Where are you heading, metric access methods?

A provocative survey☺

Tomáš Skopal

SIRET research group, Faculty of Mathematics and Physics, Charles University in Prague, Czech rep.

http://siret.ms.mff.cuni.cz

SISAP 2010, Sep 18-19, Istanbul, Turkey

Talk Outline

- 6 questions, discussed by analyzing:
- experimental practices in MAM research
  - analysis of papers on MAMs (past 4 decades)
  - weak points
- prospective applications for MAMs
  - analysis of similarity search in CBIR
  - multimedia search engines (not) using MAMs
- discussion & suggestions

Questions

1. **Isn't the metric space model too general?**
2. **Are the established MAM cost measures relevant?**
3. **Is there a real demand for general metric indexing?**
4. **Are the simple similarity queries competitive enough?**
5. **Have the real-world search engines ever used a MAM?**
6. **Isn't the metric model too restrictive?**

Metric access methods

- content-based retrieval→similarity search→**metric space model**
- metric access method (MAM):

  Set of algorithms and data structure(s) providing efficient (fast) similarity search under the metric space model.

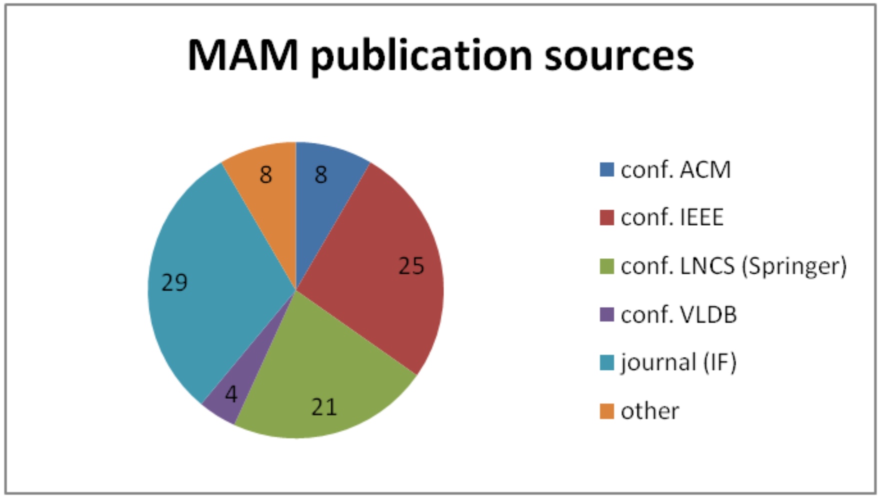
  – includes index structures and related stuff, like pivot selection techniques, metric mapping/classification/clustering, etc.
  – assuming black-box metric space – **only distances can be used**

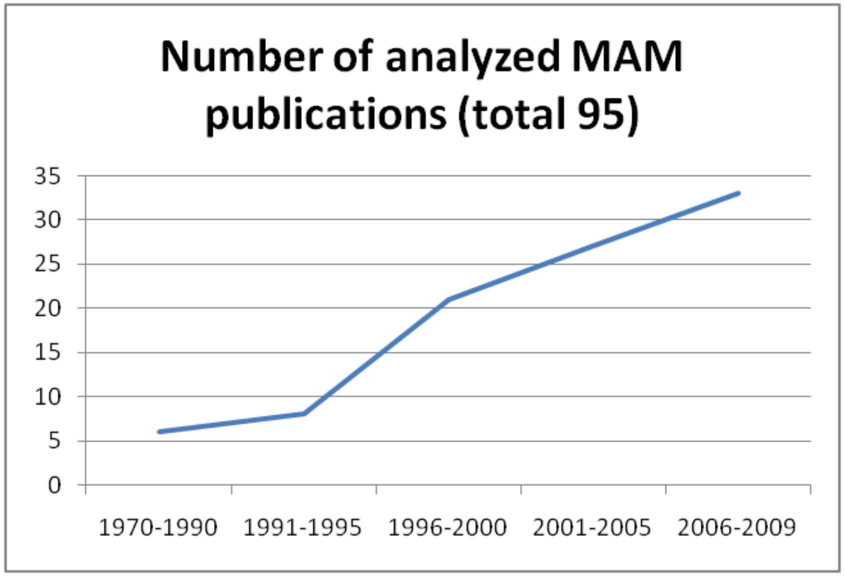
- many MAMs developed so far, various aspects
  – main vs. secondary memory, static vs. dynamic database, exact vs. approximate search, continuous vs. discrete metric, centralized/serial vs. distributed/parallel implementation, etc.

