Journal of Computer Science

Original Research Paper

A Cloud-Based Approach for Privacy-Preserving Medical Image Retrieval: Leveraging Local Features and PCA in Two Efficient Steps

M Geetha Yadav and SP Chokkalingam

Department of Computer Science and Engineering, School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

Article history
Received: 20-03-2024
Revised: 20-05-2024
Accepted: 05-07-2024

Corresponding Author:
M Geetha Yadav
Department of Computer Science and Engineering, School of Computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India
Email: geethayadav22@gmail.com

Abstract: The major challenges of content-based image retrieval are searching, ranking, and retrieving in secure mode. In this article, we propose a two-step security approach for each database image, utilizing encryption mechanisms and cloud technology. In this approach, the watermark-embedded encrypted images, along with the efficient feature vector database and the list of authenticated users, will be stored in the cloud. During retrieval, upon verification of the authenticated user in the cloud, the encrypted images will be accessed as the first step in the security process. In the second step, after further user verification, a key provided by the image owner will allow the watermark to be retrieved, enabling the end user to access the original image. Feature vector dataset for all dataset images is constructed using a dominant local pattern named RDEBP combined with PCA. In this two-step security approach for database images, the watermark-embedded system is designed to provide security from unauthenticated users and also from duplicating the images after the first stage of retrieval. Customized watermark bits are embedded into each block of encrypted images before they are stored in the cloud. Consequently, if any unauthorized duplicate image is found, then the watermark-extraction module traces the source and identifies the user, who is responsible for circulating the image. The significance of the proposed method compared to its watermark encryption accuracy by varying the size of blocks has been verified. And also verified its retrieval accuracy over many existing methods, showcased in terms of mean average precision and recall. Experimental trials and security analyses confirm that the proposed approach is both robust and efficient, ensuring a secure and reliable system.

Keywords: Secure Image Retrieval, Cloud-Based Security, Encryption, Decryption, Watermark, Local Patterns

Introduction

The rapid growth of the healthcare sector in utilizing modern imaging-based diagnostics has led to the creation of massive imaging databases. However, the lack of proper labeling and challenges in storing and retrieving these images accurately and securely have become significant issues. The proliferation of extensive image databanks across various domains has intensified the need for efficient and secure image storage and retrieval services. Over the past decade, Content-Based Image Retrieval (CBIR) has made positive strides in addressing database challenges and meeting end-user requirements. Nevertheless, as time goes on, these challenges, particularly in ensuring security, continue to grow more complex. The extensive research efforts have yielded robust algorithms for CBIR to address some of the real problems (Nitin *et al*., 2023; Sucharitha *et al*., 2023). Most of the recent articles have tried to address some of the specific issues of medical image retrieval especially. Harnessing the vast storage potential of the cloud offers a powerful solution for large-scale data management, complex

computations, and scalable resources. This approach proves highly advantageous for both image storage and the outsourcing of CBIR tasks. By adopting this strategy, database owners can free themselves from the demands of local data management and server maintenance.

Despite of utilizing the many resources of cloud, image privacy has become a primary concern with outsourcing of CBIR. For instance, a patient doesn't like to reveal his/her medical records to the public, except to his/her specific doctor. To tackle these issues, this study proposes a robust security model.

- We present a security method comprising two steps and an efficient image retrieval algorithm for database images. The image retrieval efficiency is enhanced by employing a combination of Relational Edge Binary Patterns (RDEBP) and Principal Component Analysis (PCA)
- The two-step accuracy made all encrypted images more secure by embedding the watermark bits into them. After encryption, a watermark is intricately embedded through a distinct method designed for it. After this two-step encryption on all dataset images, the entire dataset will be outsourced to the cloud
- Upon image retrieval, the decryption process entails two steps following user verification

Literature Survey

Image retrieval has been extensively researched; however, directly outsourcing image data to cloud servers poses security concerns. To address this, Searchable Encryption Schemes (SES) have been introduced to facilitate privacy preservation. SES enables data owners to encrypt images and upload them to cloud servers while retaining retrieval capabilities (Zhihua *et al*., 2017; Changyu *et al*., 2011; Ming *et al*., 2011; Kamara *et al*., 2012; Dan *et al*., 2004). Following this, numerous algorithms were developed to cater to diverse search functionalities, such as dynamic search (Minaud and Reichle, 2022; Seny and Tarik, 2017; Demertzis *et al*., 2019), similarity search (Cong *et al*., 2010; 2012; Ning *et al*., 2014) and multi-keyword ranked search (Wenhai *et al*., 2014; Zhangjie *et al*., 2016; Zhang *et al*., 2018). Abdelrhman *et al*. (2021) proposed an IND-CPA secure Content-Based Image Retrieval (CBIR) structure for cloud image retrieval, eliminating the need for continual user engagement. They employed pretrained deep learning or Convolutional Neural Networks (CNNs) to extract all necessary features from the image and applied a divide-and-conquer model for CNN based assessment procedure. Additionally, other researchers outsourced the feature vector database along with the indexing to a cloud server to achieve concealment in CBIR (Hu *et al*., 2016; Hsu *et al*., 2012; Tengfei *et al*., 2018).

