Diversity in Image Retrieval

P Vidyadhar Rao

IIIT-H

November 3, 2015

Overview

- Image Retrieval System
  - Challenges
- Image Query paradigms
  - SVMs for concept queries
  - Eigen queries
- What constitutes a good retrieval?
  - Diversity in image search results
  - Hashing Hyperplane-point queries
- Locality Senstive Hash functions
  - Conditions on similarity functions
  - Similarity Preserving Hash functions
  - Diversity Preserving Hash functions

Image Retrieval System

- Incomplete query specification.
- Semantic gap: User Interpretation vs similarity of visual concepts
- Scalability for large scale image retrieval

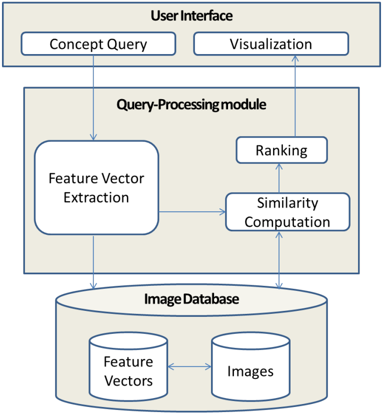

Figure : A general Image Retrieval System

Image Query Specification

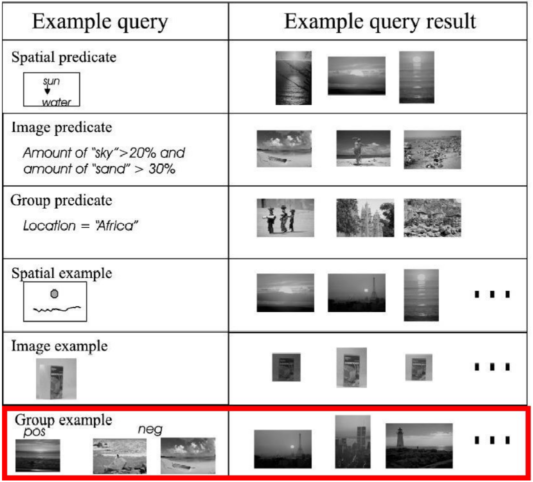

Figure : Users intent is more complex

SVMs for Image Retrieval

- Query by image content
  - point-to-point queries
  - $score(X_q, X_i) = ||X_q - X_i||$
- Query by image concept
  - (SVM)hyperplane-to-point queries
  - $score(W_q, X_i) = W_q^T X_i$
- Naive score calculation : O(nd)
- **Problem: With large image database, we cannot afford exhaustive linear scan.**

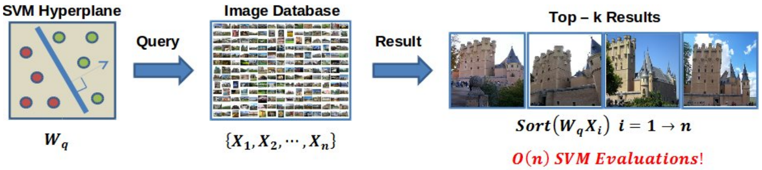

Figure : Image retrieval framework with SVM hyperplane queries

Eigen Queries

- From the query logs obtain hyperplanes, $B = \{W_1, \ldots, W_m\}$.
  - Eigen vectors, $V = \{V_1, \ldots, V_p\}$, correspond to top p eigen values of BB^T.
- Decompose W_q into set of known *eigen queries(vectors)*.
  - $W_q^{'} = \sum_{j=1}^{p} \alpha_j V_j$; use LSE optimization to solve for α.
  - $score(W_q^{'}, X_i) = \sum_{j=1}^{p} \alpha_j V_j X_i = \sum_{j=1}^{p} \alpha_j E_{ji}$
- Eigen score calculation: O(np). ($p << d$)

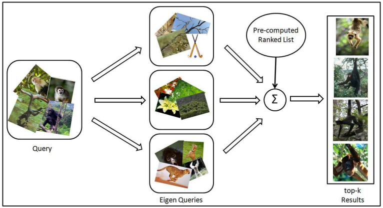

Figure : Conceptual view of ranking using Eigen Queries

- Subset Search: Choose top or bottom t scores of sorted list E_j based on whether α_j is positive or negative respectively.
- Reduces the computations to O(tp).

What constitutes a good retrieval?

- Categorical Query: Images of typical Australian animals

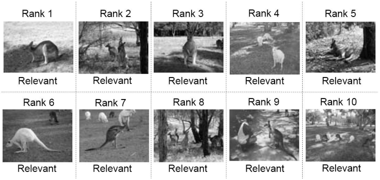

Figure : Many near duplicates in top ranked images.

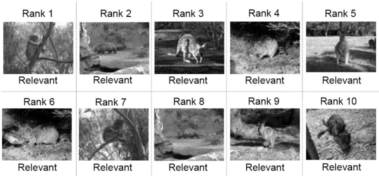

Figure : User intends for different animals in Australia.

