SIMILARITY SEARCH FOR ENCRYPTED IMAGES IN SECURE CLOUD COMPUTING

[1] RAYALLA BALAKRISHNA, [2] MR.K. JAYAKRISHNA

[1] PG Scholar in the department of MCA at QIS College of Engineering & Technology, Vengamukkapalem, Ongole, AP, India.

[2] Associate Professor in the department of MCA at QIS College of Engineering & Technology, Vengamukkapalem, Ongole, AP, India.

ABSTRACT

With the emergence of intelligent terminals, the Content-Based Image Retrieval (CBIR) technique has attracted much attention from many areas (i.e., cloud computing, social networking services, etc.). Although existing privacy-preserving CBIR scheme scan guarantee image privacy while supporting image retrieval, these schemes still have inherent defects (i.e., low search accuracy, low search efficiency, key leakage, etc.). To address these challenging issues, in this paper we provide a similarity Search for Encrypted Images in secure cloud computing (called SEI). First, the feature descriptors extracted by the Convolutional Neural Network (CNN)model are used to improve search accuracy. Next, an encrypted hierarchical index tree by using K-means clustering based on Affinity Propagation (AP) clustering is devised, which can improve search efficiency. Then, a limited key-leakage k-Nearest Neighbor (KNN) algorithm is proposed to protect key from being completely leaked to untrusted image users. Finally, SEI is extended to further prevent image users' search information from being exposed to the cloud server. Our formal security analysis proves that SEI can protect image privacy as well as key privacy. Our empirical experiments using a real-world dataset illustrate the higher search accuracy and efficiency of SEI.

Keywords: content, based, image, retrieval, CNN, security analysis

INTRODUCTION

WITH the rapid development and popularization of cloud computing, people enjoy various conveniences brought by cloud services, such as storing images on the cloud. However, directly outsourcing images to the public cloud inevitably raises privacy concerns. Once the massive images (e.g., patients' medical images) containing highly sensitive information have been leaked to unauthorize densities, it will incur serious consequences or unnecessary trouble. The encryption mechanism can alleviate image data security and privacy concerns to some extent, but it invalidates the Content-Based Image Retrieval (CBIR) technique over ciphertext, and even causes other concerns discussed in the following example.

Example. Alice outsources the encrypted images Coif the local image database M to the cloud server. The authenticated Bob generates an encrypted search request T according to query image mq by using the searchable key sk shared by Alice when he queries images similar to mq. Then, the cloud server searches C after receiving T and returns relevant search results R to Bob. Finally, Bob decrypts the R with image encryption key kie from Alice n this encrypted image search process, the cloud server is regarded as semi-trusted. Alice stores her images on this cloud server, and entrusts the cloud server to perform similarity search tasks. When Bob comes to query Alice's images, the cloud server will honestly provide Bob with search service under Alice's arrangement. The performance of search services, such as search accuracy and efficiency, will profoundly affect Bob's search experience. Assuming that Bob is a doctor who relies on search results to diagnose certain patient's condition, the incorrect search results will lead to a wrong diagnose, which endangers the patient's health and even life. Besides, the time-consuming search process will prolong the waiting time of the Bob. If Bob isa mobile image user who requires high real-time responses, such a long search time is hard to bear and easily makes search time lose timeliness. Moreover, the above search mechanism still requires Alice to shares k and kie with Bob, which cannot completely protect image privacy. This is because we cannot promise that Bob is fully trusted and does not share k and kie with other unauthorized image users due to interest incentives in practice.

Fortunately, various schemes related to privacy-preserving CBIR have been studied, like. However, n practice, these schemes still face many challenges (i.e., low search accuracy, low search efficiency, key leakage, etc.). Specifically, schemes directly distributed keys to users, leading to the risk of image users leaking keys, schemes sacrificed accuracy to improve efficiency, and schemes brought a lot of overhead to achieve high security. These challenges limiting their practical applications are shown.

