Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 44 No. 5 (2023)

Efficient & Secure similarity search for encrypted images over cloud

[1]*Komal. P. Ghorpade*, [2]*Dr. G. A. Patil*1

Abstract -- With the expansion of edge devices, the Content-Based Image Retrieval (CBIR) technique has attracted attention across sectors like distributed computing, peer-to-peer communication networks. Nevertheless, existing CBIR strategies aimed at image security and retrieval assistance exhibit certain inherent limitations including subpar search precision, restricted search capabilities, and susceptibility to key exposure. To address these concerns, our framework proposes a similar search methodology suited for secure distributed computing employing encrypted images. Data owners initially encrypt their images through keys generated by a shared key management tool. Subsequently, features are extracted via a pre-trained CNN model, ensuring security, while secure indices are constructed employing the K-Nearest Neighbors [KNN] algorithm to bolster search accuracy. The encrypted images are then transmitted to the cloud. This model guarantees heightened search precision and efficiency, simultaneously thwarting key compromise and enhancing overall image security.

Keywords: Content-Based Image Retrieval (CBIR), Deep learning

1. Introduction

With the proliferation of intelligent devices, the Content-Based Image Retrieval (CBIR) technique has captured interest across a range of domains, including cloud computing and social networking services. Nevertheless, existing CBIR strategies, while aiming to balance image confidentiality and retrieval, grapple with certain shortcomings such as compromised search precision, effectiveness, and potential key exposure. To address these concerns, we propose an analogous approach: Encrypted Image Search within secure cloud computing. Data proprietors encrypt their images using keys from a shared Key Management Tool. Security is reinforced by features extracted through pre-trained CNN models, and search accuracy is bolstered via K-Nearest Neighbors (KNN) algorithms constructing secure indices. The encrypted images are subsequently transmitted to the cloud, ensuring not only efficient and accurate searches but also mitigating key vulnerabilities and enhancing overall image security

2. Related Work

Content-Based Image Retrieval (CBIR) has significantly evolved as a focal point in interactive media applications [1]. Sumaiya and M. Armanuzzaman utilized visual cues for submerged images, enhancing perceptual clarity and cognitive aspects in image extraction for improved understanding [1]. Notably, the incorporation of a dim channel preceding "Deblurring" was applied to underwater images during preprocessing, resulting in the removal of haziness and enhancement of visual contrast. This approach led to improvements in real-time content extraction, capitalizing on visual saliency detection and dark image defogging procedures [1].

Effective management algorithms are essential for modern big data repositories. For vast image collections, robust algorithms for image comparison and similarity search are pivotal, especially considering geometric variations. The application of invariant Local Feature Detectors for image search in large repositories was explored in experimental research [2]. Image matching and retrieval represent fundamental challenges in computer vision, encompassing search, biometrics, and person re-identification. A consensus approach for harmonizing algorithms in image matching was presented by A. Barman and S. K. Shah. Their Shortest Hamiltonian Path Estimation (SHaPE) method [3] maps candidate ranking using scores onto graph theory, extended to incorporate diverse score sets from distinct algorithms.

This consensus problem is addressed through a two-step process involving a greedy algorithm and Ant Colony Optimization [3]. Locality Sensitive Hashing (LSH), commonly used for approximating high-dimensional operations, is insufficient for identifying image collections. An index structure called Bitmap-Image LSH (bImageLSH) [4], introduced by O. Jafari, P. Nagarkar, and J. Montaño, is tailored for high dimensional image processing, as evidenced by empirical evaluations on real datasets [4]. Artificial Intelligence

(AI) is a burgeoning field with extensive applications that continue to grow. Reverse image search presents an innovative utility by employing images as search queries, pioneered by A. B. Jivane [5], particularly in the fashion industry. Nonetheless, certain limitations persist. Issues include difficulties in semantic image understanding leading to retrieval inaccuracies, resource-intensive feature extraction, limited contextual awareness, inability to capture image relationships, and subjectivity in image interpretation [1].

