Information Retrieval

BITS Pilani
Pilani Campus

Abhishek
April 2020

innovate achieve lead

BITS Pilani

Pilani Campus

CS F469, Information Retrieval

Lecture topics: Multimedia IR

Content Based Retrieval

- **Objective:** Design a fast searching methods that will search database of multimedia objects to locate objects that matches the query object, exactly or approximately.

- Example:
  - Images similar to a given query image.
  - Companies whose stock prices move similarly.
  - Find X-ray images that contains something that has a texture of a tumor.

Terminologies

Distance or dissimilarity: Given two objects, O_1 and O_2 the distance of the two objects is denoted by:

$$D(O_1, O_2)$$

- Example:
  - For a two equal-length time series, the D() could be their Euclidean distance (the root of sum of squared differences).

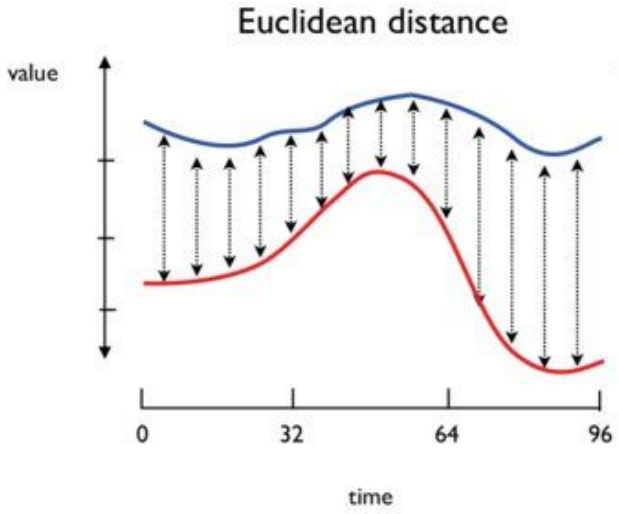


Image source: https://www.researchgate.net/publication/283458699_Scalable_time_series_similarity_search_for_data_analytics

Terminologies

- **Whole match:** Given a collection of N objects O_1, O_2, … , O_N and a query Q, we want to find those objects that are within distance **ϵ** from Q.
  - Note: Query and objects are of same type, image-image, audio-audio, etc.

- **Sub-pattern match:** Here the query is allowed to specify only part of the object.
  - Example:
    - Objects are 512x512 or larger size images (medical X-rays)
    - Query is a 32x32 images (X-ray of a tumor)

Requirements

- It should be **fast**. Sequential scanning and distance calculation with each and every object will be too slow for large databases (Objects in millions).
- It should be **"correct"**. In other words, it should return all the qualifying objects, without missing any (i.e., no false negatives). Notice that false positives are acceptable, since they can be discarded in the post processing step.
- The ideal method should require **small space overhead**.
- The method should be **dynamic**. It should be easy to insert, delete and update objects.

GEMINI (GEneric Multimedia object INdexIng)

- Two key ideas, each of which tries to avoid each of the two disadvantages of sequential scanning:
  - a '**quick-and-dirty**' test, to discard quickly the vast majority of non-qualifying objects (possibly, allowing some false alarms);
  - the use of **spatial access methods**, to achieve faster-than-sequential searching.