Local and Global Features for Secure Image Retrieval: Various secure CBIR techniques have emerged to handle image retrieval in the encrypted domain, differing in the features and indexing methods employed. In Zhihua *et al*., (2016), the authors presented a two-step secure CBIR approach aimed at security providing and copy pre-emption. A secure kNN is used for feature extraction and watermark bits are embedded into cyphertext images to deter unauthorized copying (Wong *et al*., 2009). In further research (Qian *et al*., 2018) a searchable symmetric encryption method for retrieving ciphertext images from the cloud. In this approach, color histograms and locally sensitive hashing are used for feature extraction followed by encoding image indexes before outsourcing the data to the cloud. Qi *et al*. (2022) tried to produce a solution for multi-source privacy-preserving IR based on randomized encryption and bitxor and permutation were used to certify the security of an image over the cloud. They also used the Bag of Words (BOW) model to combine the encrypted pairs in a multi-source.

Bernardo *et al*. (2017) introduced a new framework for secure content-based image retrieval in a cloud environment. The authors utilize homomorphic encryption to ensure that the cloud server can perform computations on encrypted data without needing to decrypt it. This allows for the secure extraction of features from encrypted images and the computation of similarity scores while keeping the underlying data private. The framework extracts visual features from images using techniques such as Scale-Invariant Feature Transform (SIFT). These features are then encrypted and stored in the cloud. They used a secure multi-party computation technique that allows the server to find the most similar images without revealing the content of the images or the query. Another approach by researchers (Jiaohua *et al*., 2019) optimizes the Harris corner detection algorithm to extract robust and distinctive features from images. Harris corner detection is used to identify key points in an image, which serve as the basis for feature extraction. Locality-Sensitive Hashing (LSH) is employed to speed up the image retrieval process. LSH is a method used to hash high-dimensional data in such a way that similar items are more likely to be mapped to the same buckets. Anju and Shreelekshmi (2022) proposed a method for secure CBIR that allows users to retrieve images from a cloud database without compromising the privacy of their data. The framework ensures that both the query image and the images stored in the cloud are protected through encryption. The images in the cloud database are grouped into clusters based on the similarity of their encrypted features. By organizing the database into clusters, the retrieval process can focus only on relevant clusters, reducing the search space and improving retrieval speed. When a query is submitted, the system compares the encrypted features of the query image with the encrypted

features of images in the relevant clusters. Addressing the challenges of large datasets in providing security, researchers (Yanyan *et al*., 2019) presented a hybrid encryption scheme employed to balance security and efficiency. The scheme combines symmetric encryption (for speed) with asymmetric encryption (for security). Symmetric encryption is used to encrypt the image data, while asymmetric encryption secures the encryption keys. This dual-layer encryption ensures that the images stored in the cloud are well-protected while still allowing for efficient retrieval operations. The retrieval process uses a privacy-preserving similarity measure to compare the encrypted query features with the encrypted features stored in the cloud.

In other studies, (Zhihua *et al*., 2021; Hua *et al*., 2020) local color histograms and Local Binary Patterns (LBP) were utilized for feature vector generation, with BOVW employed for indexing and AES encryption schemes for cloud security. Additionally, researchers (Zhengbai *et al*., 2019; Yingying *et al*., 2021; Lin *et al*., 2022) employed various CNN models for feature extraction from images, outsourcing the database to the cloud with encrypted indexing.

In this study, the solution for the two great challenges faced by secure image retrieval has been addressed. Efficiency and security are considered to be the great challenges for any secure image retrieval system. The presented paper has addressed the efficiency, with the help of significant texture feature extraction with the combination of relative edge directional binary patterns and PCA. The structure has been proven with significant results with the comparison of state-of-the-art techniques. The proposed method is also intended to fulfill data privacy and copy prevention.

Construction Entities

The proposed model was constructed upon 4 units: An encryption unit, a threat control unit, a retrieving unit, and a decryption unit. All the way the proposed structure is shown in Fig. (1). All these entities or units are connected sequentially for secure and efficient retrieval.

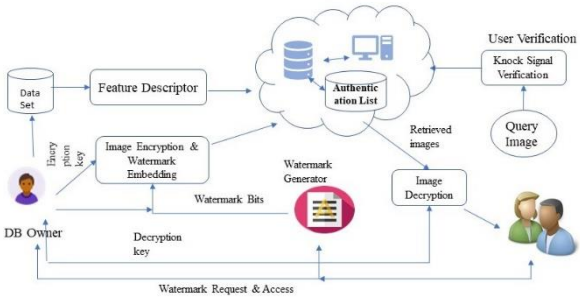


Fig. 1: Framework of the proposed model

Encryption Unit

This unit works with the database owner, where images are encrypted using an owner-defined key, and watermark bits are embedded into the blocks of each encrypted image. The original image database, denoted as $I = \{I_1, I_2, \ldots, I_n\}$ is encrypted as $E = \{E_1, E_2, \ldots . E_n\}$ and watermark bits will be embedded into the E before outsourcing these bi-encrypted images into the cloud. Before this process, a feature vector dataset F is formulated by using texture feature descriptors for all dataset Images. Subsequently, both F and E are uploaded to the cloud, along with a list of authenticated users. The image owner sends authentication information of authorized users or retrievers to the cloud server, which is responsible for verifying user identities during search queries.

Retrievers are authorized individuals, approved by the cloud server, who initiate a search by generating a Knock Signal (KS) for the query image and uploading this signal along with their identification to the cloud server. Once the images are retrieved, users can decrypt them using the secret key provided by the image owner.

Cloud server: The cloud server securely manages the encrypted image database and the feature database. Within the cloud environment, a virtual machine is tasked with extracting texture features from a query image and comparing them to the feature database of the original images to assess similarity.

Watermark generator: The watermark generator, a trusted entity, creates watermarks for images retrieved based on a query image from the list of authorized users. The image owner can decrypt the encrypted images after obtaining keys generated by the watermark generator.