As for the first challenge indicated by there are two primary reasons affecting the search accuracy. One is the feature descriptors type; the other is the similarity calculation method. As for the former, image feature descriptors are mainly classified into global features (e.g., Local Binary Pattern (LBP), Histogram of Oriented Gradients (HOG), etc.) and local features (e.g., Scale Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF), etc.). The local features can achieve higher search accuracy than global features due to the better robustness. There are also works [9],[10] that combine global features and local features with certain weights to form new features or apply Convolutional Neural Network (CNN) model mimicking human visual cognition to extract feature vectors, which achieve acceptable accuracy. For the later, the similarity between images is measured by Euclidean distance, Cosine distance, Hamming distance, and Jaccard similarity coefficient. Especially, Asymmetric Scalar-product-Preserving Encryption (ASPE) algorithm using random numbers and matrices to encrypt feature vectors can calculate the Euclidean distance of high-dimension space more accurately. At the same time, other works, using Secure Multi-party Computation (SMC), Homomorphic Encryption (HE) to calculate Euclidean distance can also improve search accuracy, but result in inefficiency due to complex garbled circuit operations.

In order to support the large-scale image search in actual applications, it is indispensable to improve the search efficiency. As far as we know, a feasible search scheme can be achieved by constructing the efficient linear index, inverted index, hash table or tree index. In particular, the hash table and tree index perform better than the other two regarding search efficiency. For example, the two-level hash table constructed based on the Locality Sensitive Hashing (LSH) algorithm greatly speeds up the search process by reducing the dimension of the high-dimension feature vectors. For tree index, scheme [10] classifies images by using K-means clustering algorithm and then builds the index tree based on clustering results, which narrows the search scope and avoids traversing the entire image database during the search phase.

Although some prevalent schemes based on ASPE and index tree achieve high search efficiency without deter-rating search accuracy, they do not solve the key leakage challenge, the image owner should share k and kie with all authorized image users. However, it is impractical and overloaded to establish a secure channel to transmit these keys. To avoid key transmission, the latest scheme focusing on improving the ASPE algorithm can guarantee that’s k used to encrypt vectors will not be completely disclosed to any image user. Unfortunately, it does not support image search. Although some image search solutions can avoid leaking k, these solutions still have low search accuracy and search efficiency due to the use of homomorphic encryption and secure multi-party computation

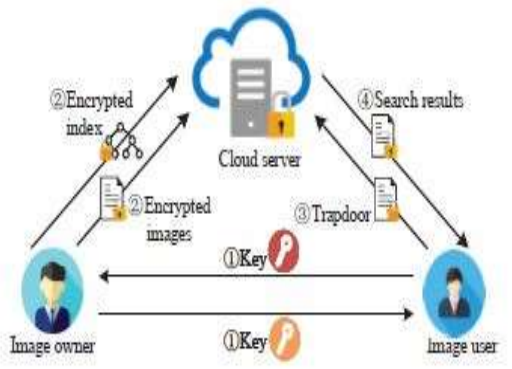
As far as we know, no existing work is dedicated to solving the above three challenges simultaneously. Hence, in this paper we propose a similarity Search for Encrypted Images in secure cloud computing (SEI) to solve the above challenges. Specifically, we employ the CNN model to extract feature vectors to improve search accuracy, and then build a hierarchical index tree in a bottom-up manner based on the clustering algorithm to improve search efficiency. Besides, we design an optimized ASPE algorithm, which does not require the image owner to shares k with image users, to achieve limited key-leakage for untrusted image users. To summarize, our contributions are set out as follows.

High search accuracy. SEI achieves a high search ac-curacy by using the pre-trained CNN model to extract feature vectors. The CNN model simulating the human visual perception process can more accurately represent the image content, which makes the similarity measurement between images more accurate and search results more accurate

High search efficiency. SEI uses the optimized K-means clustering algorithm to classify images and con-structs a hierarchical index tree in a bottom-up manner based on the clustering results. Therefore, SEI avoids traversing the entire image database when performing search operation and reduces the search time to sublinear time. Moreover, the Principal Component Analysis (PCA) algorithm is utilized to lower the dimensions of the image feature vectors extracted by CNN model, which further improves search efficiency

Limited key leakage. SEI provides a secure trapdoor generation process with limited key-leakage1, which not only prevents untrusted image users from completely leaking keys privacy but also avoids the online help of the image owner when the image user generates trapdoors locally. Moreover, even if any un-authenticated image user illegally obtains partial information about searchable keys from a certain untrusted user, he cannot generate a valid query trapdoor or decrypt the ciphertext of the image owner.