3. Methodology

Deploying images directly to the public cloud raises immediate privacy apprehensions. In order to facilitate extensive image searches within real-world contexts, the enhancement of search efficiency is imperative. Addressing the aforementioned limitations necessitates the creation of a system that incorporates dynamic index key updates, multi-owner scenario support, and verification capabilities. Our proposition involves crafting an encrypted image retrieval system. This system aims to enable precise and efficient searches for similar images while concurrently safeguarding the privacy of both the image owner and the image itself from the cloud server. Following Fig 1 shows the systematic view of the proposed system

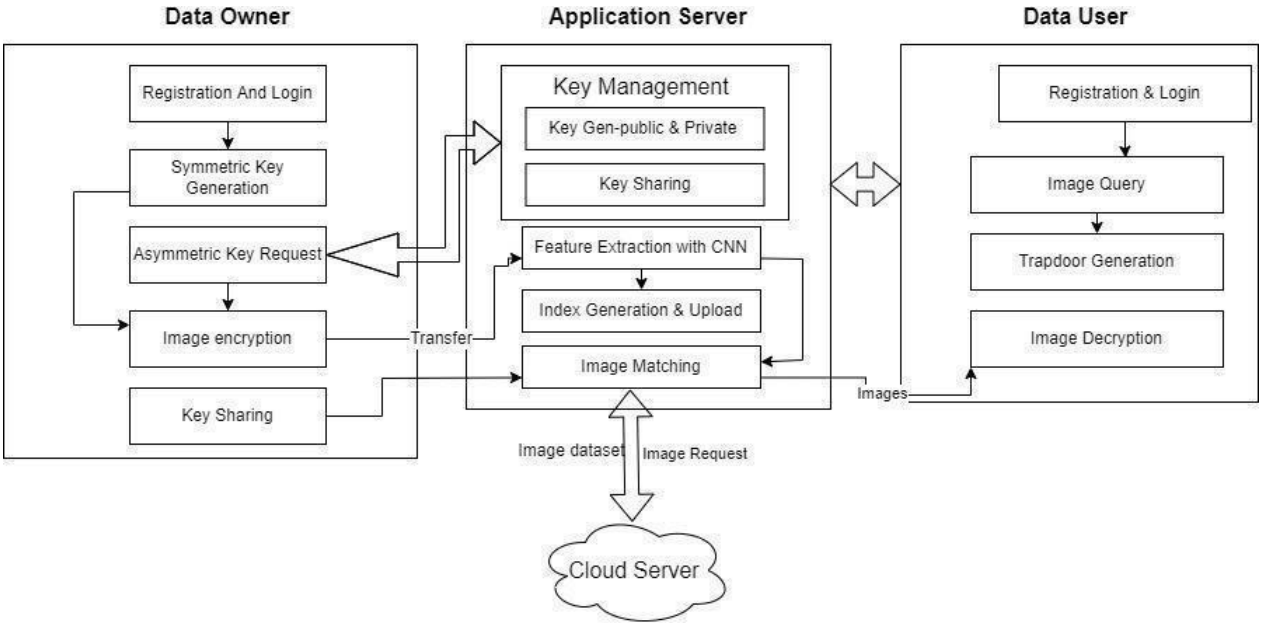

Fig 1: System Architecture for Proposed System

Module 3. 1: Client and Data Owner Processing module:

This module mainly concerns the processing of client and data owner identification. There are two end users i.e data owner and data users. Both have an application by which they register to the system and authenticate by application server.

Following are the features that are developed:

1. Registration of Data owner and data user through applications provided to them.
2. After the Login data owner generates a symmetric key on application. Image owner encrypts the image using the symmetric key AES which is of size in bits i.e., 256. A symmetric block cipher called AES is used for encryption.
3. Then the data owner requests a key to the application server.
4. Application servers use a key management tool that is responsible for creating and managing the private public keys for multiple owners.
5. The application server transmits the private key to the data owner. The private key is generated using parameters {r, S, M, π} within a secure parameter λ. Here, r and S denote random vectors specific to the image owner, while M and π represent random matrices and permutations respectively, also pertaining to the image owner.

