

Image Recognition and Indexing

Chat Fai Geoffrey Mak

Department of Electrical Engineering, Stanford University

Motivation

This work aims at using extracting sift descriptors from an image database, and building an inverted file system to index the images on a server.

An iPhone is used to scan these images, extract sift descriptors, and query the server to retrieve the nearest image (in terms of descriptor space) from the database.

Related Work

[1] Hervé Jégou, Matthijs Douze, Cordelia Schmid
Hamming Embedding and Weak Geometry Consistency for Large Scale Image Search,
10th European Conference on Computer Vision (ECCV '08) 5302 (2008) 304--317

[2] Wassim Bouachir, Mutapha Kardouchi, Nabil Belacel
Improving Bag of Visual Words Image Retrieval: A Fuzzy Weighting Scheme for Efficient Indexation,
2009 Fifth International Conference on Signal Image Technology and Internet Based Systems

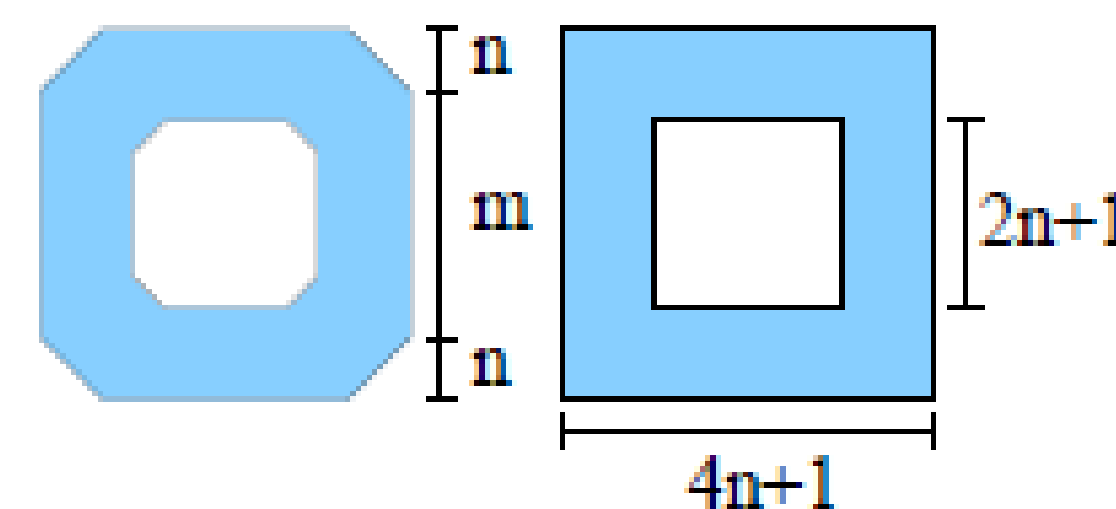
[3] Yimeng Zhang, Zhaoyin Jia, Tsuhan Chen,
Image Retrieval with Geometry-Preserving Visual Phrases

[4] Motilal Agrawal, Kurt Konolige, Morten Rufus Blas
CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching

Techniques

Censure feature detection

- Using integral image for fast LoG approximation
- Bileel filters:

