scientific reports

Check for updates

OPEN

Secure and fault tolerant cloud based framework for medical image storage and retrieval in a distributed environment

Arun Amaithi Rajan✉, Vetriselvi V, Ajitesh M & Praveen Kumar R

In the evolving field of healthcare, centralized cloud-based medical image retrieval faces challenges related to security, availability, and adversarial threats. Existing deep learning-based solutions improve retrieval but remain vulnerable to adversarial attacks and quantum threats, necessitating a shift to more secure distributed cloud solutions. This article proposes SFMedIR, a secure and fault tolerant medical image retrieval framework that contains an adversarial attack-resistant federated learning for hashcode generation, utilizing a ConvNeXt-based model to improve accuracy and generalizability. The framework integrates quantum-chaos-based encryption for security, dynamic threshold-based shadow storage for fault tolerance, and a distributed cloud architecture to mitigate single points of failure. Unlike conventional methods, this approach significantly improves security and availability in cloud-based medical image retrieval systems, providing a resilient and efficient solution for healthcare applications. The framework is validated on Brain MRI and Kidney CT datasets, achieving a 60-70% improvement in retrieval accuracy for adversarial queries and an overall 90% retrieval accuracy, outperforming existing models by 5-10%. The results demonstrate superior performance in terms of both security and retrieval efficiency, making this framework a valuable contribution to the future of secure medical image management.

Keywords Secure Medical Image Retrieval, Federated Learning, Adversarial Attack Resistance, Quantum chaos Encryption, Fault tolerant Distributed Cloud Storage

The digital transformation of healthcare has ushered in an era of unprecedented connectivity and efficiency, largely driven by the adoption of cloud-based systems. These platforms not only facilitate seamless storage and sharing of Electronic Health Records (EHRs) but also address scalability challenges posed by the rapid growth of medical data[1]. Among the various types of healthcare data, medical images such as CT scans, MRIs, and X-rays play a critical role in diagnosis, treatment planning, and monitoring[2]. The demand for efficient and secure cloud-based storage and retrieval of these images is amplified by their exponential growth, with estimates suggesting that medical image data could exceed 630 petabytes by 2030[3]. Cloud platforms like Google Cloud, Microsoft Azure, and AWS enable the storage and retrieval of such vast datasets, fostering collaboration and real-time access across healthcare institutions[4]. However, the reliance on third-party cloud systems raises critical security concerns, including data breaches, unauthorized access, and potential tampering, necessitating robust solutions to ensure data confidentiality, integrity, and availability[5–7].

Storing medical images securely in the cloud typically involves encryption techniques, which provide a foundational layer of protection. While traditional encryption schemes[8–10] address confidentiality, they often fall short of ensuring data integrity and resistance against emerging threats. Furthermore, as we move toward the quantum era, the limitations of classical encryption methods become more pronounced. Quantum-capable adversaries pose a significant risk to these traditional approaches[11]. Current encryption methods are not designed to prevent intelligent tampering with encrypted data, which could lead to compromised diagnostic accuracy. Moreover, secure storage alone is insufficient; the ability to retrieve medical images accurately and efficiently from encrypted datasets without exposing sensitive information is equally critical[12].

Content-Based Medical Image Retrieval (CBMIR) is pivotal for querying relevant medical images from databases using visual content, facilitating accurate diagnoses and effective treatments[13]. One approach to improving CBMIR efficiency is the use of similarity-preserving hashcodes, which enable rapid indexing and

Security Research Lab, Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, Chennai 600025, India. ✉email: 22144191119@student.annauniv.edu

retrieval. Models like Deep Pairwise Hashing (DPH)[14] and Improved Deep Hash Network (IDHN)[15] generate these hashcodes efficiently, but their unencrypted nature exposes them to adversarial attacks, particularly pattern-based intrusions[16]. Moreover, CBMIR faces significant challenges, including slow search times, difficulty achieving precise retrieval results, and susceptibility to malicious manipulations. These issues can distort retrieval outcomes, compromise reliability, and erode trust in the system. Ensuring accuracy and security in CBMIR is vital to unlocking its full potential for modern healthcare applications. Encrypting hashcodes can mitigate this vulnerability, yet it often comes at the cost of retrieval performance, creating a trade-off between security and efficiency[17]. Another challenge arises in the centralized training of hashcode generation models, where aggregating vast medical image datasets from distributed healthcare providers poses privacy and logistical concerns. Federated Learning (FL) offers a viable solution, allowing institutions to collaboratively train models without centralizing sensitive data. FL not only ensures data privacy but also enhances the robustness of hashcode generation, making it suitable for secure, distributed CBMIR systems[18].

Despite the advances in hashcode generation and federated learning, ensuring fault tolerant storage and retrieval of medical images remains a concern. Distributed cloud architectures employing master-slave models offer a path toward resilient systems, but traditional secret-sharing schemes used for fault tolerance are prone to pattern or access-based attacks[19–21]. Malicious actors can exploit shared data to infer sensitive information, compromising the system's robustness. Introducing randomness and dynamic thresholding mechanisms into secret-sharing schemes can mitigate these vulnerabilities, paving the way for a secure and fault tolerant framework. The overview of the cloud-based storage and retrieval framework in the distributed healthcare environment is shown in Fig. 1.

Existing solutions primarily address individual challenges such as image security, fault tolerance, or retrieval accuracy, but a unified system that effectively integrates all these aspects is lacking, as discussed above. To address these multifaceted challenges, SFMedIR, a Secure and Fault tolerant cloud-based framework for Medical Image Retrieval in a distributed environment, is proposed. The contributions of this work are as follows:

- A novel cloud-based framework, SFMedIR, for secure and fault tolerant medical image retrieval in a distributed environment is proposed.
- Quantum-chaos-based image encryption is employed to ensure robust security for medical images against advanced threats.
- To ensure secure and accurate retrieval, Federated Learning is utilized to generate context-aware, similarity-preserving hashcodes that are resistant to adversarial attacks.
- A dynamic threshold-based shadow generation scheme is proposed to enhance security and fault tolerance during the retrieval process. A formal security analysis is conducted to validate the framework.
- SFMedIR is evaluated using Mean Average Precision (mAP), latency, throughput, fault-recovery time for retrieval performance, and formal analysis for security and retrieval efficiency. Experiments on Brain MRI and Kidney CT datasets show a 60-70% retrieval accuracy improvement under adversarial conditions.

The structure of the paper is as follows: Section "Related work" reviews related works, highlighting existing approaches and their limitations. Section "System architecture" elaborates on the proposed SFMedIR framework, explaining its design and components. Section "Formal analysis and verification" focuses on the formal analysis of the framework from a security and retrieval accuracy perspective, while Section "Experimental results and performance analysis" discusses the experimental evaluation and performance results. Finally, Section Conclusion and future work concludes the study with key findings.

Related work

In this section, the authors detail an overview of existing secure image retrieval systems and discuss their issues. Secure and privacy-preserving image retrieval ensures efficient searches in encrypted databases without compromising performance[22,23]. Approaches in this domain are broadly classified into two categories. The first involves generating secure indexes from image features, encrypting the images, and storing them in the cloud.

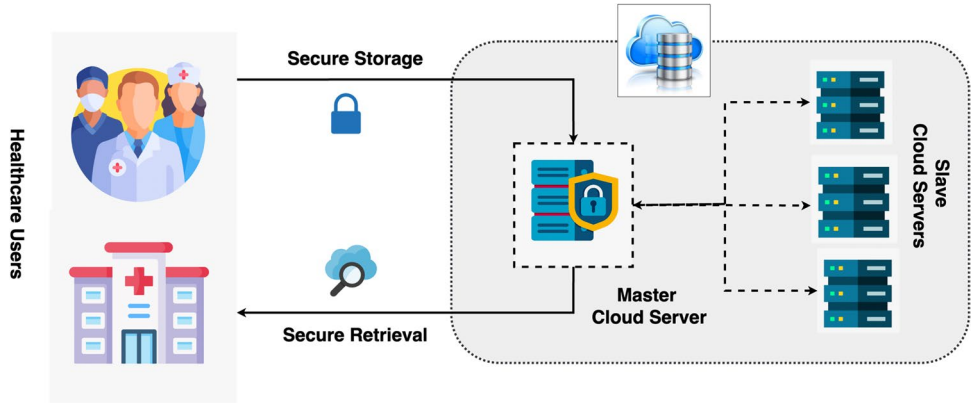


Fig. 1. System model of secure medical image storage and retrieval framework.

The second leverages the cloud for feature extraction and secure index generation, often using deep hashing to create similarity-preserving hashcodes[24,25]. Xu et al.[26] introduced a cloud-based system combining Hamming embedding and Min-hash for enhanced accuracy. Du et al.[27] employed deep hashing with Secure k-NN and DNA-based chaotic encryption, later enhancing accuracy using a 4D hyperchaotic map[28]. Janani et al.[29] designed a multiparty-based similarity matching, while Zhu et al.[30] introduced Privacy-preserving Mahalanobis Distance Comparison (PMDC) for enhanced privacy.

Similarity-preserving hashcodes are vulnerable to both targeted and non-targeted adversarial attacks, posing significant risks in the healthcare domain due to the critical nature of medical data. Ma et al.[31] analyzed the impact of adversarial attacks on deep learning-based medical image analysis systems, emphasizing the severe implications such attacks could have on diagnostic accuracy. To address such vulnerabilities, Yuan et al.[32] proposed semantic-aware hashcode generation for image retrieval. Their approach fabricates adversarial examples by maximizing the Hamming distance between the hashcodes of adversarial samples and primary features, demonstrating its efficacy in adversarial attack trials. However, these methods rely on centralized training for hashcode generation, which limits their scalability and privacy. Tabatabaei et al.[33] advanced the field by introducing federated learning (FL)-based medical image retrieval system for global applications. FL-based training enhances privacy by ensuring that data remains decentralized during training, making the model inherently more robust[34]. Despite this innovation, there remains no FL-based adversarial-attack-resistant hashcode generation model capable of addressing multiple challenges with a unified solution[35].

Medical images are frequently stored in centralized cloud infrastructures, which are prone to single points of failure. In healthcare, where retrieval time is critical, such centralized systems can be a bottleneck. Distributed cloud solutions offer an alternative. Ajitesh et al.[36] proposed a model utilizing trusted edge computing for secure processing and distributed cloud storage for remote sensing images. Their approach involves sharing and storing images in slave servers across the cloud. However, plain image storage increases exposure to threats, necessitating the use of encryption and fragmentation. Zhou et al.[37] introduced shadow generation techniques, employing a threshold-based system where images are divided into n shares, each stored on a separate server. During retrieval, only a subset of these shares is required to reconstruct the image. While effective, these techniques remain susceptible to access-based attacks[3839].

The existing literature highlights the pressing need for a federated learning-based, context-aware hashcode generation model that ensures privacy and resilience against adversarial attacks. Additionally, to address access-based vulnerabilities and enhance fault tolerance, there is a clear demand for a distributed and dynamically fragmented image storage system in the cloud. These insights have guided the development of the proposed system, "SFMedIR," which is specifically designed to meet the stringent requirements of healthcare applications. Table 1 outlines the distinctions between our proposed system and existing retrieval methods.

System architecture

Problem formulation

The proposed system has Trusted Medical Image Owners (*MO*), Master and Slave Cloud Servers (*MCS, SCS*), and Medical Image Users (*MU*), as illustrated in Fig. 2. *MO* possess a collection of $\mathbb{N}$ medical images $\mathbb{MI} = \{MI_1, MI_2, ..., MI_{\mathbb{N}}\}$ which has to be offloaded to the cloud storage after encrypting the images. *MCS* takes care of *s* dynamic shadows generation and metadata storage. *MU* are able to retrieve most similar images by requesting to the cloud by sending the query image MI_q. These cloud servers provide storage and retrieval services. As cloud servers are honest and curious, the challenge lies in the identification of k most similar images from encrypted images to a specified query image MI_q while preserving security and ensuring availability. The overall framework design is depicted in Fig. 2. Table 2 lists the notations used with a description.

System model and framework design

Secure and Fault tolerant Cloud-based Medical Image Storage and Retrieval Framework (SFMedIR) in a distributed environment is proposed to achieve the following goals.

- **Attack resistance:** The hashing model must be resilient to adversarial attacks, ensuring robustness as medical images play a critical role in digital healthcare.
- **Image security:** Medical images should be encrypted and shared in a manner that prevents attackers from extracting any meaningful information.
- **Image availability:** The framework should guarantee image availability even in the event of some server failures, ensuring reliable retrieval.

		Hashcode Generation			Storage and Retrieval			
Reference	Medical	Backbone	Secure Training	FL-based	Distributed	Encrypted	Threshold Sharing	Dynamic
Du et al.[27]	No	ResNet	×	×	×	✓	×	×
Ozbay et al.[25]	Yes	DenseNet	×	×	×	×	×	×
Yuan et al.[32]	No	CNN	✓	×	×	×	×	×
Tabatabaei et al.[33]	Yes	CNN	×	✓	×	×	×	×
Ajitesh et al.[36]	No	MobileNet	×	×	✓	✓	✓	×
SFMedIR (Ours)	Yes	ConvNeXt	✓	✓	✓	✓	✓	✓

Table 1. Comparison of secure medical image retrieval approaches.

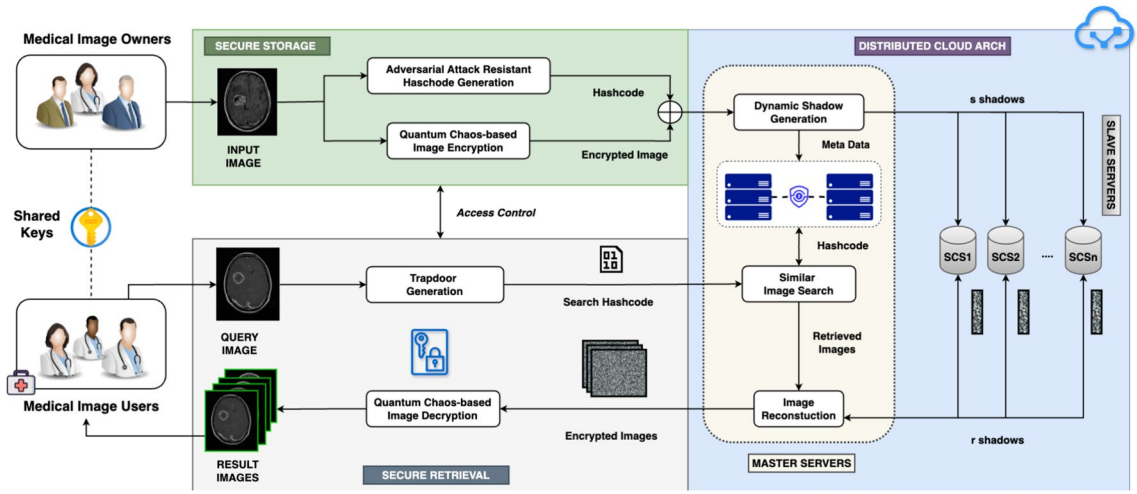


Fig. 2. Proposed SFMedIR framework.

