blockchain, and secure multi-party computation to sensitive data that is fetched during retrieval. For instance, a cross-media retrieval method that preserves privacy for encrypted data and increases the security of data on cloud platforms has been presented [17]. Furthermore, blockchain-assisted and dynamic verifiable retrieval schemes that demonstrate the vast potential of enhanced cryptographic techniques for protecting data confidentiality and integrity have been presented [19] [20]. In the case of Liang et al. [21], they developed a privacy-preserving Bloom filter-based keyword search for large encrypted cloud data which advanced the field by providing an efficient mean for similarity searches in secure environments.

The precision in retrieving images from databases is highly dependent on the extraction of meaningful features, and this applies to CBIR. Scholars have analyzed methods of feature extraction which include the application of algorithms and even deep learning models that are tailored to improve image matching and retrieval [22], [23], [24], [25] [26]. The authors proposed a CBIR approach that combines color and texture features to achieve higher retrieval results. Also, the deep learning approach to feature extraction in remote sensing image retrieval, as proposed by many scholars, marks a transition toward more sophisticated automated understanding with regard to interpreting images [27], [12], [25], [28]. Al Rahhal et al. [28] introduced the multilanguage transformer for text-to-remote sensing image retrieval, which significantly enhances cross-modal retrieval systems, merging verbal commands with spatial data.

To enhance computational efficiency and accuracy, several studies suggest optimization methods aimed at improving the performance of CBIR systems. For example, [4], [12], and [29] concentrate on clustering, thumbnail-preserving encryption, and thumbnail-accelerated retrieval algorithm improvements to reduce retrieval time and enhance precision. These optimizations also help improve the scalability of CBIR systems because of the large dataset requirements, which is important in cloud computing and remote sensing applications. The study conducted by Anju & Shreelekshmi [30] proposed PCBIR-CV, a content-based image retrieval system that preserves privacy using combined visual descriptors for cloud environments, which is another solution that achieves optimal security, efficiency, and accuracy.

Even with these notable developments in privacy preservation, feature extraction, and optimization techniques, there remains an undisputably serious gap in the literature on the seamless integration of these components into a framework tailored for remote sensing applications. It is clear that no comprehensive frameworks incorporating multiple deep learning-based feature extraction algorithms with strong privacy guarding mechanisms have been put forward despite various individual studies successfully addressing specific components of CBIR. Most existing works depend on singular deep learning models VGG [13], ResNet [14], DenseNet [15], or EfficientNet [16] which are unlikely to provide the desired results because of the differing remote sensing image spatial resolution, spectral band, and scene complexity. Moreover, the available privacy-preserving techniques concentrate on generic image datasets and do not consider unique remote-sensing data challenges such as the need to maintain spatial and spectral relationships during encryption. This research gap underscores the necessity for a comprehensive framework that not only leverages the complementary strengths of multiple deep learning models but also implements specialized privacy-preserving techniques tailored to remote sensing applications, ultimately providing a more robust and reliable solution for secure content-based remote sensing image retrieval in cloud environments.

3. Methodology

3.1. Mathematical Notation and Terminology

To ensure clarity and consistency throughout the methodology and experimental sections, Table 1 presents a comprehensive list of all mathematical symbols and notation used in this paper. This standardized notation facilitates the precise formulation of the privacy-preserving mechanisms, deep learning architectures, and security analysis that form the core of the SecureRS-CBIR framework.

Table 1: Comprehensive list of mathematical symbols used in the paper

Symbol	Description
I	Image database consisting of remote sensing images $\{i_1, i_2, \ldots, i_n\}$
i_n	An individual image in the image database
K	Set of unique encryption keys $\{k_1, k_2, \ldots, k_n\}$
k_n	An individual encryption key for an image
E	Encrypted image database $\{e_1, e_2, \ldots, e_n\}$
e_n	An individual encrypted image
ψ	Feature extraction model trained on encrypted images
E_Q	Encrypted query image

F_E	Set of features extracted from the encrypted image database $\{f_{e1}, f_{e2}, \ldots, f_{en}$
F_{E_Q}	Features extracted from the encrypted query image
E_R	Set of identifiers for the k most similar images $\{er_{ID1}, er_{ID2}, \ldots, er_{IDk}\}$
ID^R	Set of image identifiers in the query result set $\{ID_{R1}, ID_{R2}, \ldots, ID_{Rk}\}$
RK	Set of decryption keys for the retrieved images $\{rk_{IDR1}, rk_{IDR2}, \ldots, rk_{IDRk}\}$
R	Final retrieval set of decrypted images $\{r_{IDR1}, r_{IDR2}, \ldots, r_{IDRk}\}$
r_{IDRK}	An individual image in the final retrieval set
$X \epsilon R^{H \times W \times C}$	Input image tensor with height H, width W, and C channels
F_l	Feature maps at layer l
W_l	Weights for layer l
b_l	Bias terms for layer l
σ	ReLU activation function
$*$	Convolution operation
P_l	Pooling operation output at layer l
v_{VGG}	Final feature vector from VGG16 after Global Average Pooling
y_ℓ	Output of residual block at layer ℓ in ResNet50
$F(x_\ell, W_\ell)$	Residual mapping function in ResNet50
x_ℓ	Input to layer ℓ in DenseNet121
H_ℓ	Composite function of operations at layer ℓ in DenseNet121
k	Growth rate parameter in DenseNet121
$\odot$	Depthwise convolution operation
I	Original image for encryption
$subI_i$	A subblock of the original image
$subE_i$	An encrypted subblock
$subK_i$	A subblock key
π	Permutation for subblock shuffling

Φ	Color encryption function
$\lvert k_i \rvert$	Length of the key k_i in bits
$P(\pi)$	Probability of predicting the correct permutation
$P(\phi)$	Probability of guessing the correct encryption key
H', W'	Height and width dimensions after processing
F_L	Feature maps of the last convolutional layer
m	Number of divisions along each dimension for image subblocks
w, h	Width and height of the original image
num	Random number used in the shuffling process
E	Fully encrypted image
K	Comprehensive image key

3.2. Problem Formulation

This stage involved hree major entities – the data owner, the query-user, and the cloud-server. The image database $I = \{i_1, i_2, \ldots, i_n\}$, which contains n remote sensing photos, belongs to the data wner. An encrypted image database $E = \{e_1, e_2, \ldots, e_n\}$ is produced by encrypting each image contained in the database with a distinct key from the collection $K = \{k_1, k_2, \ldots, k_n\}$. To extract features from encrypted photos, the data owner additionally trains a feature extractor Ψ with E. After that, the model Ψ and the encrypted database E are sent to the cloud server for deployment and storage.