Experimental practices in MAMs

- analysis of 95 papers
  - only general MAM proposals with experimental evaluation
- 77 selected papers cited in major „bibles“ on MAMs
  - Chávez et al., ***Searching in metric spaces***, ACM Computing Surveys, 33(3), 2001
  - Zezula et al., ***Similarity Search: The Metric Space Approach***, Springer, 2006
  - Samet, ***Foundations of Multidimensional and Metric Data Structures***, Morgan Kaufmann, 2006
  - Hetland, ***The Basic Principles of Metric Indexing***, book chapter, Swarm Intelligence for Multi-objective Problems in Data Mining, Springer, 2009
- 18 selected papers presented at SISAP 2008+2009

Structure of papers

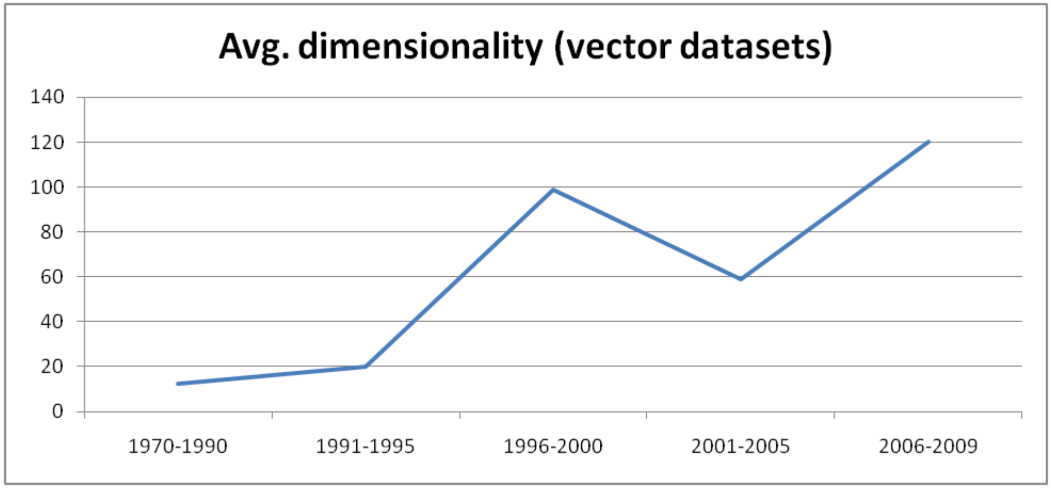




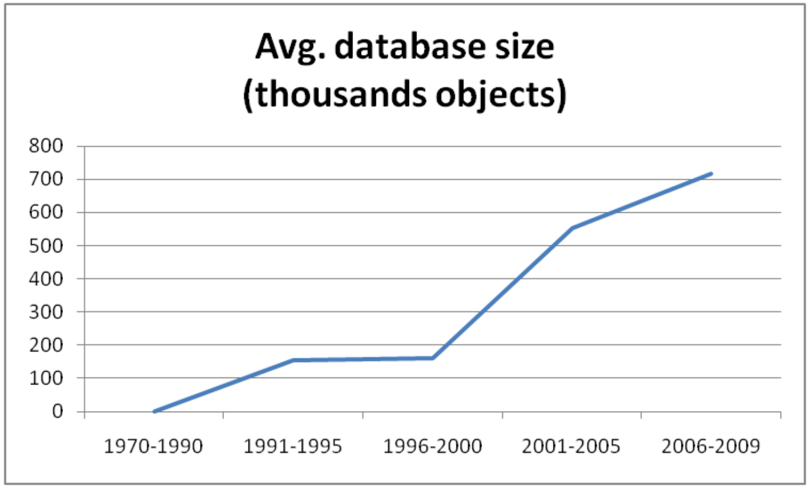
47 papers (49.5%) co-authored by somebody from SISAP PC (2008-2010, 12 people)

