CMU SCS

15-826: Multimedia Databases and Data Mining

Lecture #22: Multimedia indexing

C. Faloutsos

CMU SCS

Must-read Material

- Textbook, chapters 7, 8, 9 and 10.
- Myron Flickner, et al: *Query by Image and Video Content: the QBIC System* IEEE Computer 28, 9, Sep. 1995, pp. 23-32.
- Journal of Intelligent Inf. Systems, 3, 3/4, pp. 231-262, 1994 (An earlier, more technical version of the IEEE Computer '95 paper.)
- FastMap: Textbook chapter 11; Also in: C. Faloutsos and K.I. Lin *FastMap: A Fast Algorithm for Indexing, Data -Mining and Visualization of Traditional and Multimedia Datasets* ACM SIGMOD 95, pp. 163-174.

CMU SCS

Outline

Goal: 'Find similar / interesting things'

- Intro to DB
- Indexing - similarity search
- Data Mining

CMU SCS

Indexing - Detailed outline

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

CMU SCS

Multimedia - Detailed outline

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

CMU SCS

Problem

Given a large collection of (multimedia) records (eg. stocks)

Allow fast, similarity queries

CMU SCS

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)

CMU SCS

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’

CMU SCS

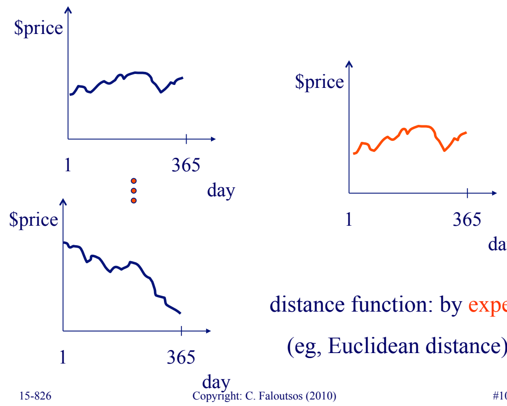
Detailed problem defn.:

Problem:

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

- for example:

CMU SCS



distance function: by expert

(eg, Euclidean distance)

CMU SCS

Types of queries

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

CMU SCS

Design goals

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

CMU SCS

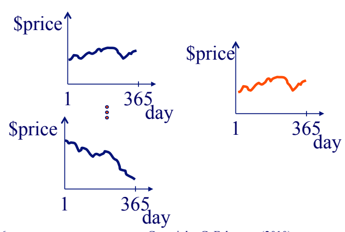
Multimedia - Detailed outline

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

CMU SCS

Main idea

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



CMU SCS

Main idea

- Seq. scanning works - how to do faster?

CMU SCS

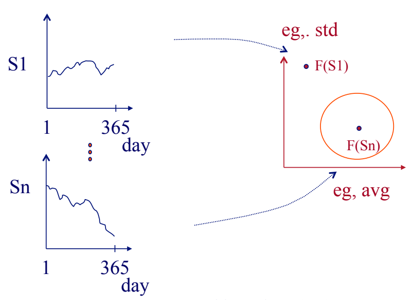
Idea: 'GEMINI'

(GEneric Multimedia INdexIng)

Extract a few numerical features, for a 'quick and dirty' test

CMU SCS

'GEMINI' - Pictorially



CMU SCS

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

CMU SCS

GEMINI

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

15-826 Copyright: C. Faloutsos (2010) #19

CMU SCS

GEMINI

Important:

Q: how to extract features?

A: "*if I have only one number to describe my object, what should this be*?"

15-826 Copyright: C. Faloutsos (2010) #20

CMU SCS

Time sequences

Q: what features?

15-826 Copyright: C. Faloutsos (2010) #21

CMU SCS

Time sequences

Q: what features?

A: Fourier coefficients (we'll see them in detail soon)

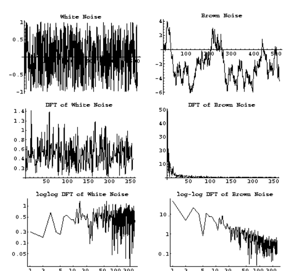
CMU SCS

Time sequences

white noise brown noise

Fourier spectrum

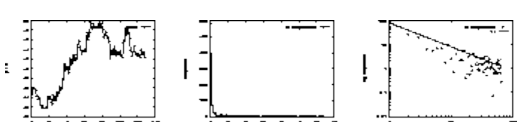
... in log-log



CMU SCS

Time sequences

- Eg.:



(a) IBM stock (b) spectrum (linear scales) (c) spectrum (log scales)

CMU SCS

Time sequences

- conclusion: colored noises are well approximated by their first few Fourier coefficients
- colored noises appear in nature:

CMU SCS

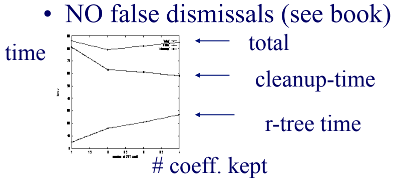
Time sequences

- 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+]

CMU SCS

Time sequences - results

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



CMU SCS

Time sequences - improvements:

- improvements/variations: [Kanellakis+Goldin], [Mendelzon+Rafiei]
- could use Wavelets, or DCT
- could use segment averages [Yi+2000]

CMU SCS

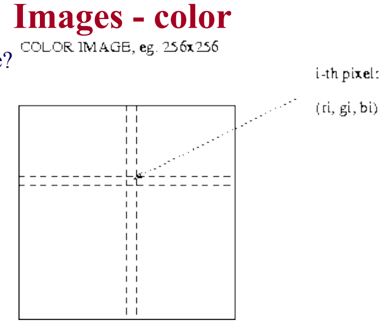
Multimedia - Detailed outline

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

CMU SCS

Images - color

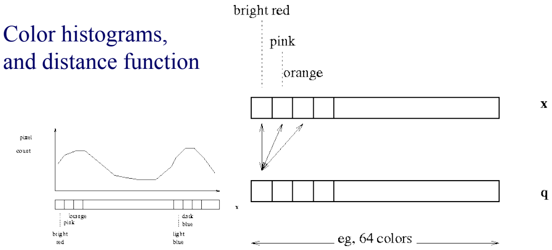
what is an image?
A: 2-d array



CMU SCS

Images - color

Color histograms,
and distance function



CMU SCS

Images - color

Mathematically, the distance function is:

$$distance_{histogram}(\vec{x}, \vec{q}) = (\vec{x} - \vec{q}) \begin{bmatrix} a_{RR} & a_{RP} & \dots \\ a_{PR} & a_{PP} & \dots \\ \dots & \dots & \dots \end{bmatrix} (\vec{x} - \vec{q})^t$$

$$\dots = (\vec{x} - \vec{q}) \mathcal{A} (\vec{x} - \vec{q})^t$$

CMU SCS

Images - color

Problem: ‘cross-talk’:

- Features are not orthogonal ->
- SAMs will not work properly

- Q: what to do?
- A: feature-extraction question

CMU SCS

Images - color

possible answers:

- avg red, avg green, avg blue

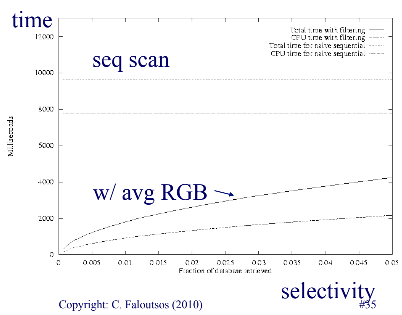
it turns out that this lower-bounds the histogram distance ->

- no cross-talk
- SAMs are applicable

CMU SCS

Images - color

performance:



CMU SCS

Multimedia - Detailed outline

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

CMU SCS

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 'moments'
- (Q: how to normalize them?

CMU SCS

Images - shapes

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

CMU SCS

Images - shapes

- distance function: Euclidean, on the area, perimeter, and 20 'moments'
- (Q: other 'features' / distance functions?

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: how to do dim. reduction?

Images - shapes

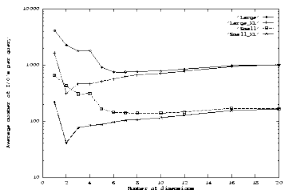
- distance function: Euclidean, on the area, perimeter, and 20 'moments'
- Q: how to do dim. reduction?
- A: Karhunen-Loeve (= centered PCA/SVD)

CMU SCS

Images - shapes

- Performance: ~10x faster

log(# of I/Os)



← all kept

# of features kept

CMU SCS

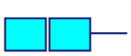
Other shape features?