SYSTEM ARCHITECTURE:



METHODOLOGY

Local Feature based Scheme:

For local feature-based scheme, a set of local features are extracted from local regions of an image. One popular approach of content-based image retrieval using local features is called bag-of-words model. In this model, local features are extracted from all images in the database and jointly clustered. The cluster centers are used as cluster identifiers, which form the vocabulary. After clustering, only the identifier of the most similar cluster center is kept for each local feature, which signifies a word. Then the cluster occurrence histogram is created for each image, which can be represented as a vector of occurrence counts of the local features. Through this way, an image can be expressed as a bag of words in a visual vocabulary. By comparing the similarity of histograms, one can retrieve similar images. As mentioned before, EMD distance is used during the search process. Denote the signature of each image in the database as $S_t = \left\{\left(s_1^{(t)}, w_1^{(t)}\right), \left(s_2^{(t)}, w_2^{(t)}\right), \dots, \left(s_m^{(t)}, w_m^{(t)}\right)\right\}$ for $t = [n]$. Let $S_q = \left\{\left(s_1^{(q)}, w_1^{(q)}\right), \left(s_2^{(q)}, w_2^{(q)}\right), \dots, \left(s_m^{(q)}, w_m^{(q)}\right)\right\}$ be the signature of a query image. The EMD distance can be converted to an LP optimization problem as follows:

minimize $\sum_{i=1}^{m_t} \sum_{j=1}^{m_q} f_{i,j} d_{i,j}$

$$f_{i,j} \geq 0, 1 \leq i \leq m_t, 1 \leq j \leq m_q;$$

$$\sum_{i=1}^{m_t} f_{i,j} = w_j^{(q)}, 1 \leq j \leq m_q;$$

subject to $\sum_{j=1}^{m_q} f_{i,j} = w_i^{(t)}, 1 \leq i \leq m_t;$

$$\sum_{i=1}^{m_t} \sum_{j=1}^{m_q} f_{i,j} = 1.$$

Similarly, we first need to build a pre-filter table for sub-linear searching time. In the EMD case, we need to filter the images using a quick estimation of EMD without computing it. We utilize an easy-to-compute EMD lower bound as the estimation. The Euclidean distance between the centroids of two signatures with the same total weights is a lower bound on the EMD between them. It is well known that

$$EMD(S_t, S_q) = \sum_{i=1}^{m_t} \sum_{j=1}^{m_q} f_{i,j}{}^{*} d_{i,j} \geq \left\| \sum_{i=1}^{m_t} s_i{}^{(t)} w_i{}^{(t)} - \sum_{j=1}^{m_q} s_j{}^{(q)} w_j{}^{(q)} \right\|_2$$

Similar to the global feature case, we apply one-way LSH functions on centroid database ε, then groups the signature ID set on all λ hash tables. After that, the data owner outsources the secure searchable index, encrypted signature database and encrypted image database to the cloud server. When image query comes, the data user first generates the signature Q, then compute the centroid ξ_q. After that, he applies λ LSH functions and one-way function to ξ_q to compute the trapdoor $T_{\xi_q} = \emptyset\left(K_j, H_j\left(\xi_q\right)\right), j = 1, \dots, \lambda$ and sends it to the cloud server. The cloud server retrieves the corresponding encrypted signatures and sends them to the data user. By building the secure searchable index, we can achieve the sub-linear filtering complexity for the large-scale image database. In the second stage, we will directly compute the underlying distance metric, which is EMD, to compare the similarities of different images. We will explain in the next section.