Datasets in experiments

- 50% papers use **only vector spaces** in their experimental settings



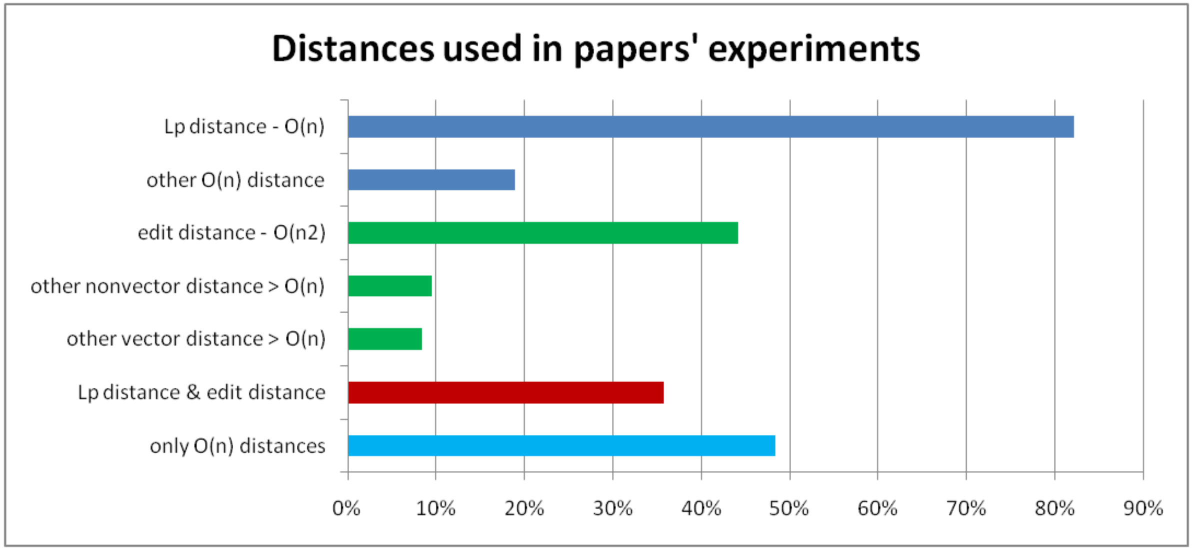
- almost 50% use (also) a string space
  - **mostly vocabulary** (English, Spanish)
  - several use biological or other DBs



- only 10% use other type of space
  - variable size descriptor
    - either embedded within block of fixed size
    - or reference to a subpart of larger entity
  - e.g., set of elements, time series, geometry

Distances in experiments

- the **vast majority** of MAM papers include **L_p spaces** in their experimental settings
  - mostly L_2, few L_∞, few L_p combinations



- almost 50% papers use **edit distance**
- almost 50% papers use **only *O(n)*** dist.
- several papers use
  - non-L_p vector distances
    - *O(n)* – Hamming dist., angle
    - > *O(n)* – quadratic form distance
  - nonvector distance (other than edit distance)
    - > *O(n)* – Hausdorff distance, string/sequence alignments

