ImageProof: Enabling Authentication for Large-Scale Image Retrieval

Shangwei Guo[†], Jianliang Xu[†], Ce Zhang[†], Cheng Xu[†], and Tao Xiang[‡]

[†]*Department of Computer Science, Hong Kong Baptist University, Hong Kong, China*

[‡]*College of Computer Science, Chongqing University, Chongqing, China*

{csswguo,xujl,cezhang,chengxu}@comp.hkbu.edu.hk, txiang@cqu.edu.cn

Problem Statement

- **Outsourced Content-Based Image Retrieval**
  - The image owner outsources its image retrieval system to a third-party service provider (SP).
  - SIFT-based image retrieval: bag-of-visual-words (BoVW) encoding and inverted index search.
  - Top-k query and involved indexes: randomized k-d tree and impact-ordered inverted index.
- **Threat Model**
  - The SP could return incorrect search results (e.g., faked or low-ranked images).
  - ✔Soundness: The results must be the outsourced images which have not been tampered with.
  - ✔Completeness: The results include the k most similar images, i.e., the similarity values of the other images are smaller than those of the returned images.
- **Challenges**
  - Designing a query authentication scheme for a large, complex retrieval system is a big challenge in itself.
  - The client usually has only limited storage, communication, and computation resources.

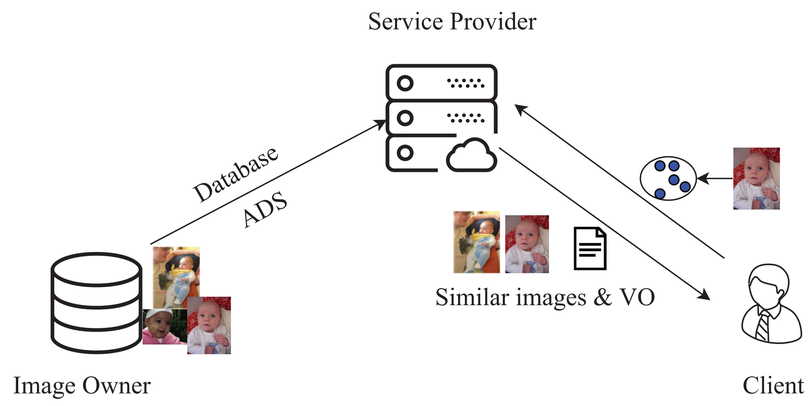

Fig. 1: Architecture of the proposed authenticated image retrieval system.

Preliminaries

- **Merkle Hash Tree**
  - An authenticated binary tree, enabling users to verify individual data objects without retrieving the entire database.
- **Cuckoo Filter**
  - A data structure supporting approximate set membership tests.

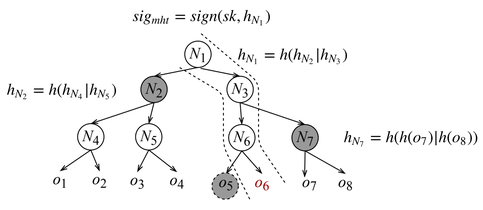

Fig. 2: An example of a Merkle hash tree.

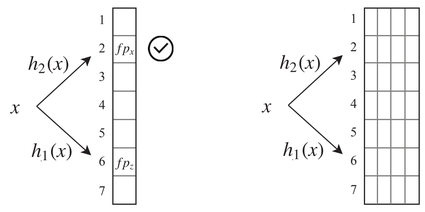

Fig. 3: A cuckoo filter, two hash values per item.

Merkle Randomlized k-d Tree (MRKD-tree)

- **Authenticated Data Structure (ADS)**
  - An internal node has three components, i.e., the splitting hyperplane, the pointers pointing to its child nodes, and a digest.
  - A leaf node records a certain number of clusters, the digests of the corresponding inverted lists, and a digest of itself.
- **Authenticated Query Processing**
  - Find the leaf nodes whose (minimum) distances to the feature vectors are shorter than the given thresholds.
  - Generate a single verification object (VO) for all feature vectors by maximizing the use of shared tree nodes.

Merkle Inverted Index With Cuckoo Filters

- **ADS**
  - Each Merkle inverted list Γ_{c_i} consists of the associated cluster c_i, cluster weight w_{c_i}, a posting list, cuckoo filter Θ_i, and digest $h_{\Gamma_{c_i}}$.

c_i	$h_{\Gamma_{c_i}}$	w_{c_i}	Θ_i	Posting Lists
c_5	$h(2\sqrt{2}\|h(\Theta_{c_5})\|h_{pos_{5,1}})$	$2\sqrt{2}$	Θ_{c_5}	$\mapsto \langle 1, 0.34, h_{pos_{5,1}}\rangle \langle 3, 0.26, h_{pos_{5,2}}\rangle \langle 4, 0.25, h_{pos_{5,3}}\rangle \langle 10, 0.17, h_{pos_{5,4}}\rangle$ ...
c_6	$h(\sqrt{2}\|h(\Theta_{c_6})\|h_{pos_{6,1}})$	$\sqrt{2}$	Θ_{c_6}	$\mapsto \langle 5, 0.41, h_{pos_{6,1}}\rangle \langle 8, 0.32, h_{pos_{6,2}}\rangle \langle 3, 0.28, h_{pos_{6,3}}\rangle \langle 6, 0.25, h_{pos_{6,4}}\rangle$ ...

