Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

SMedIR: secure medical image retrieval framework with ConvNeXt-based indexing and searchable encryption in the cloud

Arun Amaithi Rajan[1*], Vetriselvi V[1], Mayank Raikwar[2] and Reshma Balaraman[3]

Abstract

The security and privacy of medical images are crucial due to their sensitive nature and the potential for severe consequences from unauthorized modifications, including data breaches and inaccurate diagnoses. This paper introduces a method for lossless medical image retrieval from encrypted images stored on third-party clouds. The proposed approach employs a symmetric integrity-centric image encryption scheme, leveraging multiple chaotic maps and cryptographic hash techniques, to ensure lossless image reconstruction. Medical images are first encrypted by the image owners and converted into hashcodes encapsulating essential features using a deep hashing technique with the ConvNeXt network as the backbone in parallel. To ensure index privacy, these hashcodes are encrypted in a searchable manner. The encrypted medical images, along with a secure index, are subsequently uploaded to cloud storage. Authorized medical image users can request similar medical images for diagnostic purposes by submitting a query image, from which a search trapdoor is generated and sent to the cloud. The retrieval process involves a secure similar image search over the encrypted indexes, followed by decryption along with integrity verification of the retrieved images. The proposed method has been rigorously tested on three standard medical datasets, demonstrating an improvement of 5-20% in retrieval accuracy compared to standard baselines. Formal security analysis and experimental results indicate that the proposed scheme offers enhanced security and retrieval accuracy, making it an effective solution for the encrypted storage and secure retrieval of medical image data.

Keywords Encrypted medical images, Integrity-centric image encryption, Deep hashing, Searchable encryption, Secure similar image search

Introduction

In recent decades, advancements in medical imaging have substantially enhanced healthcare services, enabling physicians to make more informed decisions regarding diagnosis and treatment through detailed visualizations [1]. The exponential growth in the volume and processing of medical imaging data, coupled with the need for scalable storage solutions, has led to the incorporation of cloud technology. Despite its benefits, cloud-based medical image storage is not inherently trustworthy and poses significant risks such as data breaches, unauthorized access, and potential tampering, which compromise the security of sensitive patient information, including confidentiality, integrity, and availability [2]. To mitigate these threats and maintain trust and integrity in cloud-based medical image storage, robust security measures and compliance frameworks are essential.

Encrypting medical images before storage in the cloud addresses some aspects of secure storage. However, even slight alterations by unauthorized individuals

*Correspondence:
Arun Amaithi Rajan
arunamaithirajan@gmail.com
[1] Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, Chennai, India
[2] Department of Informatics, University of Oslo, Oslo, Norway
[3] School of Computing, National University of Singapore, Singapore, Singapore

to encrypted medical images can lead to erroneous diagnoses and treatments. Traditional retrieval methods are inadequate for the secure Content-based Image Retrieval (CBIR) [3] of encrypted images, as they fail to meet the demands for swiftly and accurately retrieving large volumes of encrypted medical image data without privacy leakage. Consequently, there is a pressing need for solutions that enable the efficient retrieval of encrypted medical images while safeguarding privacy in cloud storage. This necessitates the development of methods that ensure medical images are securely stored and retrieved without any information loss when stored in a third-party cloud.

Medical image encryption schemes primarily emphasize content confidentiality [4]. Existing methods [5, 6] address specific attack resistances such as statistical and differential attacks when developing new image encryption models. However, if an intelligent malicious user or cloud attempts to modify encrypted images in storage, it can lead to incorrect decryption and diagnosis. Some schemes [7, 8] have introduced watermarking-based authentications, but these approaches add computational overhead to the process. The challenge remains in performing these tasks efficiently. This can be addressed by incorporating integrity-centric image encryption, where the encryption algorithm inherently ensures integrity verification, eliminating the need for external processes. This approach should be executed within secure regions using trusted databases. By implementing this scheme, lossless image decryption can be achieved with ciphertext distinguishability, preventing potential adversaries from learning or modifying any information about the medical images.

Secure medical image retrieval from encrypted image databases is another critical task that ensures both confidentiality and searchability in the cloud. However, most current top-k ranked image retrieval methods suffer from limited efficiency and may inadvertently reveal the values and sequences of similarity scores to the cloud server. This exposure poses a risk, as a malicious cloud server could deduce user preferences and predict the most similar image content based on these similarity scores if it gains access to user background information through illegitimate means. Therefore, both efficiency and index security need improvement in the context of secure Content-based Medical Image Retrieval (CBMIR) [9, 10]. While some methods for efficient hashcode generation exist, they often fall short in accuracy when encrypted for secure indexing. Recently, ConvNeXt has shown superior performance in feature extraction compared to other deep learning models, making it suitable for efficient hashcode generation. To enhance index and search privacy, these hashcodes are made searchable and encrypted.

As discussed above, existing secure CBMIR systems exhibit low retrieval accuracy, a high risk of index exposure to malicious users or clouds, and less protective image encryption models. To address these issues in medical image storage and retrieval from both security and performance perspectives, a novel Secure Medical Image Retrieval system (SMedIR) in the cloud has been proposed. This system features integrity-centric image encryption and secure indexing schemes. Figure 1 provides a high-level overview of the proposed SMedIR in the cloud. The major contributions of this article are outlined below.

1. A novel secure and efficient medical image retrieval framework (SMedIR) has been proposed.
2. An integrity-centric image encryption scheme is introduced to store medical images securely, which ensures lossless image decryption.
3. The ConvNeXt-based deep hashing method is employed to extract meaningful similarity-preserving hashcodes for indexing which are encrypted using the proposed searchable encryption scheme.
4. Formal security analysis of the SMedIR framework is presented in terms of index privacy, query privacy, search privacy, and image security.
5. Experimental findings demonstrate that the proposed technique has higher security and better retrieval efficiency than existing baseline models.

The rest of the article is structured as follows, The related works have been compiled in Related works section. The proposed framework and its preliminaries are presented in System architecture section. Furthermore, Security and privacy analysis section gives a detailed theoretical analysis of security and privacy of the proposed framework. Experimental results and retrieval performance analysis are extensively discussed in Experimental results and performance analysis section. Finally, the proposed work is concluded in Conclusion section.

Related works

In this section, The authors present an overview of existing secure image encryption techniques and secure image retrieval systems.

Secure image encryption techniques

Image encryption is a technique that encodes a confidential image using an encryption method to ensure that only authorized individuals can access it [11, 12]. This technique is crucial for secure image retrieval, as images are stored in encrypted form with a secure index in the

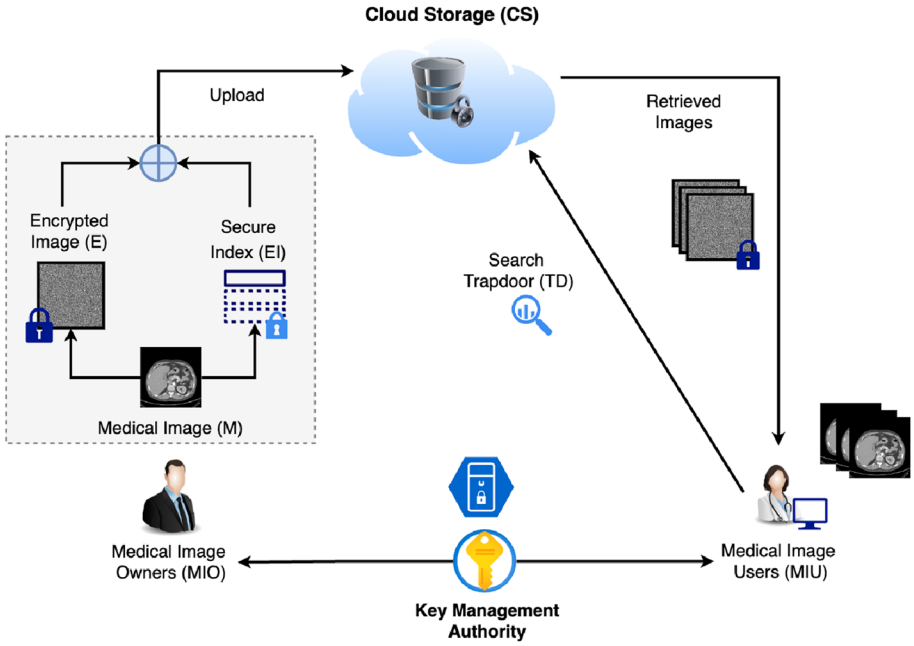


Fig. 1 System model of encrypted medical image retrieval framework

cloud. Numerous image encryption techniques have been constructed [13, 14]. Kaur et al. [14] conducted a comprehensive survey of existing image encryption methods, categorizing them into spatial, optical, transform, and compressive sensing domains, and comparing their performance metrics, advantages, and disadvantages. Furthermore, Priyanka and Singh [15] reviewed secure image encryption techniques tailored for healthcare applications.

In 2020, Wan et al. [16] introduced a new image encryption scheme combining a hyperchaotic Qi system, a one-dimensional chaotic map, and DNA coding. To enhance security, Paul et al. [17] implemented image encryption using SHA-512 and pixel-shifting based on the Zaslavskii map. Moreover, Han et al. [18] developed a fully hash-based fast image encryption system for the Internet of Things (IoT) environment, demonstrating its robustness against major attacks.

Lin et al. [19] designed a novel symmetric image encryption scheme for medical images using chaotic maps and quantum-based keys, with a focus on long-term vision. Nevertheless, recently, Dash et al. [20] proposed a unique image encryption method based on intra-inter pixel permutation for medical images in an IoT environment, showcasing its effectiveness with brain MRI images. Therefore, most existing image encryption algorithms prioritize confidentiality [21], but critical domains handling sensitive images, such as healthcare, require integrity-focused image encryption schemes.

Secure image retrieval systems

Secure and privacy-preserving image retrieval involves performing image searches within an encrypted database while maintaining high performance without any compromise [22–25]. There are various approaches for secure and privacy-enhanced content-based image retrieval, which can be categorized into two main types. The first category involves image generators constructing secure indexes from image features, encrypting the images, and using the cloud for storage. The second category delegates feature extraction and secure index calculation to the cloud, where deep hashing plays a major role. This method extracts image features and constructs similarity-preserving hashcodes [26–29].

From a security perspective, Xu et al. [30] proposed an image retrieval system for large-scale systems in the cloud, utilizing Hamming embedding to generate binary signatures. Min-hash is then performed over these binary signatures. The primary goal of this approach is to improve retrieval accuracy by combining the frequency histogram of the image with the binary signature, providing a more precise representation of the image's features. Yan et al. [31] employed the Software Guard Extensions (SGX) enclaves for secure similarity search in the IoT environment. This approach employs a simple encryption scheme, relying on the trusted environment provided by Intel SGX.

Du et al. [32] developed a secure image retrieval scheme based on deep hashing. This approach uses

Secure k-NN as a secure search index and employs DNA encoding combined with simple chaotic maps for image encryption. In 2020, the same authors enhanced the system's accuracy and speed by incorporating a 4D hyperchaotic map and deep pairwise supervised hashing [33]. Xia et al. [34] proposed a method where the Local Binary Pattern (LBP) of an image is extracted for image search. They introduced a simple encryption scheme involving big block permutation, pixel permutation, and an order-preserving polyalphabetic cipher, using the Manhattan distance to assess image similarity.

Janani et al. [35] explored secure multiparty-based similarity matching for efficient medical image retrieval, comparing it with existing similarity measuring techniques. Zhu et al. [36] advanced the field by designing a new Privacy-preserving Mahalanobis Distance Comparison (PMDC) method. This technique was compared with VFIRM [37], which uses an adapted homomorphic MAC technique to verify search result correctness and a polynomial-based access strategy for efficient fine-grained access control. However, it is noted that these methods are not IND-CPA secure, and their indexes may leak some information.

From the above studies and works, it is evident that numerous efforts have been made to develop secure image encryption schemes and create secure image retrieval systems. However, there remains a clear demand for an efficient and secure cloud-based medical image retrieval system that integrates integrity-focused image encryption and secure indexing with searchable encryption. This need has driven the authors to develop the proposed system, "SMedIR" specifically designed for healthcare applications.

System architecture

Problem formulation

The proposed system consists of Medical Image Owners (*MIO*), Cloud Storage (*CS*), and authorized Medical Image Users (*MIU*), as depicted in Fig. 1. *MIO* have a set of n medical images $\mathbb{M} = \{M_1, M_2, ..., M_n\}$ which has to be outsourced to the cloud after securing them using encryption schemes. *MIU*, who are authorized, will retrieve similar images by sending the query image M_Q to the cloud. *CS* provides services for storing, managing medical images, and handling incoming queries with secure search. As the *CS* is honest and curious, image and query privacy are the major concerns. The problem entails identifying the k most similar images from an encrypted image database to a specified query image M_Q while preserving the privacy of both the image and the query. The overall system model is depicted in Fig. 2. A secure solution proposed is detailed in this section. Table 1 lists the notations used with a description.

Table 1 Notations used

Notation	Description
$\mathbb{M} = \{M_i\}_{i=1}^n$	Medical Images
$\mathbb{E} = \{E_i\}_{i=1}^n$	Encrypted Medical Images
$\mathbb{K} = \{K_S, K_I, K_Q\}$	Keys in the system
K_S	Symmetric Key for Image Encryption and Decryption
P	Reversible Matrix
$\epsilon_1, \epsilon_2, \omega$	Random numbers
$K_I = \{P^{-1}, \epsilon_1, \epsilon_2\}$	Index Encryption Key
$\mathbb{F} = \{F_i\}_{i=1}^n$	Feature Vectors of n Medical Images
$\mathbb{H} = \{H_i\}_{i=1}^n$	Hashcodes of n Medical Images each of length d
$\hat{x}_i$	Expanded intermediate form of hashcode H_i
$\mathbb{EI} = \{EI_i\}_{i=1}^n$	Encrypted Indexes of n Medical Images
M_Q	Query Medical Image
H_Q	Hashcode of M_Q
$K_Q = \{P, \omega\}$	Query Trapdoor Generation Key
$\hat{y_q}$	Expanded intermediate form of query image hashcode H_Q
TD_Q	Search Trapdoor
λ	Security parameter
ρ	Public parameters

Proposed system model and design

An efficient and secure medical image retrieval system (SMedIR) with an integrity-centric image encryption scheme and encrypted searchable index in the cloud is proposed to achieve the following goals.

- **Privacy Preservation:** Cloud servers should not be aware of any stored and incoming query image information.
- **Image Integrity:** Reconstruction of the sensitive medical image should verify the integrity of that image.
- **Retrieval Accuracy:** The suggested system should have improved accuracy than the existing secure image retrieval models.