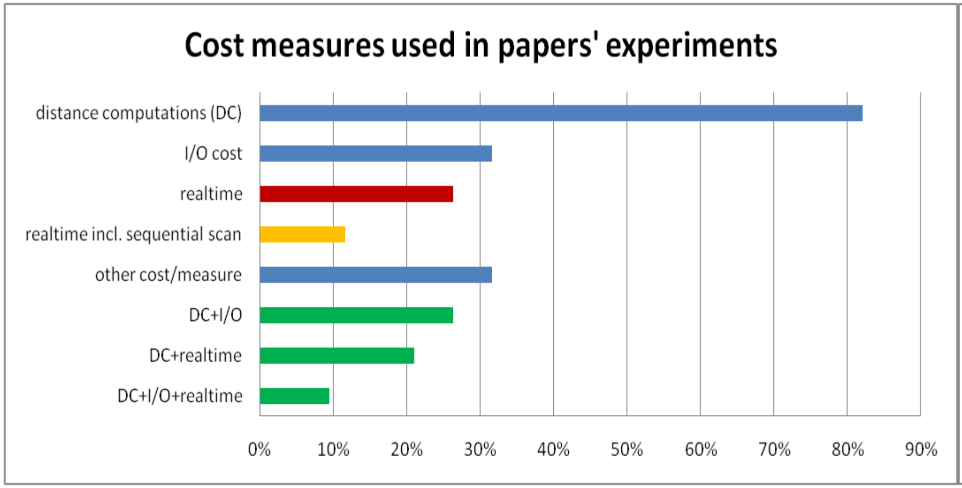
hence,

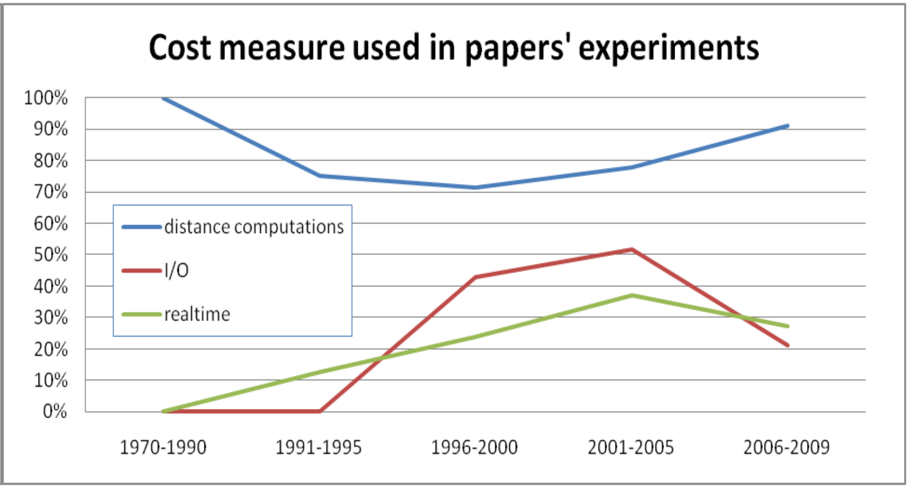
Q1:

Isn't the metric space model too general?

(when a few-hundred-dim. Lp spaces dominant)

Cost measures





- 21% papers use **only *O(n)* distances and only DC** (!!!)
  - *O(n)* distances are very cheap w.r.t. the internal overhead
  - here index organization matters (e.g., flat table vs. hierarchy)
- 25% papers show realtimes
  - 12% direct comparison with seq. search
  - 10% show all measures (DC+IO+realtime)

Distance computations (DC)

- DC alone appropriate when
  - expensive distance is used
    - ≥ $O(n^2)$ and/or large descriptor size (n)
  - rather small database is used (e.g., fits in main memory)
  - other cost contributing to realtime is negligible
    - internal time/space cost, I/Os, networking, synchronization of parallel/distributed processing
- **not respected much in the analyzed papers**
  - remember, mostly L_p distances used in experiments
  - anyways, I cannot scold anybody, my papers are (mostly) not an exception ☺ ... have to redeem

I/O cost

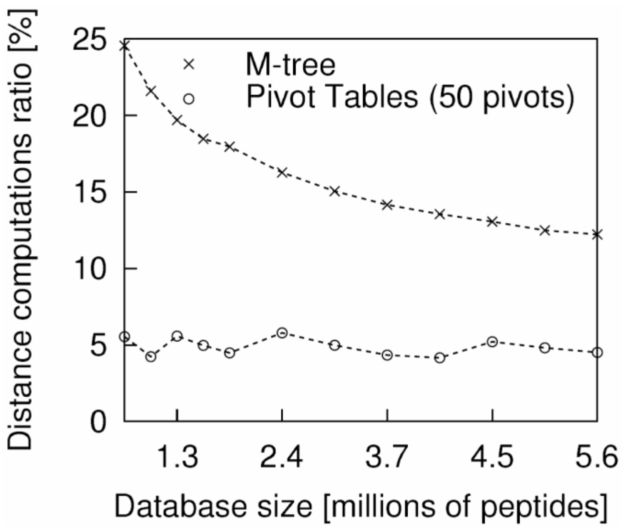
- I/O alone appropriate when
  - dominating the other cost (DC, internal, etc.)
    - assuming classic hard disk technology
  - the competitor MAMs share the same I/O access model
    - random vs. contiguous disk access
- otherwise misleading cost – **optimized sequential scan** could be a surprise!
- example
  - seek time = 8ms, transfer 50 MB/s (low-end HDD today)
  - 100 MB index, 4kB disk page, i.e., 25,600 pages
  - sequential scan, **100% pages**, contiguous access = **2 sec** (random **206 sec**)
  - a hierarchical MAM, **1% pages**, random access = **2.1 sec**
- fortunately, SSDs will change it all... random access not a problem anymore
  - renaissance of hierarchical MAM