- **Authenticated Query Processing**
  - Find top-k most similar images and generate the VO of inverted index search.
  - Ensure the integrity of top-k search with fewer postings with the help of cuckoo filters.
- **Main Idea**
  - Termination conditions:
    1. $s_k^L \geq \pi^U$, where s_k^L is the lower bound of the similarity score of the k-th most similar image and π^U is the upper bound of the similarity scores of the images not popped;
    2. $s_k^L \geq S^U(Q, I)$, where $S^U(Q, I)$ is the upper bound of the similarity scores of the images popped.
  - Take advantage of the cuckoo filters and estimate whether an image I is in a posting list with a high probability.
  - Minimize π^U and $S^U(Q, I)$.

ImageProof

- **ADS Generation**
  - Sign each image with a signature of the image ID and its raw data;
  - Invoke the same index as those in a normal SIFT-based image retrieval system;
  - Build Merkle inverted lists $\{\Gamma_{c_i}\}$;
  - Construct MRKD-trees $\{\mathcal{T}_i\}$;
  - Generate the hash of the digests of the root nodes;
  - Publish its public key and send the database and ADSs to the SP.

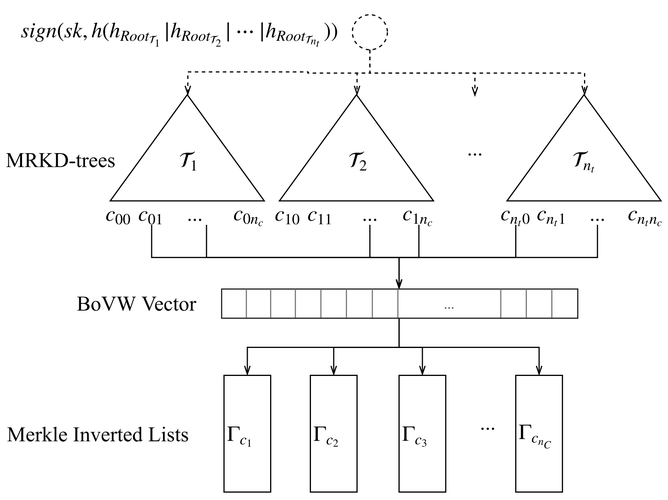

Fig. 4: An overview of ADSs for ImageProof.

- **Authenticated Query Processing**
  - Search the approximate nearest neighbors and generate the VO for the BoVW encoding;
  - Search the top-k images and generate the VO for the inverted index search;
  - Combine the VOs and the corresponding image signatures as the final VO, and send it, together with the top-k results, to the client.
- **Result Verification**
  - Check the correctness of the termination conditions and compute the digests of the posting lists;
  - Verify the integrity of the BoVW encoding and the MRKD-trees;
  - Verify the integrity of raw image data.

Optimization

- **Compressing Nearest Neighbor Candidates**
  - Drawbacks: To verify the integrity of the BoVW encoding, the client needs to check the correctness of the nearest neighbor among all the candidates.
  - Optimization: Return some partial dimensions of a cluster which are enough to prove whether the cluster is the nearest neighbor among all candidates
- **Frequency-Grouped Inverted Index**
  - Drawbacks: Most frequency counts are small and images with the same frequency count can be combined into a prefix component.
  - Optimization: Use a frequency-grouped inverted index as the underlying structure to improve the performance of ImageProof.

Experiment Results

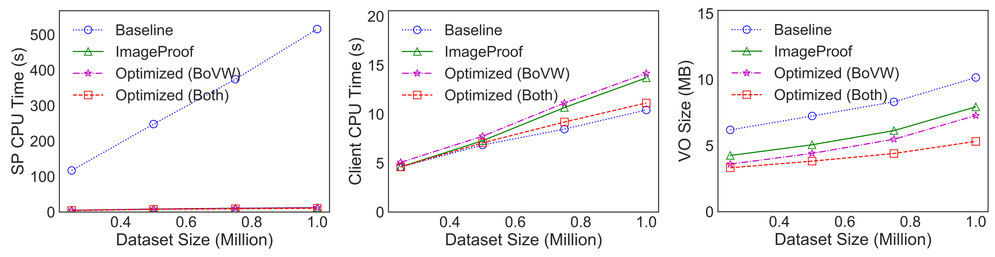

Fig. 5: Overall performance as dataset size increases