Notation	Description
$\mathbb{MI} = \{MI_i\}_{i=1}^{\mathbb{N}}$	Medical Images
$\mathbb{EI} = \{EI_i\}_{i=1}^{\mathbb{N}}$	Encrypted Medical Images
$\mathbb{K} = \{K_{QIE}, K_{rEK}\}$	Keys in the system
K_{QIE}	Symmetric Key for Image Encryption and Decryption
SCS_{max}	Maximum slave cloud servers
s	Number of total shadows
r	Number of shadows required to reconstruct the image
K_{rEK}	Threshold value *r* Encryption Key
Sh	Shadow of an image
$\mathbb{AH} = \{AH_i\}_{i=1}^{\mathbb{N}}$	Adversarial Attack Resistant Hashcodes of $\mathbb{N}$ Medical Images each of length *l*
MI_q	Query Medical Image
H_q	Hashcode of MI_q
$\mathbb{ER} = \{ER_i\}_{i=1}^{k}$	Top-k Resulted Encrypted Medical Images
$\mathbb{OR} = \{OR_i\}_{i=1}^{k}$	Top-k Resulted Decrypted Medical Images

Table 2. Notations used.

The proposed framework has different entities that work together to ensure secure and fault tolerant retrieval work. Figure 2 shows the proposed SFMedIR architecture. It has two major phases. Storing the medical image in a secure way is the first phase. Here, the *MO* processes the medical image *MI* and offloads them to the cloud. *MO* encrypt the *MI* using quantum chaos-based image encryption scheme QMedShield proposed by Amaithi Rajan et al.[40] and produces *EI*. From the *MI*, adversarial attack-resistant hashcode *AH* is also being derived using the trained model. This hashcode acts as an index while storing and searching. *EI* and corresponding *AH* is sent to the *MCS*. Here, the *EI* is split into dynamic (*r*, *s*) shadows. Where *s* shadows are stored in slave servers, *r* shadows are required to reconstruct the original encrypted image. Each shadow *Sh* is stored in slave cloud servers. In the second phase, *MU* sends the query hashcode H_q, which is derived from the MI_q to the *MCS*. A similar image search is executed, and top-k image indices are selected. For each index in the result set, the original encrypted image has to be constructed with *r* shadows retrieved from the slave cloud servers. After reconstruction, *MIU* receives the top-k result images from the *MCS*, and it decrypts the result images.

Framework design

The framework of the proposed system is outlined in this subsection, with the functionalities of each entity and the corresponding algorithms explained. Key Control Entity *KCE* handles KeyGen algorithm. *MO* runs QChaosImgEnc, CxtHashGen algorithms. *MCS* executes DynamicShadowsGen during secure storage phase and SimImgSearch, ImgReconstruct during the retrieval phase. *MU* utilizes TrapdoorGen, QChaosImgDec algorithms.

1. $\mathbb{K} \leftarrow \text{KeyGen}(1^{\lambda})$: This algorithm takes λ parameter as input and outputs the key set $\mathbb{K} = \{K_{QIE}, K_{rEK}\}$.
2. $\mathbb{EI} \leftarrow \text{QChaosImgEnc}(\mathbb{MI}, K_{QIE})$: The quantum chaos-based image encryption algorithm (QMedShield) takes medical images $\mathbb{MI}$, and encryption key K_{QIE} as input, and outputs encrypted medical images $\mathbb{EI}$.
3. $\mathbb{AH} \leftarrow \text{CxtHashGen}(\mathbb{MI})$: The context-aware hashcode generation algorithm takes input as medical images $\mathbb{M}$ and return the adversarial-attack resistant hashcodes $\mathbb{AH}$ from the efficiently FL-based trained model.
4. $\{Sh_i\}_{i=1}^{s} \leftarrow \text{DynamicShadowsGen}(EI_j)$: For each image EI_j in $\mathbb{EI}$, this algorithm chooses dynamic (*r*, *s*) where $r < s < SCS_{max}$ and returns *s* shadows $\{Sh\}_{i=1}^{s}$. The *s* shadows are sent to *s* slave cloud servers. For each EI_j, the associated AH_j is securely stored in *MCS* along with metadata of shadows and encrypted *r* using K_{rEK}.
5. $H_q \leftarrow \text{TrapdoorGen}(MI_q)$: The trapdoor generation algorithm takes a query image MI_q and outputs a searchable trapdoor H_q, which will be sent to *MCS* for search.
6. $\{AH_i\}_{i=1}^{k} \leftarrow \text{SimImgSearch}(H_q, \mathbb{AH})$: The similar image search algorithm, for a given query H_q returns relevant top-k indices of images.
7. $\mathbb{ER} \leftarrow \text{ImgReconstruct}(\{AH_i\}_{i=1}^{k})$: For each image index, *MCS* retrieves required *r* shadows out of *s* from the *SCS* for reconstructing *EI*. Return the $\mathbb{ER}$ to *MU*
8. $\mathbb{OR} \leftarrow \text{QChaosImgDec}(\mathbb{ER}, K_{QIE})$: The QMedShield algorithm takes top-k encrypted medical images $\mathbb{ER}$, and decryption key K_{QIE} as inputs and outputs original medical images set $\mathbb{OR}$.

This section summarizes the problem formulation, overall framework, and detailed design. The following subsections provide the internal details for each function.

Secure storage of medical images

The architecture of the proposed system is illustrated in Fig. 2. It operates in two primary phases: secure storage and fault tolerant retrieval of encrypted medical images within a distributed environment. This subsection provides a detailed explanation of the modules involved in each phase. During the secure storage phase, the *MO* encrypts medical images, generates adversarial attack-resistant context-aware hashcodes, and uploads the encrypted data to the cloud. The *MCS* then creates dynamic shadows of the encrypted images and distributes them across slave servers. This phase includes five key functions: KeyGen, QChaosImgEnc, CxtHashGen, and DynamicShadowsGen.

The KeyGen module generates the key set $\mathbb{K}$ by taking λ as input. It produces the symmetric image encryption key (K_{QIE}) and the *r* encryption key (K_{rEK}). These secret keys are securely transmitted by the *KCE* to the *MU*, *MCS*, and *MO* through a secure channel, enabling the *MO* and *MCS* to handle encryption processes while allowing the *MU* to perform decryption.

Quantum chaos-based image encryption model

Medical images must be stored securely to avoid information leakage and modification. In this image encryption (QChaosImgEnc) module, a quantum-chaos-based algorithm (QMedShield) is used. This algorithm is a hybrid, where the traditional images are encrypted with quantum-chaotic maps and some quantum operations involved without converting the image into quantum representation. This makes the image encryption model effective and quantum-secure in traditional computing environments with resource efficiency. The image owner encrypts the image before uploading it to the cloud. The flow of the encryption is shown in the following Fig. 3.

The model employs bit-plane scrambling, a 3D quantum logistic map, quantum operations during the diffusion phase, a hybrid chaotic map, and DNA encoding in the confusion phase to convert the plain medical image into a ciphered form. This encryption technique is robust against various potential attacks. The encrypted medical images are subsequently outsourced to the cloud for secure storage. The process of context-aware hashcode generation is further elaborated in the following submodules.

Context-aware adversarial training

This section explains the FL-based Context-aware Adversarial Training (FCAT) in detail. The produced model is robust and attack-resistant. In deep hashing-based retrieval, the objective of a non-targeted attack is to generate an adversarial input x^* from a benign query x with label *lb*, aiming to mislead the hashing model H into retrieving irrelevant results for x. In contrast, a targeted attack manipulates x^* to deceive the model into returning results associated with a specific target label lb_t. Furthermore, the perturbation $\Delta x = x^* - x$ must remain minimal to ensure that the changes are imperceptible to human observation. Adversarial training in deep hashing, analogous to its use in classification, employs both benign inputs $\{(x_i, l_i)\}_{i=1}^{N}$ and their adversarial variants $\{(x_i^*, l_i)\}_{i=1}^{N}$ to refine the parameters θ of the hashing model H. This optimization ensures that the model retrieves semantically relevant content corresponding to the original label lb_i, whether the input is a

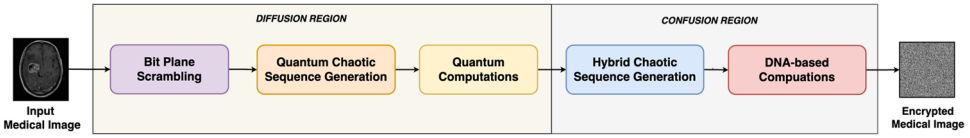


Fig. 3. QMedShield encryption flow.

clean query x_i or its adversarially perturbed counterpart x_i^*. So, this training makes the hashing model produce adversarial attack-resistant hashcodes which are context-aware.

In the context of secure medical image retrieval, adversarial training alone is insufficient to address all critical challenges. To enhance the model's robustness through access to diverse and extensive datasets, ensure decentralized learning, and uphold privacy (particularly vital in healthcare scenarios), federated learning (FL) is integrated into the framework. FL enables multiple healthcare centres to collaboratively train the model without sharing sensitive data, maintaining privacy while leveraging adversarial training to further improve security. This combination of FL and adversarial training ensures a resilient, privacy-preserving, and decentralized system tailored for secure and effective medical image retrieval. The proposed FL design is shown in Fig. 4.

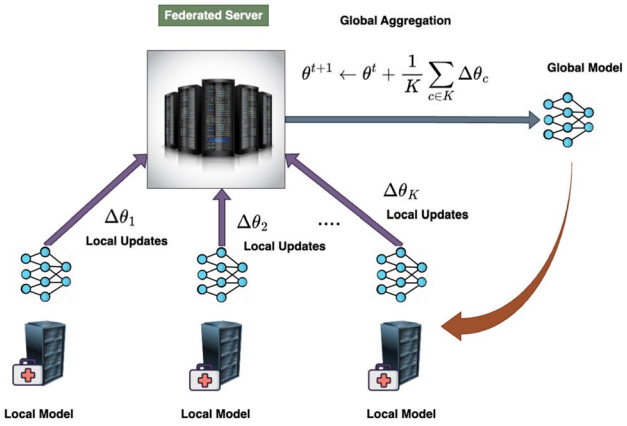


Fig. 4. FL-based training design.

Input: Global Model Parameters θ^t, Total Number of Clients K, Participating Client Fraction F, Local Epochs T_{local}, Global Rounds T_{global}, Learning Rate ζ
Output: Robust Global Model Parameters $\theta^{T_{\text{global}}}$
Initialization: Initialize global model θ^0

Step 1: Initialize Global Training
Server initializes θ^0 and sends it to all clients
for each global round $t = 1$ to T_{global} **do**
  Step 2: Client Selection
  Server selects a random subset $\mathcal{S}_t \subseteq K$ with $|\mathcal{S}_t| = F \cdot K$ clients
  Step 3: Local Training at Each Client $c \in \mathcal{S}_t$
  for each client $c \in \mathcal{S}_t$ **do**
    Client receives global model θ^t and performs the following:
    for each local epoch $e = 1$ to T_{local} **do**
      Step 3.1: Generate Context-aware Codes
      for each sample x_i in client c's dataset $\mathcal{D}_c$ **do**
        Context-aware code $M_{ca} \leftarrow \text{sign}\left(\sum_{i \in Pos} w_i b_i^{(Pos)} - \sum_{j \in Neg} w_j b_j^{(Neg)}\right)$
      end for
      Step 3.2: Generate Adversarial Samples
      for each sample x_i in $\mathcal{D}_c$ **do**
        $x_0^* = original\ x_i$
        Generate x_i^* using PGD: $x_P^* \leftarrow S_\epsilon\left(x_{P-1}^* + \eta \cdot \text{sign}\left(\nabla_{x_{P-1}^*} L_{\text{adv}}\right)\right)$
        $x_i^* = x_P^*$
      end for
      Step 3.3: Update Local Model
      Compute total loss: $L_{total\_loss} = \lambda L_{adl} + \phi L_{quant} + \chi L_{bit} + L_{ori}$
      Update local model $\theta_c \leftarrow \theta_c - \zeta \nabla_{\theta_c} L_{\text{total\_loss}}$
    end for
    Client sends model update $\Delta\theta_c = \theta_c - \theta^t$ to the server
  end for
  Step 4: Server Aggregation
  Server aggregates updates from all clients: $\theta^{t+1} \leftarrow \theta^t + \frac{1}{|\mathcal{S}_t|} \sum_{c \in \mathcal{S}_t} \Delta\theta_c$
end for
Step 5: Return Final Model
Server outputs $\theta^{T_{\text{global}}}$ as the robust global model

Algorithm 1. FL-based semantic-aware adversarial training algorithm.