Threat Control Unit

This research tried to address two significant security concerns affecting both the cloud server and retrievers.

Ensuring data confidentiality is paramount in our system, with a robust protocol meticulously crafted to safeguard sensitive information. This includes preserving the privacy of image content, characteristics, and retriever identities through stringent specifications and thorough analysis. To fortify copyright protection, we introduce a novel scheme integrating watermarking technology. This prevents unauthorized access and sharing of data by embedding watermarks into retrieved images, executed by the virtual machine within the cloud infrastructure. Central to our approach is the assumption of integrity among all stakeholders involved database owners, cloud servers, watermark generators, and retrievers. This foundational principle is critical for establishing a

coherent and dependable system. Consequently, the embedding algorithm, watermarks, and secret keys remain confidential, and accessible only through authorized channels facilitated by the database owner. Similarly, retrievers are bound by confidentiality agreements not to disclose decryption keys or any sensitive information regarding image retrieval processes.

Retrieving Unit

In the cloud server, the database owner outsources a feature dataset of all original images, constructed using a combination of RDEBP and PCA. A Virtual Machine (VM) in the cloud, equipped with the necessary software and resources, is trained with an algorithm to autonomously extract features from a query image. This VM is also trained to retrieve similar images from the encrypted image database using the d1 distance metric, as shown in Eq. (1). By running the algorithm independently, the virtual machine efficiently processes the input data and handles feature extraction and image retrieval tasks.

$$Distance\ d_1 = \sum_{i=1}^{n} \left| \frac{I_{qi} - I_{DBi}}{1 + I_{qi} + I_{DBi}} \right| \quad (1)$$

The pseudo-code in Fig. (2) outlines the framework of the local pattern RDEBP, which is used to extract texture features. The feature vector database is constructed using this algorithm.

Decryption Unit

Before the query image is transferred to the cloud, the Knock Signal (KS) module verifies the user's authentication. If authentication fails, the query image request is disregarded. Authenticated users can then utilize the retrieval module within the cloud to index and retrieve similar images for the query image. These retrieved images are encrypted and embedded with watermarks. The image database owner decrypts these images and upon request from the end user, facilitates the retrieval of decryption keys from the watermark generator via the database owner. Only after verification from the image owner's user list, the decryption keys are provided to the end retriever. This meticulous process ensures the controlled distribution of original images.

Materials and Methods

The proposed approach consists of a set of algorithms executed by various components. Figures (3a-b) provide a brief overview of the algorithms utilized in each module.

Pseudo-Code: 2

At the initial stage, the database owner applies the FeaExt algorithm to extract fine-tuned features from all dataset images *I*. In this process, a dominant local descriptor, RDEBP, is proposed and used to extract features. The pseudo-code for this algorithm is provided in Fig. (2) and a feature vector database *F* is constructed for all dataset images. In the second stage, the owner uses the KeyGen algorithm to generate keys for encrypting the images using the EncyImg algorithm. Once encryption of all dataset images is completed as the second stage of security, the WaEmb algorithm is executed with a number of watermark bits defined by this algorithm. These watermark-embedded encrypted images are then outsourced to the cloud along with the indexing of each image. Additionally, the authenticated user list defined by the owner is also uploaded to the cloud. The watermark bits for each image are uniquely generated based on the user list by the Watermark Generator (WG).

Pseudo Code: 1

For every image in the dataset
Code:
Step 1: Assume the center pixel $Id(i,j)=C_P$ for a 5X5 matrix.
Find the local differences for Round=1
Set r_1={-1,1}, r_2={-2,2}
for ki=3: n-3
for li=3; n:3
to the inner loop
Inner loop Clockwise direction transformations $R_{IE1}= C_P\text{-}I(k,l,r_1)$;
Inner Anti-Clockwise direction transformations $R_{IE2}= C_P\text{-}I(k,l,r_1)$;
to the outer loop
Outer loop Clockwise direction transformations $R_{OE1}= C_P\text{-}I(k,l,r_2)$;
Outer Anti-Clockwise direction transformations $R_{OE2}= C_P\text{-}I(k,l,r_2)$;
End
End
Step 2: Extract Binary patterns BP_{IE1}, BP_{OE1} BP_{IE2} BP_{OE2}.
Step 3: Extract individual histograms for all patterns from step2.
Step 4: Concatenate all histograms as a vector for all database images.
Step 5: PCA is applied to the aforementioned feature database to effectively reduce the dimensionality of the feature vectors while preserving fine features.

Fig. 2: Pseudo code for the RDEBP

Encryption Module

- ***Feat_Ext(I, T)*** is the texture feature extractor that précises the competent texture features ***T*** using RDEBP to all database images and generate a feature vector database.
- ***Key_Gen(1^k)*** is a procedure that generates keys based on specified security settings.
- ***Enc_Img(K, I)*** is an image encryption algorithm using a secret key generated *Key_Gen* algorithm.

Watermark Embedding

- ***WaEmb(E_k, B_w)*** is watermark embedding algorithm for every encrypted image with a exclusive key.
  WaExtrac(E_c, I_o) is the watermark extraction algorithm which extracts watermark from E_c and returns I_o.