The proposed system has different entities that work together to make this secure retrieval work. Figure 2 shows the proposed SMedIR architecture. It consists of two major phases. In the first phase, *MIO* outsources the encrypted medical images to the *CS* by securely indexing them. In the next phase, *MIU* retrieves the top-k similar medical images from the *CS*. Trusted third-party Key Management Authority, *KMA* generates the key based on the security parameter λ and distributes it to *MIO* and the *MIU*. *MIO* is responsible for securely outsourcing the medical images $\mathbb{M}$. Both encrypted medical images in $\mathbb{E}$ and the corresponding indexes in $\mathbb{EI}$ are uploaded to the cloud. The *CS* provides the storage and search service

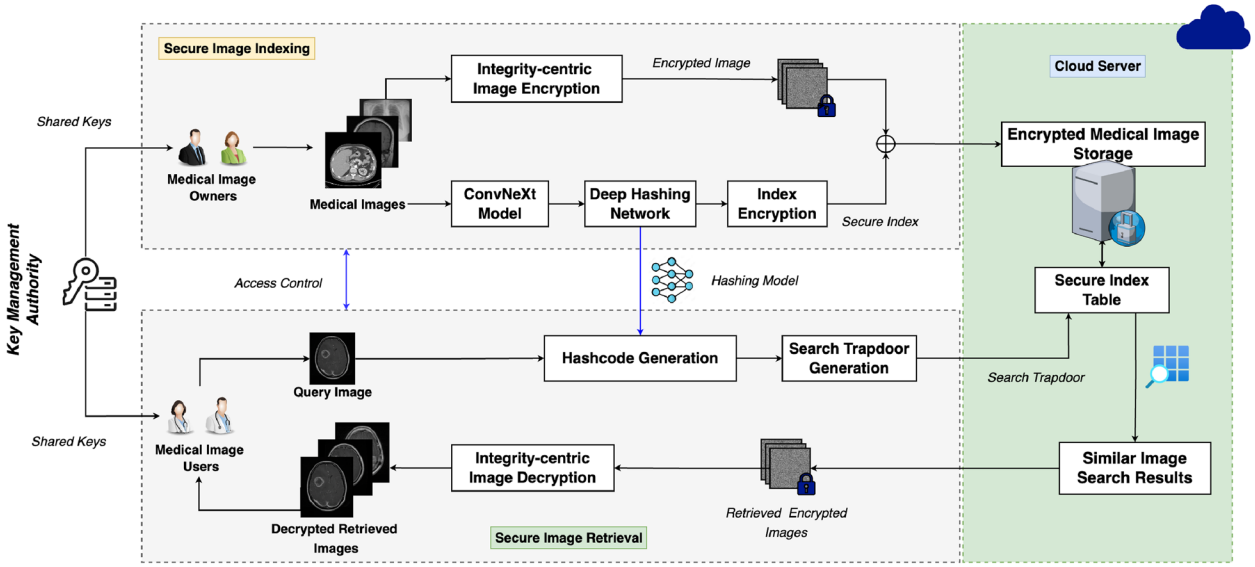


Fig. 2 Proposed SMedIR architecture

over the encrypted database when it receives the query. *MIU* creates the search trapdoor TD_Q by generating the hashcode from the query image M_Q. *CS* searches over the encrypted medical image database with TD_Q and returns top-k relevant images to the *MIU*. After *MIU* receives the top-k results from the *CS*, it decrypts the result images.

Framework design

This subsection presents the framework of proposed system by explaining the functionalities of each entity and associated algorithms. *KMA* runs EnvSetup and KeyGen algorithms. *MIO* executes ImageEnc, HashGen, and SecIndexGen algorithms. *MIU* uses TrapdoorGen, ImageDec algorithms. *CS* provides a secure similar image search with SecSimSearch algorithm.

1. $\rho \leftarrow \mathsf{EnvSetup}(1^{\lambda})$: The setup algorithm takes security parameter λ and returns the public parameters ρ of the system.
2. $\mathbb{K} \leftarrow \mathsf{KeyGen}(\rho)$: The key generation algorithm takes the public parameter ρ as input and produces the key $\mathbb{K} = \{K_S, K_I, K_Q\}$ as output.
3. $\mathbb{E} \leftarrow \mathsf{ImageEnc}(\mathbb{M}, K_S)$: The image encryption algorithm takes medical images $\mathbb{M}$, and encryption key K_S as input, and outputs encrypted medical images $\mathbb{E}$.
4. $\mathbb{H} \leftarrow \mathsf{HashGen}(\mathbb{M})$: The hashcode generation algorithm takes input as medical images $\mathbb{M}$ and return the converted meaningful hashcodes $\mathbb{H}$.
5. $\mathbb{EI} \leftarrow \mathsf{SecIndexGen}(\mathbb{H}, K_I)$: The index generation algorithm takes $\mathbb{H}$ with Index Generation key K_I and returns the searchable encrypted indexes $\mathbb{EI}$ which is offloaded to the cloud.
6. $TD_Q \leftarrow \mathsf{TrapdoorGen}(M_Q, K_Q)$: The trapdoor generation algorithm takes a query image M_Q and key K_Q, and outputs a searchable trapdoor TD_Q which will be sent to *CS* for search.
7. $\mathbb{E}_{\mathbb{Q}} \leftarrow \mathsf{SecSimSearch}(TD_Q, \mathbb{EI}, \mathbb{E})$: A secure similar image search algorithm, for a given encrypted query TD_Q returns encrypted images $\mathbb{E}_{\mathbb{Q}} \in \mathbb{E}$ as a result from the *CS* to *MIU*.
8. $\mathbb{S} \leftarrow \mathsf{ImageDec}(\mathbb{E}_{\mathbb{Q}}, K_S)$: An image decryption algorithm takes resultant encrypted medical images $\mathbb{E}_{\mathbb{Q}}$, and decryption key K_S as inputs and outputs plain medical images set $\mathbb{S}$.

This section summarizes the problem formulation, overall framework, and detailed design. Figure 3 explains the interaction among the entities of the proposed system.

Technical background details

Some prerequisites must be addressed to comprehend the SMedIR. This subsection explains DNA encoding with operations, chaotic maps, searchable encryption, and the ConvNeXt deep learning model.

DNA encoding and operations

A new area of unconventional computing called "DNA computing" replaces standard electronic computing with DNA sequences and molecular biology hardware. In DNA encoding, binary data is transformed into the DNA sequence. The Watson-Crick principle forms the basis of it [38]. Adenine (A), Guanine (G), Cytosine (C),

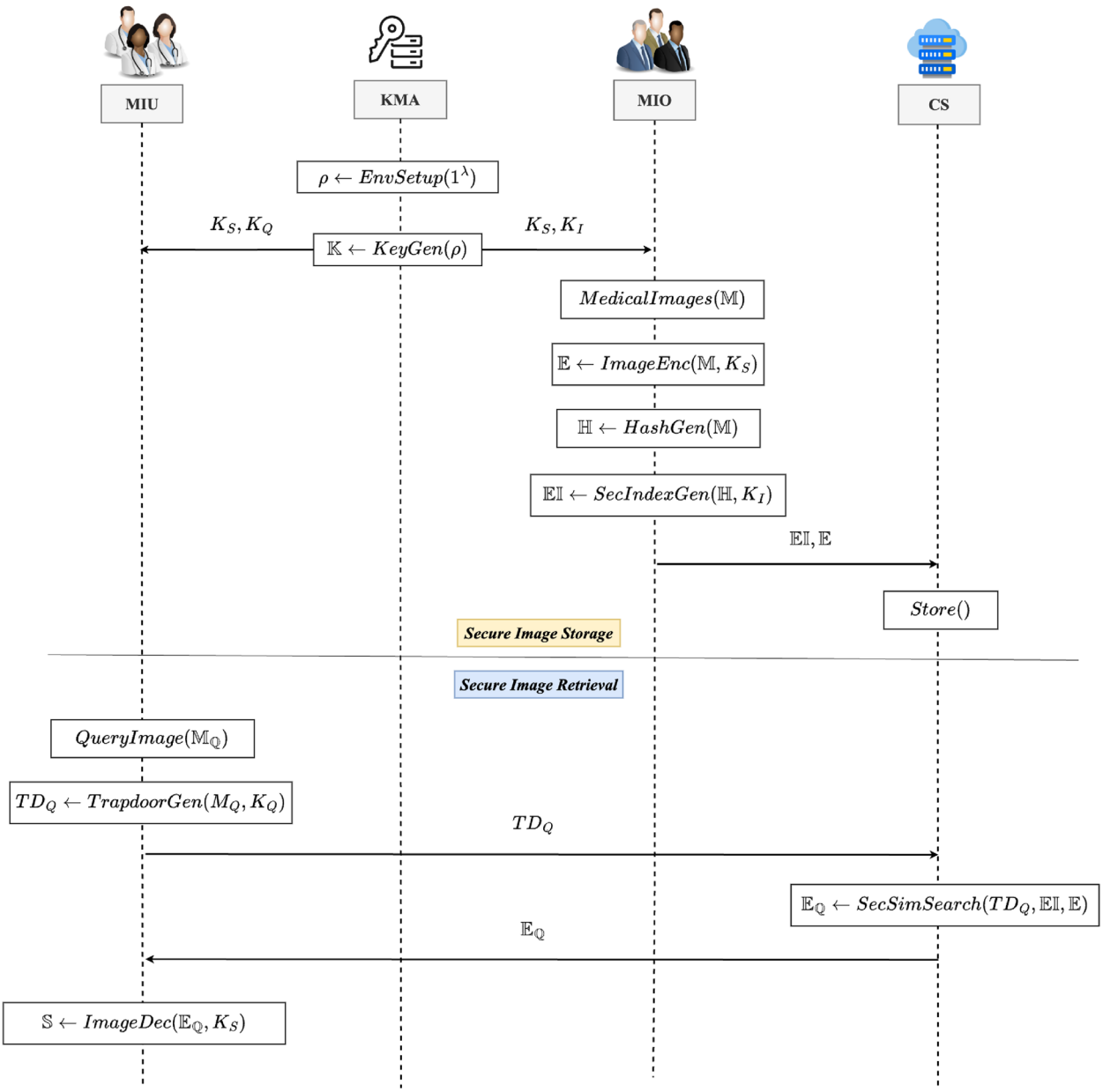


Fig. 3 End-to-end flow diagram

and Thymine (T) are the four nucleic acid bases that make up the DNA string. A and C are complementary base pairings, bonding to T and G, respectively. The four fundamental DNA building blocks are the binary codes Adenine-00, Thymine-11, Cytosine-10, and Guanine-01. The binary rule permits 24 encoding schemes, but only 8 fit the DNA pairing rule due to the complementary base pairing. Table 2 displays the eight standard rules for DNA mapping sequence.

While encoding the bit stream into a DNA-encoded sequence, the encoding has 8! possibilities. This DNA encoding technique is being utilized for substitutions in encryption methods. The tremendous storage capacity of DNA, the minimal power requirements for computation, and the rapid processing speeds of DNA encoding are its key benefits [39]. DNA encoding is a more effective means of encrypting images than bit-level and pixel-level encryption algorithms in terms of security. DNA computing supports logical and arithmetic operations. The DNA-encoded data might be subjected to addition, subtraction, XOR, and XNOR operations. XOR operation is most widely used in image encryption techniques [40].

Table 2 DNA encoding rules

Bits	i	ii	iii	iv	v	vi	vii	viii
00	A	A	G	C	G	C	T	T
01	G	C	A	A	T	T	G	C
10	C	G	T	T	A	A	C	G
11	T	T	C	G	C	G	A	A

Table 3 DNA XOR truth_table

XOR	A	G	T	C
T	T	C	A	G
G	G	A	C	T
A	A	G	T	C
C	C	T	G	A

The truth_table for the DNA XOR gate is displayed in Table 3.

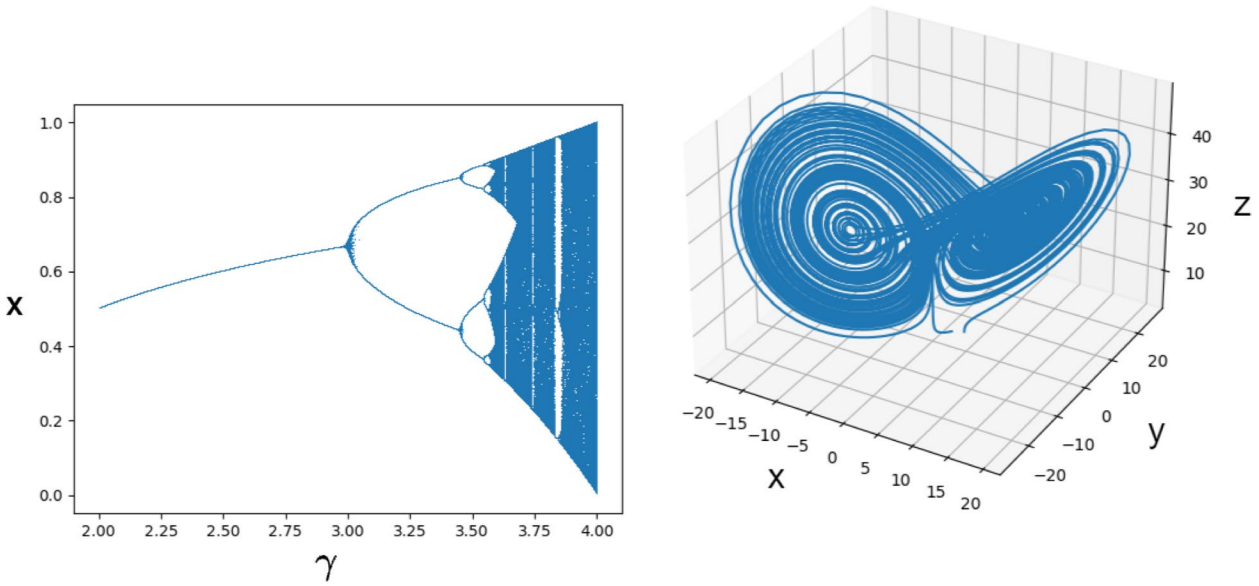
Chaotic maps

Chaotic maps are investigated in a dynamic environment and are too sensitive to initial conditions [41]. This implies that outputs can be drastically altered by slightly changing the initial parameters. These maps can be divided into discrete maps and continuous maps. These maps are used to generate keys to diffuse the image pixels. A logistic map and a 3D Lorenz map have been employed in the proposed encryption scheme.

Logistic Map: This map is a nonlinear chaotic map that yields a non-periodic pseudo-random sequence [42]. The logistic map's mathematical expression is shown in Eq. 1.

$$x_{n+1} = \gamma x_n(1 - x_n) \tag{1}$$

x_n and x_{n+1} represent the current state and next state respectively. The control parameter is γ, and x_0 is the initial value. When $\gamma \in (3.5, 4]$ and $x_0 \in [0, 1]$, the behavior becomes chaotic. This map provides randomness and complexity, making it valuable for secure cryptographic systems due to its ability to generate unpredictable sequences resistant to attacks. Its computational efficiency and simplicity make it suitable for real-time encryption applications such as in healthcare. Generally, the bifurcation diagram of any chaotic map visually shows the period doubling as the control parameter increases. Figure 4a illustrates the bifurcation diagram



(a) Logistic Map (b) 3D Lorenz Map

Fig. 4 Bifurcation diagram

of the logistic map. Here, the x becomes chaotic when γ varies from 3.5 to 4. From one chaotic map to the other, these control parameter values change.

3D Lorenz Chaotic Map: The Lorenz is a three-dimensional chaotic dynamic map [43]. Equations 2-4 below can be used to characterize the system.

$$\frac{dx}{dt} = \mu(y - x) \tag{2}$$

$$\frac{dy}{dt} = x(\eta - z) - y \tag{3}$$

$$\frac{dz}{dt} = xy - \delta z \tag{4}$$

μ, η, δ are the control parameters and x_0, y_0, z_0 are the initial values. The control parameters have a significant impact on the system. The proposed image encryption algorithm uses a system of equations that displayed chaotic behaviour for the $\mu = 10$, $\eta = 28$, and $\delta = 8/3$. Figure 4b depicts the attractor produced using this 3D Lorenz chaotic map. Secure keys generation utilizes this chaotic map in the proposed image encryption algorithm because of its higher rate of non-repetitiveness and unpredictability.

Searchable encryption

Searchable encryption (SE) enables efficient search operations over encrypted files, ensuring retrieval capability while maintaining encryption integrity and preventing decryption. SE enables outsourcing files to an untrusted cloud storage server without disclosing the files in plaintext while still enabling the server to perform searches over them. Therefore, the security model should ensure that ciphertexts are indistinguishable, preventing adversaries from discerning any meaningful information from ciphertexts. In this article, image indexes are encrypted.

ConvNeXt deep learning model

ConvNeXt, a family of pure ConvNet models introduced by Zhuang et al. [44], achieves competitive accuracy and scalability compared to Transformers while maintaining the simplicity and efficiency of standard ConvNets. Two significant changes in the architecture level are using GELU instead of ReLU and Layer Normalization instead of Batch Normalization. Both changes boost the model performance better than transformers and this model extracts features at both local and global levels. ConvNeXt models are a viable option for tasks where ConvNets are more suited, while Transformers may be more flexible for tasks requiring feature interactions across modalities or structured outputs. So, this model has been selected as a backbone of a content-based similarity-preserving deep hashing network. The ConvNeXt architecture is shown in Fig. 5. The upcoming sections explain the proposed secure indexing and retrieval framework in detail.