The feature extraction model Ψ and the encrypted picture database E are stored on the cloud server, which acts as the foundation for computation and storage. When a query user sends an encrypted query picture E_Q, the server uses Ψ to parse E and E_Q to extract their features, using F_{EQ} for the query image and $F_E = \{f_{e1}, f_{e2}, \ldots, f_{en}\}$ for the database images. The Euclidean distance is a metric for determining the k most comparable images by comparing the similarity between F_E and F_{EQ}. The query user is then provided with the image identifiers, $E_R = \{er_{ID1}, er_{ID2}, \ldots, er_{IDk}\}$.

Before sending the query image to the cloud-server, it must be first encrypted by the query-user to E_Q to retrieve photos that are comparable to the querry image Q. The query user provides the data owner with these $ID^R = \{ID_{R1}, ID_{R2}, \ldots, ID_{Rk}\}$ to get the related decryption keys $RK = \{rk_{IDR1}, rk_{IDR2}, \ldots, rk_{IDRk}\}$ after obtaining the encrypted query result set E_R from the cloud server. These keys can be used to decrypt the returned images to arrive at the final retrieval set $R = \{r_{IDR1}, r_{IDR2}, \ldots, r_{IDRk}\}$. Table 1 lists the mathematical symbols used in this formulation.

3.3. Architecture

Figure 1 depicts a conceptual diagram of the architecture; the data owner for the processing and storage of the encrypted images first uploads the encrypted images and the feature extractor to the cloud user. Following the submission of an encrypted query image by the user to the cloud, the server extracts the pertinent features from both the query image and the encrypted image database using the feature extractor. The amount of similarity between the features is computed and the identifiers of the most similar images are sent back to the query user. The query user in turn fetches the corresponding decryption keys from the data owner, retrieves the images, and performs decryption to assemble the final retrieval set. This technique strives for adequate retention of privacy, computational load, and efficiency of retrieval in a content-based image recovery system in the cloud specially designed for remote sensing use cases.

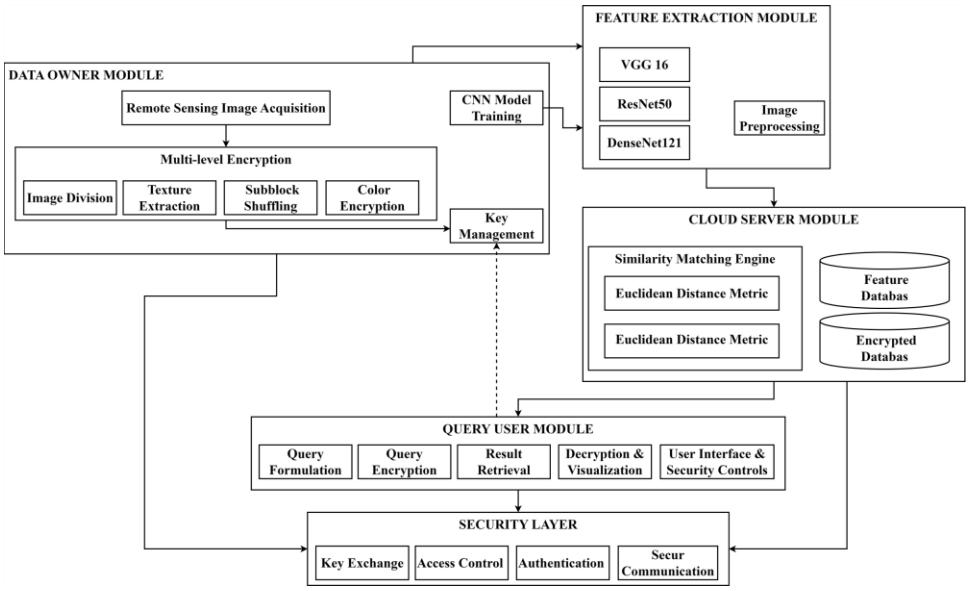

Figure 1. Diagram illustrating the process flow of an encrypted image retrieval system for remote sensing applications

3.4. SecureRS-CBIR Framework

The SecureRS-CBIR framework, illustrated in Figure 2, presents a comprehensive architecture for privacy-preserving remote sensing image retrieval. The system comprises three interconnected modules that work together to ensure secure and efficient image retrieval: (1) a Query User Module handling the encryption of query images and decryption of retrieved results, (2) a Cloud Processing Module containing query processing, similarity matching, and Top-K results generation, and (3) a Feature Extraction Module that processes remote sensing images through multiple CNN architectures. The framework employs three distinct CNN models for feature extraction: VGG16 (13 convolutional layers, 3 FC layers, 138M parameters), ResNet50 (48 convolutional layers, 1 FC layer, 23M parameters), and DenseNet121 (120 convolutional layers, dense connections, 7M parameters). These diverse architectures process the encrypted images in parallel, with their outputs feeding into a central feature database that supports the similarity-matching process. This modular design ensures end-to-end privacy preservation through the encryption-to-decryption pipeline while maintaining high retrieval accuracy through the parallel implementation of diverse deep learning models, each optimized to capture different aspects of remote sensing imagery features.