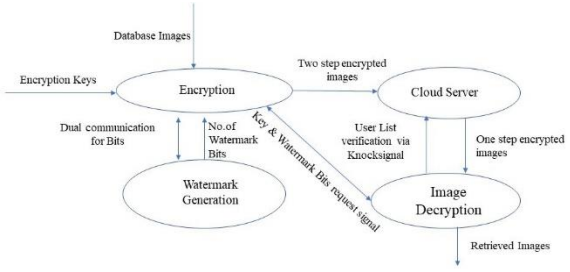
Cloud Server

- ***Auth_Verf(KS, U)*** is an algorithm to verify the knock signal *KS* to verify the authentication of the user or retriever from the user list provided by the image owner before the retrieving process.
- ***Retriev(E, D, F)*** is an algorithm in the cloud server to retrieve similar images from image encrypted database *E* using similarity metric *D* for query features to the feature database *F*.

Retriever

- ***Knock_Signal(Iqu,Fqu,Uiq)*** is an algorithm to generate a knock signal for the given query image with texture features and user information.
- ***Img_Dec(K, R)*** is an algorithm to retrieve the original images with the help of secured key K which is generated after retriever/ user authentication by the image owner.

(a)



(b)

Fig. 3: (a) Framework of the proposed algorithm; (b) Detailed framework of the proposed method

In the next phase of the proposed method, the trained Virtual Machine (VM) in the cloud verifies each Knock Signal (KS) against the users' authentication list before extracting features from the query image. This step provides preliminary security for the images. The AuthVer algorithm validates the user list. After this verification process, the VM extracts the features from the query image and compares them with the feature vector database to find similar image features using the D distance, as given in Eq. (1). With the help of image indexing, the similar images are then retrieved. Upon retrieving similar images from the cloud based on indexing, the *ImgDec* algorithm decrypts the images using the decryption key.

Report of Algorithms

In this section, the detailed nature of each algorithm is described:

- *FeaExt*: This algorithm uses the PCA combined RDEBP algorithm to extract the fine-tuned features, especially texture features from all images of the dataset. The pseudo-code for this algorithm is given in Fig. (2)
- *KeyGen*(1_k): As shown in Fig. (4), this algorithm generates a random key for each and every image of the dataset upon considering the security parameters k to encrypt and decrypt the image. This algorithm generates a unique for every image of the dataset

Pseudo-Code: 3

- *EncyImg*(K, I): This module is designed to encrypt the image with the help of key K and an XNOR bitwise operation is applied in this encryption process. The systematic procedure of this module is explained through the following equations

As we know the majority of the medical images are greyscale images and the intensity levels fall in the range from 0-255, for which 8 binary digits are used to represent each intensity level. The representation of one particular intensity at *(i,j)* is $G_{i,j}$ and the bits can be extracted as follows $G_{ijk} = \left[\frac{G_{ijk}}{2^k}\right]$, here k = 0,1,2,7.

In the encryption module, each pixel intensity of the original image will be encrypted using Eq. (2):

$$e_{i,j,k} = G_{i,j,k} \oplus K_{i,j,k} \quad (2)$$

$$e_{i,j} = \sum_{k=0}^{7} e_{i,j,k}\, 2^k \quad (3)$$

where, $K_{i,j,k}$ is the encryption key and $e_{i,j}$ is the cumulative number for the pixel *(i,j)*.

- *W_Emb* (E_k, W_k,)
- This entity is designed to embed watermark bits W_k into the encrypted image *Ek*
- Division of image into blocks: The encrypted image is divided into a number of non-overlapping blocks, each of size is *sxs*
- Watermark representation: The watermark is represented as an order of binary bits $W_k = W_{k1}, W_{k2}, W_{kN}$, where N is the no of blocks, equal to s^2
- Block selection: The blocks to be watermarked are selected using a pseudorandom number generator as part of the watermark embedding algorithm *W_Emb*
- Pixel set division: In each block, the intensity levels are divided into two sets, *Se*0 and *Se*1, as per the defined method in the algorithm
- Watermark embedding: For each binary bit:
  - If the binary bit is 0, the lower 3 bits of the pixels in set *Se*0 are flipped
  - If the binary bit is 1, the lower 3 bits of the pixels in set *Se*1 are flipped:

$$e'_{k,i,j} = \overline{e_{k,i,j}}\ (i,j) \epsilon Se0 \quad (5)$$

$$e'_{k,i,j} = \overline{e_{k,i,j}}\ (i,j) \epsilon Se1 \quad (6)$$

- *W_Extrac* (W_k) is an algorithm to extract the original image from the decrypted image. It divides the W_k into non-overlapping blocks *sxs* and places the set of blocks that carries the watermark bits $\{B_{ki}\}_{i=1}^{N}$. For each block, the pixels get divided into two sets *S*0 and *S*1 by the predefined function in the algorithm. Two blocks will be produced as B_{ki}^0, $and\ B_{ki}^1$ by flipping the last three bits of each pixel. Construct the corresponding block B_{ki} from the original image with the pseudorandom key in the algorithm and calculate the errors ϕ_0, ϕ_1:

$$\phi_0 = \sum_{p_j \epsilon B_{ki}, p_j^0 \epsilon B_{ki}^0} (p_j - p_j^0)^2 \quad (7)$$

$$\phi_1 = \sum_{p_j \epsilon B_{ki}, p_j^1 \epsilon B_{ki}^1} (p_j - p_j^1)^2 \quad (8)$$

If $\phi_0 < \phi_1$, then the watermark involved is 0, otherwise it is 1.

Take an image from database I
Procedure
Step1: Define a lookup table T_L with a random and unique value for each intensity level.
Step2: Define a zero-key matrix K of size equal to I.
Step3: Fill the Key matrix K
for i=1:M
for j=1:N
x ← I(i , j)
Search and map x to the value in T_L
K(i , j) ← x
end
end
Step4: Key matrix for every image is unique.