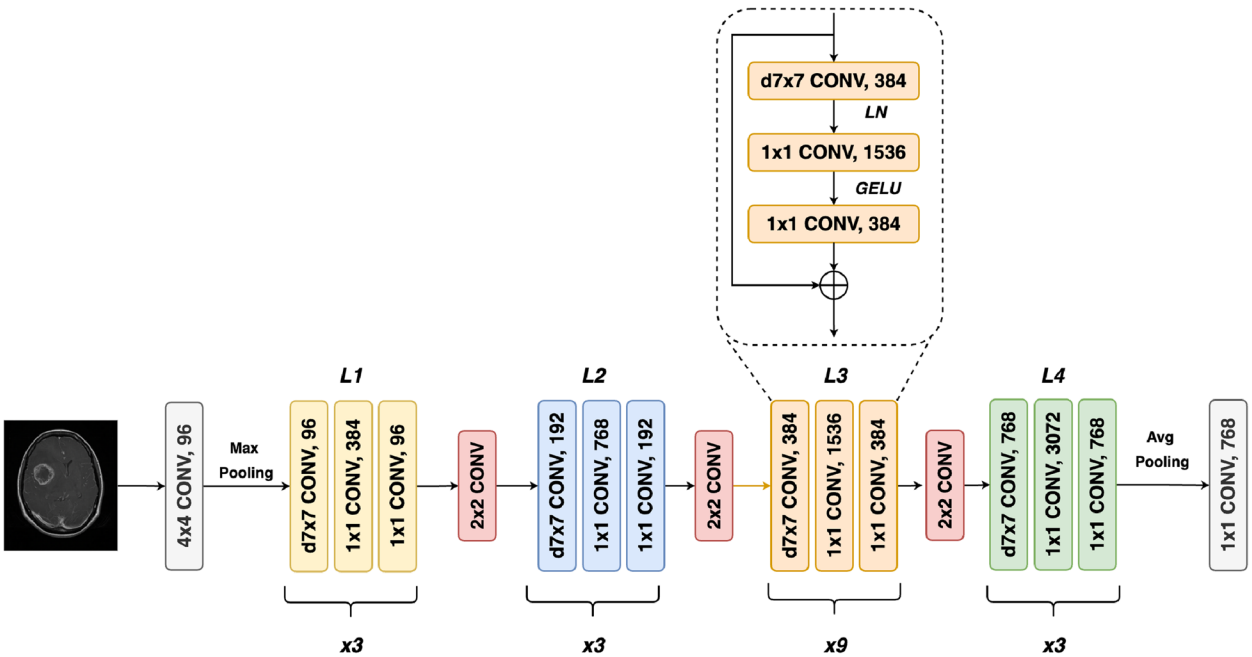


Fig. 5 ConvNeXt architecture

Secure indexing of medical images

The proposed system architecture is shown in Fig. 2. The system contains two major phases: secure indexing and secure search over encrypted medical images. This subsection details each module in each phase. In secure indexing, *MIO* encrypts the medical image, generates the encrypted searchable index, and uploads them to the cloud. This phase contains 5 different functions which are: EnvSetup, KeyGen, ImageEnc, HashGen, SecIndexGen.

EnvSetup and KeyGen

EnvSetup is the function which produces public security parameters ρ from the input parameter 1^{λ}. This module initially sets the parameters α, d as well. Here, α is the prime number, and d is the length of the hashcode. Using α, a multiplicative cycle group $\mathbb{G}_{cy}$ with order α has been selected. For a pseudorandom number generation PR_f, which is defined as $PR_f : \{0,1\}^{\lambda} \rightarrow \mathbb{G}_{cy}$ will be chosen. This PR_f maps the pseudorandom number to the prime number field $\mathbb{Z}^d$ in the order of α. The system parameters of logistic and 3D-Lorenz chaotic map, γ, μ, η, δ will be determined. Finally, the module outputs the public parameter ρ.

$$\rho = \{\alpha, d, \mathbb{G}_{cy}, PR_f, \gamma, \mu, \eta, \delta\} \tag{5}$$

KeyGen module produces the $\mathbb{K}$ by taking ρ as input. As shown in Algorithm 1, symmetric image encryption key (K_S), index encryption key (K_I), trapdoor generation key (K_Q) are generated. P represents the invertible matrix. $\epsilon_1, \epsilon_2, \omega$ are the random numbers. K_S contains the sub keys required for image encryption which are $\{k_{lx}, k_{ly}, k_{lz}, k_{lg}, k_{en}, k_{de}, k_{scr}\}$. Here k_{lx}, k_{ly}, k_{lz} are initial conditional parameters of 3D-Lorenz map. k_{lg} is the initial parameter of the logistic chaotic map. k_{en}, k_{de}, k_{scr} are encoding key, decoding key, and scrambling key accordingly Finally, the *KMS* sends the secret key $\mathbb{K}$ to the *MIO*, *MIU* through a secure channel so that the *MIU* can perform encryption, indexing, and *MIO* can perform trapdoor generation, and decryption operations.

Algorithm 1 KeyGen

Input: Public Parameter ρ
Output: Secret Keys $\mathbb{K}$

1: Choose randomly ϵ_1, ϵ_2, $\omega \in \mathbb{Z}_{\alpha}^{d}$
2: Randomly generate the reversible matrix $P \in \mathbb{Z}_{\alpha}^{d}$ with the size of $(d+3) \times (d+3)$
3: Generate keys required for image encryption from the system parameters and initial parameters of chaotic maps.
Set $K_S = \{k_{lx}, k_{ly}, k_{lz}, k_{lg}, k_{en}, k_{de}, k_{scr}\}$
4: Set $K_I = \{P^{-1}, \epsilon_1, \epsilon_2\}$
5: Set $K_Q = \{P, \omega\}$
6: Set $\mathbb{K} = \{K_S, K_I, K_Q\}$
7: **return** $\mathbb{K}$

Image encryption model

Medical images must be stored securely to avoid information leakage and modification. In this image encryption module, there is an algorithm that ensures confidentiality and integrity. The image owner encrypts the image before uploading it to the cloud. This model is designed for medical centers, which are capable of having trusted platform modules. The KeyGen module generates the keys required for image encryption and shares them with *MIO*. The proposed encryption model consists of DNA encoding, cryptographic hash generation, bit-plane scrambling, DNA-XOR, and bit-wise XOR operations.

Algorithm 2 `ImageEnc`

Input: Medical Images $\mathbb{M} = \{M_i\}_{i=1}^{n}$, Image Encryption Key K_S
Output: Encrypted Medical Images $\mathbb{E} = \{E_i\}_{i=1}^{n}$

Generate the logistic chaotic sequence K_{LG} using $\{\gamma, k_{lg}\}$ as shown in Equation 1.
Generate the 3D-Lorenz chaotic sequence K_{LZ} using $\{\mu, \eta, \delta, k_{lx}, k_{ly}, k_{lz}\}$ as shown in Equations 2- 4.
for each M_i in $\mathbb{M}$ **do**
Step 1: DNA Encoding of M_i
    Express the grayscale medical image in bit-plane format $M_i[1-8]$
    Convert the image into DNA encoded format $DM_i[1-4]$ using k_{en}
Step 2: Hash function for integrity check
    In a secure enclave, convert the DM_i into DNA sequence $D_1D_2D_3D_4$
    Find the cryptographic hash C_h value for $\{D_1, D_2\}$
    Store the pair $(C_h : D_3D_4)$ in the TPM
Step 3: Scrambling of DM_i
    Scramble the DNA encoded bit planes using k_{scr}
Step 4: DNA Encoding of K_{LZ}
    Convert the key into DNA encoded format $DK_{LZ}[1-4]$ using k_{en}
Step 5: Scrambling of DK_{LZ}
    Scramble the DNA encoded key sequence bit planes using k_{scr}
Step 6: Compute the DNA XOR
    $DX_i \leftarrow DM_i \oplus_{DNA} DK_{LZ}$
Step 7: DNA decoding
    Decode the DX_i using k_{de} results DD_i
Step 8: Compute the bit-wise XOR
    $E_i \leftarrow DD_i \oplus K_{LG}$
    Update $\mathbb{E}$ with E_i
end for
return $\mathbb{E} = \{E_i\}_{i=1}^{n}$

Algorithm 2 explains the steps in the proposed integrity-centric medical image encryption model. Initially, the required secret key sequences K_{LG}, K_{LZ} are generated from the logistic map and 3D-Lorenz map using Eqs. 1, 2-4 respectively. As bit-plane representation helps to apply bit-level operations, $\mathbb{M}$ used that format during the encryption process [45]. Input M_i is a grayscale image that uses 8-bit-plane representation. Firstly, the input image is DNA encoded ($M_i \rightarrow DM_i$) using the encoding key K_{en}. DNA-encoding rules specified in Table 2 are employed to perform pixel substitution on M_i. In this step, a process of generating cryptographic hash and corresponding DNA encoded bitplanes in a Trusted Platform Module (TPM) has been introduced. Any TPM can be used for this trusted storage. It helps to verify the integrity of the image while reconstructing. Then, the DM_i bit planes are scrambled using k_{scr}. At the same time, Lorenz chaotic sequence L_{LZ} is DNA encoded and scrambled using k_{en}, k_{scr} respectively. Both DNA encoded image and key sequence are undergoing the DNA XOR operation. The output DX_i is again DNA decoded using k_{de}. Finally, the generated image DD_i is bit-wise XOR with a generated logistic chaotic sequence K_{LZ}, which produces an encrypted image E_i.

The proposed system is resistant to multiple potential attacks. This encrypted medical image will be outsourced with encrypted images to the cloud. In traditional encryption methods, side-channel attacks exploit variations in computation time to infer sensitive information. To mitigate this vulnerability, the authors propose modifying the encryption time function to include a noise factor ϵ. This modification introduces random variability in the encryption time, thereby obfuscating the timing information that side-channel attacks rely on. This has been formally proved in Theorem 4. The hashcode generation and its encryption are detailed in the upcoming submodules.

Deep hashing using ConvNeXt model

For the secure searchable encrypted index generation, images are converted into hashcodes and then encrypted. The hashing model utilized the ConvNeXt model as a

backbone for hashcode generation. ConvNeXt models extract better features than other ConvNets. The layered architecture of this deep hashing network is shown in Fig. 6. Each level contains different convolution strides to learn the deep features. ConvNeXt block utilizes GELU instead of ReLU, LN instead of BN, Inverted Bottleneck architecture. As this module takes the concept of vision transformers, it captures both local and local features. Finally, the fully connected layer brings the features extracted for image M_i and gets the feature vector $F_i = \{f_1, f_2, ...f_d\}$. The features are converted into hashcode $H_i = \{h_1, h_2, ...h_d\}$ of length d based on the following equations:

$$h_j = H\,func(f_j) \tag{6}$$

$$H\,func(f_j) = \begin{cases} 1, & \text{if } f_j \text{ is greater than or equal to } \beta \\ 0, & \text{if } f_j \text{ is less than } \beta \end{cases} \tag{7}$$

Here, β is the median of the feature vector F_i. This model uses the contrastive learning method to generate similarity-preserving hashcodes. Inputs are paired and given while training. So, the hashing network generates the hashcodes, and based on the contrastive loss, the model updates its weights. In the proposed method, contrastive loss, quantization loss, and bit balance loss are combined in a way that the network generates better hashcodes. The contrastive loss is defined as follows:

$$\mathcal{L}_{cont}(h_i, h_j, \mathcal{S}) = \frac{1}{2}\sum_{k=1}^{N}\left(\mathcal{S}.dt^2 + (1-\mathcal{S}).MAX(0, \mathcal{M} - dt^2)\right) \tag{8}$$

Here, h_i and h_j are the hashcodes generated from the deep hashing model of i,j^{th} medical images. $\mathcal{S}$ is the label that denotes whether both images belong to the same class or not. if both i and j are falls to same class then $sim = 1$ or 0 otherwise. N in the number of training images in that minibatch. dt is the euclidian distance between the h_i and h_j. $\mathcal{M}$ is the margin hyperparameter that controls the separation between similar and dissimilar pairs. Given the inherent complexity in optimization, the current technique lacks the capability to guarantee the complete convergence of generated hashcodes. So, a pairwise quantization loss ($\mathcal{L}_{quant}$) is used to encourage the network output to be close to standard binary codes. The pairwise quantization loss is defined in Eq. 9, and a quantization function is applied to approximate the binary code to the desired hashcode.

$$\mathcal{L}_{quant} = \sum_{i,j\in N}(\| \, |h_i| - 1 \, \|_1 + \| \, |h_j| - 1 \, \|_1) \tag{9}$$

Here, 1 is a vector of all ones, $\| \quad \|$ is the L1-norm of the vector, and $| \quad |$ is the element-wise absolute value operation. Added to these two losses, bit balances loss ($\mathcal{L}_{bit}$) is also added to efficiently compute the hashcode. This means that there is a 50% chance that each hashcode will be between 0 and 1. It is possible to generate more unique hashcodes by utilizing the target function provided in Eq. 10 for d-bit hashcode generation, here, h_l is the hash layer output of l^{th} node

$$\mathcal{L}_{bit} = \frac{1}{d}\sum_{k=1}^{N}\sum_{l=1}^{d} h_l \tag{10}$$

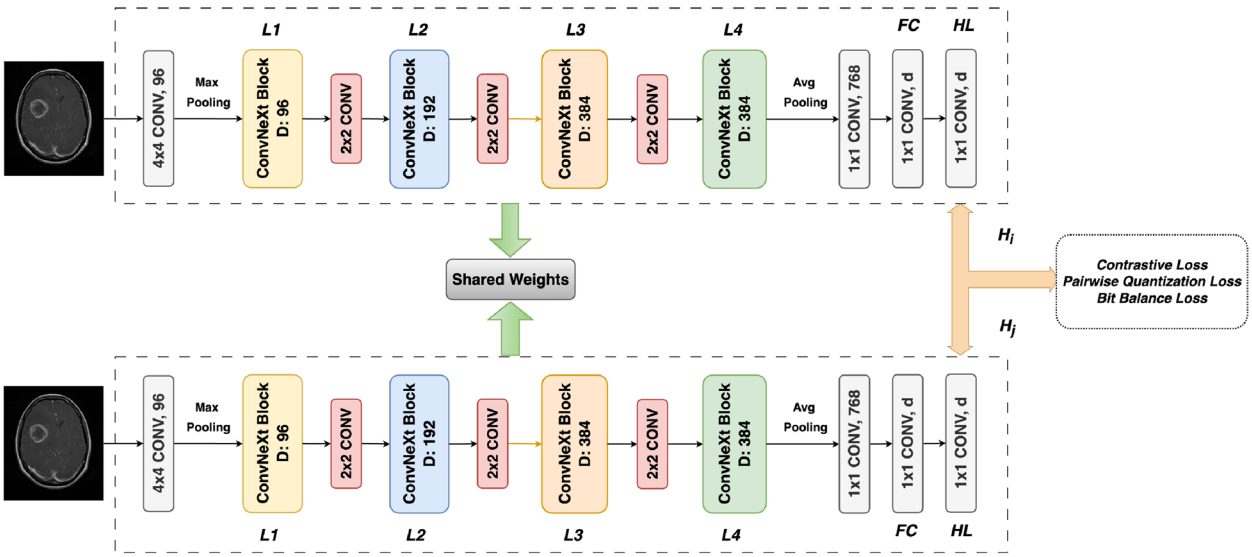


Fig. 6 ConvNeXt based deep hashing model with contrastive learning

Finally, the cumulative loss function for the discriminant hashcode generation is

$$\mathcal{L}_{total\_loss} = \mathcal{L}_{cont} + \phi\mathcal{L}_{quant} + \chi\mathcal{L}_{bit} \tag{11}$$

All losses are merged. The overall objective is to minimize the total loss. ϕ and χ are the weighting factors to control the losses. By propagating this error in each hash generation network, efficient hashcodes are generated.