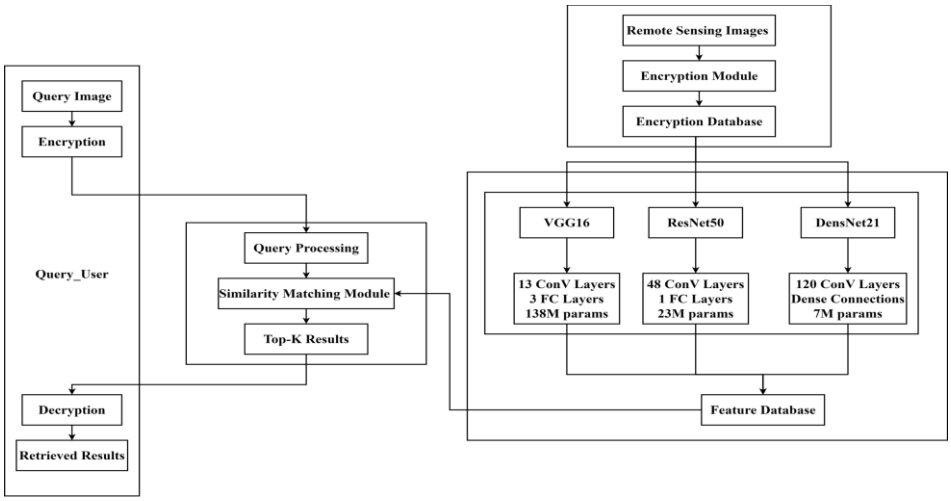

Figure 2. SecureRS-CBIR: A privacy-preserving content-based image retrieval framework leveraging deep CNN architectures and multi-level encryption for secure and efficient remote sensing image analysis.

The SecureRS-CBIR framework employs three established CNN architectures for feature extraction: VGG16 [13], ResNet50 [14], and DenseNet121 [15].

3.5. Algorithm

The encryption process of an image database, especially for remote sensing, includes multi-stage security and quick retrieval of images. The entire process consists of several steps: image division, intrablock texture extraction, subblock shuffling, and color encryption, followed by the process of merging encrypted subblocks with keys. At first, the original image, I, is partitioned into subblocks which transforms the image into m^2 subblocks $\{subl_1, subl_2, \ldots, subl_{m^2}\}$, each of size $\frac{w}{m} \times \frac{h}{m} \times 3$. The described segmentation supports parallel processing, particularly improving security through enabling local encryption. Each subblock undergoes intrablock texture extraction, resulting in an encrypted subblock subEi and a subblock key $subK_i$. This step guarantees that each segment of that image is individually encrypted and thereby increases security. To increase security, the subblocks are shuffled with their corresponding keys. For each subblock i, shuffling random number num is generated in reverse order from the range of m^2. The keys subK are thus shuffled with the blocks subE according to these random indices, which guarantees appropriate concealment of spatial relations within the image. After this shuffling, each subblock is color encrypted to add another layer of encryption towards unlicensed decryption of subblock color information. At last, all subblocks are combined into the fully encrypted image E, and all keys from the shuffle are mixed with the keys from each subblock to form the comprehensive image key K. This complex, yet efficient break, provides image retention capabilities in cloud systems while maximizing security in the encryption to achieve optimal ease of access. The remote sensing data is especially suited for sensitive data transmission because of the incorporation of texture extraction, shuffling, and color encryption. Figure 3 illustrates the corresponding sequence diagram.

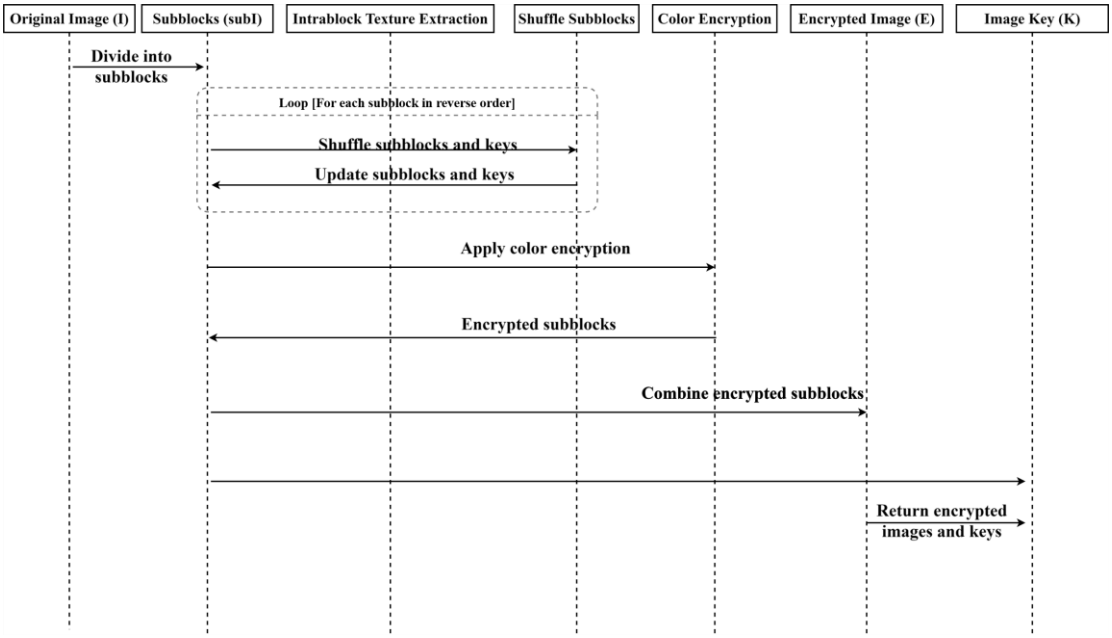

Figure 3. Block diagram illustrating the multi-step encryption process for secure image retrieval in remote sensing applications