Fig. 4: Framework for key generation algorithm

Img_Dec (*K*, *R*) is an algorithm that decrypts an image that has an embedded watermark through a structured five-step process. First, the watermark-embedded encrypted image I' is acquired. Next, each pixel in I' is converted into its binary form $e'_{(i,j,k)}$ using Eq. (2). Subsequently, the decryption key is retrieved from the key generator and also converted into binary form, resulting in $K'_{i,j,k}$ using the same equation. The pixel values are then decrypted using the formula $I'_{i,j,k} = e'_{i,j,k} \oplus K'_{i,j,k}\ k = 0,1 \ldots 7$ Finally, the output of this process is the decrypted image I'.

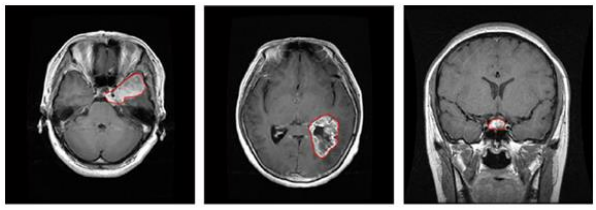
Results and Discussion

The proposed method is evaluated on the benchmark dataset known as the Brain Tumour dataset, which consists of 3 types of tumors at different stages with respect to shape location, size, and skewness as samples shown in (Figs. 5a-b). It has been collected from 234 patients with around 3065 T1-weighted contrast-boosted images and the 3 types of tumors generally named meningioma (708 slices), glioma (1426 slices), and pituitary tumor (930 slices). To run the experiment, we have taken advantage of Google Colab Pro, utilizing an NVIDIA T4 GPU with 2560 CUDA cores and 16GB of GDDR6 memory, along with an additional 100GB of disk space. Average precision and recall are used to measure the performance of the proposed method as given in Eqs. (9-10). The d1 distance metric is used to measure similarity with the query feature vector to *F*.

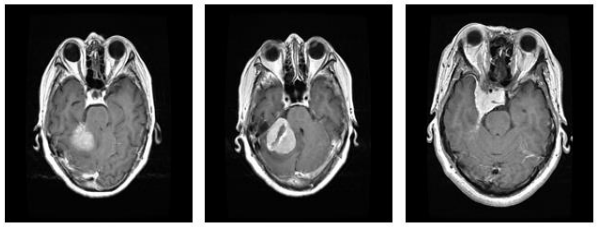
The significance of *PM* over existing in terms of retrieval accuracy based on the index stored in the cloud is evaluated using precision and recall and the comparison over existing methods is given in Table (1) and a graphical representation of the same is shown in Fig. (5):

$$ARP = \frac{1}{D_N}\sum_{i=1}^{D_N} P_i \quad (9)$$

$$ARR = \frac{1}{D_N}\sum_{i=1}^{D_N} R_i \quad (10)$$

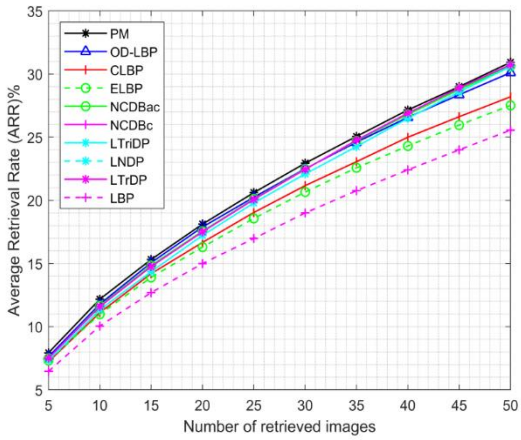


(a)



(b)

Fig. 5: (a) Meningioma, glioma, and pituitary tumor from left to right respectively (b) different glioma tumors in the database



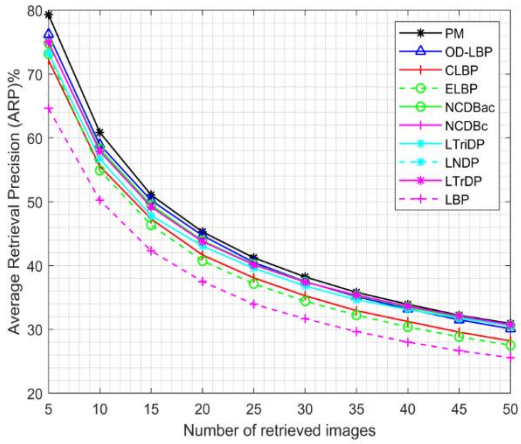


Fig. 5: Retrieval performance of PM over the standing methods in terms of precision and recall

Table 1: Comparisons over existing methods for no. of retrievals from n = 5-50 are evaluated on a specified dataset

Mean average precision

No. of images	LBP	CLBP	ELBP	LTriDP	LNDP	LTrDP	NCDBac	NCDBc	OD-LBP	PM
5	64.64	72.19	73.17	73.28	76.43	75.12	74.85	74.99	76.24	79.31
10	50.24	55.56	54.92	56.69	58.36	57.75	58.27	57.91	58.92	60.87
15	42.31	47.32	46.35	47.85	48.95	48.58	49.48	49.19	50.21	51.06
20	37.49	41.67	40.75	43.05	43.14	43.35	43.91	43.76	44.71	45.29
25	33.96	38.07	37.14	39.59	39.29	39.75	40.22	40.13	40.50	41.22
30	31.67	35.27	34.42	36.81	36.56	37.10	37.47	37.41	37.51	38.23
35	29.66	32.94	32.26	34.65	34.49	35.07	35.30	35.38	35.15	35.80
40	28.02	31.24	30.38	33.12	32.65	33.25	33.50	33.63	33.21	33.94
45	26.66	29.58	28.84	31.77	31.24	31.76	31.97	32.11	31.51	32.22
50	25.56	28.20	27.51	30.58	30.00	30.56	30.62	30.74	30.11	30.93