Searchable encrypted index generation

This module generates the secure index from the hashcode H_i leveraging the basic idea of $a.b = (aP^{-1}).(bP^T)^T$. This helps image retrieval faster and also more secure. As shown in Deep hashing using ConvNeXt model section, H_i will be generated from the medical image M_i. The length of the hashcode is d. Using the random number ϵ_1, ϵ_2 in K_I, H_i is expanded to the vector $\hat{x}_i$ with the dimension of $d + 3$ as follows:

$$\hat{x}_i = \left(\sum_{j=1}^{d} h_{ij}^2 + \epsilon_1 - \epsilon_2, h_{i1}, h_{i2}, ..., h_{id}, 1, \epsilon_2 \right) \tag{12}$$

Then, the $\hat{x}_i$ is encrypted using the inverse of reversible matrix P as in Eq. 13.

$$EI_i \leftarrow \hat{x}_i.P^{-1} \tag{13}$$

Finally, the encrypted index EI_i is sent with E_i to the *CS* to store in an encrypted indexed database. Algorithm 3 shows the step-by-step process of secure index generation.

Algorithm 3 `SecIndexGen`

Input: Hashcodes $\mathbb{H} = \{H_i\}_{i=1}^n$, Index Encryption Key $K_I = \{P^{-1}, \epsilon_1, \epsilon_2\}$
Output: Encrypted indexes $\mathbb{EI} = \{EI_i\}_{i=1}^n$

1: **for** each image hashcode in $\mathbb{H}$ **do**
2: Decompose the d-dimensional hashcode $H_i = \{h_{i1}, h_{i2}, ..., h_{id}\}$
3: Augment the H_i into $(d+3)$ using $\{\epsilon_1, \epsilon_2\} \in K_I$ $\hat{x}_i = (\sum_{j=1}^{d} h_{ij}^2 + \epsilon_1 - \epsilon_2, h_{i1}, h_{i2}, ..., h_{id}, 1, \epsilon_2)$
4: Encrypt the $\hat{x}_i$ using P^{-1} $EI_i \leftarrow \hat{x}_i.P^{-1}$
5: Update the $\mathbb{EI}$ with EI_i
6: **end for**
7: **return** $\mathbb{EI}$

Secure image retrieval

In the secure image retrieval phase, the authorized medical image user *MIU* creates the search trapdoor and sends it to the cloud for similar image retrieval. The following subsections detail trapdoor generation, secure similar image search, and image decryption.

Trapdoor generation

The medical image owner *MIO* generates an encrypted search trapdoor using Algorithm 4 on the client side. Both query medical image M_Q and query trapdoor generation key K_Q are required. The query image image hashcode H_Q is generated using the proposed

ConvNeXt-based deep hashing module explained in Deep hashing using ConvNeXt model section. The d dimensional hashcode is expanded into $(d+3)$ dimensional vector using ω as follows:

$$\hat{y}_q = (1, -2h_{q1}, -2h_{q2}, ..., -2h_{qd}, \omega, 1) \tag{14}$$

The trapdoor is encrypted using the invertible matrix P and produces TD_Q as shown in Eq. 15. Finally, the TD_Q is sent to the cloud for searching.

$$TD_Q \leftarrow \hat{y}_q.P^T \tag{15}$$

Algorithm 4 `TrapdoorGen`

Input: Query Image M_Q, Query Trapdoor Generation Key $K_Q = \{P, \omega\}$
Output: Search Trapdoor TD_Q

1: Generate the hashcode H_Q for the M_Q using the ConvNeXt based deep hashing model
2: Decompose the d-dimensional hashcode
$H_Q = \{h_{q1}, h_{q2}, ..., h_{qd}\}$
3: Expand the H_Q into $(d+3)$ using $\omega \in K_Q$
$\hat{y}_q = (1, -2h_{q1}, -2h_{q2}, ..., -2h_{qd}, \omega, 1)$
4: Encrypt the $\hat{y}_q$ using P
$TD_Q \leftarrow \hat{y}_q.P^T$
5: **return** TD_Q

Secure similar image search over encrypted indexes

The cloud server receives the TD_Q and searches over the encrypted index database $\mathbb{EI}$ and returns the similar images result set $\mathbb{E}_\mathbb{Q}$, which are encrypted to the *MIU*. Algorithm 5 shows a secure similar image search. The retrieval process loops until the whole image collection is traversed.

Algorithm 5 `SecSimSearch`

Input: Trapdoor TD_Q, Encrypted index table $\mathbb{EI}$, Encrypted images $\mathbb{E}$
Output: Encrypted Query Result $\mathbb{E}_\mathbb{Q}$

1: **for** i=1 to n **do**
2: Calculate distance $\mathcal{D}$ between EI_i and TD_Q
$\mathcal{D} = EI_i.TD_Q = \theta + constant$ (Refer to Equation 16)
3: **if** $\mathcal{D} \leq threshold$ **then**
4: Update the $\mathbb{E}_\mathbb{Q}$ with EI_i
5: **end if**
6: **end for**
7: **return** $\mathbb{E}_\mathbb{Q}$

The proof of correctness is given in Eq. 16. θ is the distance between the query hashcode H_Q and i^{th} hashcode in the encrypted index (H_i).

$$\begin{aligned}
\mathcal{D} &= EI_i.TD_Q \\
&= (\hat{x}_i.P^{-1}).(\hat{y}_q.P^T)^T \\
&= \left(\sum_{j=1}^{d} h_{ij}^2 + \epsilon_1 - \epsilon_2, h_{i1}, h_{i2}, ..., h_{id}, 1, \epsilon_2\right).(1, -2h_{q1}, -2h_{q2}, ..., -2h_{qd}, \omega, 1)^T \\
&= \sum_{j=1}^{d} h_{ij}^2 + \epsilon_1 + \omega - 2h_{i1}h_{q1} - 2h_{i2}h_{q2} - ... - 2h_{id}h_{qd} \\
&= \sum_{j=1}^{d} h_{ij}^2 - 2\sum_{j=1}^{d} h_{ij}h_{qj} + \epsilon_1 + \omega \\
&= \sum_{j=1}^{d} (h_{ij} - h_{qj})^2 + \left(\sum_{j=1}^{d} h_{qj}^2 - \epsilon_1 - \omega\right) \\
&= H_i.H_Q^T + \left(\sum_{j=1}^{d} h_{qj}^2 - \epsilon_1 - \omega\right) \\
&= \theta + constant
\end{aligned} \tag{16}$$

In the proof, $\left(\sum_{j=1}^{d} h_{qj}^2 - \epsilon_1 - \omega\right)$ term will not have an impact on the final search result. This term will be negligible. Therefore, it is shown that the similarity metric between encrypted hashcodes is equivalent to plain hashcodes, and the distance of the encrypted domain is proportional to the plaintext domain from Eq. 16. The Euclidian distance between H_i, H_Q is expressed in Eq. 17. The proposed encrypted indexing and retrieval scheme has been demonstrated in Appendix A with a working example showing proof of the working model.

$$\theta = dist(H_i, H_Q) = \sum_{j=1}^{d} (h_{ij} - h_{qj})^2 \tag{17}$$

Image decryption

Cloud servers return only retrieved encrypted images $\mathbb{E}_{\mathbb{Q}}$ to *MIU*. The users have decryption key K_S at their end to decrypt the retrieved images. Decryption is the reverse flow of encryption shown in Algorithm 2. In addition, the integrity check takes place. At the time of decryption, unique SHA-512 hash-based bit-plane verification takes place to ensure the decrypted image's integrity. If the bit planes match, the decryption process will proceed to step 5, and the decrypted image will be generated and added to $\mathbb{S}$. Otherwise, a random cipher image will be added into $\mathbb{S}$. When the attacker tries to guess the key by encrypting the same image with the same wrong key twice, the attacker will get two different random cipher images, which confuses the attacker. There could not be any pattern to guess the key. The proposed scheme is IND-CPA secure. So, information leakage of medical image data is prevented successfully.

Security and privacy model

In the proposed system, there is an assumption that *CS* which is "honest and curious" and malicious users. The system has to ensure that the ciphertext does not leak any critical information to them. For that, the security model of the system requires that no adversary can distinguish ciphertext. The authors introduce a security game to define the security model based on the ciphertext indistinguishable under a chosen plaintext attack (IND-CPA), mainly referring to two participants, challenger $\mathcal{C}$ and adversary $\mathcal{A}$. The concrete description is as follows:

- **Adversary $\mathcal{A}$:** Chooses two different medical images $M_0, M_1 \in \mathbb{M}$ and hashcodes $H_0, H_1 \in \mathbb{H}$ to send it to challenger $\mathcal{C}$
- **Challenger $\mathcal{C}$:** Chooses random $r \leftarrow \{0,1\}$ and computes the challenged ciphertext images $E^r \leftarrow \mathsf{ImgEnc}(M_r, K_S)$ and encrypted index $EI^r \leftarrow \mathsf{SecIndexGen}(H_r, K_I)$. Send them to $\mathcal{A}$.
- **Adversary $\mathcal{A}$:** Outputs the guess $r' \in \{0,1\}$.

Either adversary wins the game or loses. So, the advantage of $\mathcal{A}$ can be defined as,

$$\mathcal{ADV}_{IND-CPA} = \left| Pr[r' = r] - \frac{1}{2} \right| \tag{18}$$

The adversary repeats the process. The index encryption proposed in this system is IND-CPA secure if $\mathcal{ADV}_{IND-CPA}$ is negligible for any adversary $\mathcal{A}$. In the system, the operations are performed as specified in the protocol definitions. At the same time, the adversary can do some extra analyses to extract any meaningful information regarding the data set and queries.

Definition 1 (Index Privacy): No polynomial time adversary can decrypt the secure searchable encrypted index *EI*.

Definition 2 (Query Privacy): No polynomial time adversary can decrypt a secure search query TD_Q.

Definition 3 (Image Privacy): No polynomial time adversary can recover sensitive information, including encryption keys or image content, from the timing information of cryptographic operations.

All these claims will be proved with a solution given in the subsequent section. The security analysis section of this article will explain the theories that prove the claims.

Security and privacy analysis

This section is dedicated to analyzing the security and privacy of the proposed image encryption model and secure index encryption model. The image encryption model is analyzed from various perspectives to prove its security. The image encryption model demonstrates side-channel attack prevention. Index and query privacy are theoretically proven.

Index privacy analysis

Theorem 1 *The SMedIR provides confidentiality for the index* $\mathbb{EI}$ *of a medical dataset* $\mathbb{M}$ *in accordance with Definition* 1.

Proof The hashcodes are encrypted using the expanded random vector $\hat{x}$ and the invertible matrix P^{-1}, which is randomly generated to secure the contents of the hashcode. The secure indexing is impacted by P^{-1} and random numbers ϵ_1, ϵ_2. Because of the random number's indistinguishability, the same hashcode produces different indexes under the same key; thus, adversary $\mathcal{A}$ cannot obtain any information about the plain hashcode in a polynomial time. This guarantees the safety of the generated content-based hashcodes. It is known that the random reversible matrix $P^{-1}, \epsilon_1, \epsilon_2$ are chosen from the cyclic group $\mathbb{Z}_\alpha$. Let $|\alpha|$ be the number of elements in $\mathbb{Z}_\alpha$, then the probability advantage of adversary $\mathcal{A}$ winning the game as defined in Eq. 18 is:

$$\mathcal{ADV}_{IND-CPA} = \left| Pr[r' = r] - \frac{1}{2} \right| = \left| \left(\frac{1}{2} + \frac{1}{|\alpha|} \right) - \frac{1}{2} \right| = \frac{1}{|\alpha|} \tag{19}$$

The advantage probability of adversary $\mathcal{A}$ is negligible, so the proposed SMedIR is IND-CPA secure. No polynomial-time adversary $\mathcal{A}$ can decrypt the secure searchable encrypted index. □

Query privacy analysis

Theorem 2 *The SMedIR provides confidentiality for encrypted query* TD_Q *in accordance with Definition* 2.

Proof The medical image user always randomly selects a number ω to expand the query image hashcode H_Q into $\hat{y}$. *MIU* uses the random reversible matrix P to encrypt the $\hat{y}$. The random number ω and P are indistinguishable. Therefore, several search trapdoors TD_Q can be generated for the same query image M_Q. Even if adversary $\mathcal{A}$ gets the TD_Q, $\mathcal{A}$ cannot deduce the information about ω and P from TD_Q. Therefore, query privacy is protected, the query pattern is successfully hidden, and no information about the query image M_Q is obtained. □

Search privacy analysis

Theorem 3 *The SMedIR provides confidentiality for secure similar image search.*

Proof The honest and curious *CS* knows the operation for search, which is $EI_i.TD_Q$ yielding *D*. From the operation, *CS* cannot obtain any information about the data. From the Theorem 1, secure index EI_i is IND-CPA secure. From the Theorem 2, TD_Q is IND-CPA secure.

$$D = \theta + constant = H_i.H_Q^T + \left(\sum_{j=1}^{d} h_{qj}^2 - \epsilon_1 - \omega \right) \tag{20}$$

θ contains terms derived from the plaintext hashcode H_i and the query hashcode H_Q but also contains random numbers $\epsilon_1, \epsilon_2, \omega$. So, it does not reveal any information about the plaintext image beyond what is already known. Therefore, the encrypted representation *D* does not leak any sensitive information about the underlying plaintext

data beyond what is permissible, satisfying the definition of search privacy. So, from *D*, *CS* cannot derive any sensitive key information about the sensitive images. Thus, the search operation itself is IND-CPA secure.

Analysis of the image encryption model

The proposed integrity-centric image encryption scheme's security is evaluated using a number of metrics and proved that it is resistant to various cryptographic attacks including brute-force attacks, statistical attacks, histogram attacks, and differential attacks. Throughout the subsection, 6 sample medical images ($M_1 - M_6$) are taken to show the performance comparison. Figure 7 shows the selected sample medical images. The detailed dataset description is available in Experimental setup and datasets section.

Key space and sensitivity analysis

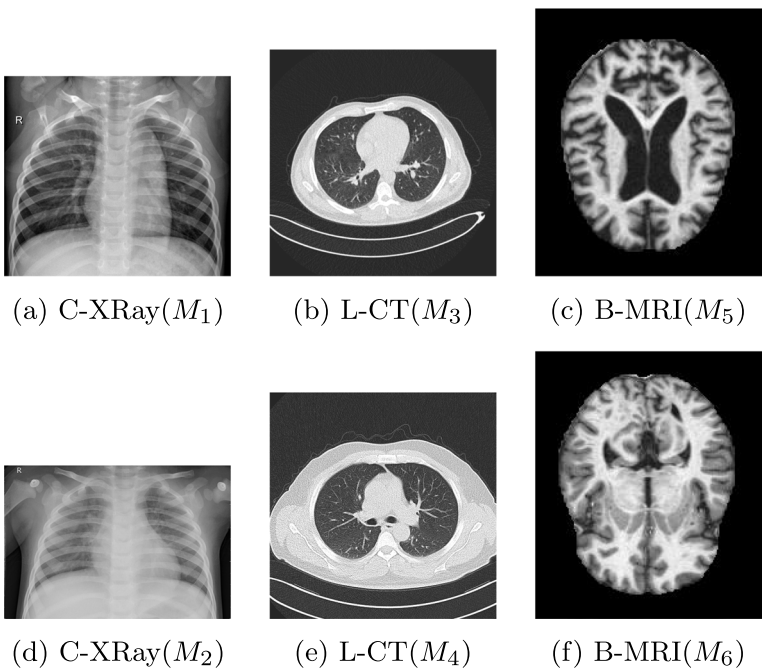
To withstand possible attacks, the encryption algorithm's key space must possess sufficient size [16]. It should be larger than 2^{128}. The proposed system does have 7 secret keys. If the precision is 10^{-14} (i.e., double precision (2^{52})), then the size of key space would be equal to $2^{52} * 7 = 2^{364} > 2^{128}$. The findings indicate that the method is significantly resilient to brute-force attacks and demonstrate the difficulty of successfully breaching it through this method. This model is highly key-sensitive as well. If k_{lx} varies by 0.001, then the process of decryption collapses.