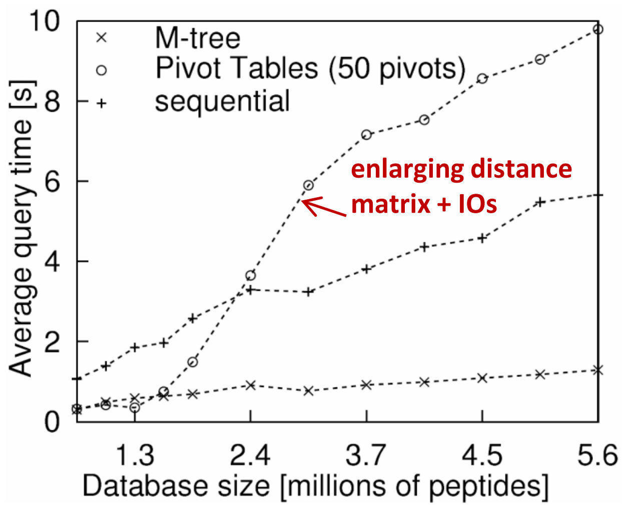
Internal cost

- the more sophisticated MAM $\rightarrow$ the more overhead
  - various auxiliary main-memory structures + processing
  - overhead data in the index + processing
- examples
  - incremental kNN processing (Hjaltason and Samet)
    - optimal in DC (w.r.t. equivalent range query), **but**
    - huge time/space overhead when managing the heap of requests
  - pivot tables (basic LAESA)
    - scanning the distance matrix
    - consider, e.g., 128 dimensional vector dataset + any L_p distance, 128 pivots $\rightarrow$ distance matrix processing means the **same or worse than simple sequential query (!)**

Realtime cost

- realtime cost (wall-clock time)
  - **cons:**
    - **optimization- and platform-dependent**
    - **harder to set up fair comparison**
  - **pros:**
    - **the only objective measure when it comes to real-world application!**
- real-world example
  - database of up to 5.6 million peptide spectra (pieces of proteins), dim ≈ 32 (intrinsic dim. ≈ 3)
  - **O(n)** variant of Hausdorff distance





hence,

Q2:

Are the established MAM cost measures relevant?

(realtime vs. DC/IO cost discrepancy due to mainly O(n) distances used)

Applications in content-based image retrieval (CBIR)

- source: *Datta et al., Image retrieval: Ideas, influences, and trends of the new age, ACM Computing Surveys, 40(2), 2008* – references almost 300 papers related to CBIR

- *„...indexing techniques largely overshadowed by research on similarity modeling...“*
  - most retrieval engines based on text-indexing research
    - automatic annotation/classification/tagging
  - or sequential similarity search
  - **i.e., indexing got not much attention in the CBIR community**

- *„...we do not have yet a universally acceptable visual model for content-based search...“*
  - **good news: relevance modeling (similarity function) mostly separated from search algorithm**

Applications in content-based image retrieval (CBIR), cont.

- common similarity measures in CBIR
  - mainly Euclidean $O(n)$, some quadratic form distance $O(n^2)$, few Earth moving distance $O(n^2)$-$O(2^n)$
    - i.e., the semantic complexity is put into descriptors, not into distances
  - **specific (L_p) indexing more appropriate?**
- *„...the richness in the mathematical formulation of signatures (descriptors) grows alongside the invention of new methods for measuring similarity...“*
  - great interest in region-based signatures (segmentation)
    - *„...global features are often too rigid to represent an image...“*
  - **i.e., hopefully will favor more general similarity search models**
    - **> *O(n)* nonvectorial/metric/nonmetric distances?**
- MAM as main-engine? many limitations... (metric modifications)

hence,

Q3:

Is there a real demand for general metric indexing?