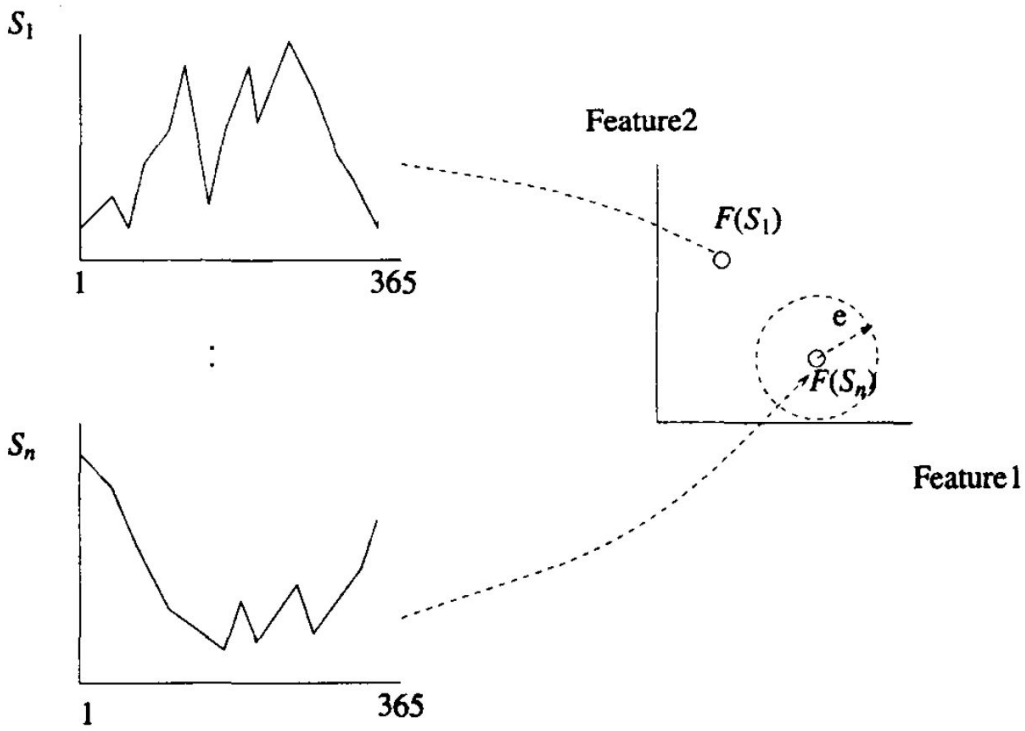
Quick-and-dirty test idea

- To categorize the object with a single or multiple numbers, which can help us discard many non-qualifying sequences.
  - For example, if the object is series of numbers (a yearly stock price movement of a company).
  - A single number can be average stock price over the year.

Features

- The numbers that contain some information about the multimedia objects.

Definition: Let F() be the mapping of objects to f-dimensional points, that is, F(O) will be the f-D point that correspond to object O.



Algorithm 1: Search

1. Map the query object Q into a point F(Q) in feature space.
2. Using a spatial access method, retrieve all points within the desired tolerance ϵ from F(Q).
3. Retrieve the corresponding objects, compute their actual distance from Q and discard the false alarms.

No False Negatives Guarantee

- We can guarantee that there will be no false negatives if the distance in the feature space matches or underestimates the distance between two objects.

Lemma 12.1 (Lower Bounding) To guarantee no false dismissals for whole-match queries, the feature extraction function F() should satisfy the following formula:

$$D_{feature}(F(O_1), F(O_2)) \leq D(O_1, O_2)$$

Spatial Access Methods

- The mapping provide the key to improve the second drawback of sequential scanning.
  - The key is spatial access methods.

- R-Trees:

http://www.mathcs.emory.edu/~cheung/Courses/554/Syllabus/3-index/R-tree.html

Algorithm 2: (GEMINI)

1. Determine the distance function **D()** between two objects.
2. Find one or more numerical feature-extraction functions, to provide a '**quick-and-dirty**' test.
3. Prove that the distance in feature space **lower-bounds** the actual distance D(), to guarantee correctness.
4. Use a **SAM** (e.g., an R-tree, R*-tree), to store and retrieve the f-D feature vectors.

More about D() and F()

- What could be good distance function?
  - A domain expert decides.
  - The GEMINI methodology focuses on speed of search only.

- What could be a a good Feature?
  - It should facilitate step 3 (distance lower bounding)
  - Should capture most of the characteristics of the objects.