Statistical attack analysis

The statistical properties of the image are depicted through the histogram. It counts the number of pixels and primarily displays the distribution of pixel values within the image. The histogram of an image will be flat when all of the pixel value numbers are almost equal. This demonstrates its strong resistance to statistical attacks. The histograms of sample plain images (M_1, M_3, M_5) are shown in Fig. 8a-c. The corresponding image's encrypted image histograms are displayed in Fig. 8d-f. The pixel values in the cipher image are uniformly distributed. This experiment demonstrates how the grey value distribution in the original image can be effectively hidden by the cipher image. It protects the data from histogram-based attacks.

An indicator for assessing an image's randomness is image correlation. The nearby pixels in a plain image had a strong correlation [46]. Encrypting the plain image may result in an increase in pixel variance. In image analysis, there are three correlation coefficients: horizontal, vertical, and diagonal. Equations 21-24 can be used to get the correlation coefficient of an image in any direction.

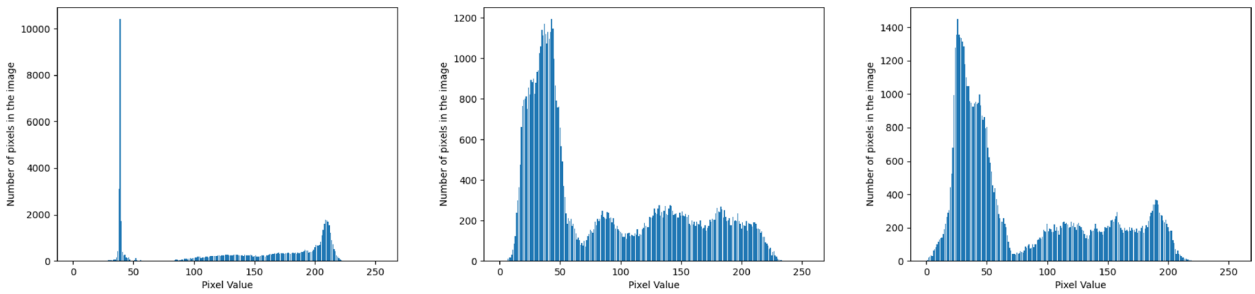
$$E(x) = \frac{1}{n}\Sigma_{i=1}^{n} x_i \tag{21}$$



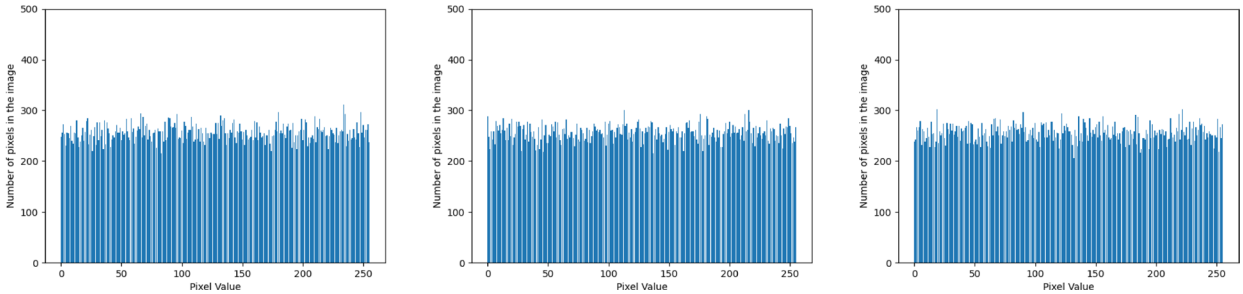
(a) C-XRay(M_1) (b) L-CT(M_3) (c) B-MRI(M_5)

(d) C-XRay(M_2) (e) L-CT(M_4) (f) B-MRI(M_6)

Fig. 7 Selected sample medical images

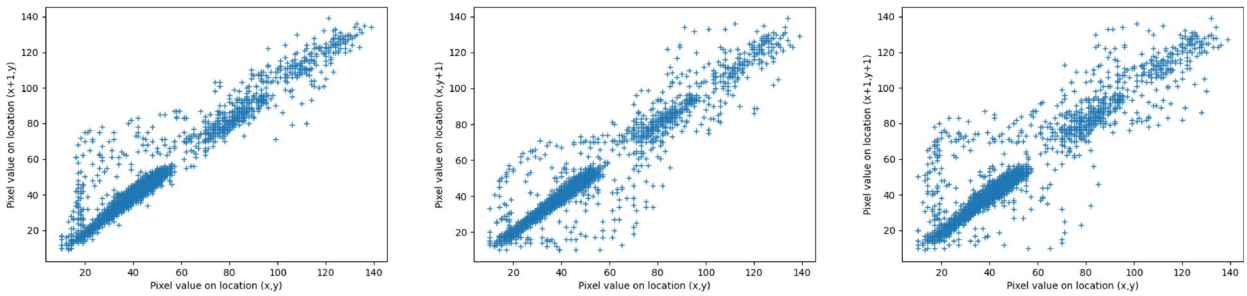


(a) Original (M_1) (b) Original (M_3) (c) Original (M_5)

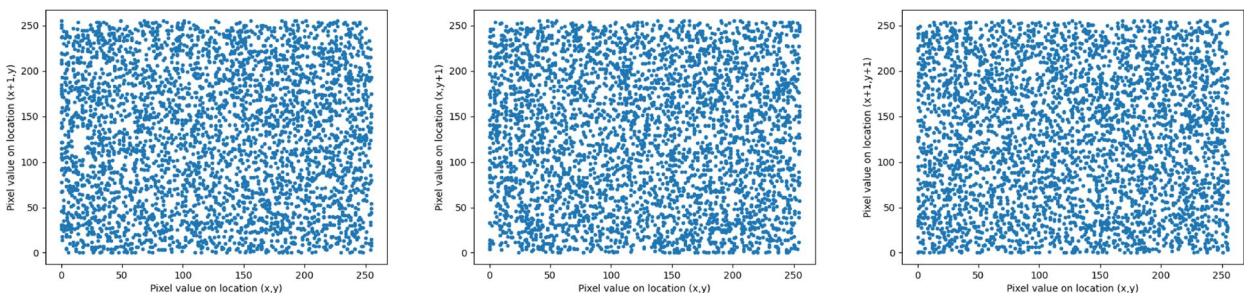


(d) Encrypted (M_1) (e) Encrypted (M_3) (f) Encrypted (M_5)

Fig. 8 Histogram analysis



(a) Plain - Horizontal (b) Plain - Vertical (c) Plain - Diagonal



(d) Encrypted - Horizontal (e) Encrypted - Vertical (f) Encrypted - Diagonal

Fig. 9 Correlation analysis (M_4)

Table 4 Correlation analysis

Image	Horizontal correlation		Vertical correlation		Diagonal correlation	
	Original	Encrypted	Original	Encrypted	Original	Encrypted
M_1	0.9544	-0.005	0.8151	0.0055	0.9923	0.1237
M_2	0.8853	-0.0006	0.8079	-0.0013	0.9964	-0.0027
M_3	0.9905	-0.0058	0.9193	0.0024	0.9840	-0.0315
M_4	0.9931	-0.0076	0.9459	0.0052	0.9809	-0.0054
M_5	0.9844	0.0074	0.9803	0.0003	0.9857	0.1331
M_6	0.9919	0.0061	0.9804	-0.0005	0.9916	0.1455

$$D(x) = \frac{1}{n}\Sigma_{i=1}^{n}(x_i - E(x))^2 \tag{22}$$

$$cov(x,y) = \frac{1}{n}\Sigma_{i=1}^{n}(x_i - E(x))(y_i - E(y)) \tag{23}$$

$$\rho_{xy} = \frac{cov(x,y)}{D(x)D(y)} \tag{24}$$

Figure 9a-f clearly project the randomness in Horizontal, Vertical, and Diagonal directions of the plain and encrypted image M_4. The correlation coefficients for horizontal, vertical, and diagonal directions are presented in Table 4, demonstrating the statistical attack resistance of the proposed encryption scheme on the selected sample medical images.

Differential attack analysis

The sensitivity of the encryption technique to even the smallest change in the plain image is evaluated using the differential attack. The Number of Pixel Change Rate (NPCR) and Unified Average Change in Intensity (UACI) are the two most important performance metrics to assess the proposed technique's resistance to differential attacks [47]. NPCR determines the pixel rate in the encrypted images whenever a single pixel of the test image is modified. It is employed to evaluate the resistance to differential attack. Its ideal value is greater than 99%. NPCR is calculated using the following Equations:

$$NPCR = \frac{\Sigma_{x,y} Diff(x,y)}{W \times H} \times 100\% \tag{25}$$

Here,

$$Diff(x,y) = \begin{cases} 1, & \text{if } M(x,y) \text{ equals } E(x,y) \\ 0, & \text{if } M(x,y) \text{ not equals } E(x,y) \end{cases} \tag{26}$$

where W and H represent the width and height of the image, respectively. $Diff(x, y)$ indicates the difference between the corresponding pixels of the original medical image M and the encrypted medical image E.

UACI is the average difference in pixel intensity between the original and encrypted image. It is yet another commonly employed efficiency indicator for evaluating the capacity to withstand a differential attack. For UACI, a value of approximately 33% is optimal. It is calculated based on the following Equation:

$$UACI = \frac{\Sigma_{x,y} M(x,y) - E(x,y)}{255 \times W \times H} \times 100\% \tag{27}$$

Table 5 shows the UACI and NPCR of the selected sample images in the proposed system and proves the withstanding power of the proposed encryption model. The extended statistical security and comparative analysis have been shown in Appendix B.

Table 5 Differential attack analysis

Image	NPCR	UACI
M_1	99.71	32.48
M_2	99.72	33.88
M_3	99.67	33.68
M_4	99.68	33.87
M_5	99.73	33.92
M_6	99.72	33.31

Side channel attack prevention

If the attacker gets the timing information of the security operations, they could deduce the key length, process length, etc. So, in the proposed model, this attack could not be succeeded as the introduced noise in the time. This is proved in the Theorem 4.

Theorem 4 *For the proposed integrity-centric image encryption scheme, it is computationally infeasible for a*

polynomial-time adversary to recover sensitive information, such as encryption keys or image content, solely from the timing information of cryptographic operations in accordance with Definition 3.

Proof Let's consider a cryptographic operations OP_c which performs encryption or decryption by taking the input M and key K. The execution time of the operation is T_{OP_c}. ϵ is the random variable that represents the noise added to the execution time. The noisy execution time T'_{OP_c} is represented by,

$$T'_{OP_c} = T_{OP_c} + \epsilon \tag{28}$$

To achieve differential privacy (DP), need to ensure that the noisy execution time satisfies the definition of DP, which states that for any pair of inputs M_1 and M_2 differing in a single data point, the probability distributions of the outputs should be similar. Mathematically, this can be expressed using the ϵ-DP definition.

$$Pr[\mathcal{M}(D_1) \in S] \leq exp(\epsilon).Pr[\mathcal{M}(D_2) \in S] \tag{29}$$

Here $\mathcal{M}$ is a mechanism that takes inputs from the dataset and produces outputs, in this case, which is a cryptographic operation, either encryption or decryption. $\mathcal{M}$ satisfies ϵ -DP if for all pairs of neighboring datasets D_1, D_2 that differ a single data point, and for all subsets of possible outputs S. The output is T'_{OP_c}. The noise ϵ should be sufficient to mask any correlation between the execution times and sensitive data while preserving the utility of the cryptographic operations.

Experimental results and performance analysis

The analysis in this section focuses on effective retrieval performance and is dedicated to demonstrating the results of the proposed SMedIR through experiments. The structure of this section is structured as follows: an explanation of the dataset with details on the experimental setup and results, an analysis of retrieval accuracy, and an assessment of retrieval time.

Experimental setup and datasets

The designed system was carried out on a PC with an Intel Xeon processor, 64 GB RAM, an NVIDIA Quadro P5000 GPU with 16 GB of memory, and a 64-bit Windows operating system. Python OpenCV libraries and Keras have been used in the development of the entire system. To experiment and evaluate the proposed retrieval model, three different medical image datasets are chosen. Details of the dataset have been explained briefly here,

- **Pneumonia Chest X-Ray Dataset (C-XRay)** [48]**:** The dataset comprises 5,863 Chest X-Ray images categorized into Pneumonia and Normal classes, sourced from pediatric patients aged one to five years at Guangzhou Women and Children's Medical Center. Quality control measures were applied, and diagnoses were graded by two expert physicians before training the system. A third expert independently reviewed the evaluation set to ensure accuracy.
- **Lung Cancer CT Dataset (L-CT)** [49]**:** The Iraq-Oncology Teaching Hospital/National Center for Cancer Diseases (IQ-OTH/NCCD) lung cancer dataset was collected over three months in fall 2019 from specialist hospitals. It includes CT scans from patients with lung cancer in different stages and healthy subjects, totaling 1190 images from 110 cases. The dataset, marked by oncologists and radiologists, categorizes cases into three classes: normal, benign, and malignant, with 40 malignant, 15 benign, and 55 normal cases.
- **Alzheimer Brain MRI Dataset (B-MRI)** [50]**:** The dataset comprises two files, Training and Testing, each containing approximately 5000 images. These images are categorized based on the severity of Alzheimer's disease into the following classes: Non Demented, Very Mild Demented, Mild Demented, and Moderate Demented.

The X-ray, CT, and MRI images are chosen for the retrieval task to illustrate the versatility and efficacy of the model across different imaging techniques, highlighting its adaptability and reliability in various clinical contexts. The datasets are divided into training and testing sets at an 80:20 ratio. Retrieval accuracy is assessed on a randomly sampled subset of 1000 images during testing. In the proposed system, medical images are encrypted and outsourced to the cloud with encrypted indexes. When the query image is given to the cloud, a secure similar image search algorithm searches over the encrypted index table and returns the top-k medical images. Figure 10 displays sample obtained top-k images for each encrypted query image from various datasets. Column 2 shows the query images for each class, whereas Column 1 shows the dataset of the chosen query image. columns 3-7 show the images retrieved as similar to the query. In the retrieved results, the exact query image retrieved is highlighted with a green border, while completely irrelevant class images, which reduce precision, are highlighted with a red border.