(keyword search, seq. search, specific indexing)

Applications in content-based image retrieval (CBIR), cont.

- today content-based retrieval models
  - **pseudo-CBIR** – add-ons of many commercial engines (as presented later)
    - ad-hoc analysis of certain feature in image, then labeling (e.g., image contains face, illustration, particular color)
  - **single-model similarity search**
    - single global descriptor + single complex similarity (range/kNN)
    - keyword-based search using visual words
  - **hybrid-model similarity search**
    - multiple (local) descriptors + multiple similarity searches $\rightarrow$ aggregation (top-k), optionally reranking

hence,

Q4:

Are the simple similarity queries competitive enough?

(MAMs mostly support range/kNN queries)

Mainstream multimedia search engines/web sites

- multimedia search engines
  - **images:** Google Image Search, Bing Image Search, AllTheWeb, PicSearch
  - **video:** Bing Video Search, Lycos, AOL Video Search, SearchForVideo, BlinkX
  - **audio:** KaZaA, FindSounds, Skreemr, Yahoo Music Search
- general image/video hosting servers
  - **images:** Flickr, PhotoBucket, ImageShack, Google Picasa, DeviantArt
  - **video:** YouTube, DailyMotion, Yahoo Video, MySpace, MetaCafe, Google Video, MSN Video
- major (micro)stock servers (cliparts for professional designers)
  - image/video/audio/vector/flash content
  - each site up to 5-20 millions hosted images
  - keyworded content, categories, controlled quality (reviewing)
  - Corbis, Getty, iStockPhoto, ShutterStock, Fotolia, DreamsTime, Alamy, Veer
- 7 of 32 content-based search (google, bing, picsearch, findsounds, flickr, picasa, shutterstock)
  - just FindSounds supports “true” similarity search (but index+similarity n/a)
  - the others simple content-based annotation (face/color/style)

Content-based image retrieval engines

- both commercial engines & research prototypes/demos

(source: http://en.wikipedia.org/wiki/List_of_CBIR_engines, June 16, 2010)

  - Elastic Vision, Gazopa, Imense, Imprezzeo, Incogna, Like.com, MiPai, idee Visual Search Lab, Empora, Shopachu, TinEye, Tiltomo, eBay More Like This, ALIPR, Anaktisi, BRISC, Caliph & Emir, CIRES, FIRE, GNU Image Finding Tool, ISSBP, img(Rummager), imgSeek, IKONA, MUVIS, PIRIA, RETIN, Retrievr, SIMBA, TagProp, MUFIN

- 25 of 29 use similarity search
  - 7 use metric similarity
    - 2 use metric access methods (MUFIN, MiPai)
  - specifications of the others n/a (patented or not documented)
    - mostly annotation of content + tag search

hence,

Q5:

Have the real-world search engines ever used a metric access method?

(some yes, but technical info mostly not available)

Beyond the metric space model

- source: *Skopal and Bustos, On Nonmetric Similarity Search Problems in Complex Domains, to appear in ACM Computing Surveys, 2012* (download here) – references almost 170 papers
  - domain experts focus on more complex similarity modeling & don’t care of other properties
    - extensive modeling $\rightarrow$ often **nonmetric distances**
    - e.g., edit distance $\rightarrow$ Smith-Waterman
  - nonmetric sequential search
    - nowadays not a problem for the initial research phase of domain expert, i.e., indexing is not a priority at all
    - could be a problem in the future, when the models will be matured and scalability demanded

Beyond the metric space model (cont.)

- fascinating opportunities for indexing by similarity, not yet discovered by the database community
  - mainstream domains – multimedia retrieval (images/video/audio/music/geometry/web)
  - recent domains – biometric identification, one-dimensional time series, XML
  - emerging domains – chemoinformatics, medical databases, (social) networks, multi-dimensional time series
- separated worlds (databases vs. domains)
  - some “evangelism” needed (as discussed later)

hence,

Q6:

Isn't the metric model too restrictive?

(all-in-one similarity & metric postulates limit the modeling)

Beyond the metric space model (cont.)

