A Classification and Survey of Multi-Dimensional Query Types

David Broneske
Student Conference on Software Engineering and Database Systems
July 18, 2012
Motivation

- Importance of multimedia databases increases
- Feature extraction
- Query types → need of information
- Missing implementation of special query types → Survey and Classification
Background – Distance Functions

- Requirement of similarity functions
- Replaced by distance functions
- Minkowski distance functions

\[
L_p(x, y) = \sqrt[p]{\sum_{i=1}^{n} (\|x_i - y_i\|^p)}
\]

- Euclidean \( p=2 \)
- Manhattan \( p=1 \)
- Supremum \( p=\infty \)

adapted from [SSH11]
Query Types – Chosen Query Types

• Exact match query
• Range query
• Partial match query
• Similarity range query
• Nearest neighbor query
• Similarity join query
• K-closest pairs query
• All-nearest-neighbor query
Query Types – Chosen Query Types

- Exact match query
- Range query
- Partial match query
  → Similarity range query
  → Nearest neighbor query
  → Similarity join query
  → K-closest pairs query
  → All-nearest-neighbor query
Query Types – Example Database

- Query types in a feature-vector space
- E.g., Image database
- Features: Average color, texture
Query Types – Similarity Range Query

- Define a distance threshold $\epsilon$
- Retrieve image $x$ if $d(x, y) \leq \epsilon$
Query Types – Nearest Neighbor Query

• Result: Nearest point to the query point
• Special case: $K$-nearest neighbor query
Query Types – Similarity Join Query

- Similarity range query for a set of query images
- Result: Pairs of images
Query Types – K-Closest Pairs Query

- Combination of $k$-nearest neighbor query and similarity join
- Retrieves $k$ pairs with the smallest distance

4-Closest Pairs Query
Query Types – All-Nearest-Neighbor Query

- Result: Nearest neighbor for every image in the query set
- Multiple usage of an image in queried set
Classification

- Query types differ in:
  - selection of results
  - amount of retrieved points
  - complexity

- Criteria influence the usage of queries in specific use cases
Classification – Selection of Results

- Retrieval of resulting points according to:
  - their values in selected dimensions → Boolean queries
  - their similarity/distance to the query → Similarity queries
Classification – Selection of Results

- Retrieval of resulting points according to:
  - their values in selected dimensions → **Boolean queries**
  - their similarity/distance to the query → **Similarity queries**

<table>
<thead>
<tr>
<th>Query Type</th>
<th>Similarity Query</th>
<th>Boolean Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact Match Query</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Range Query</td>
<td>–</td>
<td>✓</td>
</tr>
<tr>
<td>Partial Match Query</td>
<td>–</td>
<td>✓</td>
</tr>
<tr>
<td>Similarity Range Query</td>
<td>✓</td>
<td>–</td>
</tr>
<tr>
<td>Nearest Neighbor Query</td>
<td>✓</td>
<td>–</td>
</tr>
<tr>
<td>Similarity Join Query</td>
<td>✓</td>
<td>–</td>
</tr>
<tr>
<td>K-Closest Pairs Query</td>
<td>✓</td>
<td>–</td>
</tr>
<tr>
<td>All-Nearest-Neighbor Query</td>
<td>✓</td>
<td>–</td>
</tr>
</tbody>
</table>
Classification – Amount of Retrieved Points

• Query types with **fixed** and **variable** size of result set
Classification – Amount of Retrieved Points

- Query types with **fixed** and **variable** size of result set

<table>
<thead>
<tr>
<th>Query Type</th>
<th>Fixed Size of Result Set</th>
<th>Variable Size of Result Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact Match Query</td>
<td>✓</td>
<td>−</td>
</tr>
<tr>
<td>Range Query</td>
<td>−</td>
<td>✓</td>
</tr>
<tr>
<td>Partial Match Query</td>
<td>−</td>
<td>✓</td>
</tr>
<tr>
<td>Similarity Range Query</td>
<td>−</td>
<td>✓</td>
</tr>
<tr>
<td>Nearest Neighbor Query</td>
<td>✓</td>
<td>−</td>
</tr>
<tr>
<td>Similarity Join Query</td>
<td>−</td>
<td>✓</td>
</tr>
<tr>
<td>K-Closest Pairs Query</td>
<td>✓</td>
<td>−</td>
</tr>
<tr>
<td>All-Nearest-Neighbor Query</td>
<td>✓</td>
<td>−</td>
</tr>
</tbody>
</table>
Classification – Complexity

- Complexity for: dimensionality \((n)\), amount of queried points \((m)\), amount of query points \((l)\)
Classification – Complexity

- Complexity for: dimensionality \( (n) \), amount of queried points \( (m) \), amount of query points \( (l) \)

<table>
<thead>
<tr>
<th>Query Type</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact Match Query</td>
<td>( O(n \times m) )</td>
</tr>
<tr>
<td>Range Query</td>
<td>( O(n \times m) )</td>
</tr>
<tr>
<td>Partial Match Query</td>
<td>( O(n \times m) )</td>
</tr>
<tr>
<td>Similarity Range Query</td>
<td>( O(n \times m) )</td>
</tr>
<tr>
<td>Nearest Neighbor Query</td>
<td>( O(n \times m) )</td>
</tr>
<tr>
<td>Similarity Join Query</td>
<td>( O(n \times m \times l) )</td>
</tr>
<tr>
<td>K-Closest Pairs Query</td>
<td>( O(n \times m \times l) )</td>
</tr>
<tr>
<td>All-Nearest-Neighbor Query</td>
<td>( O(n \times m \times l) )</td>
</tr>
</tbody>
</table>
Conclusion and Future Work

• Different query types reviewed
• Majority of queries: Similarity queries
  → Multimedia retrieval
• Fixed or variable size of result set
  → Impact on Pastprocessing
• Different complexities
  → Factors influencing query performance
• Next step: Invent suitable algorithms
Thank you for your attention!


Saake, Gunter; Sattler, Kai-Uwe; Heuer, Andreas: 
*Datenbanken - Implementierungskonzepte.*
3. Auflage.
MITP, Bonn, 2011

Zhang, Rui; Ooi, Beng C.; Tan, Kian-Lee: 
Making the Pyramid Technique Robust to Query Types and Workloads.
Query Types – Exact Match Query

- Retrieves images with same feature vector as query
  \[ \text{Retrieves image } x \text{ for query } q \text{ if } \forall i \in [1, n] : q_i = x_i \]
Query Types – Range Query

- Define an lower \((l)\) and upper \((u)\) bound
- Retrieve image \(x\) if \(\forall i \in [1, n] : l_i \leq x_i \leq u_i\)
Query Types – Partial Match Query

- Special kind of range query
- Either $l_i = u_i$ or $l_i = 0 \land u_i = \infty$