Indexing in Large Scale Image Collections: Scaling Properties and Benchmark

Mohamed Aly
Computational Vision Lab
Electrical Engineering
Caltech, Pasadena, CA

Mario Munich
Evolution Robotics Inc.
Pasadena, CA

Pietro Perona
Computational Vision Lab
Electrical Engineering
Caltech, Pasadena, CA

IEEE Workshop on Applications of Computer Vision (WACV)
Hawaii, 5-7 January 2011
Large Scale Image Search

Query Image → Database Images

- DVD Covers
- Landmarks
- Book Covers

Result

Typical Application
Existing Applications

Barnes & Noble iPhone App

[http://www.barnesandnoble.com/iphone/]

Google Goggles App

[http://www.google.com/mobile/goggles]
Large Scale Image Search

Questions:
- What are the approaches?
- Scaling properties for billions of images?
- What is the best approach?
- What next?

Benchmark scaling properties
- Storage/Memory
- Computational Cost
- Recognition Performance
- Parallelizability
Local Features Search
Full Representation Vs Bag of Words

Full Representation Image Search
- Probe Image
- Nearest Neighbor Search
- Feature Space
- Match Counting
- Ranked List

Bag of Words Image Search
- Probe Image
- Feature Space
- Quantization
- Histogram
- Ranked List
Full Representation (FR)

Database Images

Feature Space

Probe Image

Nearest Neighbor Search

Match Counting

Ranked List

Full Representation Image Search
Bag of Words (BoW)

Database Images → Quantization → Histograms → Nearest Neighbor Search → Ranked List

WACV 2011
FR Search Methods

Kd-Trees (Kdt)

Input Points → Hash Functions → Hash Tables

Hierarchical K-Means (HKM)

Locality Sensitive Hashing (LSH)

K-d Tree in 2D

Multiple Randomized K-d Trees

HKM in 2D

Hierarchical K-Means Tree

Euclidean 2D

Spherical Simplex 2D

Spherical Orthoplex 2D
BoW Search Methods

- **Inverted File**
  - Database images → Quantized Features → Histograms → Inverted File
  - Probe image → Quantized Features → Hash Table → Ranked List

- **Min-Hash**
  - Database images → Quantized Features → Hash Functions → Hash Tables → LSH Search → Ranked List

**Bag of Words Inverted File Image Search**

**Bag of Words Min-Hash Image Search**
Methods Benchmark

• Theoretical Analysis
  • Memory/Storage
  • Computations
  • Parallelizability

• Experimental Evaluation
  • Run time
  • Recognition Performance
  • Four datasets

• All algorithms implemented in C++/Matlab
• Code available at http://vision.caltech.edu/malaa/software
Theoretical Analysis: Storage & Run Time

- BoW an order of magnitude less storage than FR
- Exhaustive search is prohibitive
- LSH, HKM, BoW grow linearly with # images
- Kdt almost constant

[Aly et al., WACV 2011]
Theoretical Analysis: Parallelizability

[Graph showing the relationship between time per image and number of machines, with different markers and lines for various methods such as kd-tree, kd-tree-adv, lsh-l2, and bow-inv-file.]
Experimental Setup

- 4 Probe Sets
- 4 Distractor Sets
Experimental Results

- FR have much better performance than BoW

[Aly et al., WACV 2011]
Experimental Results

- Kdt, HKM, BoW almost constant time
- LSH increasing time

[Aly et al., WACV 2011]
Conclusions

- FR is the way to go!
- Kdt provide the best tradeoff between run time & recognition performance

Research Directions
- Reduce storage requirements of FR methods
- Improve performance of BoW methods
THANK YOU!