One-dimensional Time Series

- Example: Stock prices of companies for a year
- Distance function
  - Euclidean distance, time-warping

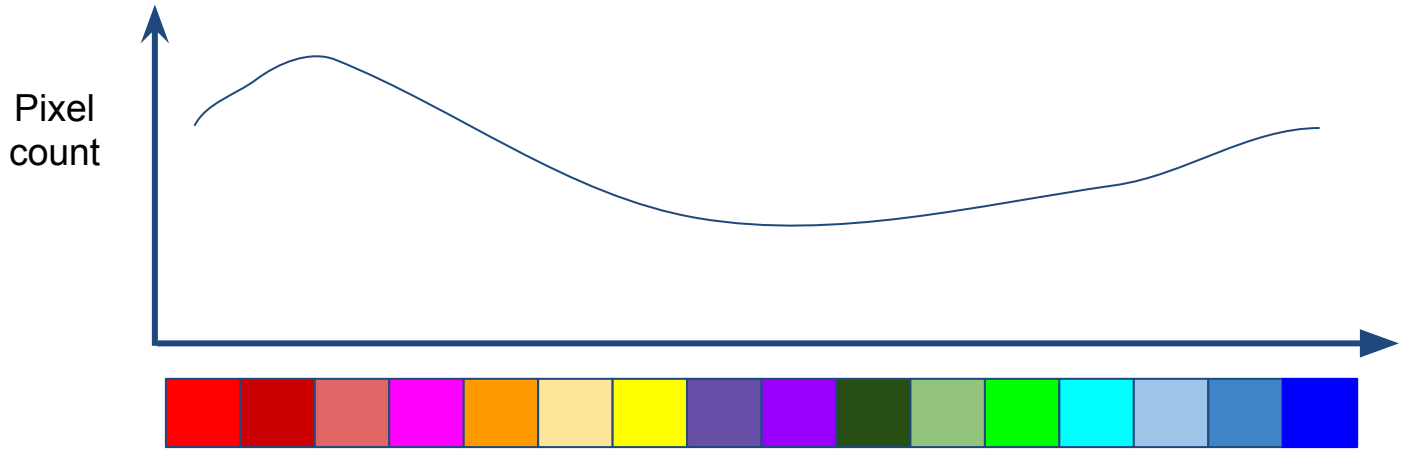
- Feature extraction and Lower-bounding
  - Bad feature:
    - First day value of the company's stock
      - Two companies might have similar stock prices, but differ a lot on day 1
      - Two companies might have same stock price on day 1, but differs a lot on all other days.

One-dimensional Time Series

- Example: Stock prices of companies for a year
- Distance function
  - Euclidean distance, time-warping
- Feature extraction and Lower-bounding
  - Bad feature:
    - First day value of the company's stock
      - Two companies might have similar stock prices, but differ a lot on day 1
      - Two companies might have same stock price on day 1, but differs a lot on all other days.
  - Good Features:
    - Average
    - Coefficients of Discrete Fourier Transform (DFT)
      - Parseval's theorem

Two-dimensional Color Images

- Distance: k-element color histogram



$$d^2_{hist}(\vec{x},\vec{y}) = (\vec{x}-\vec{y})^t \mathbf{A}(\vec{x}-\vec{y}) = \sum_{i}^{k}\sum_{j}^{k} a_{ij}(x_i - y_i)(x_j - y_j)$$

Two-dimensional Color Images

- 3 dimensional feature vector

$$\vec{x} = (R_{avg}, G_{avg}, B_{avg})^t$$

$$G_{avg} = (1/P)\sum_{p=1}^{P} G(p) \qquad R_{avg} = (1/P)\sum_{p=1}^{P} R(p) \qquad B_{avg} = (1/P)\sum_{p=1}^{P} B(p)$$

$$d^2_{avg}(\vec{x}, \vec{y}) = (\vec{x} - \vec{y})^t (\vec{x} - \vec{y})$$

Lower Bound: Quadratic Distance Bounding Theorem

Conclusion

- Two key elements to speed up the retrieval of multimedia objects are:
  - Quick-and-dirty test
  - Spatial access methods

Thank You!