Diversity in Image Retrieval

- In order to maximize the satisfaction of different search users, it is necessary to diversify search results.
- Goal is to obtain diverse images relevant to the SVM hyperplane query.
- Maximal marginal relevance(MMR) strategy
  - Jointly computing diversity and relevance scores involves hyperplane-point queries.
  - $score(w_q, x; x_{r_1}, \ldots x_{r_{i-1}}) = \gamma w_q^T x - (1-\gamma) \arg\max_{j<i} x_{r_j}^T x$
  - $score(w_q, x; x_{r_1}, \ldots x_{r_{i-1}}) = \arg\max_{j<i} (\gamma w_q^T - (1-\gamma) x_{r_j}^T) x$
- score calculations: O(nd)
- **Problem: With large image database, we cannot afford exhaustive linear scan.**

Hashing Hyperplane-point queries

- For efficient diverse image retrieval, we use hash functions for hyperplane-point queries.

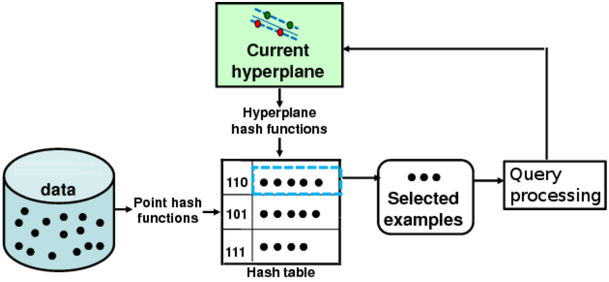

Figure : Hashing Hyperplane-point queries.

- Locality Sensitive Hash functions guarantee sub-linear retrieval time.

Locality Sensitive Hash functions

- Originally motivated by the application of eliminating near-duplicates documents.

Definition 1

A locality sensitive hashing scheme is a distribution on family F of hash functions operating on a collection of objects, such that two objects x, y under some similarity function $sim(x, y)$ hold

$$Pr_{h \in F}[h(x) = h(y)] = sim(x, y) \tag{1}$$

Conditions on similarity functions

Lemma 1

For any similarity function $sim(x, y)$ that admits a locality sensitive hash function families as defined in (1), the distance function $1 - sim(x, y)$ satisfies triangle inequality.

Lemma 2

Given a locality sensitive hash function family F corresponding to to a similarity function $sim(x, y)$, we can obtain a locality sensitive hash function family $F^{'}$ that maps objects to $\{0, 1\}$ and corresponds to the similality function $\frac{1+sim(x,y)}{2}$.

Lemma 3

For any similarity function $sim(x, y)$ that admits a locality sensitive hash function families as defined in (1), the distance function $1 - sim(x, y)$ is isometrically embeddable in Hamming cube.

Similarity preserving Hash functions

- Preserve the neighbourhood structure between the points in the original feature space.

Connections to rounding procedures used in approx. algorithms

Procedures used for rounding fractional solutions to semi-definite programs can be used to derive similarity preserving hash functions for interesting class of similarity functions.

Example

Random hyperplane rounding technique, to round vector solutions for MAX-CUT problem, naturally gives a family of hash functions F for vectors such that

$$Pr_{h \in F}[h(u) = h(v)] = 1 - \frac{\theta(u, v)}{\pi} \tag{2}$$

Diversity preserving Hash functions

Jointly computing similarity and diversity

$$sim(w_q, x; x_{r_1}, \ldots x_{r_{i-1}}) = \arg\max_{j<i}(\gamma w_q^T - (1-\gamma)x_{r_j}^T)x \tag{3}$$

Quadratic Programming problem

$$\begin{aligned} \max \ & \Sigma_{i=1}^{n} \alpha_i w^T x_i - \Sigma_{ij} \alpha_i \alpha_j x_i^T x_j \\ \text{s.t.} \ & \Sigma_{i=1}^{n} \alpha_i = k; \alpha_i \in \{0, 1\} \end{aligned}$$

with some substitutions takes the form of

$$\begin{aligned} \max \ & \alpha^T Q \alpha + c^T \alpha \\ \text{s.t.} \ & \alpha^T 1 = k; \alpha \in Z^n \end{aligned} \tag{4}$$

- **Problem: What hash functions preserve diversity in retrieval?**

References I

- Jaime Carbonell and Jade Goldstein, *The use of mmr, diversity-based reranking for reordering documents and producing summaries*, ACM SIGIR, 1998.
- Moses S Charikar, *Similarity estimation techniques from rounding algorithms*, STOC, 2002.
- Thomas Deselaers, Tobias Gass, Philippe Dreuw, and Hermann Ney, *Jointly optimising relevance and diversity in image retrieval*, ACM CIVR, 2009.
- Prateek Jain, Sudheendra Vijayanarasimhan, and Kristen Gorauman, *Hashing hyperplane queries to near points with applications to large-scale active learning*, NIPS, 2010.
- Nisarg Raval, Rashmi Vilas Tonge, and CV Jawahar, *Image retrieval using eigen queries*, ACCV, 2013.
- Lei Zhang, Fuzong Lin, and Bo Zhang, *Support vector machine learning for image retrieval*, ICIP, 2001.