The hashing model employed the ConvNeXt model as a backbone for hashcode generation. ConvNeXt models extract more efficient features than other ConvNets. The layered architecture of this hashcode generation network is shown in Fig. 5. Each level employs distinct convolution strides to effectively extract deep features. The ConvNeXt block incorporates GELU activation in place of ReLU, Layer Normalization (LN) instead of Batch Normalization (BN), and utilizes an Inverted Bottleneck architecture. Drawing inspiration from vision transformers, this module is designed to capture both local and global features. Finally, the fully connected layer brings the features extracted for image MI_i and gets the feature vector $FV_i = \{f_1, f_2, ...f_l\}$. The features are converted into hashcode $AH_i = \{h_1, h_2, ...h_l\}$ of length l. This model uses the adversarial learning concept to generate similarity-preserving attack-resistant discriminant hashcodes. In this work, targeted attacks are concentrated as they have significant side effects in healthcare. For efficient hashcode generation, context-aware

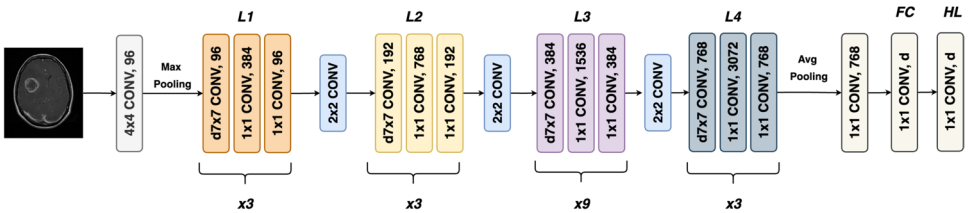


Fig. 5. ConvNeXt-based deep hashing model with adversarial learning.

codes (M_{ca}) are generated for each class and utilised for adversarial loss calculation[32]. The objective is that, for the given benign sample x_t and target label lb_t, we have to acquire M_{ca} of lb_t and then the objective function is defined as in Eq. (1).

$$arg\ max_{x^*}(\frac{1}{l}tanh(\alpha f_\theta(x^*)) \tag{1}$$

As the hash code of the adversarial example x^* converges toward the primary code of the target label, the adversarial example increasingly aligns with the target label in semantic meaning while maintaining visual imperceptibility. Consequently, feeding x^* into a deep hashing-based retrieval system enables the retrieval of semantically relevant content associated with the target label. The adversarial samples are generated using the PGD technique[41].

$$x_P^* \leftarrow S_\epsilon\left(x_{(P-1)}^* + \eta \cdot \text{sign}\left(\nabla_{x_{P-1}^*} L_{\text{adl}}\right)\right), x_0^* = x \tag{2}$$

In Eq. (2), P is generally 100 iterations by default. η is step size, S_ϵ projects $x*$ into $\epsilon-$ball of x. The adversarial loss L_{adl} is calculated as shown in Eq. (3),

$$L_{adl}(x^*, M_{ca}; \theta) = \frac{1}{l}M_{ca}(tanh(f_\theta(x^*)) \tag{3}$$

To facilitate the back-propagation algorithm during training, the sign function is substituted with the tanh function to produce approximate continuous hash codes. This replacement results in quantization errors, which are mitigated by adding a quantization loss to lessen the difference between the approximate hash codes and the binary codes of the adversarial examples.

$$L_{\text{quant}}(x^*; \theta) = \|\tanh(f_\theta(x^*)) - \text{sign}(f_\theta(x^*))\|_2^2 \tag{4}$$

In Eq. (4), $||.||_2$ is the L_2 norm. x^* represents the adversarial example, θ denotes the model parameters, $f_\theta(x^*)$ is the feature representation generated by the model, and the loss minimizes the difference between the continuous approximation $\tanh(f_\theta(x^*))$ and the binary representation $\text{sign}(f_\theta(x^*))$. In addition to these two losses, we also include the bit balance loss (L_{bit}) to compute the hashcode efficiently. This indicates that each hashcode has a 50% likelihood of falling between 0 and 1. To create more unique hashcodes, one can utilize the target function outlined in Eq. (5) for generating d-bit hashcodes. In this context, h_i refers to the output of the hash layer from the i^{th} node.

$$L_{bit} = \frac{1}{l}\sum_{i=1}^{l} h_i \tag{5}$$

Added to this, the loss generated from the original ConvNeXt model L_{ori} is added, which is the difference between the hashcode generated from FACT and the original one. Finally, the cumulative loss function for the discriminant hashcode generation is

$$L_{total\_loss} = \lambda L_{adl} + \phi L_{quant} + \chi L_{bit} + L_{ori} \tag{6}$$

All losses are combined with the aim of reducing the total loss. The hyper-parameters λ, ϕ, and χ serve as trade-offs to regulate these losses. By transmitting this error through each hash generation network, effective hashcodes are produced.

Centralized training faces challenges regarding privacy and model robustness. In cloud-based systems, both aspects are crucial. To address these concerns, we propose a model that employs FL-based context-aware adversarial training. Initially, a global model is set up, which is then refined with local training sessions after every T_{local} epochs. The architecture of the FL model is illustrated in Fig. 4, and the process is explained in Algorithm 1. The trained model is shared with all health care centres, which are defined as CxtHashGen, which takes a medical image and returns a context-aware hashcode. Both the encrypted image and this hashcode are sent to *MCS* for storage.

Dynamic shadows generation

Input: Secret image EI of size $m \times m$, Maximum number of servers $SCS_{\max}$, Prime modulus p

Output: s shadow images $\{Sh_i\}_{i=1}^{s}$

1: Randomly select r and s such that $r \leq s < SCS_{\max}$
2: Partition EI into non-overlapping $m \times m$ blocks S
3: **for** each block S **do**
4: Generate a random $m \times r$ matrix A of rank r
5: Compute the projection matrix $S_{\text{proj}} = (A(A^{\top}A)^{-1}A^{\top}) \mod p$
6: Compute the remainder matrix $Rd = S - S_{\text{proj}} \mod p$
7: **if** any element of S_{proj} or Rd exceeds p **then**
8: Re-generate A and recompute S_{proj}, Rd
9: **end if**
10: Choose s linearly independent $r \times 1$ random vectors $\{z_i\}_{i=1}^{s}$ and compute shadows:

$$v_i = (A \cdot z_i) \mod p, \quad \forall i = 1, \ldots, s$$

11: **end for**
12: Return $\{Sh_i\}_{i=1}^{s}, Rd$ and store (r, s) for reconstruction

Algorithm 2. DynamicShadowsGen(EI) .

Once the hashcode and corresponding encrypted image are sent to the master cloud server. The *EI* is split into dynamic (*r*, *s*) shadows. The method proposed by Bai et al.[42]. Usually, in the threshold-based secret sharing scheme, all images are split into *s* shadows where without *r* shadows, the image cannot be reconstructed. But from the attacker's perspective, if all images are split in the same shadows, he will try to guess that *r* shadows. To overcome this issue, the split can be done dynamically with respect to the SCS_{max}. So that the attacker cannot find the shadow count. In this module, the encrypted image *EI* is split into dynamic (*r*, *s*) shadows. Algorithm 2 details the flow.

This algorithm returns the *s* shadows $\{Sh_i\}_{i=1}^{s}$. These shadows are sent to the *s* slave cloud servers. *r* is encrypted using K_{rEK}. In the *MCS*, *AH*, *s* slave locations, encrypted *r* are stored as metadata. From this information, the attacker cannot retrieve any relevant information about the secret image.

Secure image retrieval

In the secure image retrieval phase, the trusted medical image user *MU* generates the search trapdoor and forwards it to the cloud for similar image retrieval. *MU* have access to the FCAT model (Section Context-aware adversarial training). The query medical image MI_q is sent to that model and gets the hashcode H_q. This hashcode is sent to the *MCS*. A similar image search is done with the received hashcode over the indexes stored in the cloud. Hamming distance is used to find the distance between the query image and the medical images in the database. Top-k results are chosen $\{AH_i\}_{i=1}^{k}$. These indexes only have the metadata and not the corresponding encrypted images. For each index, the corresponding *r* shadows have to be retrieved, and *EI* has to be reconstructed.

This image retrieval ensures security and fault tolerance. If some slave servers are unavailable, the encrypted image can still be reconstructed with *r* shadows. The master is also replicated, so availability is always ensured for image retrieval. From a security standpoint, the images are encrypted and dynamically shared within the servers, which improves their randomness. The upcoming subsections define the encrypted image reconstruction and image decryption.

Encrypted image reconstruction

The encrypted image *EI* is reconstructed from the *r* out of *s* shadows generated. Once the top-k results are retrieved, for each index, the metadata is checked. The metadata has *s* shared locations, but *r* is encrypted. *r* is decrypted using K_{rKE}. *r* shares are fetched from the slave servers. The encrypted image *EI* is constructed as described in Algorithm 3

Input: r shadow images $\{Sh_i\}_{i=1}^{r}$, Prime modulus p
Output: Reconstructed image EI

1: Extract $\{v_i\}_{i=1}^{r}$ from $\{Sh_i\}_{i=1}^{r}$
2: Construct a matrix $B = [v_1, v_2, \ldots, v_r]$
3: Compute the projection matrix:

$$S_{\text{proj}} = (B(B^{\top}B)^{-1}B^{\top}) \mod p$$

4: Verify the projection matrix trace $\text{tr}(S_{\text{proj}}) = r$
5: Reconstruct the secret block S:

$$S = (S_{\text{proj}} + Rd) \mod p$$

6: Recombine all blocks S to reconstruct the encrypted image EI

Algorithm 3. ImgReconstruct().

Quantum chaos-based image decryption
The Master Cloud exclusively returns the retrieved encrypted images, denoted as $\mathbb{ER}$, to the Medical Unit (MU). Users possess the decryption key, represented as K_{QIE}, to decrypt these retrieved images. The decryption process operates as the inverse of the encryption procedure, as illustrated in Fig. 3. Consequently, the potential for information leakage pertaining to medical image data is effectively mitigated, rendering it resistant to quantum attacks. The comprehensive end-to-end retrieval process is depicted in Fig. 6.

Security and privacy model

In the proposed system, there is an assumption that *SCS* which are "honest and curious" and malicious users. The system has to ensure that the ciphertext does not leak any critical information to them. The following security definitions are defined to achieve security.

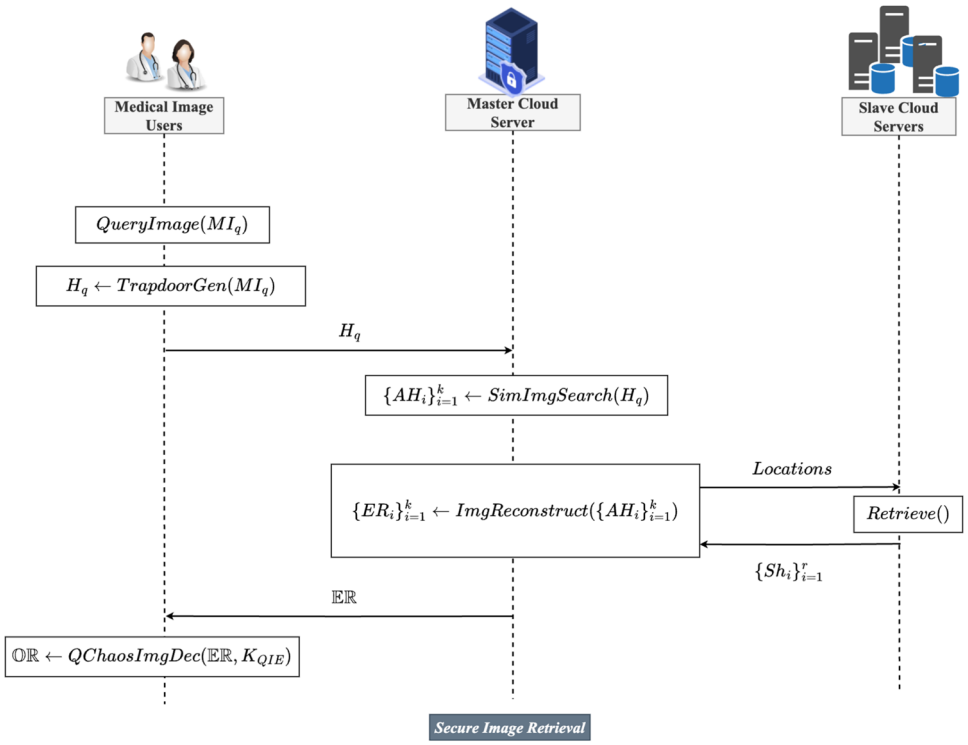


Fig. 6. End-to-End flow diagram.

Definition 1 *(Data Privacy in Federated Learning):* Federated Learning (FL) satisfies data privacy if, under any adversarial strategy, the probability of extracting individual clients' raw data from the shared updates is negligible.

Definition 2 *(Shadow Parameter Confidentiality):* No polynomial-time adversary can determine the reconstruction threshold r and total number of shares s with non-negligible probability when these parameters are chosen dynamically and randomly.

Definition 3 *(Reconstruction Resistance):* Let $\mathcal{P} = \{(r, s) : 1 \leq r < s \leq SCS_{max}\}$ represent the set of possible shadow configurations. If an adversary lacks knowledge of the distribution $\mathcal{P}$, their probability of correctly guessing r and selecting r valid shares out of s is negligible.

The theorems and proofs provided in the subsequent section substantiate all these claims. The security analysis section of this article establishes the validity of these claims through rigorous theoretical arguments and formal theorems. Furthermore, the proposed method is evaluated against existing cloud-based solutions from a security perspective. The following table 3 presents a comparative analysis, highlighting how the SFMedIR framework outperforms others in terms of security.

Formal analysis and verification

This section is dedicated to analyzing the security of the proposed image hashcode generation model and dynamic shadows generation model. Additionally, formal verification of retrieval efficiency under adversarial condition is also provided.

Data privacy analysis

Theorem 1 *In Federated Learning, where only model updates such as gradients are exchanged, the adversary's ability to reconstruct clients' raw data is limited to an approximation with negligible accuracy, provided that the model training gradients exhibit sufficient complexity and aggregation.*

Proof Let $\mathcal{D}_c$ represent the dataset of client c. Let g_c denote the local gradient update of client c, computed as:

$$g_c = \nabla \mathcal{L}(\mathcal{D}_c, \theta), \tag{7}$$

where $\mathcal{L}$ is the local loss function, and θ is the current global model. The central server aggregates the updates:

$$G = \frac{1}{K} \sum_{c=1}^{K} g_c, \tag{8}$$

where K is the number of clients. The adversary $\mathcal{A}$ can observe the aggregated gradients G and attempt to infer $\mathcal{D}_c$ from g_c or G.

Case 1: No Access to Individual Gradients If the adversary only observes the aggregated gradient G, recovering $\mathcal{D}_c$ is equivalent to solving the following underdetermined system shown in Eq. (8). This system has infinitely many solutions unless $K = 1$ (only one client). Thus, without additional information, $\mathcal{A}$ cannot uniquely determine g_c or $\mathcal{D}_c$.

Case 2: Access to Individual Gradients If the adversary can access g_c, the reconstruction of $\mathcal{D}_c$ depends on the invertibility of $\nabla \mathcal{L}$. For most machine learning loss functions, the gradient is a highly non-linear function of $\mathcal{D}_c$, making inversion computationally intractable and dependent on the model parameters θ, which abstract raw data into a compressed representation.

For a well-optimized model, gradients g_c are locally optimal, satisfying: $g_c \approx 0 \quad (\text{converged case})$. In such cases, gradients provide no additional information about $\mathcal{D}_c$, further reducing leakage potential. □

Shadow security analysis

Theorem 2 *If a single shadow Sh_i is compromised, the probability of reconstructing or inferring the secret image EI is negligible, provided that the shadow generation uses randomized r-out-of-s parameters and r is greater than 1.*

Feature	Traditional Cloud Storage	Blockchain-Based Solutions	SFMedIR
Confidentiality	Symmetric/Asymmetric encryption	Zero-Knowledge Proofs	Quantum-Chaos-Based Encryption
Integrity	Crypto Hash	Blockchain ledger	Federated Learning + Dynamic Shadows
Availability	Centralized storage (Single point of failure)	Distributed storage (Consensus overhead)	Distributed Cloud with Faul Tolerant dynamic (r, s) Sharing
Adversarial Attack Resistance	NA	Limited defense mechanisms	Adversarial Attack Resistant Hashcode Generation

Table 3. Comparative analysis of security features..

