Qs Rank and efficient ϵ-neighbor search

Arthur Darcet & Yohann Salaun

March 26, 2013

Summary

Introduction

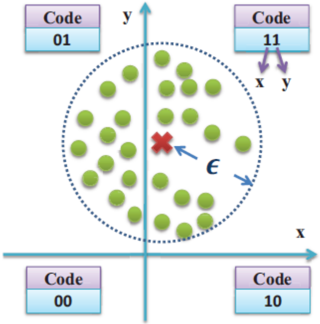

Figure : ϵ-neighbor search

QsRank: Query-sensitive Hash Code Ranking for Efficient Epsilon-neighbor Search, CVPR 2012

Hash Codes

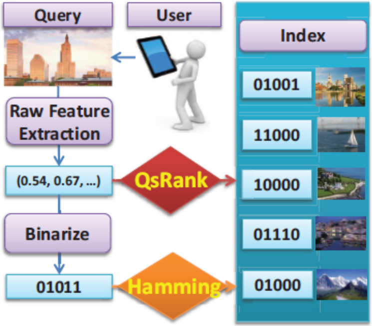

Figure : data binarization

1. Hash code generation of the data for faster retrieval
2. Ranking of the hash code depending on a query

Hash Codes generation

Principal Component Analysis:

$\forall j \in [1; d], \ h_j =$

- 1 if $(PCA(x, \{x_i\}_i))_j > 0$
- 0 if $(PCA(x, \{x_i\}_i))_j \leq 0$

Hash Codes generation

Principal Component Analysis:

$\forall j \in [1; d],\ h_j =$

- 1 if $(PCA(x, \{x_i\}_i))_j > 0$
- 0 if $(PCA(x, \{x_i\}_i))_j \leq 0$

1. dimension decrease but keep information
2. orthogonal projection that preserves the L^2-norm
3. uncorrelated PCA values which leads to an efficient ranking with Qs Rank.

Qs Rank Formula

$$\mathrm{QsRank}(q, h, \epsilon) = \frac{\int_{NN(q,\epsilon)\cap S(h)} p(y)dy}{\int_{NN(q,\epsilon)} p(y)dy} = \mathbb{P}(y \in S(h) | y \in NN(q, \epsilon))$$

where:

- $NN(q, \epsilon) = \{y \in \mathbb{R}^d \text{ s.t. } ||y - q|| < \epsilon\}$ is the ϵ-ball around the query q
- $S(h) = \{y \in \mathbb{R}^d \text{ s.t. } \forall i \in [1; d] y_i h_i > 0\}$ is the set described by the hash code h

Qs Rank Approximations

$$\mathrm{QsRank}(q, h, \epsilon) \quad \approx \quad \frac{\int_{NN(q^k,\epsilon)\cap S(h^k)} p(y^k)dy^k}{\int_{NN(q^k,\epsilon)} p(y^k)dy^k}$$

Qs Rank Approximations

$$\text{QsRank}(q, h, \epsilon) \approx \frac{\int_{NN(q^k,\epsilon)\cap S(h^k)} p(y^k)dy^k}{\int_{NN(q^k,\epsilon)} p(y^k)dy^k}$$

$$\approx \frac{\int_{HC(q^k,\epsilon)\cap S(h^k)} p(y^k)dy^k}{\int_{HC(q^k,\epsilon)} p(y^k)dy^k}$$

where $HC(q^k, \epsilon) = \{y^k \in \mathbb{R}^k \text{ s.t. } \forall i \in [1; k], |y_i^k - q_i^k| < \epsilon\}$

Qs Rank Approximations

$$
\begin{aligned}
\mathrm{QsRank}(q, h, \epsilon) &\approx \frac{\int_{NN(q^k,\epsilon)\cap S(h^k)} p(y^k)dy^k}{\int_{NN(q^k,\epsilon)} p(y^k)dy^k} \\
&\approx \frac{\int_{HC(q^k,\epsilon)\cap S(h^k)} p(y^k)dy^k}{\int_{HC(q^k,\epsilon)} p(y^k)dy^k} \\
&\approx \prod_{i=1}^{k} \frac{\int_{|y_i^k - q_i^k|<\epsilon, y_i^k h_i^k > 0} p(y_i^k)dy_i^k}{\int_{|y_i^k - q_i^k|<\epsilon} p(y_i^k)dy_i^k}
\end{aligned}
$$

Qs Rank Approximations

$$\mathrm{QsRank}(q, h, \epsilon) \approx \prod_{i=1}^{k} \mathbb{P}(y_i^k \in S(h_i^k) | y_i^k \in HC(q_i^k, \epsilon))$$

We suppose also that $(y_i^k)_i \sim \mathcal{U}(\mathbb{R}^k)$:

$$\begin{aligned}\mathrm{QsRank}(q, h, \epsilon) &\approx \prod_{i=1}^{k} \frac{\int_{|y_i^k - q_i^k| < \epsilon, y_i^k h_i^k > 0} dy_i^k}{\int_{|y_i^k - q_i^k| < \epsilon} dy_i^k} \\ &\approx \prod_{i=1}^{k} \mathrm{clamp}(\tfrac{1}{2}(1 + \frac{h_i^k q_i^k}{\epsilon}), [0; 1])\end{aligned}$$

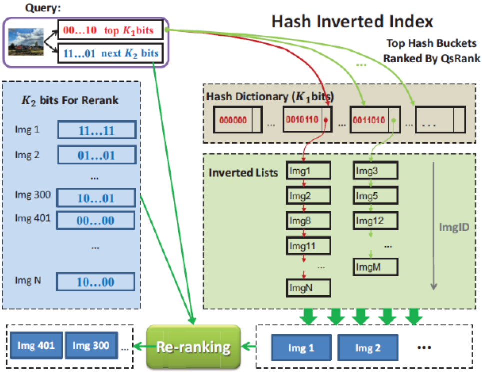

Figure : Retrieval procedure: Buckets ranking (K_1 bits). Neighbors re-ranking ($K_1 + K_2$ bits)

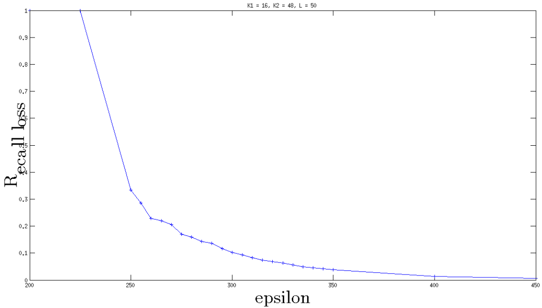

Figure : Recall Loss function of epsilon.

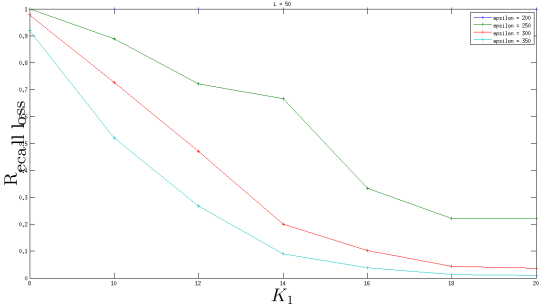

Figure : Recall Loss function of K_1.