CMU SCS

Other shape features

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

shape



"structuring element"

R=1

Other shape features

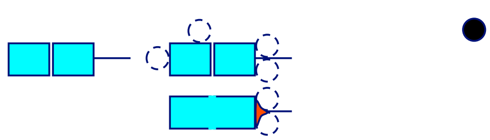
- Morphology (dilations, erosions, openings, closings) [Korn+, VLDB96]

Other shape features

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

Morphology: closing

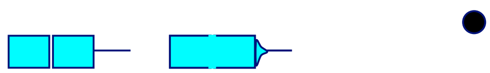
- fill in small gaps
- **very similar** to ‘alpha contours’



CMU SCS

Morphology: closing

- fill in small gaps

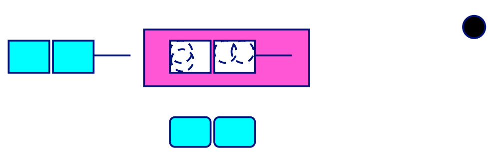


'closing',
with R=1

CMU SCS

Morphology: opening

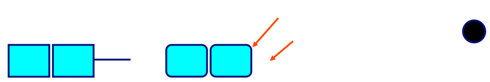
- 'closing', for the complement =
- trim small extremities



CMU SCS

Morphology: opening

- 'closing', for the complement =
- trim small extremities

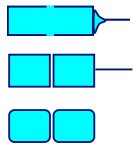


'opening'
with R=1

CMU SCS

Morphology

- Closing: fills in gaps
- Opening: trims extremities

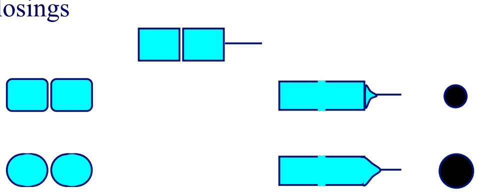


- All wrt a structuring element: 

CMU SCS

Morphology

- Features: areas of openings (R=1, 2, …) and closings



CMU SCS

Multimedia - Detailed outline

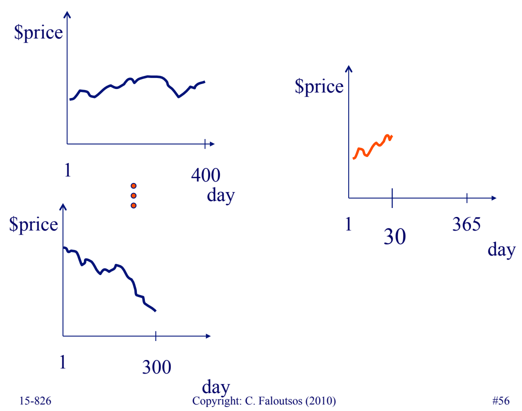
- multimedia
  - Motivation / problem definition
  - Main idea / time sequences
  - images (color; shape)
  - sub-pattern matching
  - automatic feature extraction / FastMap

CMU SCS

Sub-pattern matching

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

CMU SCS



CMU SCS

Sub-pattern matching

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

CMU SCS

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?)

CMU SCS

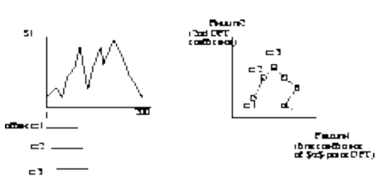
Sub-pattern matching

- 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:

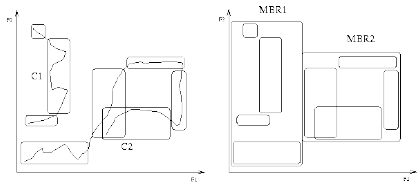
CMU SCS

Sub-pattern matching



CMU SCS

Sub-pattern matching

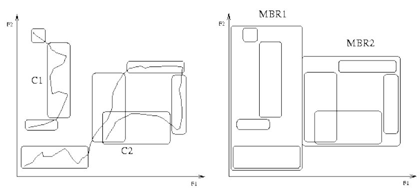


sequences -> trails -> MBRs in feature space

15-826 Copyright: C. Faloutsos (2010) #61

CMU SCS

Sub-pattern matching

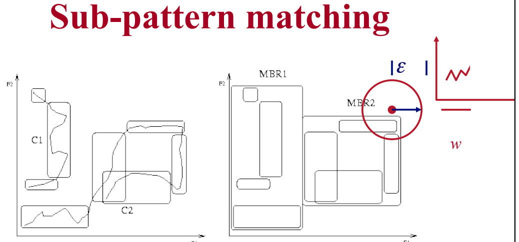


Q: do we store all points? why not?