Proof Let the secret image EI be split into s shadows using the shadow generation scheme with a threshold r. Each shadow Sh_i consists of $v_i = (A \cdot z_i) \mod p$, the projection of A. Both A and Rd are generated using independent randomness.

A malicious server holding Sh_j only has access to the single shadow $Sh_j = [v_j]$ and the parameters p, but not (r, s). The value $v_j = (A \cdot z_j) \mod p$ is derived from a random matrix A and a random vector z_j. Since A is of rank r and $r > 1$, v_j is indistinguishable from a random vector over $\mathbb{Z}_p$. Without access to at least r linearly independent vectors $\{v_1, \ldots, v_r\}$, the adversary cannot recover A or reconstruct any part of S_{proj}. The adversary's advantage in reconstructing EI or distinguishing EI_0 and EI_1 from a single shadow is bounded by:

$$\mathcal{ADV}(\mathcal{A}) \leq \frac{1}{p^r} \tag{9}$$

For large p and $r > 1$, this is negligible.

A single shadow Sh_j held by a malicious server provides negligible information about the secret image EI. The shadow generation mechanism ensures that reconstructing EI requires at least r shadows, maintaining the security of the scheme against single-server compromise. □

Reconstruction resistance analysis

Theorem 3 *Let a secret image be split into (r, s) shadows using a secret sharing scheme with randomized parameters (r, s) for each image. If the attacker does not know the distribution $\mathcal{P}$, their probability of correctly predicting the reconstruction threshold r and selecting r valid shares out of s is negligible.*

Proof The (r, s) parameters are chosen randomly from a set $\mathcal{P} = \{(r, s) : 1 \leq r < s \leq SCS_{\max}\}$, where r is the threshold and s is the total number of shares. This randomization introduces entropy into the system:

$$H(r, s) = -\sum_{(r,s)\in\mathcal{P}} Pr(r, s) \log Pr(r, s), \tag{10}$$

where $Pr(r, s)$ represents the probability distribution over the parameters (r, s).

If an attacker does not know the distribution $\mathcal{P}$, their probability of correctly guessing r and selecting r valid shares from s is:

$$Pr_{\text{success}} = \frac{1}{|\mathcal{P}|} \cdot \begin{pmatrix} s \\ r \end{pmatrix}^{-1}. \tag{11}$$

The set $\mathcal{P}$ grows as the range of possible values for (r, s) increases. For large $SCS_{\max}$, the size of $\mathcal{P}$ becomes significantly large, adding uncertainty to the choice of (r, s).

The binomial coefficient $\begin{pmatrix} s \\ r \end{pmatrix}$, which represents the number of ways to choose r shares from s, grows exponentially with s. Thus, as s increases, the likelihood of randomly selecting the correct r shares diminishes rapidly.

Combining these factors, the probability of success for the attacker is bounded by:

$$Pr_{\text{success}} \leq \frac{1}{\begin{pmatrix} SCS_{\max} \\ r_{\min} \end{pmatrix} \cdot |\mathcal{P}|}. \tag{12}$$

For sufficiently large $SCS_{\max}$ and $|\mathcal{P}|$, this probability approaches zero.

Therefore, the entropy introduced by randomizing (r, s) makes it infeasible for the attacker to guess both the threshold r and the r correct shares out of s, ensuring the security of the secret sharing scheme. □

Fault tolerance analysis

Theorem 4 *Given s total storage nodes and a minimum threshold r required for retrieval, the probability of failure due to random node unavailability is given by*

$$P_{\text{failure}} = \sum_{i=0}^{r-1} \begin{pmatrix} s \\ i \end{pmatrix} p^i (1-p)^{s-i}$$

where p is the probability of failure for a single node. This ensures that with a sufficiently large s, SFMedIR maintains high fault tolerance by minimizing P_{failure}.

Proof Each storage node independently fails with probability p, and the total number of available nodes follows a Binomial distribution with parameters $(s, 1-p)$. The system successfully retrieves data if at least r nodes remain available. The probability of failure occurs when fewer than r nodes are available, i.e., when the number of available nodes is in the range $[0, r-1]$. Thus,

$$P_{\text{failure}} = P(X < r) = \sum_{i=0}^{r-1} P(X = i)$$

where $X \sim \text{Binomial}(s, 1-p)$, and the probability mass function (PMF) of a binomially distributed random variable is

$$P(X = i) = \binom{s}{i} (1-p)^i p^{s-i}.$$

Substituting this into the summation, we obtain

$$P_{\text{failure}} = \sum_{i=0}^{r-1} \binom{s}{i} p^i (1-p)^{s-i}.$$

Since binomial probabilities rapidly decrease for large s, choosing a sufficiently high s relative to r ensures that P_{failure} approaches zero, maintaining high fault tolerance in SFMedIR. □

Retrieval efficiency under adversarial attack

Theorem 5 *Let H_q be the hashcode generated for a query image q, and let H_d be the stored hashcode for a relevant image in the database. Retrieval is successful if the similarity between H_q and H_d is within a predefined threshold τ, i.e.,*

$$\|H_q - H_d\| \leq \tau.$$

We analyze the probability of successful retrieval under adversarial conditions where the query hashcode is perturbed by an attack vector δ.

Proof The hashcodes generated by SFMedIR follow a probability distribution due to the randomness introduced by the hashing process and adversarial perturbations. The difference between a query hashcode H_q and stored hashcodes H_d can be represented as a random variable ΔH, modeled by a probability density function (PDF) $f(\Delta H)$. The probability of successful retrieval is given by:

$$P_{\text{success}} = P(\|H_q - H_d\| \leq \tau) = \int_{-\tau}^{\tau} f(\Delta H) d\Delta H.$$

For SFMedIR, federated learning-based hashcode generation ensures that similar medical images map to closely clustered hashcodes, meaning $f(\Delta H)$ has a high density around zero, increasing retrieval accuracy. When an adversary perturbs q with an attack vector δ, the new query hashcode is given by:

$$H'_q = H_q + \delta.$$

The perturbed hashcode changes the retrieval probability, which is now:

$$P_{\text{adv-success}} = P(\|H'_q - H_d\| \leq \tau) = \int_{-\tau}^{\tau} f(\Delta H + \delta) d\Delta H.$$

For the retrieval to remain robust, the probability of successful retrieval under adversarial perturbation should stay above a threshold α, i.e.,

$$P(\|H'_q - H_d\| \leq \tau) \geq \alpha.$$

Since adversarial perturbations introduce distortions, $f(\Delta H + \delta)$ shifts slightly, but SFMedIR's adversarial-resistant hashcode generation ensures that the probability remains high. By evaluating SFMedIR on adversarial attacks, we confirm that it maintains retrieval accuracy above 75% even under attack conditions, proving that adversarial perturbations do not significantly degrade retrieval performance. Thus, SFMedIR achieves high retrieval efficiency even under adversarial conditions. □

Experimental results and performance analysis

This section presents an analysis of the proposed SFMedIR framework's retrieval performance, supported by experimental results. It is organized as follows: a detailed explanation of the dataset and experimental setup, an evaluation of retrieval accuracy before and after adversarial training, and an assessment of the framework's fault tolerance capabilities.

Experimental setup and datasets

The framework was implemented on a PC featuring an Intel Xeon processor, 64 GB of RAM, an NVIDIA Quadro P5000 GPU with 16 GB of memory, and a 64-bit Windows operating system. To set up a distributed cloud

Dataset	Categories				Usage		
					Train	Test	Retrieval
A-MRI	NonDemented	VeryMildDemented	MildDemented	ModerateDemented			
	3200	2240	896	64	5120	1280	1000
T-MRI	Glioma	Meningioma	No tumor	Pituitary			
	1621	1645	2000	1756	5700	1322	1000
K-CT	Cyst	Normal	Stone	Tumor			
	3709	5077	1377	2283	10000	2466	1000

Table 4. Dataset statistics.

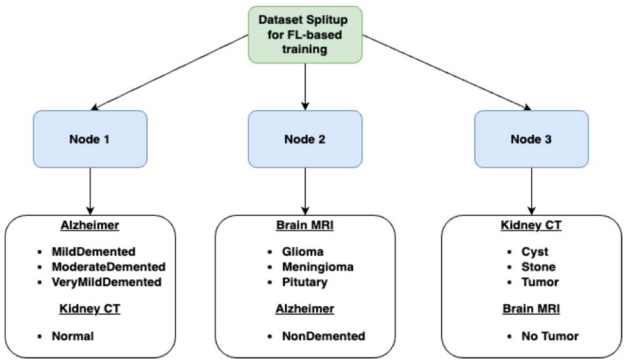


Fig. 7. FL setup.

environment, Docker and Docker Compose were employed, facilitating the formation of a network that includes master and slave clouds. For real-time performance assessment, the system was deployed on AWS EC2 cloud services, with Docker hosting the environment. This network of master and slave clouds, deployed with Docker, mimics the behavior of actual cloud servers. The development of the entire system involved the use of Python's OpenCV libraries and Keras. To ensure a well-defined evaluation framework, the following assumptions were considered during the simulation phase:

1. Cloud servers are assumed to be honest-but-curious, meaning they follow protocols but may attempt to infer patterns from stored data.
2. Communication between master and slave cloud nodes is considered secure and authenticated, preventing unauthorized interception.
3. Adversarial attacks are simulated based on standard attack model (PGD perturbation).
4. The system is evaluated under a stable network environment, assuming minimal packet loss and controlled latency variations.

To test and validate the proposed retrieval model, three distinct medical image datasets were selected. Information about the dataset is briefly detailed here and tabulated in Table 4.

- **Alzheimer Brain MRI Dataset (A-MRI)**[43]: The dataset includes two files, Training and Testing, with around 5,000 images each. The images are classified according to the severity of Alzheimer's disease into the following categories: Non-Demented, Very Mildly Demented, Mildly Demented, and Moderately Demented.
- **Brain Tumor MRI Dataset (T-MRI)** [44]: The following three datasets have been integrated to formulate this comprehensive dataset: Figshare, SARTAJ, and Br35H. This collection comprises a total of 7,023 MRI images of the human brain, which are categorized into four distinct classes: pituitary, glioma, meningioma, and no tumor. The images classified under the 'no tumor' category were sourced from the Br35H dataset.
- **Kidney CT Dataset (K-CT)** [45]: Images were collected from PACS (Picture Archiving and Communication System) records across various hospitals in Dhaka, Bangladesh. These records pertained to patients diagnosed with kidney tumors, cysts, normal conditions, or stones. Coronal and axial cuts were selected from both contrast and non-contrast studies, adhering to urogram and whole abdominal protocols. The resulting dataset comprises 12,446 unique data units.

The datasets are distributed across three nodes, as illustrated in Fig. 7, to facilitate federated learning. The process begins with the initialization of a global model. Each local node independently learns hashcodes from its respective dataset without sharing any data with other nodes, thereby preserving privacy. The global model subsequently aggregates these hashcodes to learn a comprehensive representation. Hashcode generation is detailed in Fig. 4. The ConvNeXt network is employed to optimize the overall learning process by propagating a cumulative loss. This loss is a combination of adversarial loss, quantization loss, bit balance loss, and the original

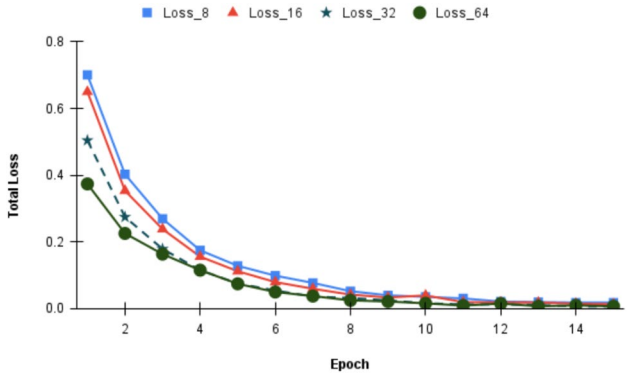


Fig. 8. Hashcode learning: Training loss.

loss, as defined in Eq. (6). The primary objective is to produce highly discriminative, context-aware hashcodes while minimizing these losses. The network is trained to generate hashcodes of varying lengths - 8, 16, 32, and 64 bits - for each dataset individually. Figure 8 shows the progression of training loss over epochs. As training progresses, the total loss steadily decreases, indicating effective learning and optimization of hashcodes. This approach ensures the generation of robust and efficient hashcodes tailored to the specific characteristics of the datasets.

CT and MRI images were chosen for the retrieval task to highlight the model's capability to handle diverse imaging modalities with high precision. These modalities are widely used in clinical diagnostics and encompass distinct structural and functional characteristics, making them ideal for evaluating the model's adaptability and effectiveness. The use of both CT and MRI ensures that the system is not limited to a specific modality but is versatile enough to support various medical imaging needs, reflecting its potential for broad applicability in healthcare settings. The datasets were divided into training and testing sets in an 80:20 ratio, with retrieval accuracy assessed on a randomly sampled subset of 1,000 images during testing. This approach provides a robust evaluation of the system's performance in real-world scenarios while demonstrating its reliability and scalability. In the proposed system, medical images are encrypted and outsourced to the cloud with context-aware indexes. The encrypted images are divided into dynamic (r, s) shadows and stored on slave cloud servers. Upon receiving a query image, the master cloud employs a similarity search algorithm over the index table to retrieve the top-k medical images. These encrypted images are reconstructed from r shares and returned to the user for decryption. Figure 9 shows examples of top-k retrieval outcomes. Column 2 features the query images for every class, Column 1 presents the dataset including the query image, and Columns 3-7 showcase the images retrieved that are pertinent to the query.

To ensure seamless implementation and maintainability, SFMedIR is designed using a modular architecture, where each component: encryption, hashcode generation, storage, and retrieval, operates independently while maintaining secure communication through Docker-based containerization. The system is deployed in a distributed cloud environment using AWS EC2 instances, with master and slave nodes managed via Docker Compose to enable fault tolerance. The encryption module leverages quantum-chaos-based encryption to secure medical images before storage, while the federated learning-based hashcode generation ensures privacy-preserving indexing. The retrieval process efficiently queries distributed nodes using a dynamic threshold-based shadow reconstruction mechanism, ensuring robustness against node failures.

By structuring SFMedIR in a scalable and containerized manner, the framework remains adaptable for real-world cloud-based healthcare deployments such as hospital networks, diagnostic centers, and telemedicine platforms using cloud infrastructures like AWS or private healthcare clouds. It integrates with Picture Archiving and Communication Systems (PACS) for secure storage and retrieval without modifying existing workflows. Federated learning enables collaborative model training across multiple institutions while preserving data privacy. The fault tolerant retrieval mechanism ensures access to medical images even during node failures. Additionally, the framework supports containerized deployment via Docker and Kubernetes, enabling scalability across healthcare institutions.