Mean average recall

No.of images	LBP	CLBP	ELBP	LTriDP	LNDP	LTrDP	NCDBac	NCDBc	ODLBP	PM
1	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00
2	14.88	18.25	17.65	16.60	16.08	15.25	17.35	17.73	18.88	19.45
3	18.90	24.88	24.08	22.40	20.90	19.03	24.10	24.00	26.40	28.00
4	22.00	30.80	29.60	26.58	25.15	22.78	29.83	29.78	32.90	36.03
5	25.20	36.03	34.25	30.35	28.08	25.48	34.25	34.35	38.35	42.48
6	27.93	40.00	38.25	33.60	30.93	28.05	38.08	38.08	42.88	47.38
7	30.25	43.40	41.73	36.25	33.20	30.23	41.38	40.83	46.83	51.93
8	32.58	46.53	44.55	38.78	35.28	32.25	44.33	43.30	50.18	55.78
9	34.13	48.78	47.10	40.68	37.68	34.30	46.80	45.80	53.00	58.80
10	35.90	50.78	49.28	42.68	39.35	36.33	48.50	47.83	54.98	61.48

Watermark Extraction Accuracy

A user who obtains a watermarked image can employ various standard image processing techniques, such as compression, to modify the image. These modifications may distort or degrade the embedded watermark. Additionally, if the user has knowledge of the watermarking process, they might attempt to crack and remove the watermark by manipulating the image at the pixel level, specifically by flipping certain image pixel bits. However, it is important to note that even with such manipulations, it may not be possible to completely remove 100% of the watermark bits, particularly from an encrypted image.

The effectiveness of such an attack is closely tied to the size of the blocks used in the watermarking process. The pixel flipping rate, which represents how frequently pixel values are altered, is highly dependent on the block size. When the block size is minimized, the number of blocks increases, leading to a higher flipping rate for a fixed watermark bit size. This suggests that smaller block sizes make the watermark more susceptible to pixel-level attacks, as more blocks are available for manipulation.

For instance, in Fig. (6), the watermarked versions of the original "Lena" image are displayed for two different block sizes: 64 and 32. The images in Figs. (6b-c) shows how the watermark is embedded with these block sizes, respectively. The impact on the watermark's robustness can be observed by comparing these images. The accuracy of watermark extraction is quantified by the ratio of correctly acquired watermark bits to the total number of watermark bits that were initially embedded, as shown in Eq. (11). This accuracy metric is crucial for evaluating the effectiveness of the watermarking technique and its

resilience against various forms of image processing and manipulation. For the fixed watermark bits with different block sizes, the accuracy of watermark extraction is given in Fig. (7) VM:

$$Accuracy = \frac{B_{ke}}{B_{kw}} \quad (11)$$

(a)

(b)

(c)

Fig. 6: (a) The original LEENA image; (b-c) are the watermarked images for the block of pixels of 64-32

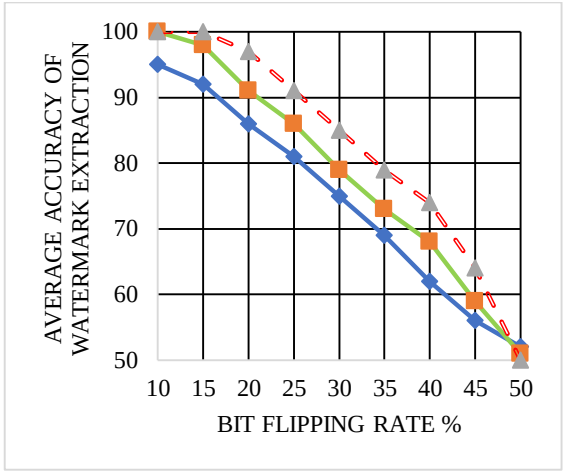


Fig. 7: The typical extraction accuracy for watermarks following a bit-flipping attack. The outcomes are an average of 500 photos

Table 2: Time consumed by the whole image encryption process

	Feature extraction+	Block size s	
	Encryption Time(s)	32	64
Enc. with mxn matrix	3.411+1.212 = 4.623	-	-
Watermark Embedding(ms)	-	0.542	0.783

Time Utilizations

In this section, we try to address the time complexities for the PM in image encryption and search operations. The time consumption for the modules is calculated using Google Colab Pro, utilizing an NVIDIA T4 GPU with 2560 CUDA cores and 16GB of GDDR6 memory, along with an additional 100GB of disk space:

- Encryption time: The encryption time has been calculated before storing the images in the cloud. This has been calculated in two components. One is the time taken to encrypt an image and the second is the time taken for watermark bits embedding into the encrypted images. Table (2) is given for two different sizes of blocks considered for watermark encryption and the results are given
- Similarity search time: This time is generally considered the second phase of any image retrieval process. In this phase, the Virtual Machine (VM) extracts the texture features from the query image and performs similarity verification against the feature vector database F using the d1 distance metric. Once the distances are calculated, the VM identifies similar images by indexing the dataset images based on these calculated distances

The two-step security approach for CBIR systems addresses the key challenges of secure image searching,

ranking and retrieval in a cloud environment. Through the use of encryption, watermarking, and feature extraction techniques like RDEBP and PCA, the system safeguards images from unauthorized access, duplication, and tampering. This method strengthens both the security and integrity of the system, while also ensuring efficient image retrieval.