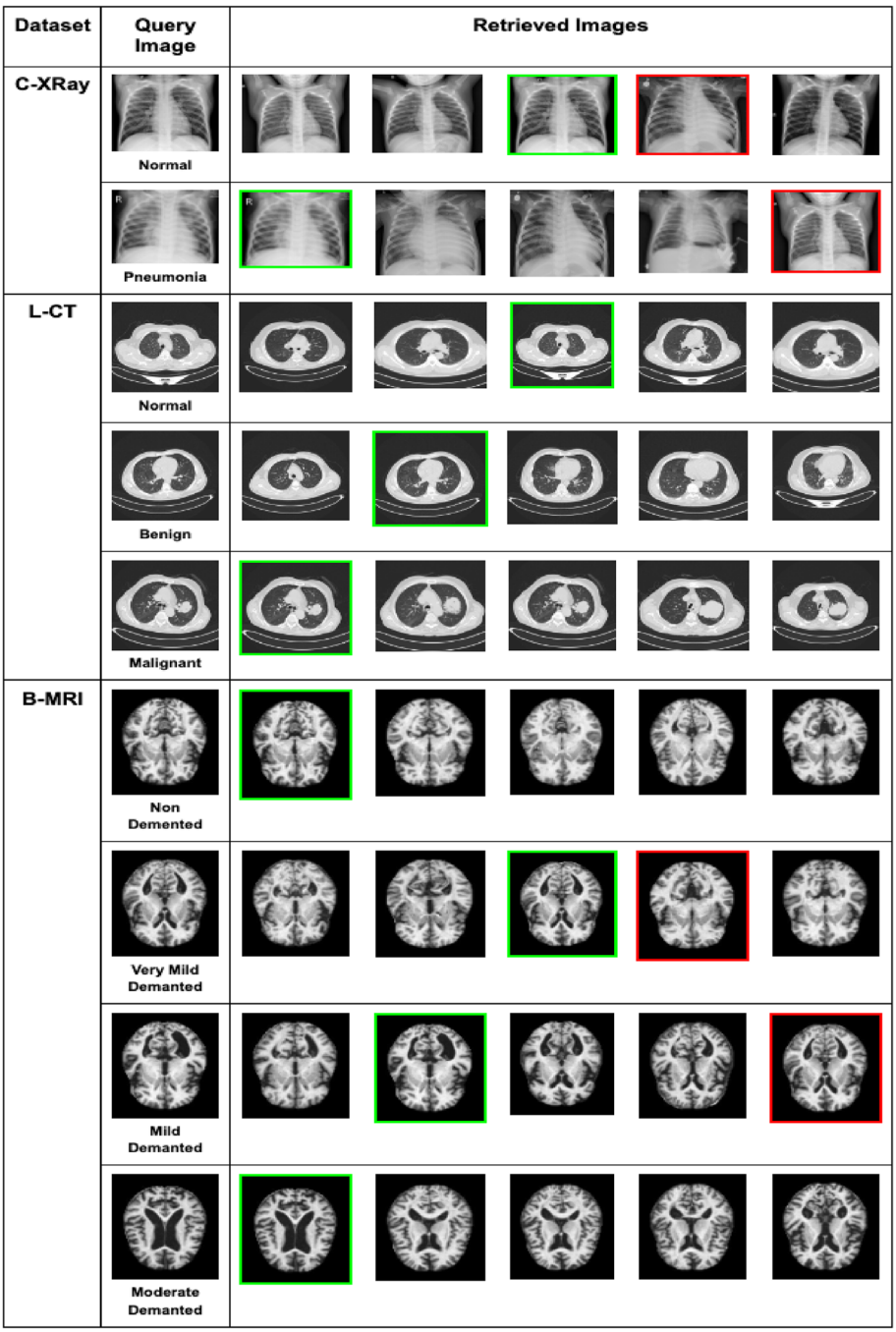


Fig. 10 Sample retrieval results: Query image is in column 2 and top-5 retrieved results for that query in subsequent columns. The most relevant retrieved image is highlighted in green border and the most irrelevant retrieved image is highlighted in red border

Retrieval performance analysis

In order to evaluate the proposed SMedIR which is a secure medical image retrieval system, selected two metrics Mean Average Precision, and PR Curve (ROC) Analysis. The proposed method has been tested with three different medical datasets and compared with four previous state-of-the-art secure image retrieval models IES-CBR [51], LBP-BOW [34], VFIRM [37], and TAMMIA [36] as baseline models. Top-k retrieved images are used to estimate the retrieval accuracy. Image retrieval accuracy can be measured using Precision (Pr@k), Recall (Re@k), and Mean Average

Table 6 Comparative analysis of deep hashing models

Name	Backbone Net	Loss used	mAP @ 100		
			C-XRay	L-CT	B-MRI
UDTH [26]	VGG-16	Triplet Loss	0.61	0.62	0.58
CDHN [27]	ResNet50	Reconstruction Loss	0.65	0.68	0.64
IFFH [28]	DenseNet121	Binary Cross Entropy Loss	0.68	0.74	0.67
IPH [29]	DenseNet201	Binary Cross Entropy Loss	0.71	0.76	0.66
Ours	ConvNeXt	Contrastive Loss	0.75	0.81	0.72

Precision (mAP@k) metrics. Precision refers to the ratio of relevant retrieved images to the total number of retrieved images in relation to the query image.

$$Pr = \frac{|\text{relevant images} \cap \text{retrieved images}|}{|\text{retrieved images}|} \tag{30}$$

Recall denotes the proportion of relevant retrieved images to the query image, considering the number of identical images in the entire dataset.

$$Re = \frac{|\text{relevant images} \cap \text{retrieved images}|}{|\text{total relevant images in the dataset}|} \tag{31}$$

Mean Average Precision (mAP) is the standard measure for assessing and comparing the accuracy of image retrieval. The calculation of mAP can be performed using the Equation below.

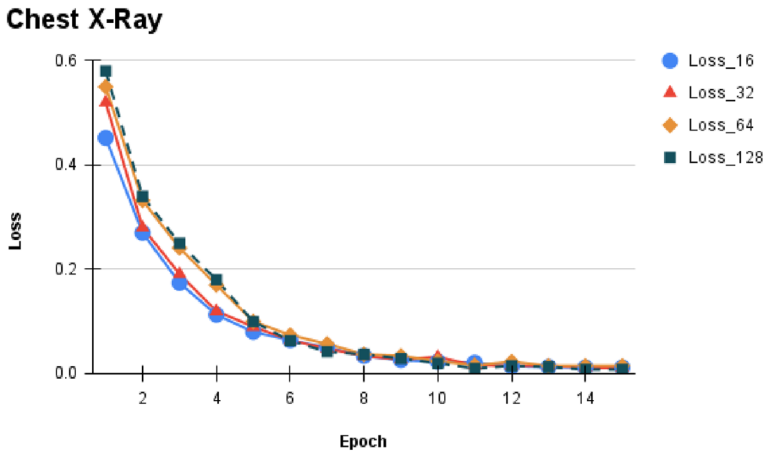
$$mAP = \frac{1}{R}\sum_{m=1}^{R}\left(\frac{1}{q_m}\sum_{n=1}^{q_m} Pr_n\right) \tag{32}$$

Here, R is a number of queries, q_m is a number of relevant images for query n and the Pr_n is precision at n^{th} relevant image. For the hashcode generation, the ConvNeXt network is used for feature extraction, and the contrastive loss is primarily used for learning. The hashcode is then encrypted using SE before outsourcing. ConvNeXt model is used because it extracts both local and global features better than other pre-trained DL models. This leads to the production of more meaningful hashcodes than other models. The experiment was carried out with different deep hashing models and finally the ConvNext, the best backbone model was chosen. When the hashcodes are generated with other backbone networks and encrypted with the proposed SE model, they produce less retrieval accuracies in all medical datasets. This leads the authors to choose the ConvNeXt-based deep hashing for better-encrypted image retrieval. Table 6 shows the comparative analysis. For the comparison, 4 different deep hashing models: UDTH [26], CDHN [27], IFFH [28], IPH [29] are taken. During the comparison of models, changes are made to the backbone and loss functions, but the encrypted indexing remains as proposed in our model. The details of the models are explained in Appendix C.

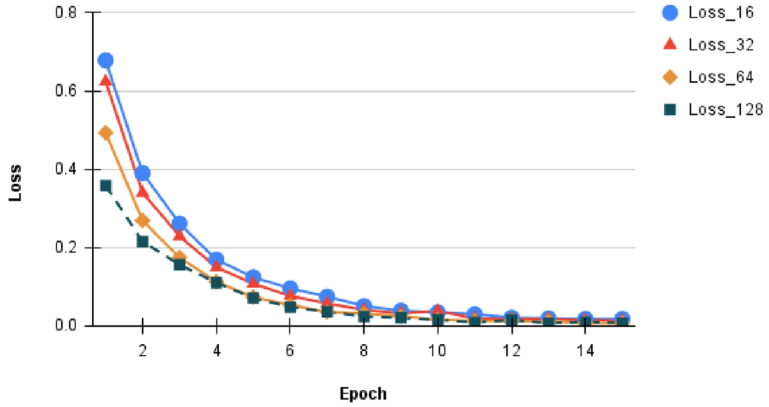
Hashcode is generated as illustrated in Fig. 6. The ConvNeXt network is designed to propagate cumulative loss, combining contrastive loss, quantization loss, and bit balance loss. The objective is to generate a discriminative hashcode while minimizing the loss defined in Eq. 11. The network undergoes training for hashcodes of lengths 16, 32, 64, and 128 bits for each dataset separately. Over epochs, the training loss is depicted in Fig. 11a-c. As epochs increase, the total loss decreases.

Randomly selected 1000 samples from each medical dataset are used for the analysis of retrieval accuracy. The experimental results of the proposed model, including mAP values across three datasets with varying hashcode lengths and different values of k, are documented in Table 7. Figure 12 illustrates the impact of hashcode length on image retrieval accuracy using mAP@100. The observation indicates that the 32-bit hashcode yields the highest retrieval accuracy. With less number of bits, the hashcodes are not able to capture the features with respect to classes. However, when the number of bits is high then the hashcodes get sparsed, and submerged into different classes. In our model, 32 bits provide better results. From Table 7, it is observable that the authors have tested the performance at different k values as well. When the k is less then the performance is better. A decrease in mAP with increasing k values in top-k retrieval suggests that the retrieval system's precision decreases when more images are retrieved (higher k values). Possible contributing aspects to this phenomenon include the dispersion of relevant items, challenges in discerning relevant from irrelevant images, or intrinsic features present in the medical image data. From the analysis, the d=32 and k=100 are fixed to compare SMedIR framework with other baseline models.

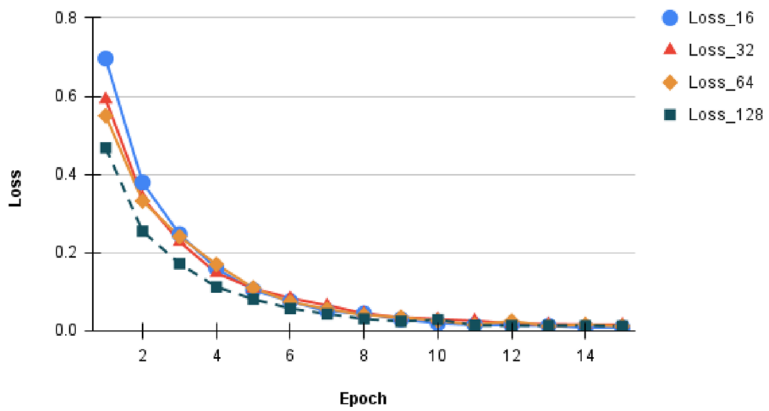
To underscore the effectiveness of the method, Precision curves at k retrieved images (P@k) and Precision-Recall (PR) curves are produced across three datasets. While outcomes may differ across various domain datasets, P@k curves depict precision at fixed numbers of retrieved images. Figure 13 illustrates the P@k curve for all datasets, demonstrating that SMedIR consistently attains superior precision compared to other methods across all three datasets, notably excelling in the case of L-CT. At k=100, with C X-ray images, our model exhibits a 25% improvement over IES-CBR and a 14% enhancement compared to VFIRM. Both IES-CBR and LBP-BOW achieve identical mAP @ 500 scores. On average, across the X-ray image dataset, our model surpasses TAMMIE, the latest encrypted image



(a)



(b)



(c)

Fig. 11 Hashcode learning: training loss

Table 7 Detailed retrieval results: three different dataset retrieval mAPs are compared under varying conditions

Medical Dataset	SMedIR			
	Hash Code Length	16b	32b	64b
	k			
C-XRay	100	0.69	0.75	0.72
	200	0.65	0.70	0.69
	500	0.55	0.65	0.64
	800	0.45	0.60	0.56
L-CT	100	0.71	0.80	0.76
	200	0.65	0.72	0.71
	500	0.59	0.69	0.61
	800	0.52	0.60	0.55
B-MRI	100	0.65	0.72	0.66
	200	0.64	0.69	0.58
	500	0.53	0.57	0.51
	800	0.45	0.41	0.48

retrieval model, by 5%, and IES-CBR by 24%. In Lung Cancer CT images, SMedIR demonstrates superior performance, showcasing differences ranging from 5.9% to 19.8% compared to other models. Despite lower MRI image quality, our model excels in capturing features, leading to a higher mAP @ 100 score of 72%, surpassing IES-CBR by 27%. Additionally, our model exhibits substantial performance improvements ranging from 6.8% to 21.8% in the Brain MRI images dataset. Within the B-MRI dataset, our curves exhibit slight superiority over TAMMIE's, particularly noticeable when the count of retrieved images is under 800. This finding underscores our approach's efficacy in retrieving a greater number of accurate images compared to alternative methods, particularly evident with a constrained retrieval quantity, affirming its suitability for image retrieval tasks.

The Precision-Recall (PR) curve is a vital metric for contrasting the proposed methods with baseline approaches, providing a comprehensive view of precision and recall across various retrieval scenarios. It offers valuable insights into system performance across different sensitivity levels. A larger area under the PR curve typically indicates a more effective retrieval system capable of maintaining a balance between precision and recall. As demonstrated in Fig. 14, our method consistently outperforms other methods across all PR curves.

The summary of the retrieval accuracy is tabulated in Table 8. The table indicates that X-ray and CT images exhibit better retrieval accuracy compared to MRI images. This discrepancy can be attributed to various factors [52]. X-ray images focus on capturing density variations in bones and tissues, which may contribute to their superior performance in retrieval tasks. Conversely, while MRI images provide high-resolution details of soft tissues, their overall image quality may not be conducive to accurate feature extraction and deep hashing techniques. Additionally, CT images, combining X-ray with computer processing, offer a balanced view of both bones and soft tissues, contributing to their improved retrieval accuracy. Therefore, in this deep hashing-based retrieval, X-ray images are likely to outperform MRI images due to their more favorable characteristics for feature representation and extraction. Through a comprehensive analysis across diverse conditions on three distinct medical datasets and benchmarking against four baseline models

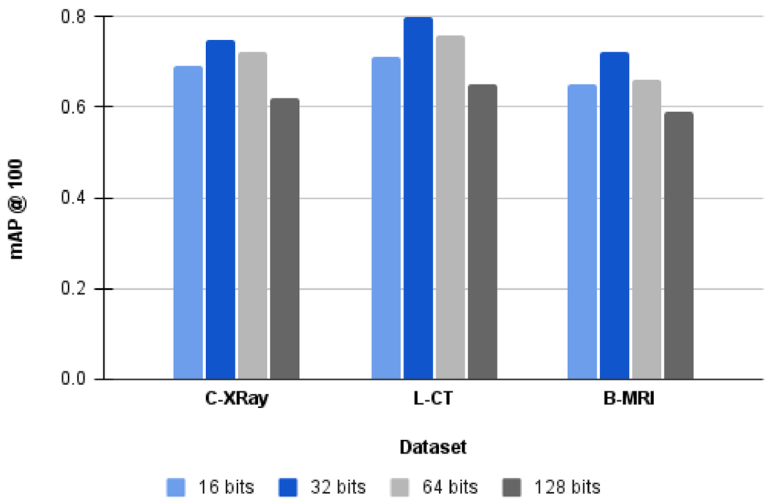
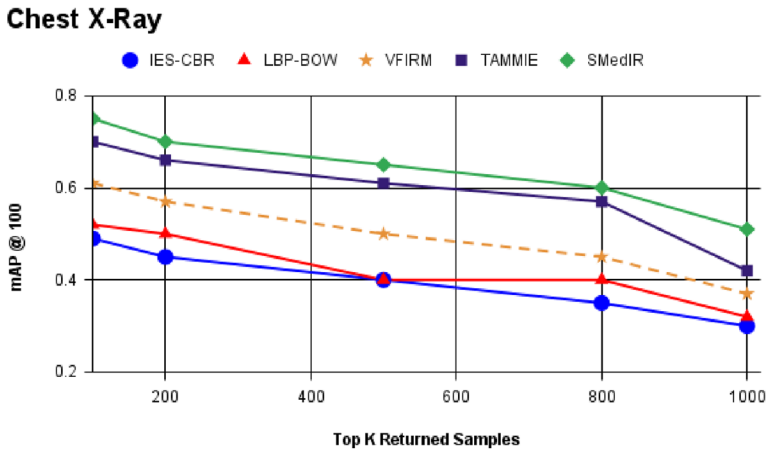
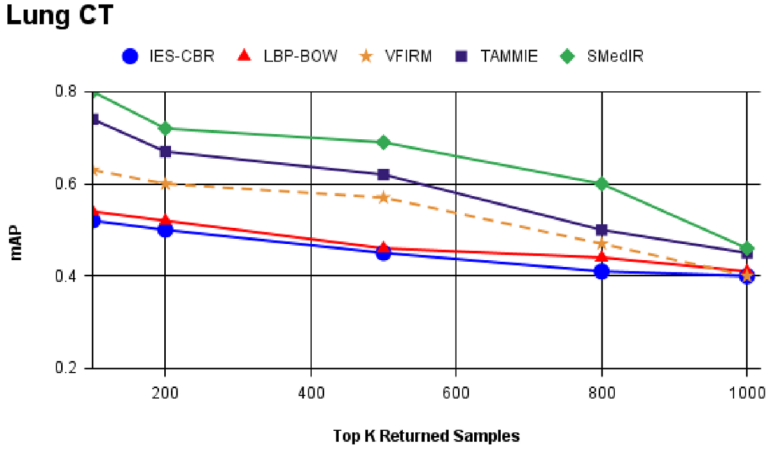