6. Image owner uses this key to encrypt the image again.

$$C = Enc(M, kie)$$
$$= (Enc(m1, kie), \cdots, Enc(mn, kie))$$
$$= (c1, \cdots, cn),$$

Where,

C=Cipher Image set
M=Image Set
Kie= symmetric Key

7. Then the data owner sends the encrypted images to the application server.

Module 3.2: Application Server End Processing Module:

This module mainly concerns all the activities to be completed at the end of the application server. Here application server is responsible for user authentication and Key Handling.

Following are the features that are developed:

1.The authenticated data owner requests for the key.

2. Application server generates the private and public key pair for that user or can select the pair from existing key pairs already generated for different users randomly. And then send the private key to data owners.

3. Upon reception of an encrypted image set from the data owner, the Application server undertakes feature extraction via a pretrained CNN Model. These features encompass Local Binary Pattern (LBP) and Histogram of Oriented Gradients. Local features encompass identifiable patterns or distinctive structures within an image, including points, edges, or small image patches.

4. These features are further used to build the index tree for all images in which leaf nodes store cluster centers of encrypted images.

5. Upon deriving feature vectors from images M, the application server proceeds to determine the count K of clusters and the corresponding cluster centers C = {C1, C2, · ··, CK} through the utilization of the Affinity Propagation Algorithm (AP). Subsequently, both K and C are employed to initialize the K-means algorithm.

6. Eventually, the K-means algorithm yields a fresh collection of cluster centers C* {C1, C2, · ··, C*K}.

Subsequently, the index tree is constructed, and indexes for encrypted images are uploaded to the cloud server. Upon receiving the image set from the cloud server, the application server undertakes a top-down search for matched nodes within the hierarchical index tree.

This is achieved by successively locating the entry point for the subsequent level, which involves identifying the node with the vector having the minimal distance. Upon locating matched leaf nodes, candidate search outcomes are consolidated within the clusters associated with these nodes. The application server then computes the inner product values between the feature vector of each image within the clusters and that of the query image. Subsequent to accurately sorting these inner product values, identifiers for k encrypted images showcasing the highest relevance to query image Mq are determined. Ultimately, the application server transmits the most similar top k images corresponding to the query back to the data user.

Module 3.3: Data User Application Module:

This module mainly concerns data user application where data user will receive the key and processing at his side will be carried out. Data user who already has been authenticated by application server login to application server. Data users receive the symmetric key from the data owner.

Following are the features that are developed:

1. The user chooses the image and encrypts it with a symmetric key before asking the application server for a private key when they want to query a certain image's data to find comparable photos.

2. The data user creates the trapdoor T and delivers it to the application server after encrypting the data with the key after receiving it.

3. After receiving the trapdoor, the application server requests a cloud server for encrypted images of the particular organization. After receiving images, the data user decrypts the image first using a public key and then using the symmetric key given by the data owner.

$$R0 = Dec(R, kie)$$
$$= (Dec(c1, kie), \cdots, Dec(ck, kie))$$
$$= (m1, \cdots, mk).$$

Where,

Kie: Key

R: Ciphertext/plaintext of search results

R0: Encrypted image

C1: Cluster exemplars of AP clustering

4. Results Analysis

Accuracy

Accuracy is one measurement for assessing correctness of results in total related to output. Casually, exactness is the small portion of expectations our model got right. Officially, exactness has the accompanying definition:

Accuracy=Number of correct Predictions/Total No of predictions

We have uploaded a different set of images and we have tested the accuracy by providing a different set of query images to test the accuracy.

The accuracy is calculated with following formula:

$$\textbf{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Where,

FP = False Positives *TN = True Negatives*

TP = True Positives *FN = False Negatives*

Table 1. Experimental Results for Accuracy

Image Dataset (No of Images)	Search Accuracy of Image Data
50	66%
100	69%
200	82%
300	88%
500	94%