3.6. VGG16 model

The VGG16 model processes images in the form of tensors $X \in R^{H \times W \times C}$. Here, H is the height and W is the width; C is the number of channels. Each convolutional layer in the network performs the following operation:

$$F_l = \sigma(W_l * F_{i-1} + b_l) \quad (1)$$

This equation represents the fundamental building block of VGG16, where W_l denotes the layer weights, b_l represents the bias terms, and σ is the ReLU activation function. The convolution operation $(*)$ is explicitly defined as:

$$F_{l(i,j)} = \sum_{\{m=0\}}^{\{k-1\}} \sum_{\{n=0\}}^{\{k-1\}} W_l(m,n)\, F_{I-1}(i+m, j+n) \tag{2}$$

The max pooling operation, crucial for reducing spatial dimensions, is defined as:

$$P_l(i,j) = \max_{0m,n1} F_l(2i+m, 2j+n) \tag{3}$$

The final feature vector is obtained through Global Average Pooling (GAP):

$$v_{VGG} = \frac{1}{H'W'} \sum_{i=1}^{\{H'\}} \sum_{j=1}^{\{W'\}} F_L(i,j) \tag{4}$$

3.7. WResNet50 Architecture

ResNet50 introduces the revolutionary residual learning framework, where each block computes:

$$y_\ell = F(x_\ell, W_\ell) + x_\ell \tag{5}$$

This formulation represents the core innovation of ResNet, where x_ℓ is the input to layer ℓ, and $F(x_\ell, W_\ell)$ is the residual mapping. The bottleneck structure consists of three operations:

$$F_1 = \sigma(W_1 * x_\ell)(1 \times 1\text{conv}) \tag{6}$$

$$F_2 = \sigma(W_2 * F_1)(3 \times 3\text{conv}) \tag{7}$$

$$F_3 = W_3 * F_2(1 \times 1\text{conv}) \tag{8}$$

The final output includes the skip connection:

$$y_\ell = \sigma(F_3 + x_\ell) \tag{9}$$

3.8. DenseNet121 Architecture

DenseNet121 implements dense connectivity, where each layer receives input from all preceding layers:

$$x_\ell = H_\ell([x_0, x_1, \ldots, x_{\ell-1}]) \tag{10}$$

The growth rate k determines how the feature maps expand:

$$|x_\ell| = |x_0| + k(\ell - 1) \tag{11}$$

3.9. Security Analysis

To evaluate the security of the encryption algorithm for remote sensing applications mathematically, we need to examine each step's contribution to the overall security and rigorously identify potential vulnerabilities.

3.9.1. Intrablock Texture Extraction

Let I be the original image, divided into m^2 subblocks $\{subI_1, subI_2, \ldots, subI_{m^2}\}$. Each subblock $subI_i$ undergoes a transformation via the function ψ:

$$(subE_i, subK_i) = \Psi(subI_i) \tag{12}$$

where $\mathrm{subE_i}$ is the encrypted subblock, and $\mathrm{subK_i}$ is the key associated with $\mathrm{subI_i}$. Assuming ψ is a bijective function with high entropy, the probability of two different subblocks resulting in the same encrypted subblock is negligible. Mathematically, if $\mathrm{subI_i} \neq \mathrm{subI_j}$, then:

$$\Pr\left(\Psi(subI_i) = \Psi\left(subI_j\right)\right) \approx 0 \tag{13}$$

3.9.2. Subblock Shuffling

The shuffling process involves generating a permutation π of the set $\{1, 2, \ldots, \mathrm{m}^2\}$ such that:

$$\pi: \{1,2,\ldots,m^2\} \rightarrow \{1,2,\ldots,m^2\} \tag{14}$$

For each subblock $\mathrm{subE_i}$ and key $\mathrm{subK_i}$, the algorithm swaps positions based on the permutation π. The security of this process relies on the randomness of π. The number of possible permutations is $\mathrm{m}^2!$. If π is generated using a secure random number generator, the probability $\mathrm{P}(\pi)$ of predicting the permutation without knowing the key is:

$$P(\pi) = \frac{1}{m^2!} \tag{15}$$

3.9.3. Color Encryption

The color encryption function Φ applied to each subblock $\mathrm{subE_i}$ is assumed to be a secure encryption function. For simplicity, let us consider it as a block cipher with a key $\mathrm{k_i}$:

$$subE_i' = \Phi(subE_i, k_i) \tag{16}$$

Assuming Φ is secure; the probability $\mathrm{P}(\Phi)$ of an attacker guessing the correct key $\mathrm{k_i}$ for a single subblock depends on the key length. Let $|\mathrm{k_i}|$ represent the length of the key $\mathrm{k_i}$ in bits. For a secure block cipher, the probability of correctly guessing the key by brute force is:

$$P(\Phi) = \frac{1}{2^{|k_i|}} \tag{17}$$

For instance, if the key length $|\mathrm{k_i}|$ is 128 bits (a common length for secure encryption), the probability of an attacker guessing the correct key for a single subblock is:

$$P(\Phi) = \frac{1}{2^{128}} \tag{18}$$

Commonly used lengths are 128, 192, or 256 bits.