15-826 Copyright: C. Faloutsos (2010) #62

CMU SCS

Sub-pattern matching

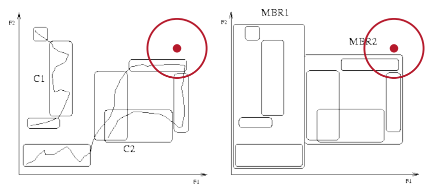


Q: how to do range queries of duration *w*?

15-826 Copyright: C. Faloutsos (2010) #63

CMU SCS

Sub-pattern matching

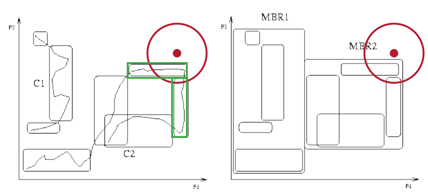


Q: how to do range queries of duration *w*?

A: R-tree; find qualifying stocks and intervals

CMU SCS

Sub-pattern matching

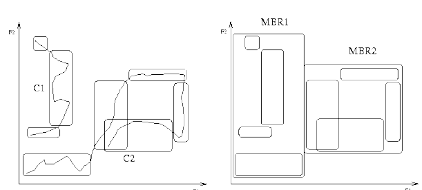


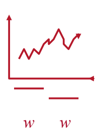
Q: how to do range queries of duration *w*?

A: R-tree; find qualifying stocks and intervals

CMU SCS

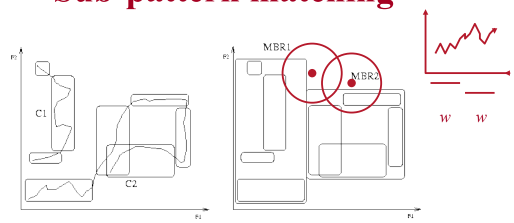
Sub-pattern matching





Q: how to do range queries of duration >*w* (say, *2*w*)?

Sub-pattern matching



Q: how to do range queries of duration >*w* (say, *2*w*)?
A: Two range queries of radius epsilon and intersect
(or two queries of smaller radius and union – see paper)

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?)

CMU SCS

Multimedia - Detailed outline

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

CMU SCS

FastMap

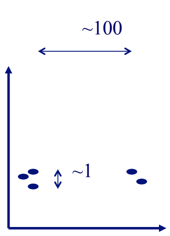
Automatic feature extraction:

- Given a dissimilarity function of objects
- Quickly map the objects to a (k-d) `feature' space.
- (goals: indexing and/or visualization)

CMU SCS

FastMap

	O1	O2	O3	O4	O5
O1	0	1	1	100	100
O2	1	0	1	100	100
O3	1	1	0	100	100
O4	100	100	100	0	1
O5	100	100	100	1	0



CMU SCS

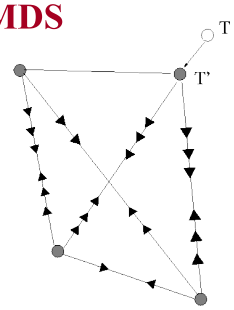
FastMap

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

CMU SCS

MDS

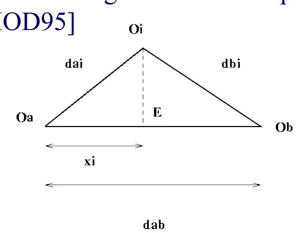
Multi Dimensional Scaling



CMU SCS

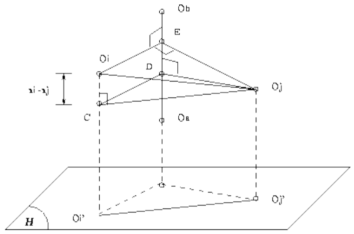
Main idea: projections

We want a linear algorithm: FastMap [SIGMOD95]