LP Transformation:

The local features always involve in the more complex distance metrics, which need more delicate security design. In the second stage, after receiving the encrypted signatures from the cloud in the first stage, the data user decrypts the signatures and computes the distance C_{ij} between the cluster s_i , q_j. Then he formulates the EMD optimization problem as in Eq. (2). The data user wants to leverage the computation power of cloud server to compute EMD with privacy protection. Next, we will explain how to make the cloud server securely compare the EMD distance between different images without revealing the sensitive information. For more concise explanation, we use the matrix expression for Eq. (2):

$$\begin{aligned} &\text{Minimize } c^T x \\ &\text{Subject to } Ux = \tau \\ &\qquad Vx \le E \qquad (2) \end{aligned}$$

where **c** is an ab×1 distance vector, **x** is an ab×1 flow vector. **U** is an 1×ab known structure matrix and **V** is an $(a+b) \times ab$ known structure matrix. τ is the minimal total weight and $E = \left(w_{s_i}, w_{q_j}\right)$ is the weight vector, where $1 \le i \le a, 1 \le j \le b$. We denote the problem (2) by a tuple $\Psi = (c, U, \tau, V, E)$. Our design goal is to find a secure and efficient transformation, using a secret key K_r, to make cloud compute the EMD optimization problem while protecting the input and output privacy. It means that the cloud solves the randomly transformed EMD optimization problem without knowing the input distance vector ***c***, weight vector **E** and the output flow vector **x**. After that, we also want cloud to compute the EMD distance and sort the results. Our secure transformation contains two main steps. First of all, to protect the output privacy, we perform the affine mapping that $\boldsymbol{x} = \text{y}\Lambda - \gamma$, where $\boldsymbol{\Lambda}$ is an $ab \times ab$ non-singular matrix and $\boldsymbol{\gamma}$ is an $ab \times 1$ vector. The original problem is transformed to:

$$\begin{aligned} &\text{minimize } \boldsymbol{c^T}\boldsymbol{\Lambda}\mathbf{y} - \boldsymbol{c^T}\gamma \\ &\text{subject to } \boldsymbol{U}\boldsymbol{\Lambda}\mathbf{y} = \tau + \mathbf{U}\gamma, \\ &\qquad \boldsymbol{V}\boldsymbol{\Lambda}\mathbf{y} \le \mathbf{E} + \mathbf{V}\gamma \qquad (3) \end{aligned}$$

Next, we multiply the $((a+b) \times (a+b))$ generalizes permutation **G** to the inequality constraints to protect **E** and multiply a real positive value r to protect optimal value. We transform the problem (3) to another problem:

$$\begin{aligned} &\text{minimize } r\mathbf{c}^T\Lambda\mathbf{y} - r\mathbf{c}^T\gamma \\ &\text{subject to } \mathbf{U}\Lambda\mathbf{y} = \tau + \mathbf{U}\gamma, \qquad (4) \\ &\qquad \mathbf{G}\mathbf{V}\Lambda\mathbf{y} \le \mathbf{G}(\mathbf{E} + \mathbf{V}\gamma) \end{aligned}$$

Because the constant term $rc^T\gamma$ does not affect the optimal solution, the final transformation problem can be formed as:

$$\begin{aligned} &\text{minimize } \mathbf{c}'^T\mathbf{y} \\ &\text{subject to } \mathbf{U}'\mathbf{y} = \tau', \qquad (5) \\ &\qquad \mathbf{V}'\mathbf{y} \le \mathbf{E}' \end{aligned}$$

Where $c'^T = rc^T\mathbf{\Lambda}, U' = U\mathbf{\Lambda}, \tau' = \tau + U\gamma,\ V' = GV\mathbf{\Lambda}, E' = G(E + V\gamma)$. This problem has similar structure to the original problem (2). We use $\varphi_{K_r} = (c', U', \tau', V', E')$ to denote the secure transformed problem, where the secret transformation key $K_r = (G, \mathbf{\Lambda}, \gamma, \text{r})$. The whole transformation process is illustrated in Algorithm 1. After solving the EMD optimization problem, we also want cloud to compute EMD as in Eq. (1) and sort the results. Observe that the numerator of EMD equation is the optimal values of the original problem (2) and the denominator is the sum of the elements of optimal solution.