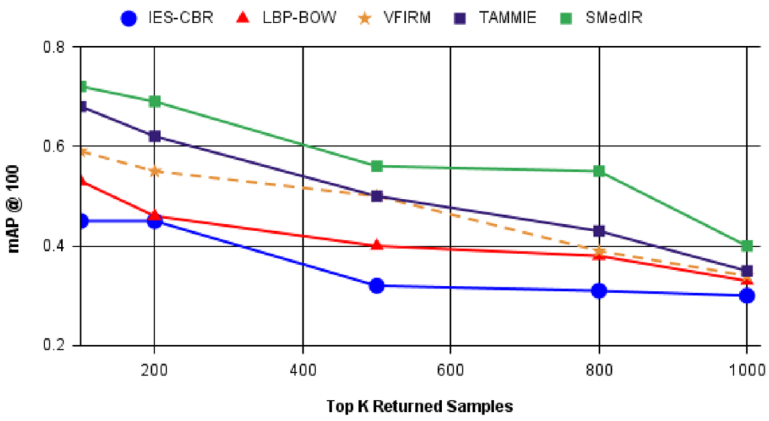
Fig. 12 Hashcode length vs mAP@100



(a)

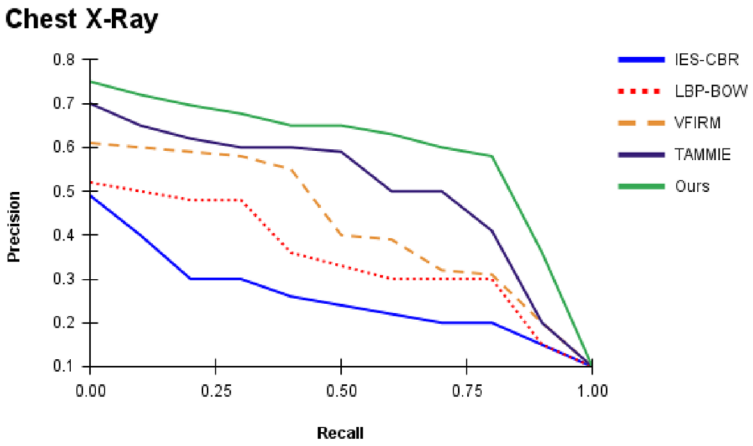


(b)

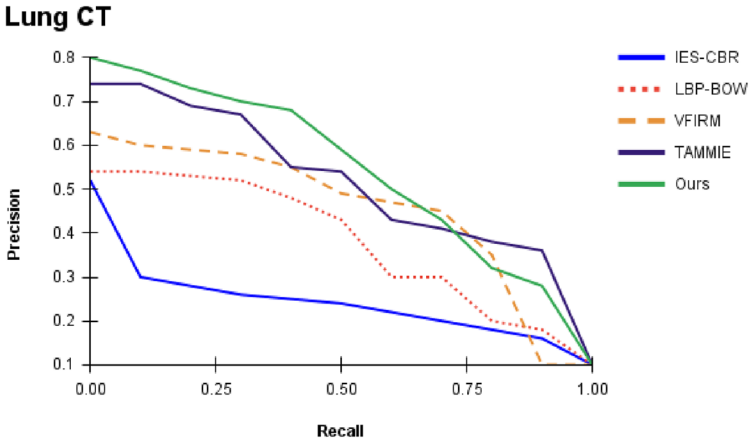


(c)

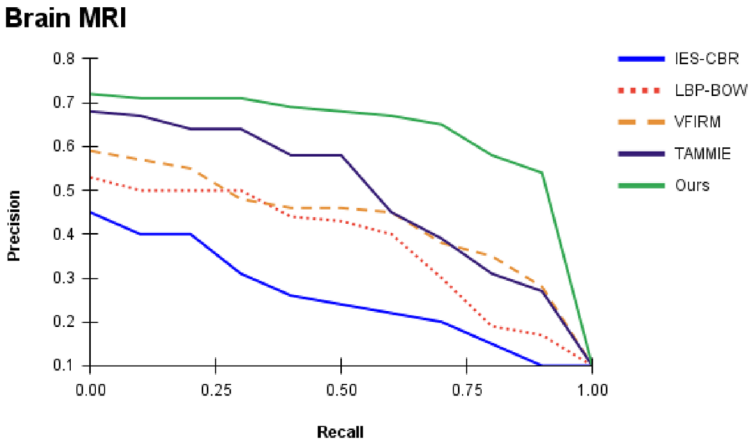
Fig. 13 Top k retrieved results vs mAP (P@k Curves)



(a)



(b)



(c)

Fig. 14 PR curves for all datasets

Table 8 Retrieval accuracy mAP@100: comparison with baseline models

Name	Security level	Method	mAP @ 100		
			C-XRay	L-CT	B-MRI
IES-CBR [51]	Adaptive Secure	Encrypted images as indexes	0.49	0.52	0.45
LBP-BOW [34]	COA and KBA Secure	Local Binary Pattern and Bag-of-Words	0.52	0.54	0.53
VFIRM [37]	Non-adaptive Semantic Secure	Verifiable Homomorphic encryption with Access control	0.61	0.63	0.59
TAMMIE [36]	IND-KPA Secure	Mahalanobis distance based Fuzzy C-Means	0.7	0.74	0.68
SMedIR (Ours)	**IND-CPA Secure**	**ConvNext-based indexing with searchable encryption**	**0.75**	**0.8**	**0.72**

Fig. 15 Search time analysis

using various performance metrics, the authors assert that the proposed model consistently outperforms other existing models in terms of retrieval accuracy.

Search time analysis

This work considers a linear index. Thus the time consumption is linear to the size of the image set and the dimension of the feature vector, as shown in Fig. 15. SMedIR is compared with LBP-BOW and shows an improvement of 10x times. For 1000 images, LBP-BOW takes 0.1s whereas our model takes 0.01s as it compares only encrypted hashcodes.

Conclusion

Medical images stored on third-party cloud platforms were vulnerable to multiple attacks, leading to potential information leakage. Given the high sensitivity of medical images, secure storage and retrieval in digital healthcare were imperative. This paper proposed SMedIR, a system designed to ensure better retrieval efficiency, privacy, and integrity of medical images. To achieve robust security and high retrieval accuracy, an integrity-centric image encryption scheme was employed for secure medical image storage. Additionally, ConvNeXt-based indexing with searchable encryption was used for efficient and secure image retrieval. The encryption scheme's security was thoroughly verified and validated through various experiments, including histogram analysis, differential attack analysis, entropy analysis, and key sensitivity analysis. Theoretical proofs for index privacy and query privacy were also provided. The performance of SMedIR was evaluated using accuracy metrics and search time, and it was compared with existing secure image retrieval models. Future work could involve introducing a tree-based encrypted index to further enhance performance measures.

Appendix A: Proof of working model

In this subsection, the toy example has been taken to demonstrate how SMedIR works. The public parameters are $\alpha = 3, d = 2, \mathbb{G}_{cy}$, which is a multiplicative cyclic group with order 3. KeyGen generates K_I, K_Q from the parameters. The matrix P and P^{-1} of dimension $(d+3) \times (d+3)$ which is 5×5 from $\mathbb{G}_{cy}$ are,

$$P = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 \\ 2 & 0 & 2 & 2 & 1 \\ 1 & 2 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 2 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix} P^{-1} = \begin{bmatrix} -0.125 & 0.25 & -0.5 & 0 & 1 \\ 0.125 & -0.25 & 0.5 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ 0.875 & 0.25 & 0.5 & -1 & -1 \\ 0.5 & 0 & 0 & 0 & 0 \end{bmatrix}$$

As P is invertible matrix, $PP^{-1} = P^{-1}P = \mathcal{I}$. If *MIO* are outsourcing 4 images with hashcodes [0, 0], [0, 1], [1, 0], [1, 1], Firstly, the hashcodes are expanded into a hashcode with dimension 5 using random numbers $\epsilon_1 = 0, \epsilon_2 = 1$. These are all the corresponding $\hat{x}_i$ of H_i. Then each $\hat{x}_i$ is multiplied with P^{-1} to get the secure indexes EI_i. Table 9 shows the secure indexing process of all hashcodes. The secure indexes are offloaded to the cloud with corresponding encrypted images E_i. If an authorized image user wants to search the image with hashcode $H_Q = [1, 1]$. Then, the trapdoor is generated using $\omega = 2$ and P. First, H_Q is expanded as shown in Eq. 14: $\hat{y} = [1, -2, -2, 2, 1]$. Then $\hat{y_q}$ is multiplied with P^T to get $TD_Q = [2, 3, -3, 0, -1]$. This TD_Q is sent to the cloud for secure similar image searching. Here, the distance between all the encrypted index and the query index is calculated as in Eq. 16. Table 10 shows the distance with each encrypted index in the cloud with TD_Q. Table 10 vividly shows that the search fetches [1, 1] as a most similar image, which is the system's goal. If *MIU* search for an image with hashcode [0, 1] then $\hat{y_q} = [1, 0, -2, 2, 1]$ and $TD_Q = [2, 3, 1, 0, 1]$. Secure search is shown in Table 11. Here also the search algorithm fetches the image with hashcode [0, 1] first.

Table 9 Secure indexing

H_i	$\hat{x}_i$	$EI_i = \hat{x}_i.P^{-1}$
[0, 0]	[−1, 0, 0, 1, 1]	[1.5,0,1,-1,-2]
[0, 1]	[0, 0, 1, 0, 1]	[-0.5,0,0,1,0.]
[1, 0]	[0, 1, 0, 0, 1]	[0.625,-0.25,0.5,0,0]
[1, 1]	[1, 1, 1, 0, 1]	[-0.5,0,0,1,1]

Table 10 Secure search for [1, 1]

H_i	$dist(H_i, H_Q = EI_i.TD_Q)$	EI_i
[0, 0]	2.0	[1.5,0,1,-1,-2]
[0, 1]	1.0	[-0.5,0,0,1,0.]
[1, 0]	1.0	[0.625,-0.25,0.5,0,0]
[1, 1]	-2.0	[-0.5,0,0,1,1]

Table 11 Secure search for [0, 1]

H_i	$dist(H_i, H_Q = EI_i.TD_Q)$	EI_i
[0, 0]	2.0	[1.5,0,1,-1,-2]
[0, 1]	-1.0	[-0.5,0,0,1,0.]
[1, 0]	1.0	[0.625,-0.25,0.5,0,0]
[1, 1]	0.0	[-0.5,0,0,1,1]

Appendix B: Security analysis of integrity-centric image encryption model

Chi-square test

The chi-square (χ^2) test is used to assess the histogram's evenness. It is calculated using the following Eq. 33.

$$\chi^2 = \Sigma_{i=0}^{255} \frac{(Observed_i - Expected_i)^2}{Expected_i} \tag{33}$$

Here the null hypothesis is that "Pixels are evenly distributed". Critical value is $\chi^2(255, 0.05) = 293$. If the χ^2 test value is less than 293, then the null hypothesis can be accepted. Table 12 shows the χ^2 test done over the sample images.

Table 12 χ^2 test

Image	χ^2 Value	Critical value	Decision (H=0)
M_1	268.53	293	Pass
M_2	257.05	293	Pass
M_3	278.97	293	Pass
M_4	254.36	293	Pass
M_5	276.48	293	Pass
M_6	292.12	293	Pass

Entropy analysis

Entropy is a term used to describe how unpredictable the image information is. E denotes the entropy of the image. It is used to assess the uncertainty of the proposed encryption technique and calculated using the following Eq. 34.

$$E = -\Sigma_{i=1}^{255} p_i log(p_i) \tag{34}$$

p_i denotes the probability of occurrence of pixel value i. The value of $E \in [0, 8]$. An 8-bit image should have an entropy value close to 8. Table 13 displays the entropy values of the selected sample images, indicating that the entropy of all encrypted images is in close proximity to 8.

Table 13 Entropy analysis

	Entropy	
Image	Original	Encrypted
M_1	6.3177	7.9970
M_2	5.3486	7.9971
M_3	6.9745	7.9969
M_4	7.3090	7.9971
M_5	7.1528	7.9969
M_6	6.9926	7.9968

Error metrics

There are few metrics available to assess the error of the encryption model. MAE, RMSE, and PSNR are standard metrics to assess whether the encryption scheme produces acceptable errors. MAE is used to measure the difference between encrypted and original images. MAE $\in [0, 2^n - 1]$, where n is the number of bits to represent each pixel. For a good encryption model, MAE should be maximum. It can be evaluated using Eq. 35.

$$MAE = \frac{1}{W \times H} \Sigma_{x,y} |E_M(x,y) - M(x,y)| \tag{35}$$

MSE is useful for comparing exact pixel values between an original image and a decrypted image. The error is the difference between the original image's and the decrypted image's pixel values. In order to provide more precise and reliable data, RMSE assesses the MSE root. A desirable encryption algorithm would yield a minimal RMSE value. Equations 36 and 37 can be used to calculate these measures.

$$MSE = \frac{1}{W \times H} \Sigma_{x,y} (E_M(x,y) - M(x,y))^2 \tag{36}$$

$$RMSE = \sqrt{MSE} \tag{37}$$

The range of RMSE $\in [0, \infty]$. PSNR is used as a quality measurement between the original and decrypted images. PSNR is mathematically computed as follows in Eq. 38.

$$PSNR = 10log_{10} \frac{(2^n - 1)^2}{MSE} \tag{38}$$

where n represents the number of bits per pixel. PSNR is measured in decibels (dB). Table 14 shows the error metrics of the sample images in the proposed model.

Table 14 Error metrics

Image	MAE	RMSE	PSNR(dB)
M_1	127.86	101.47	8.00
M_2	127.80	105.81	7.64
M_3	127.38	105.22	7.68
M_4	127.34	105.76	7.64
M_5	127.86	105.94	7.63
M_6	127.41	104.03	7.78

Comparative security analysis

Proposed Integrity-centric image encryption is compared with other existing image encryption models [16, 47, 53, 54]. The correlation coefficients, NPCR, UACI, and Entropy metrics are taken to compare our proposed method with existing encryption models. Table 15 reveals the better correlations and NPCR of the proposed model. Thus the proposed integrity-centric image encryption contains DNA-encoding, chaotic maps, and SHA-512 yields an improved NPCR value and reduces the correlation coefficient among the pixels in the encrypted image.