1. 128-bit Key:

$$P(\Phi) = \frac{1}{2^{128}} \approx 2.94 \times 10^{-39} \tag{19}$$

2. 192-bit Key:

$$P(\Phi) = \frac{1}{2^{192}} \approx 6.28 \times 10^{-58} \tag{20}$$

3. 256-bit Key:

$$P(\Phi) = \frac{1}{2^{236}} \approx 1.57 \times 10^{-77} \quad (21)$$

The final encrypted image E is formed by combining all $subE_i'$ subblocks. The image key K is a combination of all subblock keys and the shuffle key. Let $K = \{k_i, \pi\}$ represent the combined key set. The security of K relies on both the individual subblock keys k_i and the permutation π. The total key space size is:

$$|K| = 2^{|k_i| \cdot m^2} \cdot m^2! \quad (22)$$

This indicates that the difficulty of breaking the encryption grows exponentially with the number of subblocks and the key size. However, there are some of the potential vulnerabilities:

1. Key Management: The security of the algorithm is contingent on the secure management of K. If an attacker gains access to K, they can decrypt the entire image. The mathematical security relies on ensuring |K| remains secret.
2. Random Number Generator: The security of the shuffling process depends on the entropy of the random number generator used to generate π. A weak random number generator reduces $P(\pi)$, compromising the shuffle's security.
3. Implementation Security: Practical implementation must avoid side-channel attacks and ensure that operations like Ψ and Φ are resistant to timing attacks, differential power analysis, etc. Mathematical security assumes ideal implementations.

3.9.4. Complexity Analysis

The encryption algorithm for the image database, particularly designed for remote sensing applications, involves several distinct steps: image division, intrablock texture extraction, subblock shuffling, color encryption, and the combination of encrypted subblocks and keys.

Below is a detailed complexity analysis for each of these steps. The image division step involves partitioning the original image I into m^2 subblocks. The complexity of this operation is $O(w \cdot h)$, as each pixel in the image needs to be assigned to a subblock. Given the size of the image is $w \times h$, this step scales linearly with the number of pixels.

In the intrablock texture extraction step, the texture and key for each subblock are extracted. Let t be the time taken to perform texture extraction on a single subblock. The total time burden will now be $O(m^2 \cdot t)$ for m^2. Each subblock is of size $\frac{w}{m} \times \frac{h}{m}$, thus the entire texture extraction algorithm is dominated by the subblock size which leads to $O\left(\frac{w \cdot h}{m^2} \cdot m^2\right) = O(w \cdot h)$.

The shuffling of subblocks consists of two steps: first, iterating through the m^2 subblocks in reverse order; second, swapping each subblock with a randomly chosen subblock. For this specific shuffling step, the time burden equates to $O(m^2)$, which is linear in proportion to the subblocks.

The step of color encryption is done by encrypting the color information of every subblock. If we let c be the time complexity for color encryption for a single subblock, and considering there are m^2 subblocks, the total complexity is $O(m^2 \cdot c)$. Since each subblock's size is $\frac{w}{m} \times \frac{h}{m}$, the color encryption process has a complexity that corresponds to the subblock size, hence it is $O\left(\frac{w \cdot h}{m^2} \cdot m^2\right) = O(w \cdot h)$.

In the last step, the encrypted subblocks are combined to derive the final encrypted image and the keys to form the final image. The process of combining m^2 subblocks and their keys are given as $O(m^2)$; it is a linear operation in relation to the number of subblocks.

As evaluated earlier, the major contributor to the complexity of the encryption algorithm is mainly the steps involved in image division, intrablock texture extraction, and color encryption; the complexity for each step is given as $O(w \cdot h)$, meaning that the total complexity of the algorithm is $O(w \cdot h$. Hence, the algorithm's efficiency is directly a function of the pixel count of the image that is an important factor for remote sensing applications where images frequently exceed terabyte sizes.

4. Experimental Results and Analysis

The purpose of the experiment on the SecureRS-CBIR framework was to test its security and retrieval accuracy features. In this sub-section, the details of the selected dataset, experimental configurations, methods of measuring performance, and analysis of several deep learning models in the context of the framework are provided.

4.1. Experimental Design

This research developed and evaluated a privacy-preserving CBIR system using an elaborate multi-stage methodology. The AID (Aerial Image Dataset) used in this study contained three representative classes: Beach, Dense Residential, and Desert. The selection of these classes was based on their sufficiency in feature diversity and adequate computational cost. The first phase of the system-processing pipeline is data augmentation to add more complementing images to the dataset. Images were retrieved at the 224 x 224-pixel size as described in Table 2, and augmentation parameters included a rotational range of 20 degrees as well as a brightness zoom level ranging from 0.8 to 1.2. This augmented dataset then underwent a two-level encryption scheme, implementing block-wise encryption with 2x2 sub-blocks, followed by normalization using the standard ImageNet statistics (mean: [0.485, 0.456, 0.406], standard deviation: [0.229, 0.224, 0.225]).

For feature extraction, the system employed three pre-trained CNN architectures: VGG16, ResNet50, and DenseNet121. Each model underwent fine-tuning on the encrypted dataset over 10 epochs with a batch size of 32, as specified in Table 2. The training process involved freezing early layers while keeping the last 10 layers trainable and adding a Global Average Pooling layer with a task-specific classification head.

Table 2: Configuration parameters for the enhanced privacy-preserving CBIR system including image pre-processing, model training, and fusion settings.

Parameter	**Value**
Image dimensions	(224, 224)
Rotation range	20
Brightness range	(0.8, 1.2)
Zoom range	(0.8, 1.2)
Epochs	10
Batch size	32
Number of sub-blocks	2
Mean	[0.485, 0.456, 0.406]
Standard deviation	[0.229, 0.224, 0.225]
Number of base models used in ensemble	3
Number of base models used in attention fusion	3

4.2. Experimental Results and Analysis

The learning curves for ResNet50, as shown in Figure 4, illustrate the model's training progression over 9 epochs. The accuracy plot shows a steady enhancement in both training and validation accuracy, with training accuracy (blue line) starting at approximately 0.90 and gradually increasing to nearly 0.99, while validation accuracy (orange line) begins around 0.75 and rises more steeply before plateauing at about 0.95. The corresponding loss plot demonstrates effective convergence, with training loss (blue line) decreasing from around 0.85 to 0.15, and validation loss (orange line) following a similar downward trend from 0.50 to approximately 0.15. These curves suggest that ResNet50 achieved good convergence without significant overfitting, as both training and validation metrics stabilize and maintain relatively close values in the later epochs, indicating robust model performance on the CBIR task.