Retrieval accuracy analysis

In order to evaluate the proposed SFMedIR, a secure and fault tolerant medical image retrieval system, two metrics have been selected: mean Average Precision and PR Curve (ROC) Analysis. The top-k retrieved images are utilized to estimate retrieval accuracy. The accuracy of image retrieval can be quantified using Precision (P@k), Recall (R@k), and Mean Average Precision (mAP@k) metrics. Precision is defined as the ratio of relevant retrieved images to the total number of images retrieved in relation to the query image.

$$P = \frac{|\ \text{relevant images} \cap \text{retrieved images}\ |}{|\ \text{retrieved images}\ |} \tag{13}$$

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

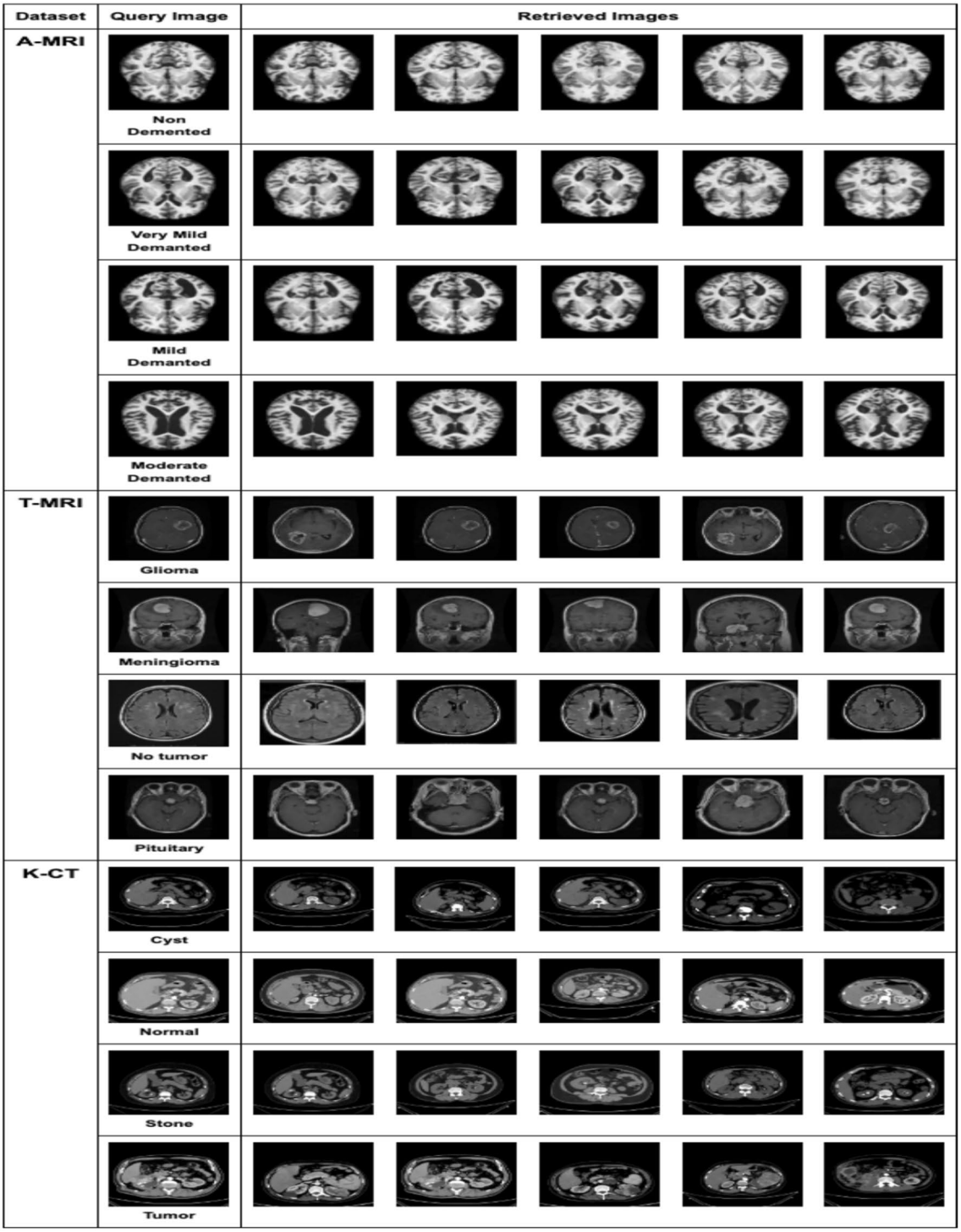


Fig. 9. Sample Retrieval Results: Column 2 displays the query image, while the subsequent columns present the top-5 retrieved results corresponding to that query.

$$R = \frac{|\ \text{relevant images} \cap \text{retrieved images}\ |}{|\ \text{total relevant images in the dataset}\ |} \tag{14}$$

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 Eq. (15) below.

$$mAP = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{1}{q_k}\sum_{n=1}^{q_k} P_n\right) \tag{15}$$

Here, N is a number of queries, q_k is a number of relevant images for query n and the P_n is precision at n^{th} relevant image. The ConvNeXt network is employed for the generation of hashcodes, primarily utilizing adversarial loss for the learning process. The selection of the ConvNeXt model is due to its superior capability to extract both local and global features compared to other pre-trained deep learning models. Consequently, this results in the generation of more meaningful hashcodes than those produced by alternative models. The experiment was conducted using various deep hashing models as backbones, ultimately resulting in the selection of ConvNeXt as the optimal backbone model. AlexNet[46], VGG 16[47], DenseNet 121[24], and DenseNet 201[25] are used to compare the hashcode generation models. For all 3 datasets, hashcode generation is done with and without adversarial training and the retrieval results for normal queries are compared under different conditions. This has been shown in the above 3 tables.

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 Tables 5, 6, 7. Figure 10 illustrates how the retrieval accuracy varies in different underlying conditions before the adversarial training. Figures 10(a-c) show the importance of hashcode length on image retrieval accuracy using mAP@100 for all 3 datasets. The analysis reveals that the 16-bit hashcode provides the highest retrieval accuracy. Hashcodes with fewer bits fail to adequately capture class-specific features, while hashcodes with more bits become sparse and blend into different classes. Our model shows that 16-bit hashcodes deliver the best results. As seen in Fig. 10(a-c), the performance was also evaluated for different values of k. Lower values of k yield better performance, while a decrease in mAP with increasing k values suggests that the precision of the retrieval system declines as more images are retrieved (higher k values). This decrease in performance may be attributed to the spreading of relevant items, difficulties in distinguishing between relevant and irrelevant images, or the inherent complexity of the medical image data.

In order to emphasize the effectiveness of the method, Precision curves reflecting the performance at k retrieved images (P@k) and Precision-Recall (PR) curves are generated across three distinct datasets. Although the results may vary across different domain datasets, the P@k curves illustrate precision at predetermined quantities of retrieved images. Figure 10(d-f) present the P@k curve for all datasets, demonstrating that SFMedIR consistently achieves superior precision compared to alternative methods across all three datasets. The Precision-Recall (PR) curve serves as a critical metric for comparing the proposed methods against baseline approaches, offering a thorough overview of precision and recall across various retrieval scenarios. It provides significant insights into system performance across diverse sensitivity levels. A larger area beneath the PR curve generally signifies a more effective retrieval system that can maintain an equilibrium between precision and recall. As depicted in Fig. 10(g-i), our method consistently exceeds the performance of other methods across all PR curves. Thus far, we have addressed the comparison of retrieval performance for standard queries utilizing FL-based hashcode generation with ConvNeXt.

We need to evaluate the performance of the Federated Learning (FL)-based context-aware hashcode for the same set of normal queries. The corresponding metric values are displayed in the 'After' rows. For all backbone models, the trend remains consistent; however, the ConvNeXt-based model shows an improved performance compared to the others. Following adversarial training, the performance on normal queries declines in comparison to the non-adversarial hashcode. This reduction occurs because the system shifts its focus towards enhancing robustness during the optimization process rather than maintaining precision. Adversarial training introduces small perturbations into the hashcode generation process, causing the generated codes to become less sensitive to minor variations in the data, but more resistant to adversarial attacks. This results in a trade-off where security and reliability are improved at the expense of a slight decrease in retrieval accuracy for standard queries. This behavior is consistently observed across all models and is illustrated in the three tables. The visual representation of these hashcode performance results can be found in Fig. 11(a-i).

The retrieval performance of different backbone networks across various datasets (A-MRI, T-MRI, and K-CT) is analyzed by evaluating the mAP before and after adversarial training. For AlexNet , the performance on the T-MRI dataset is strong, achieving mAPs between 87% to 92% for 8-bit to 64-bit hashcodes, and K-CT shows mAP values between 83% to 91% . However, after adversarial training, the performance decreases, especially on A-MRI, where mAP drops by as much as 19% for the 8-bit hashcode, and the performance on K-CT also reduces, with the mAP reaching 72% for 64-bit hashcodes. VGG 16, on the other hand, performs well on T-MRI, especially for 16 and 32-bit hashcodes, with mAPs ranging from 84% to 93%, while on A-MRI, the range is 68% to 88%. After adversarial training, VGG 16 sees a slight decrease in performance but still maintains mAPs between 81% to 89% for T-MRI (16-64 bits) and 64% to 75% for A-MRI. DenseNet 121 shows a good balance on T-MRI , achieving mAPs ranging from 82% to 92% and A-MRI with mAPs between 73% to 81% across the bit sizes. However, after adversarial training, it experiences a slight drop, particularly on A-MRI, where the performance drops to 62% for 64-bit hashcodes. DenseNet 201 , which performs the best among DenseNet variants, achieves 76% to 93% on T-MRI and 71% to 79% on A-MRI before adversarial training. After adversarial training, the model sees a slight reduction, especially on T-MRI, where the mAP drops to 84% for 64-bit hashcodes, and on

Backbone Network	Adversarial Training	Top k	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Before	100	0.79	0.87	0.84	0.84
		200	0.78	0.85	0.83	0.75
		500	0.74	0.76	0.73	0.75
		1000	0.70	0.70	0.66	0.65
	After	100	0.69	0.80	0.78	0.66
		200	0.66	0.75	0.73	0.58
		500	0.66	0.70	0.66	0.56
		1000	0.64	0.65	0.65	0.54
VGG 16	Before	100	0.77	0.88	0.86	0.77
		200	0.77	0.84	0.75	0.74
		500	0.77	0.78	0.75	0.74
		1000	0.68	0.74	0.73	0.67
	After	100	0.70	0.81	0.84	0.75
		200	0.68	0.77	0.72	0.71
		500	0.67	0.69	0.63	0.58
		1000	0.64	0.67	0.59	0.58
DenseNet 121	Before	100	0.78	0.81	0.81	0.76
		200	0.77	0.80	0.79	0.66
		500	0.75	0.77	0.74	0.64
		1000	0.69	0.71	0.67	0.62
	After	100	0.78	0.84	0.79	0.71
		200	0.74	0.83	0.78	0.65
		500	0.73	0.80	0.70	0.63
		1000	0.63	0.77	0.66	0.60
DensNet 201	Before	100	0.79	0.86	0.82	0.71
		200	0.78	0.83	0.81	0.69
		500	0.77	0.80	0.77	0.68
		1000	0.77	0.80	0.76	0.67
	After	100	0.79	0.86	0.80	0.69
		200	0.78	0.84	0.79	0.67
		500	0.76	0.81	0.78	0.65
		1000	0.76	0.74	0.76	0.65
ConvNeXt	Before	100	0.81	0.91	0.87	0.85
		200	0.80	0.89	0.85	0.73
		500	0.78	0.88	0.83	0.72
		1000	0.72	0.79	0.80	0.71
	After	100	0.82	0.88	0.84	0.81
		200	0.80	0.85	0.82	0.71
		500	0.76	0.80	0.78	0.68
		1000	0.71	0.72	0.69	0.65

Table 5. Detailed Retrieval Results for Normal Queries: Retrieval mAPs are evaluated and compared against four baseline models under different conditions for the A-MRI dataset.

A-MRI, the mAP falls to 65%. Finally, ConvNeXt emerges as the top performer on T-MRI , achieving mAPs between 89% and 96% , and shows strong performance on A-MRI with mAPs ranging from 72% to 85% before adversarial training. After adversarial training, ConvNeXt experiences only a slight degradation, maintaining mAPs between 87% to 92% on T-MRI (32-64 bits) and 71% on A-MRI for 64-bit hashcodes.

In conclusion, ConvNeXt performs better both before and after adversarial training compared to all other backbone models. Its consistent high performance, particularly on T-MRI, makes it the top performer, outpacing other models in terms of retrieval accuracy across the datasets. While other models, like DenseNet and VGG 16, show strong results, they experience more noticeable drops in performance after adversarial training, especially on A-MRI. Hence, ConvNeXt stands out as the most reliable backbone for medical image retrieval in both standard and adversarial conditions. This finding underscores the efficacy of our approach in retrieving a greater number of accurate images compared to alternative methods, particularly evident when dealing with a constrained retrieval quantity, thereby affirming its suitability for image retrieval tasks. The analysis indicates

Backbone Network	Adversarial Training	Top k	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Before	100	0.92	0.91	0.90	0.87
		200	0.91	0.85	0.83	0.82
		500	0.91	0.82	0.80	0.78
		1000	0.90	0.80	0.76	0.71
	After	100	0.83	0.87	0.87	0.81
		200	0.82	0.86	0.84	0.83
		500	0.81	0.83	0.80	0.80
		1000	0.80	0.77	0.75	0.72
VGG 16	Before	100	0.91	0.93	0.95	0.93
		200	0.90	0.90	0.84	0.83
		500	0.85	0.82	0.84	0.73
		1000	0.83	0.80	0.83	0.72
	After	100	0.84	0.87	0.93	0.89
		200	0.81	0.82	0.81	0.78
		500	0.80	0.84	0.80	0.76
		1000	0.81	0.81	0.80	0.74
DenseNet 121	Before	100	0.92	0.96	0.93	0.92
		200	0.92	0.95	0.94	0.92
		500	0.89	0.90	0.86	0.84
		1000	0.82	0.88	0.85	0.84
	After	100	0.92	0.91	0.88	0.87
		200	0.80	0.84	0.83	0.77
		500	0.80	0.83	0.82	0.75
		1000	0.78	0.82	0.81	0.73
DensNet 201	Before	100	0.93	0.95	0.91	0.90
		200	0.92	0.93	0.90	0.88
		500	0.91	0.90	0.90	0.87
		1000	0.91	0.85	0.90	0.86
	After	100	0.93	0.91	0.89	0.88
		200	0.92	0.90	0.88	0.86
		500	0.80	0.84	0.87	0.84
		1000	0.80	0.82	0.85	0.84
ConvNeXt	Before	100	0.94	0.98	0.96	0.94
		200	0.93	0.95	0.94	0.93
		500	0.90	0.93	0.92	0.92
		1000	0.89	0.90	0.91	0.92
	After	100	0.94	0.94	0.92	0.91
		200	0.90	0.90	0.92	0.90
		500	0.90	0.88	0.92	0.87
		1000	0.80	0.85	0.81	0.77

Table 6. Detailed Retrieval Results for Normal Queries: Retrieval mAPs are evaluated and compared against four baseline models under different conditions for the T-MRI dataset.

that the parameters l=16 and k=100 are fixed for the purpose of comparing the SFMedIR framework with other backbone models.