Conclusion

This article has proposed a two-step approach for secure-based image retrieval using the cloud. A texture feature extractor RDEBP integrated with PCA was used for fine-grained texture feature extraction for all dataset images and constructed feature vector database F. The two-step encrypted images are outsourced to the cloud along with the users' authentication list and image indexing. For a query image, the trained virtual machine verifies the users' authentication list before extracting the features from the query image. Then the VM performs similarity measurement using d1 distance and retrieves the images based on the index. Here the two-step verification process gets executed to provide encryption and watermark extraction. In this approach, the database owner used to have full control of all modules involved in the algorithm. The proposed technique can be employed in various security-based applications that have a demand for safe outsourcing of huge image repositories with strict privacy, retrieval speed, and precision requirements. The two-step security of the proposed scheme prevents data leakages through unreliable query users. As a result, the method is ideal for sensitive applications that can't afford to lose information due to image loss.

Acknowledgment

All authors would like to thank the Dean, School of computing, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

M Geetha Yadav: Defining the research concept, analysis, implementation, and interpretation of results.

SP Chokkalingam: Collecting related literature, analysis, interpretation of results.

Ethics

This article is original and unpublished. Corresponding authors confirm that all other authors have read and agree that the manuscript does not involve ethical issues.

References

Abdelrhman, H., Fei, L., Fanchuan, W., & Yong, W. (2021). Secure Content Based Image Retrieval for Mobile Users with Deep Neural Networks in the Cloud. *Journal of Systems Architecture*, *116*, 102043. https://doi.org/10.1016/j.sysarc.2021.102043

Anju, J., & Shreelekshmi, R. (2022). A Faster Secure Content-Based Image Retrieval Using Clustering for Cloud. *Expert Systems with Applications*, *189*, 116070. https://doi.org/10.1016/j.eswa.2021.116070

Bernardo, F., Joao, R., Joao, L., & Henrique, D. (2017). Practical Privacy-Preserving Content-Based Retrieval in Cloud Image Repositories. *IEEE Transactions on Cloud Computing*, *7*(3), 784–798. https://doi.org/10.1109/tcc.2017.2669999

Changyu, D., Giovanni, R., & Naranker, D. (2011). Shared and Searchable Encrypted Data for Untrusted Servers. *Journal of Computer Security*, *19*(3), 367–397. https://doi.org/10.3233/jcs-2010-0415

Cong, W., Kui, R., Shucheng, Yu, & Karthik Mahendra Raje, U. (2012). Achieving Usable and Privacy-Assured Similarity Search over Outsourced Cloud Data. *2012 Proceedings IEEE INFOCOM*, 451–459. https://doi.org/10.1109/infcom.2012.6195784

Cong, W., Ning, C., Jin, L., Kui, R., & Wenjing, L. (2010). Secure Ranked Keyword Search over Encrypted Cloud Data. *2010 IEEE 30th International Conference on Distributed Computing Systems*, 253–562. https://doi.org/10.1109/icdcs.2010.34

Dan, B., Di Crescenzo, G., Ostrovsky, R., & Persiano, G. (2004). Public Key Encryption with Keyword Search. *International Conference on the Theory and Applications of Cryptographic Techniques*, 506–522. https://doi.org/10.1007/978-3-540-24676-3_30

Demertzis, I., Chamani, J. G., Papadopoulos, D., & Papamanthou, C. (2019). Dynamic searchable encryption with small client storage. *Cryptology ePrint Archive*.

Hsu, C.-Y., Lu, C.-S., & Pei, S.-C. (2012). Image Feature Extraction in Encrypted Domain with Privacy-Preserving SIFT. *IEEE Transactions on Image Processing*, *21*(11), 4593–4607. https://doi.org/10.1109/tip.2012.2204272

Hu, S., Wang, Q., Wang, J., Qin, Z., & Ren, K. (2016). Securing SIFT: Privacy-preserving outsourcing computation of feature extractions over encrypted image data. *IEEE Transactions on Image Processing*, *25*(7), 3411-3425 https://doi.org/: 10.1109/TIP.2016.2568460

Hua, W., Zhihua, X., Jianwei, F., & Fengjun, X. (2020). An AES-Based Secure Image Retrieval Scheme Using Random Mapping and BOW in Cloud Computing. *IEEE Access*, *8*, 61138–61147. https://doi.org/10.1109/access.2020.2983194

Jiaohua, Q., Hao, L., Xuyu, Xiang, Yun, T., Wenyan, P., Wentao, M., & Xiong, N. N. (2019). An Encrypted Image Retrieval Method Based on Harris Corner Optimization and LSH in Cloud Computing. *IEEE Access*, *7*, 24626–24633. https://doi.org/10.1109/access.2019.2894673

Kamara, S., Papamanthou, C., & Roeder, T. (2012). Dynamic Searchable Symmetric Encryption. *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, 965–976. https://doi.org/10.1145/2382196.2382298

Lin, Song, Yinbin, M., Jian, W., Kim-Kwang Raymond, C., Ximeng, L., & Deng, R. H. (2022). Privacy-Preserving Threshold-Based Image Retrieval in Cloud-Assisted Internet of Things. *IEEE Internet of Things Journal*, *9*(15), 13598–13611. https://doi.org/10.1109/jiot.2022.3142933