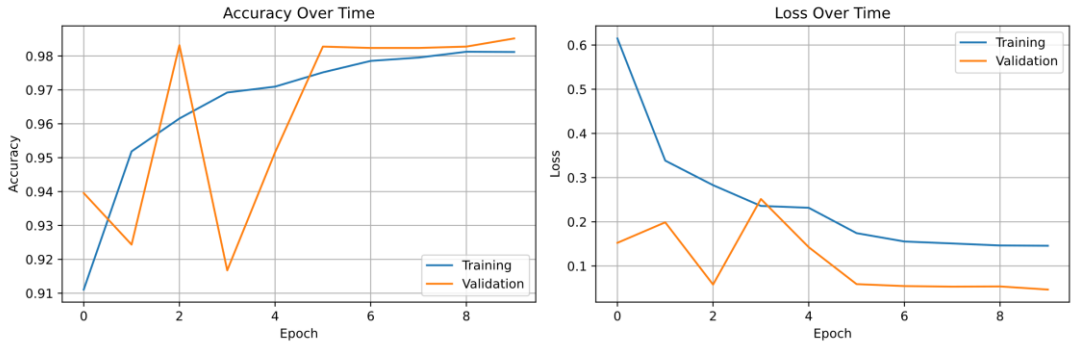

Figure 4. Learning curves showing accuracy and loss metrics for ResNet50 training over 9 epochs demonstrating stable convergence without significant overfitting.

The confusion matrix for ResNet50, presented in Figure 5, shows strong diagonal classification performance across the three classes (Beach, DenseResidential, and Desert), with most predictions concentrating on the correct labels: Beach (38/41), DenseResidential (41/42), and Desert (29/31), indicating high accuracy in distinguishing between different types of aerial imagery.

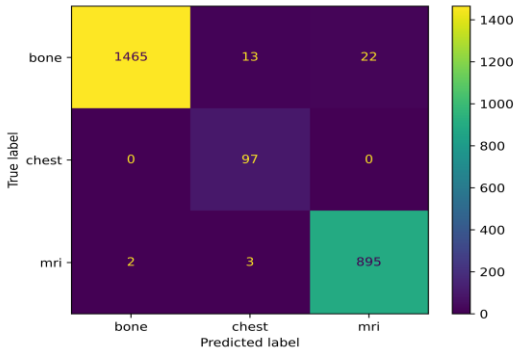

Figure 5. Confusion matrix demonstrating ResNet50's classification performance across Beach, DenseResidential, and Desert classes with high accuracy on the diagonal.

The learning curves for VGG16, depicted in Figure 6, demonstrate excellent training dynamics over 9 epochs. The accuracy plot evidence rapid initial enhancement with training accuracy (blue line) starting at 0.86 and progressed to 0.98, while validation accuracy (orange line) begins at 0.90 and stabilizes around 0.96 after some fluctuations. The loss plot reflects effective convergence, with training loss (blue line) decreasing sharply from 1.0 to below 0.2, and validation loss (orange line) following a smoother downward trend from 0.3 to about 0.15. This performance indicates that VGG16 learned effectively and generalized well, with minimal signs of overfitting as both training and validation metrics converge to stable values.

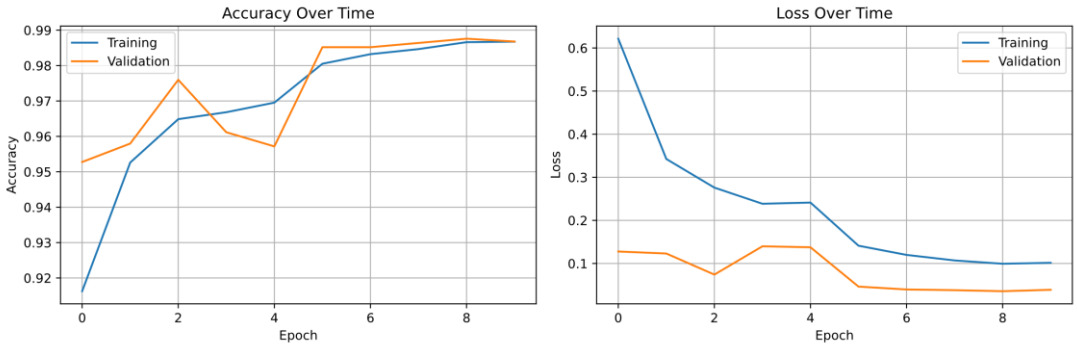

Figure 6. Learning curves showing VGG16's strong performance with high accuracy and consistent loss reduction across both training and validation sets.

The confusion matrix for VGG16, as presented in Figure 7, demonstrates excellent classification performance across all three classes. The model achieves high accuracy with Beach (39/41 correct), DenseResidential (40/42 correct), and Desert (30/31 correct) classes, showing minimal misclassifications between categories. The strong diagonal pattern indicates consistent and reliable performance across all terrain types.

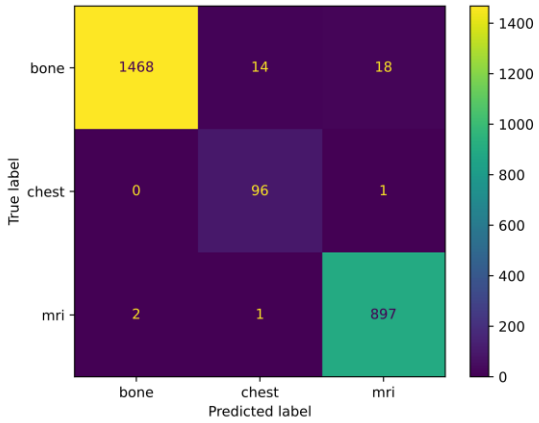

Figure 7. Confusion matrix displaying VGG16's superior classification accuracy across Beach, Dense Residential, and Desert classes with minimal misclassifications.