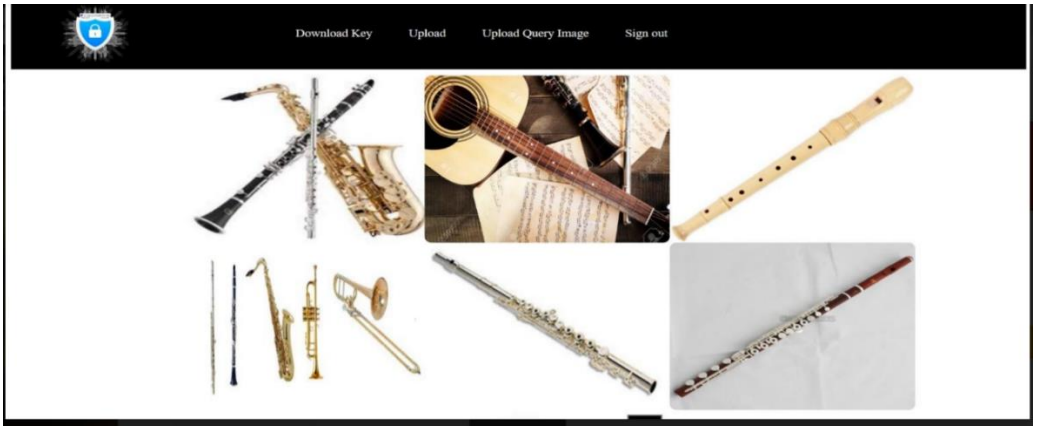

Fig 2. Final set of Images obtai

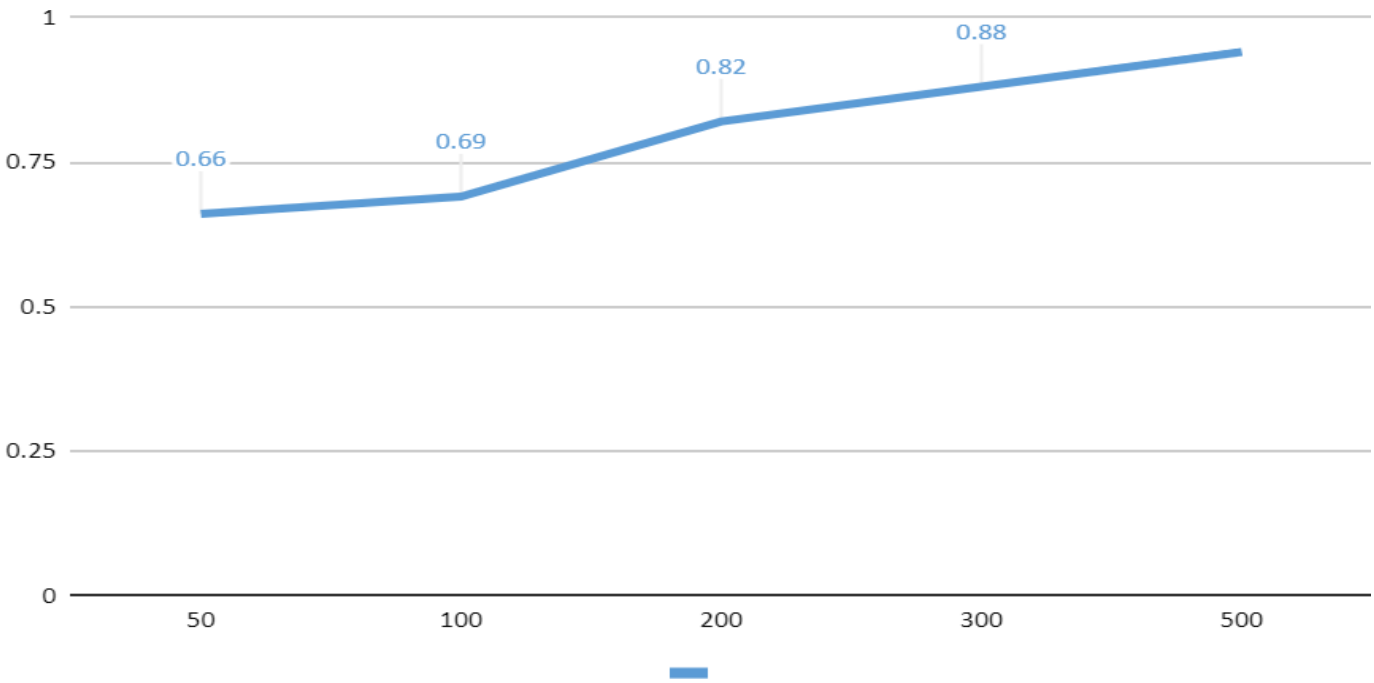

Fig 3. Image search Accuracy achieved against each Image dataset.

