15-826: Multimedia Databases and Data Mining

 Multimedia indexing
     C. Faloutsos

Outline

Goal: ‘Find similar / interesting things’
  • Intro to DB
  • Indexing - similarity search
  • Data Mining

Indexing - Detailed outline
  • primary key indexing
  • secondary key / multi-key indexing
  • spatial access methods
  • fractals
  • text
  • Singular Value Decomposition (SVD)
  • multimedia
  • ...

Multimedia - Detailed outline
  • multimedia
    – Motivation / problem definition
    – Main idea / time sequences
    – images
    – sub-pattern matching
    – automatic feature extraction / FastMap

Problem

Given a large collection of (multimedia) records (eg. stocks)
Allow fast, similarity queries

Applications

  • time series: financial, marketing (click-streams!), ECGs, sound;
  • images: medicine, digital libraries, education, art
  • higher-d signals: scientific db (eg., astrophysics), medicine (MRI scans), entertainment (video)
**Sample queries**

- find medical cases similar to Smith’s
- Find pairs of stocks that move in sync
- Find pairs of documents that are similar (plagiarism?)
- find faces similar to ‘Tiger Woods’

**Detailed problem defn.:**

Problem:
- given a set of multimedia objects,
- find the ones similar to a desirable query object
- for example:

**Types of queries**

- **whole match** vs **sub-pattern match**
- **range query** vs **nearest neighbors**
- all-pairs query

**Design goals**

- Fast (faster than seq. scan)
- ‘correct’ (ie., no false alarms; no false dismissals)

**Multimedia - Detailed outline**

- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images
  - sub-pattern matching
  - automatic feature extraction / FastMap
Main idea

• Eg., time sequences, ‘whole matching’, range queries, Euclidean distance

GEMINI

Solution: Quick-and-dirty’ filter:
• extract \( n \) features (numbers, eg., avg., etc.)
• map into a point in \( n \)-d feature space
• organize points with off-the-shelf spatial access method (‘SAM’)
• discard false alarms

‘GEMINI’ - Pictorially

Idea: ‘GEMINI’

(GEneric Multimedia INdexIng)
Extract a few numerical features, for a ‘quick and dirty’ test

Important: Q: how to guarantee no false dismissals?
A1: preserve distances (but: difficult/impossible)
A2: Lower-bounding lemma: if the mapping ‘makes things look closer’, then there are no false dismissals

CMU SCS
Important:
Q: how to extract features?
A: “if I have only one number to describe my object, what should this be?”

Q: what features?
A: Fourier coefficients (we’ll see them in detail soon)

• Eg.:

• conclusion: colored noises are well approximated by their first few Fourier coefficients
• colored noises appear in nature:
### Time sequences

- **brown noise:** stock prices ($1/f^2$ energy spectrum)
- **pink noise:** works of art ($1/f$ spectrum)
- **black noises:** water reservoirs ($1/f^b$, $b>2$)
- (slope: related to ‘Hurst exponent’, for self-similar traffic, like eg. Ethernet/web
  [Schroeder], [Leland+])

### Time sequences - results

- keep the first 2-3 Fourier coefficients
- faster than seq. scan
- NO false dismissals (see book)

### Multimedia - Detailed outline

- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images (color, shapes)
  - sub-pattern matching
  - automatic feature extraction / FastMap

### Images - color

**what is an image?**

A: 2-d array

**Color histograms, and distance function**
Images - color

Mathematically, the distance function is:

\[ d(x, y) = \sqrt{(x - \bar{x})^2 + (y - \bar{y})^2 + \ldots} \]

possible answers:
- avg red, avg green, avg blue

it turns out that this lower-bounds the histogram distance ->
- no cross-talk
- SAMs are applicable

Images - color

Problem: ‘cross-talk’:
- Features are not orthogonal ->
- SAMs will not work properly
- Q: what to do?
- A: feature-extraction question

Multimedia - Detailed outline

- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images (color; shape)
    - sub-pattern matching
    - automatic feature extraction / FastMap

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
  - Q: how to normalize them?
Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: how to normalize them?
  A: divide by standard deviation)

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: other ‘features’ / distance functions?
  A1: turning angle
  A2: dilations/erosions
  A3: ... )

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 ‘moments’
- (Q: other ‘features’ / distance functions?
  A: Karhunen-Loeve (= centered PCA/SVD)

Images - shapes

- Performance: ~10x faster

[Graph showing performance improvement with log(# of I/Os)]

log(# of I/Os)

# of features kept

all kept
Other shape features?

Other shape features

• Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

Other shape features

• Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

Morphology: closing

• fill in small gaps

• very similar to ‘alpha contours’

Morphology: closing

• fill in small gaps

‘closing’, with R=1
Morphology: opening

- ‘closing’, for the complement =
- trim small extremities

Morphology

- Closing: fills in gaps
- Opening: trims extremities
- All wrt a structuring element:

Multimedia - Detailed outline

- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images (color; shape)
  - sub-pattern matching
  - automatic feature extraction / FastMap

Sub-pattern matching

- Problem: find sub-sequences that match the given query pattern
Sub-pattern matching

- Q: how to proceed?
- Hint: try to turn it into a “whole-matching” problem (how?)

Sub-pattern matching

- Assume that queries have minimum duration $w$; (eg., $w=7$ days)
- divide data sequences into windows of width $w$ (overlapping, or not?)

Sub-pattern matching

- Assume that queries have minimum duration $w$; (eg., $w=7$ days)
- divide data sequences into windows of width $w$ (overlapping, or not?)
- A: sliding, overlapping windows. Thus: trails Pictorially:

Sub-pattern matching

sequences -> trails -> MBRs in feature space
Sub-pattern matching

Q: do we store all points? why not?

Sub-pattern matching

Q: how to do range queries of duration \( w \)?

Sub-pattern matching

(improvement [Moon+2001])
- use non-overlapping windows, for data

Conclusions

- GEMINI works for any setting (time sequences, images, etc)
- uses a ‘quick and dirty’ filter
- faster than seq. scan
- (but: how to extract features automatically?)

Multimedia - Detailed outline

- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images (color; shape)
  - sub-pattern matching
  - automatic feature extraction / FastMap

FastMap

Automatic feature extraction:
- Given a dissimilarity function of objects
- Quickly map the objects to a (k-d) ‘feature’ space.
- (goals: indexing and/or visualization)
FastMap

- Multi-dimensional scaling (MDS) can do that, but in $O(N^2)$ time

MDS

Multi Dimensional Scaling

Main idea: projections

We want a linear algorithm: FastMap

SIGMOD95

Results

Documents /cosine similarity -> Euclidean distance (how?)
Results

Applications: time sequences
- given n co-evolving time sequences
- visualize them + find rules [ICDE00]

Applications - financial
- currency exchange rates [ICDE00]

Conclusions
- GEMINI works for multiple settings
- FastMap can extract ‘features’ automatically (-> indexing, visual d.m.)

References
References