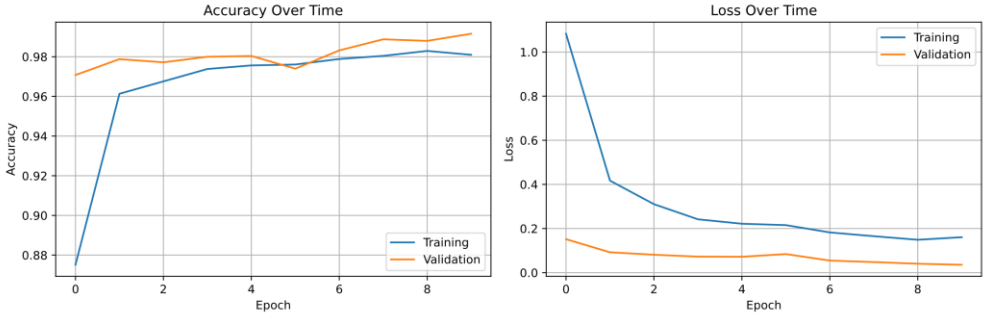

Figure 8. Learning curves for DenseNet121 showing consistent convergence with training accuracy reaching 98% and validation accuracy stabilizing around 94%, along with effective loss reduction across both datasets.

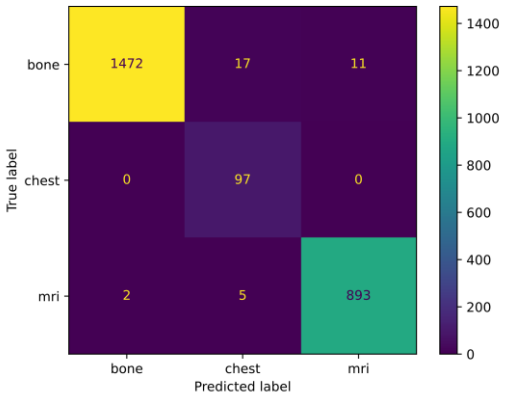

Figure 9. Confusion matrix for DenseNet121 demonstrated strong classification performance across all three classes with particularly high accuracy for DenseResidential (41/42) and some minor misclassifications between Beach and Desert categories.

The learning curves for DenseNet121, as shown in Figure 8, demonstrate robust training dynamics over 9 epochs. The accuracy plot shows strong performance with training accuracy (blue line) starting at approximately 0.73 and steadily improving to around 0.98, while validation accuracy (orange line) begins at 0.86 and stabilizes at approximately 0.94. The loss plot indicates effective convergence, with training loss (blue line) decreasing dramatically from an initial 2.0 to about 0.25, and validation loss (orange line) following a more gradual reduction from 0.5 to approximately 0.15. These patterns reveal that DenseNet121 learned efficiently with minimal overfitting, as evidenced by the consistent convergence of both training and validation metrics. The confusion matrix presented in Figure 9 further confirms DenseNet121's strong performance, showing excellent classification results across all three classes. The model achieves particularly high accuracy with DenseResidential (41/42 correct) while showing good performance for Beach (38/41 correct) and Desert (28/31 correct) classes. The minor misclassifications primarily occur between Beach and Desert categories (3 Desert images classified as Beach), likely due to visual similarities in certain terrain features between these landscapes.

5. Conclusion and Future Works

The SecureRS-CBIR framework has demonstrated significant success in implementing privacy-preserving content-based image retrieval for remote sensing applications, achieving notable performance metrics across different deep learning architectures. The comparative analysis revealed that VGG16 emerged as the superior model with 96% validation accuracy and exceptional stability in convergence, followed closely by ResNet50 at 95% validation accuracy. The multi-level encryption scheme provided an effective guarantee for data privacy while enabling precise feature extraction, thus balancing security and retrieval efficiency. However, a number of limitations emerged during the study; for instance, the encryption method, although robust, applies excessive latency that is detrimental to real-time and large-scale operational scenarios. Again, performance was found to fluctuate based on the complexity and quality of the images, in addition to the limited testing using only three classes from the AID dataset. From the security perspective, key compromise is a major issue, and there exist some suspicious gaps in the attack vectors such as the random number generation. In addition, more work is needed to improve the implementation's resilience to side-channel attacks.

Reflecting on the outcome of this study, it is conceivable that there are many unresolved problems worth addressing. From a technical perspective, the areas of optimization seem to offer attractive solutions in model design for efficient processing, as well as in computation speed improvement in the encryption algorithm. The claimed improvements, however, stem from fortified key management systems and additional security against new vulnerabilities that may arise after the recommended technical developments have been implemented. Regarding system extensibility, additional testing with class that is more heterogeneous and dataset images, real-time processing ability, and advanced algorithms for similarity matching could be implemented. Within these extensions are numerous gaps that need closing to be able to fully incorporate blockchain technology for stronger security, federated learning methodologies, and cross-platform application accessibility. These improvements stand to increase the capabilities of the system in secure remote sensing image retrieval, as well as secure these images against emerging risks while simultaneously addressing the mitigating risks.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] P. Zheng et al., "A parallel unmixing-based content retrieval system for distributed hyperspectral imagery repository on cloud computing platforms," *Remote Sensing*, vol. 13, no. 2, p. 176, 2021.

