Image-based Retrieval using a Camera Phone
Sam S. Tsai*, David M. Chen*, Jatinder P. Singh†, and Bernd Girod*
*Information Systems Laboratory, Stanford University, †Deutsche Telekom Laboratories

Mobile Image Matching Application

Query
Wireless Network
Information

Feature-based Image Matching

Scalable Vocabulary Tree (SVT)
[Nister and Stewenius, CVPR 2006]
Speeded Up Robust Features (SURF)
[Bay et al., ECCV 2006]

Rate-efficient Image Retrieval

“Send Feature” Mode

Camera | SURF Extraction | Coding | Client
Transform | Quantization | Entropy Coding
Interest Point Detection | SURF Descriptor Generation | Scalable Vocabulary Tree (SVT) | Geometric Consistency Check

“Send Image” Mode

Camera | Client ↔ Server | SURF Extraction | Image Matching
Mode | Size of query
Feature | 3-6 KB
Image | 30-50 KB
200-400 feature per images

Robust Multiview SVTs

Front View SVT
Top View SVT

Mobile CD/DVD Recognition System

Client
Nokia N95
ARM11 332MHz 64 MB RAM
Network
EDGE < 236 kbps
Server
Xeon 2GHz 4 GB RAM
Database
10k+ CD entries
3k+ DVD entries

Reference
Inverted Index Coding for Scalable Image Matching

David Chen*, Sam Tsai*, Vijay Chandrasekhar*, Gabriel Takacs*, Ramakrishna Vedantham†, Radek Grzeszczuk†, and Bernd Girod*

*Department of Electrical Engineering, Stanford University †Nokia Research Center, Palo Alto

Mobile Image Search

Query

Wireless Network

Information

Inverted Index Coding

Coding of Image Identifiers

Differential coding of image identifiers

\[ d_{ij} = i_{ij} - i_{i(j-1)} \]

\[ d_1 = i_1 - i_0, d_2 = i_2 - i_1, \ldots, d_N = i_N - i_{N-1} \]

Entrophy coding of consecutive differences

Very non-uniform and peaky distribution suggests that variable-length coding is much more efficient than fixed-length coding:

- Arithmetic coding
- Carryover coding
- RBUC coding

Coding of Fractional Counts

Fractional counts from soft binning

\[ w_i = \frac{1}{C} \exp \left( \frac{d_i^2}{\sigma^2} \right) \]

Lloyd-Max quantization of fractional counts

Vocabulary Tree and Inverted Index

Uncompressed index

4 bytes for image identifier
4 bytes for fractional count

\[ \sum_{i=1}^{N_i} \text{list lengths} \times \text{total memory usage} \]

Example: 1 million image database requires 10 GB

Inverted Index

Experimental Results

Database of 1 million CD/DVD/book cover images

\[ 9.8 \text{ GB} \to 2.0 \text{ GB} \sim 5x \text{ savings} \]

\[ 5.7 \text{ sec} \to 1.0 \text{ sec} \sim 5 \text{ sec savings} \]