Minaud, B., & Reichle, M. (2022). *Dynamic Local Searchable Symmetric Encryption* (pp. 91–120). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-15985-5_4

Ming, L., Shucheng, Y., Ning, C., & Wenjing, L. (2011). Authorized Private Keyword Search over Encrypted Data in Cloud Computing. *2011 31st International Conference on Distributed Computing Systems*, 383–392. https://doi.org/10.1109/icdcs.2011.55

Ning, C., Cong, W., Ming, L., Kui, R., & Wenjing, L. (2014). Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data. *IEEE Transactions on Parallel and Distributed Systems*, *25*(1), 222–233. https://doi.org/10.1109/tpds.2013.45

Nitin, A., Sucharitha, G., & Sharma, S. C. (2023). MVM-LBP: Mean−Variance−Median Based LBP for Face Recognition. *International Journal of Information Technology*, *15*(3), 1231–1242. https://doi.org/10.1007/s41870-023-01204-y

Qi, G., Zhihua, X., & Xingming, S. (2022). MSPPIR: Multi-Source Privacy-Preserving Image Retrieval in Cloud Computing. *Future Generation Computer Systems*, *134*, 78–92. https://doi.org/10.1016/j.future.2022.03.040

Qian, Wang, Meiqi, He, Minxin, Du, Chow, S. S. M., Lai, R. W. F., & Qin, Z. (2018). Searchable Encryption over Feature-Rich Data. *IEEE Transactions on Dependable and Secure Computing*, *15*(3), 496–510. https://doi.org/10.1109/tdsc.2016.2593444

Seny, K., & Tarik, M. (2017). Boolean Searchable Symmetric Encryption with Worst-Case Sub-Linear Complexity. *Advances in Cryptology Eurocrypt 2017*, 94–124.

Sucharitha, G., Kalyani, B. J. D., Chandra Sekhar, G., & Srividya, Ch. (2023). Efficient Image Retrieval Technique with Local Edge Binary Pattern Using Combined Color and Texture Features. *Computational Intelligence for Engineering and Management Applications*, 261–276. https://doi.org/10.1007/978-981-19-8493-8_21

Tengfei, Y., Jianfeng, M., Qian, W., Yinbin, M., Xuan, W., & Qian, M. (2018). Image Feature Extraction in Encrypted Domain with Privacy-Preserving Hahn Moments. *IEEE Access*, *6*, 47521–47534. https://doi.org/10.1109/access.2018.2866861

Wenhai, S., Bing, W., Ning, C., Ming, L., Wenjing, L., Y. Thomas, H., & Hui, L. (2014). Verifiable Privacy-Preserving Multi-Keyword Text Search in the Cloud Supporting Similarity-Based Ranking. *IEEE Transactions on Parallel and Distributed Systems*, *25*(11), 3025–3035. https://doi.org/10.1109/tpds.2013.282

Wong, W. K., Cheung, D. W., Ben, K., & Nikos, M. (2009). Secure kNN computation on encrypted databases. *Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data*, 139–152. https://doi.org/10.1145/1559845.1559862

Yanyan, X., Xiao, Z., & Jiaying, G. (2019). A Large-Scale Secure Image Retrieval Method in Cloud Environment. *IEEE Access*, *7*, 160082–160090. https://doi.org/10.1109/access.2019.2951175

Yingying, Li, Jianfeng, M., Yinbin, M., Huizhong, L., Qiang, Y., & Yue, W. (2021). DVREI: Dynamic Verifiable Retrieval over Encrypted Images. *IEEE Transactions on Computers*, *71*(8), 1755–1769. https://doi.org/10.1109/tc.2021.3106482

Zhang, Q., Fu, S., Jia, N., & Xu, M. (2018). A verifiable and dynamic multi-keyword ranked search scheme over encrypted cloud data with accuracy improvement. *In International Conference on Security and Privacy in Communication Systems* (pp. 588-604). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-01701-9_32

Zhangjie, F., Kui, R., Jiangang, S., Xingw, S., & Fengxiao, H. (2016). Enabling Personalized Search over Encrypted Outsourced Data with Efficiency Improvement. *IEEE Transactions on Parallel and Distributed Systems*, *27*(9), 2546–2559. https://doi.org/10.1109/tpds.2015.2506573

Zhengbai, H., Meng, Z., & Yi, Z. (2019). Toward Efficient Encrypted Image Retrieval in Cloud Environment. *IEEE Access*, *7*, 174541–174550. https://doi.org/10.1109/access.2019.2957497

Zhihua, X., Lan, W., Jian, T., Xiong, N. N., & Jian, W. (2021). A Privacy-Preserving Image Retrieval Scheme Using Secure Local Binary Pattern in Cloud Computing. *IEEE Transactions on Network Science and Engineering*, *8*(1), 318–330. https://doi.org/10.1109/tnse.2020.3038218

Zhihua, X., Xinhui, W., Liangao, Z., Zhan, Q., Xingming, S., & Kui, R. (2016). A Privacy-Preserving and Copy-Deterrence Content-Based Image Retrieval Scheme in Cloud Computing. *IEEE Transactions on Information Forensics and Security*, *11*(11), 2594–2608. https://doi.org/10.1109/tifs.2016.2590944

Zhihua, X., Xiong, N. N., Vasilakos, A. V., & Xingming, S. (2017). EPCBIR: An Efficient and Privacy-Preserving Content-Based Image Retrieval Scheme in Cloud Computing. *Information Sciences*, *387*, 195–204. https://doi.org/10.1016/j.ins.2016.12.030