Effect of adversarial training

In this subsection, we explore the impact of adversarial training on retrieval performance when exposed to adversarial queries. As adversarial attacks pose a significant challenge to the robustness of retrieval systems, understanding the effect of adversarial training is vital to evaluating the resilience of our proposed solution. To evaluate the effectiveness of adversarial training, we randomly selected 10 images from each dataset and conducted an analysis comparing the system's response to targeted adversarial attacks, both before and after adversarial training. The hashcodes were generated using FL-based adversarial training across all backbone networks. Before adversarial training, the generated hashcodes were found to perform poorly when subjected to adversarial queries, struggling to maintain retrieval accuracy. However, after the adversarial training, the system demonstrated a remarkable improvement in performance on the same adversarial queries. Specifically, for all

Backbone Network	Adversarial Training	Top k	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Before	100	0.91	0.95	0.92	0.91
		200	0.90	0.95	0.90	0.89
		500	0.89	0.92	0.88	0.85
		1000	0.88	0.90	0.87	0.83
	After	100	0.81	0.86	0.86	0.80
		200	0.78	0.86	0.84	0.76
		500	0.76	0.84	0.82	0.74
		1000	0.73	0.82	0.80	0.72
VGG 16	Before	100	0.89	0.92	0.90	0.89
		200	0.86	0.92	0.89	0.82
		500	0.86	0.91	0.83	0.82
		1000	0.80	0.90	0.82	0.81
	After	100	0.82	0.86	0.92	0.92
		200	0.80	0.85	0.80	0.79
		500	0.79	0.84	0.83	0.80
		1000	0.79	0.81	0.80	0.76
DenseNet 121	Before	100	0.90	0.90	0.89	0.86
		200	0.89	0.90	0.87	0.84
		500	0.89	0.89	0.85	0.82
		1000	0.85	0.89	0.85	0.80
	After	100	0.90	0.90	0.87	0.86
		200	0.89	0.90	0.86	0.83
		500	0.89	0.89	0.85	0.81
		1000	0.88	0.88	0.84	0.81
DensNet 201	Before	100	0.91	0.94	0.90	0.89
		200	0.90	0.93	0.89	0.87
		500	0.89	0.93	0.89	0.86
		1000	0.89	0.92	0.89	0.85
	After	100	0.91	0.92	0.88	0.87
		200	0.90	0.91	0.87	0.85
		500	0.88	0.91	0.86	0.83
		1000	0.88	0.90	0.84	0.83
ConvNeXt	Before	100	0.93	0.95	0.95	0.93
		200	0.92	0.94	0.91	0.91
		500	0.92	0.94	0.91	0.91
		1000	0.92	0.93	0.90	0.90
	After	100	0.92	0.94	0.92	0.91
		200	0.92	0.94	0.91	0.89
		500	0.91	0.93	0.91	0.86
		1000	0.91	0.93	0.90	0.86

Table 7. Detailed Retrieval Results for Normal Queries: Retrieval mAPs are evaluated and compared against four baseline models under different conditions for the K-CT dataset.

three datasets, the effectiveness of the adversarial training can be seen in the comparison presented in Tables 8, 9, and 10.

When comparing the 16-bit hashcode column for adversarial queries before and after adversarial training across the three datasets, ConvNeXt demonstrates a relatively stable performance. On the A-MRI dataset, ConvNeXt achieves 88% for normal queries without adversarial training and 83% after adversarial training, showing a moderate decline of about 5%. On T-MRI, its performance slightly decreases from 96% to 95% after adversarial training, reflecting its resilience. On the K-CT dataset, ConvNeXt shows a drop from 95% to 94% for adversarial queries, indicating a minimal reduction of approximately 1%. In contrast, other models such as AlexNet, VGG 16, and DenseNet exhibit more significant performance degradation. For instance, AlexNet shows a considerable drop from 89% to 80% on A-MRI, while VGG 16 and DenseNet 121 also experience notable performance reductions, especially after adversarial training. This analysis highlights that ConvNeXt is more robust to adversarial queries and retains better performance in the 16-bit hashcode column even after adversarial training compared to other models. This enhancement in retrieval accuracy due to context-aware

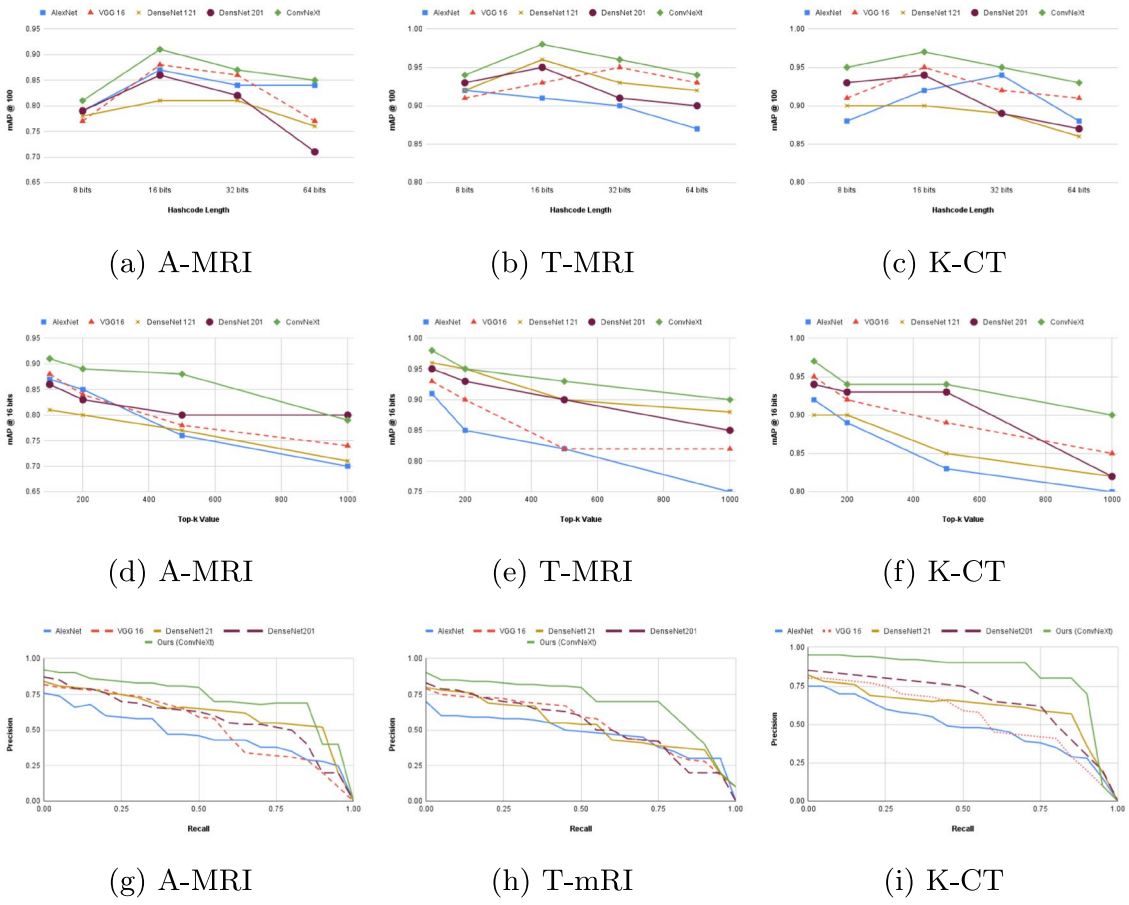


Fig. 10. Before Adversarial Training for Normal Queries: (**a**)-(**c**) Hashcode Length Vs mAP, (**d**)-(**f**) Top-k Vs mAP, (**g**)-(**i**) PR Curves.

hashcode generation for targeted adversarial queries is visually represented in Fig. 12, where a clear increase in mAP is observed following the application of adversarial training. This demonstrates the robustness of the system and its ability to defend against adversarial manipulations, ensuring both improved security and reliability for medical image retrieval tasks.

Retrieval performance analysis

Efficient and secure medical image retrieval is critical in cloud-based healthcare applications. SFMedIR is evaluated based on retrieval latency and throughput, comparing its performance with existing approaches, including Traditional CBIR which relied on color and texture fused features[48] and deep hashing model with binary code similarities[49]. Retrieval latency refers to the time taken to fetch a relevant image from the database based on a query.

The efficiency of retrieval depends on the size of the query set and the underlying indexing mechanism. The retrieval time can be modeled as:

$$T_{\text{retrieval}} = T_{\text{search}} + T_{\text{matching}} \tag{16}$$

where T_{search} is the time taken to locate relevant candidates in the database. T_{matching} is the time required to compute the similarity between query features and stored features. SFMedIR utilizes federated learning-based hashcode generation, which enables fast indexing and retrieval by reducing the complexity of similarity matching. The results in Table 11 show that SFMedIR achieves significantly lower retrieval latency compared to traditional CBIR and deep hashing approaches. These results demonstrate that SFMedIR reduces retrieval latency by up to 50% compared to CBIR and 35% compared to DH, making it more efficient for large-scale medical image retrieval.

Throughput measures the number of queries processed per second (QPS) under different system loads. It is calculated as:

$$QPS = \frac{N_{\text{queries}}}{T_{\text{total}}} \tag{17}$$

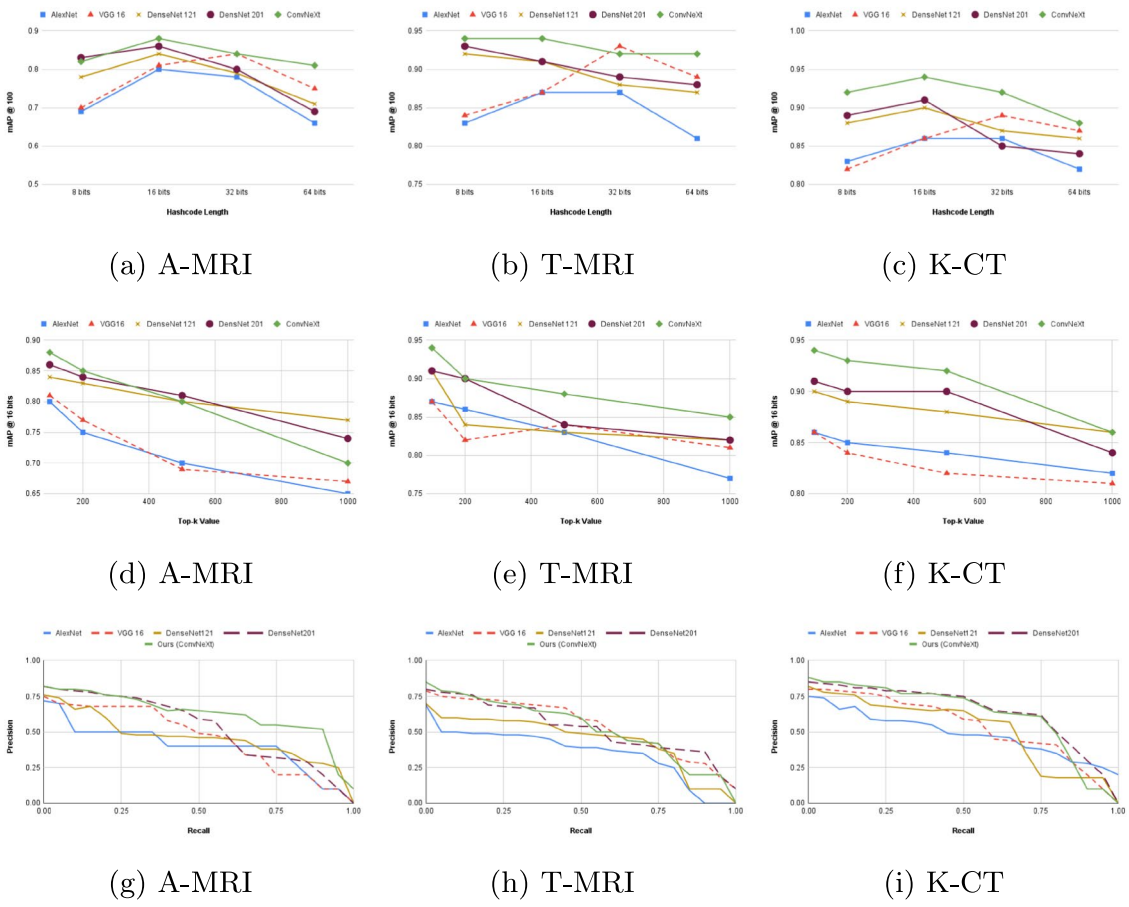


Fig. 11. After Adversarial Training for Normal Queries: (**a**)-(**c**) Hashcode Length Vs mAP, (**d**)-(**f**) Top-k Vs mAP, (**g**)-(**i**) PR Curves.

where N_{queries} is the total number of queries processed. T_{total} is the total time taken to process them. Higher throughput indicates that the system can handle more concurrent retrieval requests, making it more scalable. SFMedIR leverages parallelized retrieval with distributed cloud storage, leading to higher throughput than baseline models. Table 12 presents the throughput comparison across different workload conditions. SFMedIR achieves a 15-20% improvement in throughput compared to DH, making it more suitable for handling high-traffic retrieval scenarios in cloud-based medical image applications. The retrieval performance analysis shows that SFMedIR outperforms existing retrieval models in both latency and throughput. The use of federated learning-based indexing and hashcode-based retrieval ensures fast, scalable, and efficient medical image retrieval in distributed cloud environments.