AWS Cloud, a prominent cloud computing platform, offers an extensive array of services that can be accessed when needed. Its offerings include scalable computing capabilities, storage options, and database solutions that foster innovation and expansion for enterprises. The presence of worldwide data centers facilitates rapid deployment of applications and services for businesses, complemented by a pay-as-you-go pricing structure that enhances cost-effectiveness. Consequently, AWS Cloud provides organizations the means to construct, launch, and oversee applications within an adaptable and expandable framework. Using the S3 storage service of AWS cloud achievement of encrypted images and indexing files has been successful.

5. Conclusion

Our goal was to create an encrypted image retrieval system that achieves two important objectives. Firstly, it enables image users to conduct precise and efficient searches for similar images. Secondly, it ensures the privacy and security of both image owners and image users, safeguarding their sensitive information from potential risks posed by the cloud server. The achievement is reflected through results. The results provide information on the "Image Dataset" and its corresponding "Search Accuracy" for different sizes of the dataset. The "Image Dataset" column represents the number of images in the dataset, while the "Accuracy" column represents the search accuracy on encrypted images, accuracy achieved by a specific system or model in analyzing and classifying these images.

As the number of images in the dataset increases, the "Search Accuracy" also improves, indicating that the system becomes more effective at correctly identifying and classifying the images. For example, with 50 images in the dataset, the accuracy is 0.66, which means the system correctly identifies around 66% of the images. However, as the dataset size increases to 500 images, the accuracy significantly improves to 0.94, indicating that the system can now accurately identify around 94% of the images in the dataset. The data in the table demonstrates a positive correlation between the size of the dataset and the accuracy of the system. As the dataset grows, the system becomes more capable of accurately processing and analyzing the images, resulting in higher accuracy rates.

The future scope of this encrypted image retrieval system lies in its potential for further improvement and expansion. By achieving precise and efficient image searches while maintaining privacy and security, the system can be developed to handle larger and more diverse image datasets. As evident from the results, increasing the dataset size enhances the search accuracy of encryption, indicating room for optimization and

refinement. To enhance its practical applications, the system could be fine-tuned and optimized to handle even larger datasets, improving its search accuracy and performance for real-world image retrieval tasks. Additionally, research and development efforts can focus on exploring novel encryption techniques, advanced feature extraction methods, and context-awareness to further enhance the system's capabilities and ensure it remains at the forefront of secure and efficient content-based image retrieval technology.

References

[1] Sumaiya and M. Armanuzzaman, "DURISE- Deblurring of Underwater Image Search Engine by CBIR,"2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 2021, pp. 1-5, Doi: 10.1109 /ICEEICT 53905.2021.9667835.

[2] K. Smelyakov, A. Chupryna, O. Ponomarenko and M. Kolisnyk, "Search by Image Engine using Local Feature Detectors," 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences(eStream),2020, pp.1-4, Doi:10.1109/eStream 50540.2020.9108884.

[3] A. Barman and S. K. Shah, "A Graph-Based Approach for Making Consensus-Based Decisions in Image Search and Person Re-Identification," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 3, pp. 753-765, 1 March 2021, Doi: 10.1109/TPAMI.2019.2944597.

[4] O. Jafari, P. Nagarkar and J. Montaño, "Efficient Bitmap-based Indexing and Retrieval of Similarity Search Image Queries," 2020 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI), 2020, pp. 58-61, Doi: 10.1109/SSIAI49293.2020.9094616.

[5] A. B. Jivane, "Time efficient privacy-preserving multi-keyword ranked search over encrypted cloud data," 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), 2017, pp. 497-503, Doi: 10.1109 /ICPCSI. 2017.8392345.