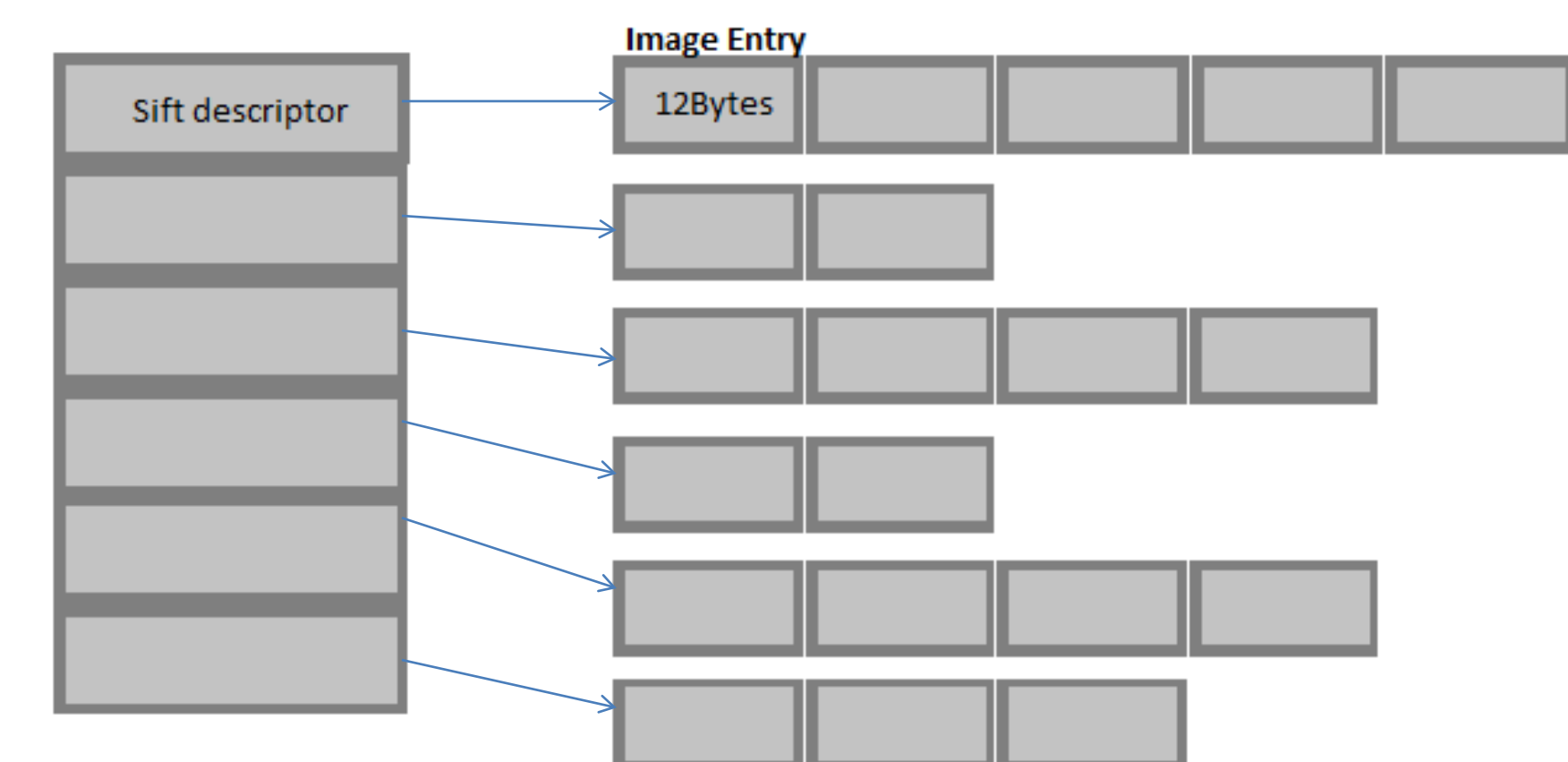


Upright descriptors (no rotation invariance)

- Hope to minimize false recognition
- Rely on perspective transform

Inverted file system (IFS)

- Keeping the whole IFS in RAM for fast query
- Image entry size keeps to the minimal (12 bytes/entry) to support large database (scalability)
- Server (32GB RAM) can support 2B entries
- Hamming embedding for refined matching



Experiment

In this work, I used 40 sample images from nikon and canon web sites, and indexed them using the above method.

Actual server implementation is a Java servlet hosted on apache tomcat. Server is a old laptop machine with a single core 2GHz CPU, 1GB RAM. Client is implemented on iPhone4S with objective C.

Result: almost 100% correct retrieval rate. Error mainly due to motion blurring.

Some sample images