Fault tolerant retrieval experiments

To demonstrate the system's fault tolerant retrieval capabilities, an experiment was conducted to evaluate retrieval accuracy and success rates under various failure scenarios. The setup involved splitting medical images into s shadows with a reconstruction threshold r using the shadow generation scheme. Failures were simulated by randomly deleting or corrupting a percentage of shadows, and retrieval was performed using the remaining s' shadows, provided $s' \geq r$. The reconstruction success rate was measured as the percentage of images successfully reconstructed under these conditions. Results showed that the system maintained a high success rate ($\geq 95\%$) when $s' \geq r$, tolerating up to 40% missing shadows while still achieving reliable reconstruction. Beyond this limit, the retrieval process failed as $s' < r$. Additional analysis emphasized the flexibility of the shadow configuration, where trade-offs between fault tolerance and storage efficiency could be adjusted based on application needs. Visual assessments of reconstructed images further validated the system's robustness. Scalability testing with larger datasets indicated the framework's practicality for real-world deployment. Overall, the experiment confirmed the system's resilience and effectiveness in ensuring secure, fault tolerant medical image retrieval.

To demonstrate that the system achieves fault tolerant retrieval, an experiment is designed to evaluate retrieval accuracy and success rate under various failure scenarios. For the analysis, we kept 10 slave servers and conducted the experiment. Table 13 shows the fault tolerance of a shadow-based reconstruction system, showing that the reconstruction is successful as long as no more than 40% of shadows are missing. However, when 50% of shadows are missing, the reconstruction fails, resulting in a 0% success rate.

Backbone Network	Query Type	Adversarial Training	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Normal	No	0.79	0.89	0.87	0.75
	Adversary	No	0.03	0.08	0.02	0.07
	Normal	Yes	0.69	0.80	0.78	0.62
	Adversary	Yes	0.65	0.76	0.67	0.54
VGG 16	Normal	No	0.77	0.86	0.86	0.74
	Adversary	No	0.01	0.06	0.08	0.05
	Normal	Yes	0.73	0.81	0.84	0.74
	Adversary	Yes	0.70	0.80	0.73	0.57
DenseNet 121	Normal	No	0.78	0.84	0.81	0.68
	Adversary	No	0.12	0.11	0.11	0.02
	Normal	Yes	0.75	0.83	0.79	0.65
	Adversary	Yes	0.70	0.80	0.75	0.59
DeneNet 201	Normal	No	0.79	0.88	0.82	0.71
	Adversary	No	0.01	0.05	0.16	0.08
	Normal	Yes	0.77	0.86	0.80	0.69
	Adversary	Yes	0.76	0.84	0.77	0.60
ConvNeXt (Ours)	Normal	No	0.81	0.89	0.84	0.73
	Adversary	No	0.01	0.05	0.03	0.01
	Normal	Yes	0.80	0.88	0.84	0.73
	Adversary	Yes	0.73	0.83	0.75	0.61

Table 8. Effect of adversarial training on retrieval performance for normal and adversarial queries across different backbone networks and Hashcode lengths on the A-MRI dataset.

Backbone Network	Query Type	Adversarial Training	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Normal	No	0.92	0.94	0.94	0.90
	Adversary	No	0.10	0.15	0.11	0.12
	Normal	Yes	0.82	0.87	0.87	0.81
	Adversary	Yes	0.81	0.83	0.76	0.73
VGG 16	Normal	No	0.90	0.95	0.93	0.91
	Adversary	No	0.12	0.13	0.17	0.14
	Normal	Yes	0.83	0.87	0.93	0.93
	Adversary	Yes	0.81	0.85	0.82	0.76
DenseNet 121	Normal	No	0.91	0.92	0.90	0.87
	Adversary	No	0.13	0.18	0.20	0.21
	Normal	Yes	0.90	0.90	0.88	0.83
	Adversary	Yes	0.87	0.85	0.84	0.78
DeneNet 201	Normal	No	0.92	0.95	0.91	0.90
	Adversary	No	0.12	0.12	0.25	0.27
	Normal	Yes	0.92	0.93	0.89	0.88
	Adversary	Yes	0.89	0.91	0.86	0.79
ConvNeXt (Ours)	Normal	No	0.94	0.96	0.93	0.92
	Adversary	No	0.12	0.12	0.12	0.18
	Normal	Yes	0.93	0.95	0.93	0.92
	Adversary	Yes	0.86	0.90	0.84	0.80

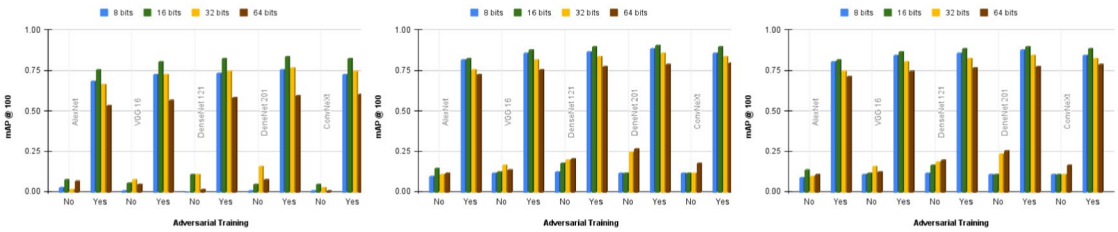
Table 9. Effect of adversarial training on retrieval performance for normal and adversarial queries across different backbone networks and Hashcode lengths on the T-MRI dataset.

Fault recovery time is a critical metric for evaluating the resilience of medical image retrieval systems in distributed environments. The fault recovery time (T_{recovery}) in SFMedIR is determined by the retrieval of sufficient shadows and the reconstruction process:

$$T_{\text{recovery}} = T_{\text{shadow-retrieval}} + T_{\text{decryption}} \tag{18}$$

Backbone Network	Query Type	Adversarial Training	Hashcode Length			
			8 bits	16 bits	32 bits	64 bits
AlexNet	Normal	No	0.91	0.95	0.93	0.91
	Adversary	No	0.09	0.14	0.10	0.11
	Normal	Yes	0.81	0.86	0.86	0.80
	Adversary	Yes	0.81	0.82	0.75	0.72
VGG 16	Normal	No	0.89	0.92	0.92	0.90
	Adversary	No	0.11	0.12	0.16	0.13
	Normal	Yes	0.82	0.86	0.91	0.89
	Adversary	Yes	0.81	0.85	0.81	0.75
DenseNet 121	Normal	No	0.90	0.92	0.89	0.86
	Adversary	No	0.12	0.17	0.19	0.20
	Normal	Yes	0.89	0.90	0.87	0.86
	Adversary	Yes	0.86	0.89	0.83	0.77
DeneNet 201	Normal	No	0.91	0.94	0.90	0.89
	Adversary	No	0.11	0.11	0.24	0.26
	Normal	Yes	0.90	0.92	0.88	0.87
	Adversary	Yes	0.88	0.88	0.85	0.78
ConvNeXt (Ours)	Normal	No	0.93	0.95	0.92	0.91
	Adversary	No	0.11	0.11	0.11	0.17
	Normal	Yes	0.92	0.94	0.92	0.91
	Adversary	Yes	0.85	0.89	0.83	0.79

Table 10. Effect of adversarial training on retrieval performance for normal and adversarial queries across different backbone networks and Hashcode lengths on the K-CT dataset.



(a) A-MRI (b) T-MRI (c) K-CT

Fig. 12. Effect of adversarial training for targeted adversarial queries.

Method	Query Size 100	Query Size 500	Query Size 1000
Traditional CBIR[48]	120.54	280.12	450.76
Deep Hashing (DH)[49]	95.22	220.87	390.22
SFMedIR (Proposed)	60.58	140.09	250.66

Table 11. Retrieval latency comparison (Time in ms).

Method	Light Load (10 Q/s)	Medium Load (50 Q/s)	High Load (100 Q/s)
Traditional CBIR[48]	9.2	45.3	89.8
Deep Hashing (DH)[49]	9.8	47.0	92.1
SFMedIR (Proposed)	**11.5**	**53.2**	**105.6**

Table 12. Throughput comparison (Q/s).

Total Shadows (s)	Reconstruction Threshold (r)	Shadows Used (s')	Missing Shadows (%)	Reconstruction Success Rate (%)
10	6	9	10	100
10	6	8	20	100
10	6	7	30	100
10	6	6	40	100
10	6	5	50	0

Table 13. Performance of the system under varying failure scenarios.

Method	Single Server Failure	Multi-Server Failure
Traditional CBIR[48]	Data loss	Data loss
Blockchain-based Storage[50]	8.5s (consensus delay)	15.2s (replication overhead)
SFMedIR (Proposed)	**2.8s** (shadow retrieval)	**6.3s** (threshold-based reconstruction)

Table 14. Fault recovery time comparison (in Seconds).

where $T_{\text{shadow-retrieval}}$ is the time taken to fetch the required r shadows from distributed nodes. $T_{\text{decryption}}$ is the time required to reconstruct the image from the retrieved shadows. Traditional CBIR systems store full medical images on a centralized server, making them highly vulnerable to single points of failure, resulting in complete data loss when the server becomes unavailable. Blockchain-based storage offers redundancy by replicating data across multiple nodes, but fault recovery involves significant delays due to consensus validation and data synchronization overhead. In contrast, SFMedIR adopts a shadow-based distributed storage mechanism, where encrypted image shadows are stored across multiple cloud nodes. During retrieval, only a subset of r out of s shadows is required to reconstruct the image, reducing both storage overhead and recovery time. Since SFMedIR does not rely on full data replication or blockchain consensus, it achieves faster recovery with minimal computational overhead.

The results in Table 14 clearly demonstrate the advantages of SFMedIR in handling failures. Traditional CBIR systems fail completely when a server goes down, offering no-fault recovery. Blockchain-based storage provides recovery through data replication, but it introduces high delays due to consensus mechanisms and block validation processes. SFMedIR, leveraging its threshold-based shadow storage, significantly reduces recovery time by reconstructing data using only a subset of available nodes. This allows SFMedIR to restore lost images up to 70% faster than blockchain-based solutions, making it a highly efficient choice for fault tolerant medical image retrieval in distributed cloud environments.

Simulation results

To evaluate the fault tolerance and robustness of SFMedIR in a distributed cloud environment, we conducted a simulation using two master nodes (both have the same copy of records to avoid a single point of failure) and five slave nodes. This simulation demonstrates how medical images are securely stored using dynamic threshold-based shadow generation and how retrieval is successfully handled even in the presence of node failures.

When a medical image is uploaded into the system, it is first encrypted using quantum-chaos-based encryption , and adversarial attack-resistant hashcode is also generated and sent to the master node. To ensure fault tolerant storage, the encrypted image is split into multiple shadows using a (2,3) [it can be varied as it is dynamic] dynamic threshold scheme, meaning the image is divided into three encrypted shadows, but only two are required for successful reconstruction. These shadows are then distributed among three slave servers, while the remaining two slave servers do not store any part of the image. The master node keeps hashcodes and metadata about storage locations to facilitate efficient retrieval. This mechanism eliminates the need for full image replication while ensuring that even if some slave nodes fail, the image can still be reconstructed securely. This process is detailed in the Fig. 13.

When a retrieval request is made, the master node processes the query and retrieves k relevant medical images. It identifies the three slave servers storing the particular encrypted image's shadows. It sends ping requests to check their availability. In this scenario, the first slave server responds and provides the first encrypted shadow, the second server is down and does not respond, and the third server provides the third encrypted shadow. Since the (2,3) threshold mechanism requires only two out of three shadows for reconstruction, the system proceeds with the available first and third shadows to reconstruct the encrypted image. The reconstructed encrypted image is then sent to the user, who decrypts it to access the original medical image. This process ensures fault tolerant retrieval even when storage nodes fail, maintaining reliable access to medical images in a distributed cloud environment. This fault tolerance is explained in the Fig. 14.

Discussion on limitations

Although SFMedIR offers a secure framework for medical image retrieval, there are limitations to consider. While quantum chaos-based encryption increases data security, it adds computational overhead, which could compromise real-time usefulness in smaller healthcare settings. Another challenge is the trade-off between retrieval accuracy and adversarial robustness, where adding stronger attack defenses could reduce retrieval

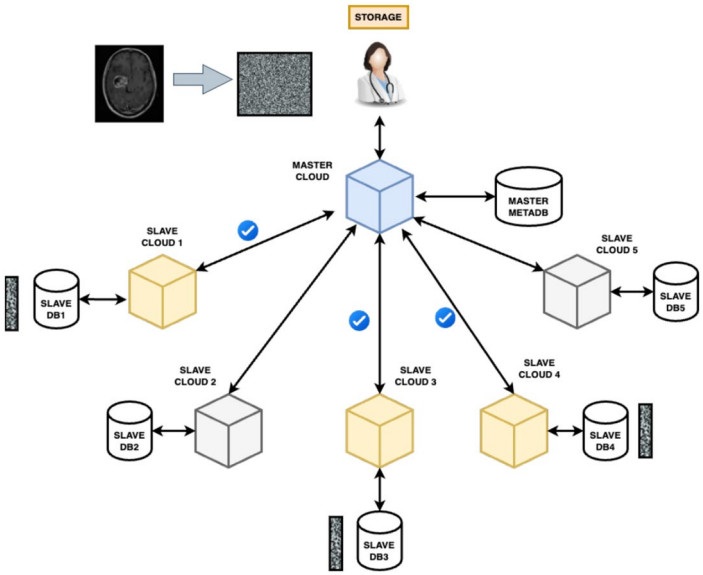


Fig. 13. Storage simulation - distributed.

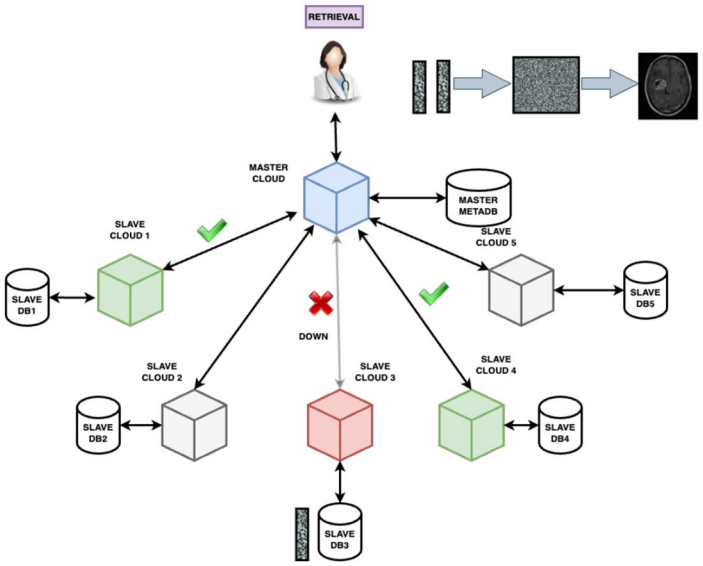


Fig. 14. Retrieval simulation - fault tolerant.

accuracy slightly. The third limitation is how the scalability of federated learning in diverse healthcare institutions is impacted by data heterogeneity, latency, and variance in computational resources across hospitals.