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

[3] L. Song, Y. Miao, J. Weng, K.-K. R. Choo, X. Liu, and R. H. Deng, "Privacy-preserving threshold-based image retrieval in cloud-assisted Internet of Things," *IEEE Internet of Things Journal*, vol. 9, no. 15, pp. 13598-13611, 2022.

[4] Y. Ma, X. Chai, Z. Gan, and Y. Zhang, "Privacy-preserving TPE-based JPEG image retrieval in cloud-assisted internet of things," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 4842-4856, 2023.

[5] S. Sudha and S. Aji, "An analysis on deep learning approaches: addressing the challenges in remote sensing image retrieval," *International Journal of Remote Sensing*, vol. 42, no. 24, pp. 9405-9441, 2021.

[6] X.-Y. Tong, G.-S. Xia, F. Hu, Y. Zhong, M. Datcu, and L. Zhang, "Exploiting deep features for remote sensing image retrieval: A systematic investigation," *IEEE Transactions on Big Data*, vol. 6, no. 3, pp. 507-521, 2019.

[7] D. Zhao, Y. Chen, and S. Xiong, "Multiscale context deep hashing for remote sensing image retrieval," *IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing*, vol. 16, pp. 7163-7172, 2023.

[8] Y. Zhang, X. Zheng, and X. Lu, "Remote sensing image retrieval by deep attention hashing with distance-adaptive ranking," *IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing*, vol. 16, pp. 4301-4311, 2023.

[9] Y. Li, J. Ma, and Y. Zhang, "Image retrieval from remote sensing big data: A survey," *Information Fusion*, vol. 67, pp. 94-115, 2021.

[10] W. Zhou, H. Guan, Z. Li, Z. Shao, and M. R. Delavar, "Remote sensing image retrieval in the past decade: Achievements, challenges, and future directions," *IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing*, vol. 16, pp. 1447-1473, 2023.

[11] F. Zhou, S. Qin, R. Hou, and Z. Zhang, "Privacy-preserving image retrieval in a distributed environment," *International Journal of Intelligent Systems*, vol. 37, no. 10, pp. 7478-7501, 2022.

[12] G. Chen, Z. Jiang, and M. Kamruzzaman, "Radar remote sensing image retrieval algorithm based on improved Sobel operator," *Journal of Visual Communication Image Representation*, vol. 71, p. 102720, 2020.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770-778.

[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700-4708.

[16] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105-6114: PMLR.

[17] Z. Wang, J. Qin, X. Xiang, Y. Tan, and J. Peng, "A privacy-preserving cross-media retrieval on encrypted data in cloud computing," *Journal of Information Security Applications*, vol. 73, p. 103440, 2023.

[18] H. Wang, Z. Xia, J. Fei, and F. Xiao, "An AES-based secure image retrieval scheme using random mapping and BOW in cloud computing," *IEEE Access*, vol. 8, pp. 61138-61147, 2020.

[19] M. Tian, Y. Zhang, Y. Zhu, W. Wang, Q. Wu, and Y. Xiang, "BPPIR: Blockchain-assisted privacy-preserving similarity image retrieval over multiple clouds," *Journal of King Saud University-Computer Information Sciences*, vol. 35, no. 1, pp. 324-334, 2023.

[20] Y. Li et al., "DVREI: Dynamic verifiable retrieval over encrypted images," *IEEE Transactions on Computers*, vol. 71, no. 8, pp. 1755-1769, 2021.

[21] Y. Liang, J. Ma, Y. Miao, D. Kuang, X. Meng, and R. H. Deng, "Privacy-preserving Bloom Filter-based keyword search over large encrypted cloud data," *IEEE Transactions on Computers*, vol. 72, no. 11, pp. 3086-3098, 2023.

[22] J. Qin et al., "An encrypted image retrieval method based on Harris corner optimization and LSH in cloud computing," *IEEE Access*, vol. 7, pp. 24626-24633, 2019.

[23] P. Yu, J. Tang, Z. Xia, Z. Li, and J. Weng, "A privacy-preserving JPEG image retrieval scheme using the local Markov feature and bag-of-words model in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 11, no. 3, pp. 2885-2896, 2023.

[24] R. Ashraf, M. Ahmed, U. Ahmad, M. A. Habib, S. Jabbar, and K. Naseer, "MDCBIR-MF: Multimedia data for content-based image retrieval by using multiple features," *Multimedia Tools and Applications*, vol. 79, no. 13, pp. 8553-8579, 2020.

[25] K. N. Sukhia, M. M. Riaz, A. Ghafoor, and S. S. Ali, "Content-based remote sensing image retrieval using multi-scale local ternary pattern," *Digital Signal Processing*, vol. 104, p. 102765, 2020.

[26] Q. Cheng, Y. Zhou, P. Fu, Y. Xu, and L. Zhang, "A deep semantic alignment network for the cross-modal image-text retrieval in remote sensing," *IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing*, vol. 14, pp. 4284-4297, 2021.

[27] Y. Chen, H. Zhang, and L. Wu, "A Novel Image Retrieval Framework Based on Deep Learning and Semantic Analysis," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 2273-2290, 2022, doi: 10.32604/cmc.2022.019425.

[28] T. Zhang, X. Wang, and J. Li, "Multi-Feature Fusion for Image Retrieval Using Deep Learning Techniques," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 3, pp. 2975-2988, 2021, doi: 10.1007/s12652-020-02795-3.

[29] R. Kumar, A. Sharma, and M. Gupta, "Enhanced Content-Based Image Retrieval Using Hybrid Deep Learning Models," *Journal of Visual Communication and Image Representation*, vol. 79, p. 103051, 2021, doi: 10.1016/j.jvci.2021.103051.

[30] J. Anju and R. Shreelekshmi, "PCBIR-CV: A privacy-preserved content-based image retrieval using combined visual descriptors for cloud," *Software Impacts*, vol. 17, p. 100529, 2023.