CMU SCS

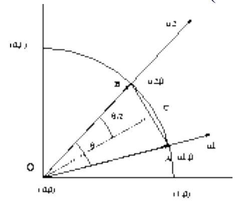
FastMap - next iteration



CMU SCS

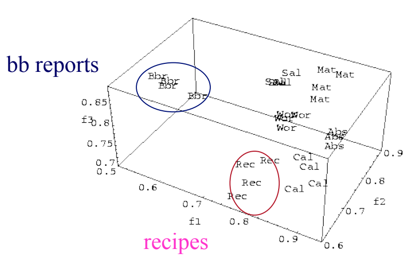
Results

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



CMU SCS

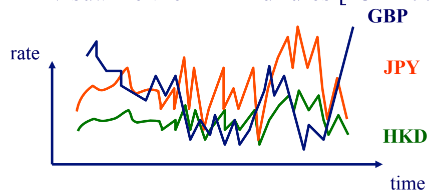
Results



CMU SCS

Applications: time sequences

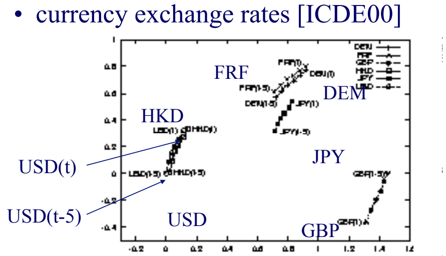
- given *n* co-evolving time sequences
- visualize them + find rules [ICDE00]



CMU SCS

Applications - financial

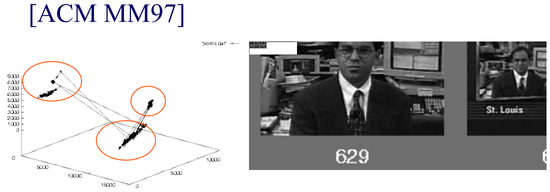
- currency exchange rates [ICDE00]



CMU SCS

VideoTrails

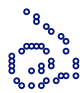
[ACM MM97]



CMU SCS

Variations

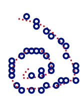
- Isomap [Tenenbaum, de Silva, Langford, 2000]
- LLE (Local Linear Embedding) [Roweis, Saul, 2000]
- MVE (Minimum Volume Embedding) [Shaw & Jebara, 2007]



CMU SCS

Variations

- Isomap [Tenenbaum, de Silva, Langford, 2000]
- LLE (Local Linear Embedding) [Roweis, Saul, 2000]
- MVE (Minimum Volume Embedding) [Shaw & Jebara, 2007]



CMU SCS

Conclusions

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

References

- Faloutsos, C., R. Barber, et al. (July 1994). "*Efficient and Effective Querying by Image Content*." J. of Intelligent Information Systems 3(3/4): 231-262.
- Faloutsos, C. and K.-I. D. Lin (May 1995). *FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets*. Proc. of ACM-SIGMOD, San Jose, CA.
- Faloutsos, C., M. Ranganathan, et al. (May 25-27, 1994). *Fast Subsequence Matching in Time-Series Databases*. Proc. ACM SIGMOD, Minneapolis, MN.

References

- Flickner, M., H. Sawhney, et al. (Sept. 1995). "*Query by Image and Video Content: The QBIC System*." IEEE Computer 28(9): 23-32.
- Goldin, D. Q. and P. C. Kanellakis (Sept. 19-22, 1995). *On Similarity Queries for Time-Series Data: Constraint Specification and Implementation*. Int. Conf. on Principles and Practice of Constraint Programming (CP95), Cassis, France.
- Flip Korn, Nikolaos Sidiropoulos, Christos Faloutsos, Eliot Siegel, Zenon Protopapas: *Fast Nearest Neighbor Search in Medical Image Databases*. VLDB 1996: 215-226

References

- Leland, W. E., M. S. Taqqu, et al. (Feb. 1994). "*On the Self-Similar Nature of Ethernet Traffic*." IEEE Transactions on Networking 2(1): 1-15.
- Moon, Y.-S., K.-Y. Whang, et al. (2001). *Duality-Based Subsequence Matching in Time-Series Databases*. ICDE, Heidelberg, Germany.
- Rafiei, D. and A. O. Mendelzon (1997). *Similarity-Based Queries for Time Series Data*. SIGMOD Conference, Tucson, AZ.

References

- Lawrence Saul & Sam Roweis. *An Introduction to Locally Linear Embedding* (draft)
- Sam Roweis & Lawrence Saul. *Nonlinear dimensionality reduction by locally linear embedding*. Science, v.290 no.5500 , Dec.22, 2000. pp.2323--2326.
- Schroeder, M. (1991). *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. New York, W.H. Freeman and Company.
- B. Shaw and T. Jebara. "*Minimum Volume Embedding*" . Artificial Intelligence and Statistics, AISTATS, March 2007.

References

- Josh Tenenbaum, Vin de Silva and John Langford. *A Global Geometric Framework for Nonlinear dimensionality Reduction*. Science 290, pp. 2319-2323, 2000.
- Yi, B.-K. and C. Faloutsos (2000). *Fast Time Sequence Indexing for Arbitrary Lp Norms*. VLDB, Cairo, Egypt.