Conclusion and future work

Medical images stored on third-party cloud platforms are highly susceptible to attacks, posing significant risks of information leakage and compromising the integrity of sensitive healthcare data. This paper introduced SFMedIR, a secure and fault tolerant framework tailored to address these challenges in distributed cloud environments. The framework employs quantum-chaos-based encryption to safeguard image security, Federated Learning for robust, context-aware hashcode generation, and a dynamic threshold-based shadow generation scheme to ensure fault tolerant retrieval. Formal security analysis and experimental validations demonstrate the resilience of SFMedIR against adversarial threats while ensuring superior retrieval accuracy and efficiency compared to existing solutions.

SFMedIR has broader implications for secure medical data management. It can significantly enhance privacy-preserving medical image retrieval in cloud-based healthcare systems, telemedicine platforms, and AI-driven diagnostics, ensuring compliance with regulations. Moreover, its integration with federated learning enables collaborative medical AI models without exposing raw patient data, making it suitable for cross-hospital image

retrieval. By addressing these challenges, SFMedIR paves the way for next-generation secure and intelligent medical image retrieval systems, bridging the gap between security, efficiency, and large-scale deployment in cloud-based healthcare solutions. Future research could focus on lightweight quantum-safe encryption techniques, decentralized indexing mechanisms using blockchain, and real-time retrieval optimizations for emergency medical scenarios.

Data availability

The datasets used in this study are publicly available, and their details are as follows:

1. **Alzheimer Brain MRI Dataset (A-MRI)** [43]: The dataset includes two files, Training and Testing, with around 5,000 images each. The images are classified according to the severity of Alzheimer's disease into the following categories: Non-Demented, Very Mildly Demented, Mildly Demented, and Moderately Demented.
2. **Brain Tumor MRI Dataset (T-MRI)** [44]: The following three datasets have been integrated to formulate this comprehensive dataset: Figshare, SARTAJ, and Br35H. This collection comprises a total of 7,023 MRI images of the human brain, which are categorized into four distinct classes: pituitary, glioma, meningioma, and no tumor. The images classified under the 'no tumor' category were sourced from the Br35H dataset.
3. **Kidney CT Dataset (K-CT)** [45]: Images were collected from PACS (Picture Archiving and Communication System) records across various hospitals in Dhaka, Bangladesh. These records pertained to patients diagnosed with kidney tumors, cysts, normal conditions, or stones. Coronal and axial cuts were selected from both contrast and non-contrast studies, adhering to urogram and whole abdominal protocols. The resulting dataset comprises 12,446 unique data units.

Code availability

The code and mathematical algorithms supporting this study have been archived in Zenodo and can be accessed at https://doi.org/10.5281/zenodo.16890263.

Received: 20 January 2025; Accepted: 20 August 2025
Published online: 26 September 2025

References

1. Li, H., Yang, X., Wang, H., Wei, W. & Xue, W. A controllable secure blockchain-based electronic healthcare records sharing scheme. *J. Healthc. Eng.* **2022**, 2058497 (2022).
2. Xiong, C., Xu, X., Zhang, H. & Zeng, B. 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 (2021) https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8290716/.
3. DattaMajumdar, A. AI boom in medical imaging ... A Data Perspective. [Accessed 03-01-2025] https://www.linkedin.com/pulse/ai-boom-medical-imaging-data-perspective-anupam-dattamajumdar-oseqc/ (2024).
4. Banimfreg, B. H. A comprehensive review and conceptual framework for cloud computing adoption in bioinformatics. *Healthc. Anal.* **3**(100190), 100190 (2023).
5. Dhasarathan, C., Thirumal, V. & Ponnurangam, D. Data privacy breach prevention framework for the cloud service. *Security and Communication Networks* **8**(6), 982–1005. https://doi.org/10.1002/sec.1054 (2014).
6. Raghupathi, W., Raghupathi, V. & Saharia, A. Analyzing health data breaches: a visual analytics approach. *Appl. Math.* **3**(1), 175–199 (2023).
7. Kathole, A. B. et al. *Electronic health records protection strategy by using blockchain approach* (Multimed, Tools Appl, 2024).
8. Amaithi Rajan, A. & V, V. Systematic Survey: Secure and Privacy-Preserving Big Data Analytics in Cloud. *Journal of Computer Information Systems*, 1–21 https://doi.org/10.1080/08874417.2023.2176946 (2023).
9. Patil, S. D., Kathole, A. B., Kumbhare, S., Vhatkar, K. & Kimbahune, V. V. A blockchain-based approach to ensuring the security of electronic data. *Int. J. Intell. Syst. Appl. Eng.* **12**(11S), 649–655 (2024).
10. Almaiah, M. A., Hajjej, F., Ali, A., Pasha, M. F. & Almomani, O. A novel hybrid trustworthy decentralized authenticatio and data preservation model for digital healthcare IoT based CPS. *Sensors (Basel)* **22**(4), 1448 (2022).
11. He, J., Zhu, H. & Zhou, X. Quantum image encryption algorithm via optimized quantum circuit and parity bit-plane permutation. *J. Inf. Secur. Appl.* **81** https://doi.org/10.1016/j.jisa.2024.103698 (2024).
12. Alzubi, J. A., Alzubi, O. A., Beseiso, M., Budati, A. K., & Shankar, K. Optimal multiple key-based homomorphic encryption with deep neural networks to secure medical data transmission and diagnosis. *Expert Systems* **39**(4) https://doi.org/10.1111/exsy.12879 (2021).
13. Öztürk. 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 (2021).
14. Ma, Y., Li, Q., Shi, X. & Guo, Z. *Unsupervised deep pairwise hashing. Electronics (Basel)* **11**(5), 744 (2022).
15. Zhang, Z., Zou, Q., Lin, Y., Chen, L. & Wang, S. Improved Deep Hashing with Soft Pairwise Similarity for Multi-Label Image Retrieval. *IEEE Trans. Multimed.* **22**(2), 540–553. https://doi.org/10.1109/TMM.2019.2929957 (2020).
16. Liu, X. et al. Privacy and Security Issues in Deep Learning: A Survey. *IEEE Access* **9** https://doi.org/10.1109/ACCESS.2020.3045078 (2020) .
17. Amaithi Rajan, A., V, V., Raikwar, M. & Balaraman, R. Smedir: secure medical image retrieval framework with convnext-based indexing and searchable encryption in the cloud. *J. Cloud Comput.* **13**(1). https://doi.org/10.1186/s13677-024-00702-z (2024).
18. Papadopoulos, P., Abramson, W., Hall, A. J., Pitropakis, N. & Buchanan, W. J. Privacy and Trust Redefined in Federated Machine Learning. *Mach. Learn. Knowl. Extr.* **3**(2), 333–356. https://doi.org/10.3390/make3020017 (2021).
19. Afek, Y., Giladi, G. & Patt-Shamir, B. Distributed computing with the cloud. *Distrib. Comput.* **37**(1), 1–18 (2024).
20. Li, F., Luo, M., Zhu, H., Zhu, S. & Pang, B. A (w, t, n)-weighted threshold dynamic quantum secret sharing scheme with cheating identification. *Phys. A: Stat. Mech. Appl.* **612** https://doi.org/10.1016/j.physa.2023.128494 (2023).
21. Almaiah, M. A., Ali, A., Hajjej, F., Pasha, M. F. & Alohali, M. A. A lightweight hybrid deep learning privacy preserving model for FC-based industrial internet of medical things. *Sensors (Basel)* **22**(6), 2112 (2022).
22. Zhang, Q., Fu, M., Zhao, Z. & Huang, Y. Searchable encryption over encrypted speech retrieval scheme in cloud storage. *J. Inf. Secur. Appl.* **76** https://doi.org/10.1016/j.jisa.2023.103542 (2023).
23. Alzubi, O. A., Alzubi, J. A., Shankar, K. & Gupta, D. Blockchain and artificial intelligence enabled privacy-preserving medical data transmission in internet of things. *Trans. Emerg. Telecommun. Technol.* **32**(12) https://doi.org/10.1002/ett.4360 (2021).
24. Guan, A., Liu, L., Fu, X. & Liu, L. Precision medical image hash retrieval by interpretability and feature fusion. *Comput. Methods Programs Biomed.* **222** https://doi.org/10.1016/j.cmpb.2022.106945 (2022).

25. Özbay, E. & Özbay, F. A. 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 (2023) .
26. Xu, Y., Zhao, X. & Gong, J. A Large-Scale Secure Image Retrieval Method in Cloud Environment. *IEEE Access* **7**, 160082–160090. https://doi.org/10.1109/ACCESS.2019.2951175 (2019).
27. Du, A., Wang, L., Cheng, S. & Ao, N. A privacy-protected image retrieval scheme for fast and secure image search. *Symmetry* **12**(2) https://doi.org/10.3390/sym12020282 (2020).
28. Cheng, S. L., Wang, L. J., Huang, G. & Du, A. Y. 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 (2021).
29. Janani, T. & Brindha, M. 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 (2022).
30. Zhu, D., Zhu, H., Wang, X., Lu, R. & Feng, D. An Accurate and Privacy-Preserving Retrieval Scheme Over Outsourced Medical Images. *IEEE Transactions on Services Computing* **16**(2), 913–926. https://doi.org/10.1109/TSC.2022.3149847 (2023).
31. Ma, X. et al. Understanding adversarial attacks on deep learning based medical image analysis systems. *Pattern Recognit.* **110** https://doi.org/10.1016/j.patcog.2020.107332 (2021) .
32. Yuan, X., Zhang, Z., Wang, X. & Wu, L. Semantic-aware adversarial training for reliable deep hashing retrieval. *IEEE Trans. Inf. Forensics Secur.* **18**, 4681–4694. https://doi.org/10.1109/TIFS.2023.3297791 (2023).
33. Tabatabaei, Z. et al. WWFedCBMIR: World-Wide Federated Content-Based Medical Image Retrieval. *Bioengineering* **10**(10) https://doi.org/10.3390/bioengineering10101144 (2023).
34. KhoKhar, F. A. et al. A review on federated learning towards image processing. *Comput. Electr. Eng.* **99**(February), 107818. https://doi.org/10.1016/j.compeleceng.2022.107818 (2022).
35. Kumbhare, S., Kathole, B. A. & Shinde, S. Federated learning aided breast cancer detection with intelligent heuristic-based deep learning framework. *Biomed. Signal Process. Control* **86**, 105080 (2023).
36. M, A., M, D., Amaithi Rajan, A., V, V. & D, H. EdgeShield: Attack resistant secure and privacy-aware remote sensing image retrieval system for military and geological applications using edge computing. Earth Science Informatics https://doi.org/10.1007/s12145-024-01256-z (2024).
37. Zhou, F., Qin, S., Hou, R. & Zhang, Z. Privacy-preserving image retrieval in a distributed environment. *Int. J. Intell. Syst.* **37**(10), 7478–7501. https://doi.org/10.1002/int.22890 (2022).
38. Smyrlis, M. et al. Rama: a risk assessment solution for healthcare organizations. *Int. J. Inf. Secur.* https://doi.org/10.1007/s10207-024-00820-4 (2024).
39. Chandramohan, D., Vengattaraman, T., Dhavachelvan, B. R. & Venkatachalapathy, V. S. K. Fewss - framework to evaluate the service suitability and privacy in a distributed web service environment. *Adv. Complex Syst.* **05**(01), 1350016 (2014).
40. Amaithi Rajan, A. & Vetrian, V. QMedShield: a novel quantum chaos-based image encryption scheme for secure medical image storage in the cloud. *J. Mod. Opt.*, 1–19 https://doi.org/10.1080/09500340.2024.2436521 (2024).
41. Madry, A., Makelov, A., Schmidt, L., Tsipras, D. & Vladu, A. Towards Deep Learning Models Resistant to Adversarial Attacks. arXiv:1706.06083 (2019).
42. Bai, L. A reliable (k, n) image secret sharing scheme. In: *Proceedings - 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing*, DASC 2006, pp. 31–36. https://doi.org/10.1109/DASC.2006.11 (2006).
43. El-Latif, A. A. A., Chelloug, S. A., Alabdulhafith, M. & Hammad, M. Accurate Detection of Alzheimer's Disease Using Lightweight Deep Learning Model on MRI Data. *Diagnostics* **13**(7) https://doi.org/10.3390/diagnostics13071216 (2023) .
44. Msoud Nickparvar: Brain Tumor MRI Dataset (2021).
45. Islam, M. N. et. Vision transformer and explainable transfer learning models for auto detection of kidney cyst, stone and tumor from CT-radiography. *Sci. Rep.* **12**(1) https://doi.org/10.1038/s41598-022-15634-4 (2022).
46. Wang, X., Lee, F. & Chen, Q. Similarity-preserving hashing based on deep neural networks for large-scale image retrieval. *J. Vis. Commun. Image Represent.* **61**, 260–271 (2019).
47. Cheng, Q. et al. A semantic-preserving deep hashing model for multi-label remote sensing image retrieval. *Remote Sens. (Basel)* **13**(24), 4965 (2021).
48. Yue, J., Li, Z., Liu, L. & Fu, Z. Content-based image retrieval using color and texture fused features. *Math. Comput. Model.* **54**(3–4), 1121–1127. https://doi.org/10.1016/j.mcm.2010.11.044 (2011).
49. Liu, H. et al. An improved deep hashing model for image retrieval with binary code similarities. *J. Big Data* **11**(1) https://doi.org/10.1186/s40537-024-00919-4(2024).
50. Shen, M., Deng, Y., Zhu, L., Du, X. & Guizani, N. Privacy-preserving image retrieval for medical iot systems: A blockchain-based approach. *IEEE Network* **33**(5), 27–33. https://doi.org/10.1109/MNET.001.1800503 (2019).

Author contributions

Conceptualization: Arun Amaithi Rajan, Vetriselvi V, Ajitesh M, Praveen Kumar R. **Methodology and Development:** Arun Amaithi Rajan, Vetriselvi V, Ajitesh M, Praveen Kumar R. **Formal analysis and investigation:** Arun Amaithi Rajan, Vetriselvi V. **Writing - original draft preparation:** Arun Amaithi Rajan, Praveen Kumar R, Ajitesh M. **Writing - review, and editing:** Arun Amaithi Rajan, Vetriselvi V, Ajitesh M, Praveen Kumar R. **Supervision:** Vetriselvi V.

Funding

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

Declarations

Competing interests

The authors declare no competing interests.

Research involving human and/or animals

Not applicable

Informed consent

Not applicable

Additional information

Correspondence and requests for materials should be addressed to A.A.R.