However, the cloud server solves the transformed problem (5) instead of the original one for security reason. The difference of optimal value between original problem and transformed problem is a constant term $rc^T\gamma$, which can be computed by data users before outsourcing. We also want the sum of elements of optimal solutions between original problem transformed problem differs in a constant term after the affine mapping $x = \Lambda y - \gamma$. So, we apply one additional constraint when constructing Λ such that for Λ^{-1} each column's sum is one. Then we can derive that $\sum_{i=1}^{a}\sum_{j=1}^{b} x_{ij} = \sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij} - \sum_{k=1}^{ab} \gamma_k$. The data user can compute $\sum_{k=1}^{ab} \gamma_k$ before outsourcing. When outsourcing the secure transformed problems to cloud server, the data users also send two offset constants $rc^T\gamma$ and $\sum_{k=1}^{ab} \gamma_k$ to cloud server. Then the cloud server can solve the transformed problems and use these constants to compute the right EMD in an order preserving way. The offset constants themselves will reveal little information. After that, the cloud server sorts the results and returns the more accurate ranked order encrypted images.

Global Feature vs. Local Feature

Under our two stages similarity image search system, the global and local feature-based solutions have certain differences. The global feature-based solution only needs one round communication between data user and cloud server to retrieve the ranked order encrypted images, while local feature based solution needs two rounds. There are two reasons. The first reason is that the underlying distance metric for local feature based solution, which is EMD, has more complicated problem structure and needs more delicate security design. Our encryption method should preserve the LP problem's structure and make sure the cloud server can solve the right optimal solution. We need to compute the distance vector c_{zq}^T for each retrieved and querying signature pair first, then encrypt this distance vector using transformation key Λ,r . We cannot outsource these two operations to cloud server at the same time without revealing the distance vector and the transformation key. The second reason is that there is a trade-off between communication round and security. For one round solution, every feature vector encrypted use the same invertible matrix W. It is vulnerable to know plaintext attack, when cloud server has certain number of pairs of plaintexts and ciphertext feature vectors, it can derive the invertible matrix W. For two rounds solution, the data owner adopts the standard encryption techniques, which can resist the known plaintext attack. So, we favor the two rounds design for our system, which also has higher scalability for different traditional and underlying distance metrics.

User:

In this application user is a module, here user should register with the application then login. After user successful login he can perform some operations, such as generate Key, search Images, request Status and logout

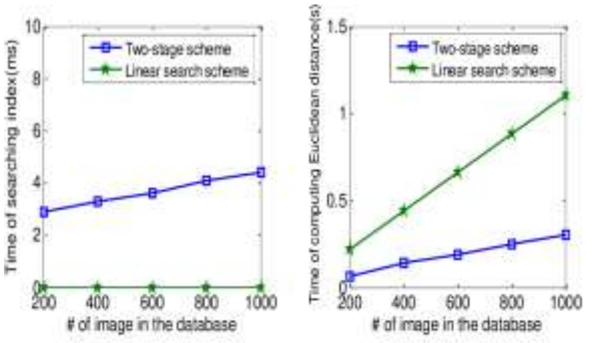
Owner:

In this application user is a module, here user should register with the application then login. After user successful login he can perform some operations, such as generate Key, upload Image, view Request and logout.

Cloud:

Here cloud can directly login with the application and after cloud successful login he can perform some operations such as view all images and view all users requests and logout.

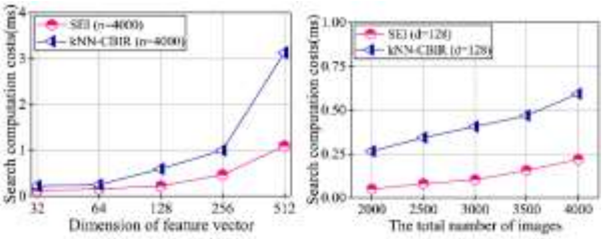
OUTPUTS



(a) Time of Searching Index

(b) Time of Computing Euclidean Distance

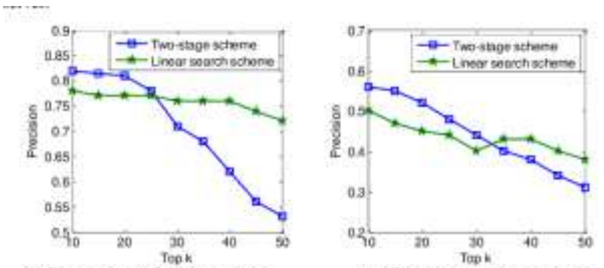
Fig 1: Search Efficiency of Global Feature Scheme.