Table 15 Encryption model: comparative analysis

Reference	HCorr	VCorr	Dcorr	NPCR	UACI	Entropy
Wan et al. [16]	0.0105	0.0020	0.0019	99.61	33.52	7.9971
Khan et al. [47]	0.0008	0.0005	-0.0006	99.62	33.49	7.9990
Brahim et al. [53]	0.0007	0.0049	0.0030	99.58	33.49	7.9970
Wang et al. [54]	0.0070	0.0065	0.0067	99.61	33.48	7.9993
Our Model	**-0.0009**	**-0.0019**	**-0.0605**	**99.70**	**33.52**	**7.9972**

Appendix C: Baseline models explanation

1. **Unsupervised Deep Triplet Hashing (UDTH)** [26]: A novel framework for scalable image retrieval. UDTH leverages pseudo triplets based on high-dimensional visual features, employing a unique objective function to maximize binary representation distance across classes and preserve structural information through Autoencoder and Binary quantization.
2. **Class Driven Hashing Network (CDHN)** [27]: An efficient framework (CDHN) for indexing and retrieving MRI and CT medical images using deep features, aiming for optimal and minimally parameterized hashcodes. Utilizing a CNN for automatic feature extraction, the acquired deep features undergo effective reduction for optimal retrieval speed. Feature selection algorithms are applied to address the limitations of medical image datasets, producing better class-driven hashcodes.
3. **Interpretable Feature Fusion based Hashing (IFFH)** [28]: This approach combines interpretability and feature fusion by pre-training a DenseNet-121 network using the comparison to learn (C2L) method. An interpretable saliency map is obtained to locate focal regions, and features are fused to avoid information omission. The model incorporates a hash layer with classification and bit-balanced loss functions to generate high-quality hashcodes, improving retrieval accuracy.
4. **Interpretable Precise Hashing (IPH** [29]: It is a precision hashing method that combines interpretability and feature fusion to address the issue of low image resolution in brain tumor detection using the Brain-Tumor-MRI (BT-MRI) dataset. Initially, the pretrained the dataset with the DenseNet201 network employs the Comparison-to-Learn method.

Authors' contributions
Conceptualization: Arun Amaithi Rajan, Vetriselvi V, Mayank Raikwar, Reshma Balaraman; Methodology and Development: Arun Amaithi Rajan, Vetriselvi V; Formal analysis and investigation: Arun Amaithi Rajan, Vetriselvi V, Reshma Balaraman; Writing - original draft preparation: Arun Amaithi Rajan, Vetriselvi V, Mayank Raikwar, Reshma Balaraman; Writing - review, and editing: Arun Amaithi Rajan, Mayank Raikwar, Vetriselvi V, Reshma Balaraman; Supervision: Vetriselvi V.

Authors' information
Arun Amaithi Rajan received his Master's in Computer Science from the National University of Singapore in 2020. He worked as a Security Firmware Engineer at Micron, Singapore. He is doing his PhD at the College of Engineering Guindy, Anna University, Chennai. His research interests include Cryptography, Multimedia Security, and Secure Multimedia Retrieval.
Vetriselvi V received a PhD in Computer Science and Engineering from Anna University, Chennai, in 2008. She is a professor at the Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, Chennai. Her primary area of research is Cryptography and Network Security.
Mayank Raikwar is currently a Post-doctoral Fellow at the University of Oslo (UiO), Norway. He previously served as a researcher at the Norwegian University of Science and Technology (NTNU), where he earned his Ph.D. with a focus on Cryptography in Innovative Blockchain Services. His current research explores Directed Acyclic Graph (DAG) based distributed ledger technology (DLT) and identifies suitable cryptographic primitives to enhance security and privacy.
Reshma Balaraman received her Master's in Computer Science from the National University of Singapore in 2020. She is currently working as a research assistant at the Fintech Research Lab, National University of Singapore. Her research interests lie in Blockchain Technology, and Security in Financial Applications.

Funding
The authors did not receive support from any organization for the submitted work.

Availability of data and materials
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 7 June 2024 Accepted: 31 August 2024
Published online: 14 September 2024

References

1. Sharma H, Jain JS, Bansal P, Gupta S (2020) Feature extraction and classification of chest x-ray images using cnn to detect pneumonia. In: 2020 10th international conference on cloud computing, data science & engineering (Confluence). pp 227–231. https://doi.org/10.1109/Confluence47617.2020.9057809
2. Suraj MV, Kumar Singh N, Tomar DS (2018) Big data Analytics of cyber attacks: A review. In: 2018 IEEE International Conference on System, Computation, Automation and Networking, ICSCA 2018. pp 1–7. https://doi.org/10.1109/ICSCAN.2018.8541263
3. Gudivada VN, Raghavan VV (1995) Content based image retrieval systems. Computer 28(9):18–22. https://doi.org/10.1109/2.410145
4. Kamal ST, Hosny KM, Elgindy TM, Darwish MM, Fouda MM (2021) A new image encryption algorithm for grey and color medical images. IEEE Access 9:37855–37865. https://doi.org/10.1109/ACCESS.2021.3063237
5. Sarosh P, Parah SA, Bhat GM (2022) An efficient image encryption scheme for healthcare applications. Multimed Tools Appl 81(5):7253–7270. https://doi.org/10.1007/s11042-021-11812-0
6. Ding Y, Wu G, Chen D, Zhang N, Gong L, Cao M, Qin Z (2020) Deepedn: A deep-learning-based image encryption and decryption network for internet of medical things. IEEE Internet Things J 8(3):1504–1518. https://doi.org/10.1109/JIOT.2020.3012452
7. Priya S, Santhi B (2021) A novel visual medical image encryption for secure transmission of authenticated watermarked medical images. Mob Netw Appl 26(6):2501–2508. https://doi.org/10.1007/s11036-019-01213-x
8. Kannammal A, Subha Rani S (2014) Two level security for medical images using watermarking/encryption algorithms. Int J Imaging Syst Technol 24(1):111–120. https://doi.org/10.1002/ima.22086
9. Cai TW, Kim J, Feng DD (2008) 4 - content-based medical image retrieval. In: Feng DD (ed) Biomedical Information Technology, Biomedical Engineering. Academic Press, Burlington, pp 83–113. https://doi.org/10.1016/B978-012373583-6.50008-6
10. Das P, Neelima A (2017) An overview of approaches for content-based medical image retrieval. Int J Multimed Inf Retr 6(4):271–280. https://doi.org/10.1007/s13735-017-0135-x
11. Loukhaoukha K, Chouinard JY (2012) Berdai A (2012) A secure image encryption algorithm based on rubik's cube principle. J Electr Comput Eng 1:173931. https://doi.org/10.1155/2012/173931
12. Çavuşoğlu Ü, Kaçar S, Pehlivan I, Zengin A (2017) Secure image encryption algorithm design using a novel chaos based s-box. Chaos Solitons Fractals 95:92–101. https://doi.org/10.1016/j.chaos.2016.12.018
13. Kumari M, Gupta S, Sardana P (2017) A survey of image encryption algorithms. 3D Res 8:1–35. https://doi.org/10.1007/s13319-017-0148-5
14. Kaur M, Kumar V (2020) A Comprehensive Review on Image Encryption Techniques. Arch Comput Methods Eng 27(1):15–43. https://doi.org/10.1007/s11831-018-9298-8
15. Priyanka Singh AK (2022) A survey of image encryption for healthcare applications. Evol Intel. https://doi.org/10.1007/s12065-021-00683-x
16. Wan Y, Gu S, Du B (2020) A new image encryption algorithm based on composite chaos and hyperchaos combined with DNA coding. Entropy 22(2). https://doi.org/10.3390/e22020171
17. Paul LSJ, Gracias C, Desai A, Thanikaiselvan V, Suba Shanthini S, Rengarajan A (2022) A novel colour image encryption scheme using dynamic DNA coding, chaotic maps, and SHA-2. Multimedia Tools Appl. https://doi.org/10.1007/s11042-022-13095-5
18. Han R (2022) A Hash-Based Fast Image Encryption Algorithm. Wirel Commun Mob Comput. 2022. https://doi.org/10.1155/2022/3173995
19. Lin CH, Wu JX, Chen PY, Lai HY, Li CM, Kuo CL, Pai NS (2021) Intelligent Symmetric Cryptography with Chaotic Map and Quantum Based Key Generator for Medical Images Infosecurity. IEEE Access 9:118624–118639. https://doi.org/10.1109/ACCESS.2021.3107608
20. Dash S, Padhy S, Anjali Devi S, Sachi S, Patro KAK (2023) An efficient Intra-Inter pixel encryption scheme to secure healthcare images for an IoT environment. Expert Syst Appl 231:120622. https://doi.org/10.1016/j.eswa.2023.120622
21. Amaithi Rajan A, V V, (2023) Systematic Survey: Secure and Privacy-Preserving Big Data Analytics in Cloud. J Comput Inf Syst 1–21. https://doi.org/10.1080/08874417.2023.2176946
22. Qayyum A, Anwar SM, Awais M, Majid M (2017) Medical image retrieval using deep convolutional neural network. Neurocomputing 266:8–20. https://doi.org/10.1016/j.neucom.2017.05.025
23. Lu W, Varna AL, Swaminathan A, Wu M (2009) Secure image retrieval through feature protection. In: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, pp 1533–1536. https://doi.org/10.1109/ICASSP.2009.4959888
24. Shen M, Deng Y, Zhu L, Du X, Guizani N (2019) Privacy-preserving image retrieval for medical iot systems: A blockchain-based approach. IEEE Netw 33(5):27–33. https://doi.org/10.1109/MNET.001.1800503
25. Zhang Q, Fu M, Zhao Z, Huang Y (2023) Searchable encryption over encrypted speech retrieval scheme in cloud storage. J Inf Secur Appl 76. https://doi.org/10.1016/j.jisa.2023.103542
26. Gu Y, Zhang H, Zhang Z, Ye Q (2020) Unsupervised deep triplet hashing with pseudo triplets for scalable image retrieval. Multimed Tools Appl 79(47–48):35253–35274. https://doi.org/10.1007/s11042-019-7687-0
27. Öztürk Şaban (2021) Class-driven content-based medical image retrieval using hash codes of deep features. Biomed Signal Process Control 68:102601. https://doi.org/10.1016/j.bspc.2021.102601

28. Guan A, Liu L, Fu X, Liu L (2022) Precision medical image hash retrieval by interpretability and feature fusion. Comput Methods Prog Biomed 222. https://doi.org/10.1016/j.cmpb.2022.106945
29. Özbay E, Özbay FA (2023) Interpretable pap-smear image retrieval for cervical cancer detection with rotation invariance mask generation deep hashing. Comput Biol Med 154. https://doi.org/10.1016/j.compbiomed.2023.106574
30. Xu Y, Zhao X, Gong J (2019) A Large-Scale Secure Image Retrieval Method in Cloud Environment. IEEE Access 7:160082–160090. https://doi.org/10.1109/ACCESS.2019.2951175
31. Yan H, Chen Z, Jia C (2019) SSIR: Secure similarity image retrieval in IoT. Inf Sci 479:153–163. https://doi.org/10.1016/j.ins.2018.11.046
32. Du A, Wang L, Cheng S, Ao N (2020) A privacy-protected image retrieval scheme for fast and secure image search. Symmetry 12(2). https://doi.org/10.3390/sym12020282
33. Cheng SL, Wang LJ, Huang G, Du AY (2021) A privacy-preserving image retrieval scheme based secure kNN, DNA coding and deep hashing. Multimed Tools Appl 80(15):22733–22755. https://doi.org/10.1007/s11042-019-07753-4
34. Xia Z, Wang L, Tang J, Xiong NN, Weng J (2021) A Privacy-Preserving Image Retrieval Scheme Using Secure Local Binary Pattern in Cloud Computing. IEEE Trans Netw Sci Eng 8(1):318–330. https://doi.org/10.1109/TNSE.2020.3038218
35. Janani T, Brindha M (2022) SEcure Similar Image Matching (SESIM): An Improved Privacy Preserving Image Retrieval Protocol over Encrypted Cloud Database. IEEE Trans Multimed 24:3794–3806. https://doi.org/10.1109/TMM.2021.3107681
36. Zhu D, Zhu H, Wang X, Lu R, Feng D (2023) An Accurate and Privacy-Preserving Retrieval Scheme Over Outsourced Medical Images. IEEE Trans Serv Comput 16(2):913–926. https://doi.org/10.1109/TSC.2022.3149847
37. Tong Q, Miao Y, Chen L, Weng J, Liu X, Choo KKR, Deng RH (2022) VFIRM: Verifiable Fine-Grained Encrypted Image Retrieval in Multi-Owner Multi-User Settings. IEEE Trans Serv Comput 15(6):3606–3619. https://doi.org/10.1109/TSC.2021.3083512
38. Namasudra S (2022) A secure cryptosystem using DNA cryptography and DNA steganography for the cloud-based IoT infrastructure. Comput Electr Eng 104. https://doi.org/10.1016/j.compeleceng.2022.108426
39. Monika Upadhyaya S (2015) Secure communication using dna cryptography with secure socket layer (ssl) protocol in wireless sensor networks. Procedia Comput Sci 70:808–813. https://doi.org/10.1016/j.procs.2015.10.121
40. Amaithi Rajan A, Vetrian V, Gladys A (2023) Secure image encryption model for cloud-based healthcare storage using hyperchaos and dna encoding. In: International Conference on Computational Intelligence in Data Science. pp 89–103. https://doi.org/10.1007/978-3-031-38296-3_8
41. Zia U, McCartney M, Scotney B, Martinez J, AbuTair M, Memon J, Sajjad A (2022) Survey on image encryption techniques using chaotic maps in spatial, transform and spatiotemporal domains. Int J Inf Secur 21(4):917–935. https://doi.org/10.1007/s10207-022-00588-5
42. Mu Z, Liu H (2020) Research on digital media image encryption algorithm based on Logistic chaotic map. In: Proceedings - 2020 International Conference on Robots and Intelligent Systems, ICRIS 2020, vol 3. pp 108–111. https://doi.org/10.1109/ICRIS52159.2020.00035
43. Rakheja P, Vig R, Singh P (2020) Double image encryption using 3D Lorenz chaotic system, 2D non-separable linear canonical transform and QR decomposition. Opt Quant Electron 52(2). https://doi.org/10.1007/s11082-020-2219-8
44. Liu Z, Mao H, Wu CY, Feichtenhofer C, Darrell T, Xie S (2022) A convnet for the 2020s. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp 11966–11976. https://doi.org/10.1109/CVPR52688.2022.01167
45. Yoo CH, Kim SW, Jung JY, Ko SJ (2017) High-dimensional feature extraction using bit-plane decomposition of local binary patterns for robust face recognition. J Vis Commun Image Represent 45:11–19. https://doi.org/10.1016/j.jvcir.2017.02.009
46. Zhang Q, Han J, Ye Y (2021) Multi-image encryption algorithm based on image hash, bit-plane decomposition and dynamic DNA coding. IET Image Process 15(4):885–896. https://doi.org/10.1049/ipr2.12069
47. Khan M, Khan L, Hazzazi MM, Jamal SS, Hussain I (2022) Image encryption scheme for multi-focus images for visual sensors network. Multimed Tools Appl 81(12):16353–16370. https://doi.org/10.1007/s11042-022-12441-x
48. Kermany DS, Goldbaum M, Cai W, Valentim CC, Liang H, Baxter SL, McKeown A et al (2018) Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell 172(5):1122–1131. https://doi.org/10.1016/j.cell.2018.02.010
49. Nitha VR, Vinod Chandra SS (2023) ExtRanFS: An Automated Lung Cancer Malignancy Detection System Using Extremely Randomized Feature Selector. Diagnostics 13(13). https://doi.org/10.3390/diagnostics13132206
50. El-Latif AA, Chelloug SA, Alabdulhafith M, Hammad M (2023) Accurate Detection of Alzheimer's Disease Using Lightweight Deep Learning Model on MRI Data. Diagnostics 13(7). https://doi.org/10.3390/diagnostics13071216
51. Ferreira B, Rodrigues J, Leitão J, Domingos H (2019) Practical Privacy-Preserving Content-Based Retrieval in Cloud Image Repositories. IEEE Trans Cloud Comput 7(3):784–798. https://doi.org/10.1109/TCC.2017.2669999
52. Xiong C, Xu X, Zhang H, Zeng B (2021) An analysis of clinical values of mri, ct and x-ray in differentiating benign and malignant bone metastases. Am J Transl Res 13(6):7335
53. Brahim AH, Pacha AA, Said NH (2021) A new image encryption scheme based on a hyperchaotic system & multi specific S-boxes. Inf Secur J. https://doi.org/10.1080/19393555.2021.1943572
54. Mm Wang, Nr Zhou, Li L, Mt Xu (2022) A novel image encryption scheme based on chaotic apertured fractional Mellin transform and its filter bank. Expert Syst Appl 207. https://doi.org/10.1016/j.eswa.2022.118067

Publisher's Note