[6] P. R. Kumar, T. Arunprasath, M. P. Rajasekaran, and G. Vishnuvarthanan, ‘‘Computer-aided automated discrimination of Alzheimer’s disease and its clinical progression in magnetic resonance images using hybrid clustering and game theory-based classification strategies,’’ Comput. Electr. Eng., Vol. 72, pp. 283–295, Nov. 2018.

[7] A. V. Dastjerdi, H. Gupta, R. N. Calheiros, S. K. Ghosh, and R. Buyya, ‘Fog computing: Principles, architectures, and applications,’’ 2016, arXiv:1601.02752. [Online]. Available: http://arxiv.org/abs/1601.02752

[8] H.-J. Cha, H.-K. Yang, and Y.-J. Song, ‘‘A study on the design of fog computing architecture using sensor networks,’’ Sensors, vol. 18, no. 11, p. 3633, Oct. 2018.

[9] H. F. Atlam, R. J. Walters, and G. B. Wills, ‘‘Fog computing and the Internet of Things: A review,’’ Big Data Cogn. Comput., vol. 2, no. 10, pp. 1–18, Apr. 2018.

[10] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, ‘‘Fog computing and its role in the Internet of Things,’’ in Proc. 1st MCC Workshop Mobile Cloud Comput. - MCC, Aug.2012, pp. 13–15.

[11] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, ‘‘An overview of blockchain technology: Architecture, consensus, and future trends, ’in Proc. IEEE Int. Congr. Big Data (BigData Congress), Jun. 2017, pp. 557–564.

[12] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, ‘‘Mobile fog: A programming model for large-scale applications on the Internet of Things,’’ in Proc. 2nd ACM SIGCOMM Workshop Mobile Cloud Comput. - MCC, 2013, pp. 15–20.

[13] S. Yi, Z. Hao, Z. Qin, and Q. Li, ‘‘Fog computing: Platform and applications,’’ in Proc. 3rd IEEE Workshop Hot Topics Web Syst. Technology (HotWeb), Nov. 2015, pp. 73–78.

[14] N. Rifi, E. Rachkidi, N. Agoulmine, and N. C. Taher, ‘‘Towards using blockchain technology for eHealthdata access management,’’ in Proc. 4th Int. Conf. Adv. Biomed. Eng. (ICABME), Oct. 2017, pp. 1–4.

[15] M. Ahmad, M. B. Amin, S. Hussain, B. H. Kang, T. Cheong, and S. Lee, ‘‘Health fog: A novel framework for health and wellness applications,’’ J. Supercomput., vol. 72, no. 10, pp. 3677–3695, Oct. 2016.

[16] P. Verma and S. K. Sood, ‘‘Fog assisted-IoT enabled patient health monitoring in smart homes,’’ IEEE Internet Things J., vol. 5, no. 3, pp. 1789–1796, June. 2018.

[17] T. N. Gia, M. Jiang, A.-M. Rahmani, T. Westerlund, P. Liljeberg, and H. Tenhunen, ‘‘Fog computing in healthcare Internet of Things: A case study on ECG feature extraction,’’ in Proc. IEEE Int. Conf. Comput. Inf. Technol.; Ubiquitous Comput. Commun.; Dependable, Autonomic Secure Comput.; Pervas. Intell. Comput., Oct. 2015, pp. 1

[18] B. Negash, A. Anzanpour, I. Azimi, M. Jiang, T. Westerland, A. M. Rahmani, P. Liljeberg, and H. Tenhunen, ‘‘Leveraging fog computing for healthcare IoT,’’ in Fog computing in the Internet of Things Intelligence at the edge. Cham, Switzerland: Springer, 2017, pp. 145–169.

[19] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, ‘‘Exploiting smart e-health gateways at the edge of healthcare Internet-of-Things: A fog computing approach,’’ FutureGener. Comput. Syst., vol. 78, pp. 641–658, Jan. 2018

[20] Azimi, A. Anzanpour, A. M. Rahmani, T. Pahikkala, M. Levorato, P. Liljeberg, and N. Dutt, ‘‘HiCH: Hierarchical fog-assisted computing architecture for healthcare IoT,’’ ACM Trans. Embedded Comput. Syst., vol. 16, no. 5s, pp. 1–20, Oct. 2017.