(a) *Search* computation costs.

(b) *Search* computation costs.

Fig 2: Search Computation Costs.



(a) Precision of Global feature scheme

(b) Precision of Local feature scheme

Fig 3: Search Precision.

CONCLUSION

In this paper, we investigate similarity search for encrypted images in secure cloud computing. Concretely, we intro-duce a clustering improvement method and give the design method of the hierarchical index tree. With these two techniques, SEI can efficiently perform the retrieval process and achieve high accuracy based on features extracted by the CNN model. Further, we consider un trusted image users in SEI and hence propose a similarity calculation method with limited key-leakage. We also give strict security analysis and conduct experiments on a real-world dataset, which indicate that SEI is secure and feasible in practice.

FUTURE ENHANCEMENT

Efficiency Optimization: Explore techniques like selective encryption or partial feature indexing to further reduce computational overhead during the search process, especially for very large datasets.

Advanced Searchable Encryption: Investigate the integration of emerging SE schemes that offer additional functionalities like multi-keyword search or fuzzy search, allowing for more complex user queries.

Privacy Amplification: Implement techniques like random masking or homomorphic encryption to further strengthen privacy guarantees against potential key leakage scenarios.

Dynamic Data Management: Develop mechanisms for efficient insertion, deletion, and update of image data while maintaining the integrity and searchability of the encrypted index structure.

Scalability Evaluation: Conduct extensive scalability tests on even larger and more diverse image datasets to assess the system's performance under high load.

REFERENCE

1. Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” International Journal of Computer Vision, vol. 40, no. 2, (2000), pp. 99–121.
2. 2.H. Ling and K. Okada, “An efficient earth mover’s distance algorithm for robust histogram comparison,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 5, (2007), pp. 840–853.
3. E. Levina and P. Bickel, “The earth mover’s distance is the mallows distance: some insights from statistics”, in Computer Vision, 2001, ICCV 2001, Proceedings, Eighth IEEE International Conference on, vol. 2. IEEE, (2001), pp. 251–256.
4. 4.P. Indyk and R. Motwani, “Approximate nearest neighbors: towards removing the curse of dimensionality”, in Proceedings of the thirtieth annual ACM symposium on Theory of computing, ACM, (1998), pp. 604–613.

A. Gionis, P. Indyk and R. Motwani, “Similarity search in high dimensions via hashing”, in VLDB, vol. 99, (1999), pp. 518–529.

5. M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, “Locality-sensitive hashing scheme based on pstable distributions”, in Proceedings of the twentieth annual symposium on Computational geometry. ACM, (2004), pp. 253–262.

6. 7.A. Rajaraman and J. D. Ullman, “Mining of massive datasets”, Cambridge University Press, (2011).

7. Andoni and P. Indyk, “Near-optimal hashing algorithms for approximate nearest neighbour in high dimensions”, in Foundations of Computer Science, 2006. FOCS’06. 47th Annual IEEE Symposium on. IEEE, (2006), pp. 459–468.

8. D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data”, in Security and Privacy, 2000, S&P 2000, Proceedings, 2000 IEEE Symposium on. IEEE, (2000), pp. 44–55.

9. E.-J. Goh, “Secure indexes”, IACR Cryptology enprint Archive, vol. 2003, (2003), p. 216.

AUTHOR PROFILE

Mr. K. Jaya Krishna, currently working as an Associate Professor in the Department of Master of Computer Applications, QIS College of Engineering and Technology, Ongole, Andhra Pradesh. He did his MCA from Anna University, Chennai, M.Tech (CSE) from JNTUK, Kakinada. He published more than 10 research papers in reputed peer reviewed Scopus indexed journals. He also attended and presented research papers in different national and international journals and the proceedings were indexed IEEE. His area of interest is Machine Learning, Artificial intelligence, Cloud Computing and Programming Languages.

Mr. Rayalla Bala Krishna, currently pursuing Master of Computer Applications at QIS College of engineering and Technology (Autonomous), Ongole, Andhra Pradesh. He Completed B.Sc. in Computer Science from Sri Nagarjuna Degree College, Ongole, Andhra Pradesh. His areas of interests are Cloud Computing & Machine learning.