- nonmetric access methods
  - single (global) descriptor + nonmetric measure
  - transformation to metric space + indexing by MAMs
    - concave function enforces triangle inequality
      Skopal, Unified Framework for Fast Exact and Approximate Search in Dissimilarity Spaces, ACM TODS 32(4), 2007
  - alternative indexing schemes
    - fuzzy logic, ptolemaic indexing

Discussion & Suggestions

Balancing model complexity

- complex descriptor vs. complex distance
- high-level descriptor + cheap distance is better for performance ,
  i.e., not god news for MAMs
  but
- can always be the complexity put into „canonized“ descriptors?
  - do they exist problems inherently requiring complex distance?

- example – robust shape matching based on time series
  - **windowing produces many fragments + L2** (*Ye and Keogh, Time Series Shapelets: A New Primitive for Data Mining, ACM SIGKDD 2009*)
  - **single time series + nonmetric DTW** (*Keogh et al., LB_Keogh Supports Exact Indexing of Shapes under Rotation Invariance with Arbitrary Representations and Distance Measures, VLDB 2006*)

addressing Q1, Q2, Q6

MAMs in search engine architectures

- MAM as **single-model engine**, where MAM is essential,
  - complex similarity produces single (final) ranking
    - simple kNN/range search
  - more complex query types?
    - reverse kNN, skylines, multi-example queries, joins
  - mapping from more complex (nonmetric) spaces

addressing Q3, Q4, Q5

MAMs in engine architectures (cont.)

- MAMs in **hybrid-model engine,** MAMs still essential, but kind of „middleware“
  - multiple (local) descriptors + metric measures = multiple MAM indexes
  - allows to include also keyword search
  - aggregation system produces the final result from the intermediate results produced by MAMs
    - top-k, reranking, user preferences, learning, user feedback
  - e.g., *Berreti et al., Retrieval by Shape Similarity with Perceptual Distance and Effective Indexing, IEEE Tran. on Mult. 2(4), 2000*
    - multi-query using M-tree + nonmetric ranking of partial results

addressing Q3, Q4, Q5

MAMs in engine architectures (cont.)

- MAM as **low-level tool**, i.e., MAM is just a support
- example – MAM as implementation of visual words vocabulary
  - *Philbin et al., Object retrieval with large vocabularies and fast spatial matching, CVPR, IEEE, 2007*
  - MAM could be used to organize the vocabulary of visual words
    - an image consist of segments (> 3000), each is transformed to 128D SIFT descriptor
    - each segment is mapped to a visual word (a representative SIFT descr.)
      - metric similarity used: L_2 or L_1
    - vocabulary of visual words (10^6) serves for "second feature extraction", producing vector of linear combination of visual words (tf-idf weights)
    - the vocabulary needs fast building/access
      - MAM
  - the main search engine is based on classic vector model of IR
    - retrieval using inverted list + cosine measure

addressing Q3, Q4, Q5

Bidirectional motivation

- two separate worlds (databases vs. domains)
  - need to bridge the gaps
    - terminology (big problem!)
    - separation of similarity model from the search algorithm
- MAM-side requirements
  - expensive metric distances and/or large databases
- domain-side requirements
  - effective retrieval (sophisticated similarity model)
  - reasonably cheap search
- interdisciplinary research crucial
  - top expertise in databases + conceptual knowledge in domain (and vice versa)
  - otherwise "no interface"
    - danger for database research: solving **toy problems**
    - danger for a domain research: **quantitative** limits imply **qualitative** limits

Bidirectional motivation (cont.)

Usual thinking stereotype:

variant (a) all-in-one algorithm

monolithic retrieval solution
(e.g., BLAST - protein search)

variant (b) separated similarity

modeling **cheap** similarity
(due to sequential search)

efficient indexing
(optional bonus)

Modeling augmented by (metric) indexing:

modeling **expensive** similarity
(future indexing required)

efficient indexing
(necessary)

addressing Q1, Q3, Q6

One more provocation at the end ☺

- many papers claim their new MAM is *„an order of magnitude faster than the others“*
  - after the decades the similarity search should transitively become costless!
  - hmm, probably just not proper experimental practices
  - fair comparison needed
    - standardized datasets, queries and code (SISAP library)
    - do optimize/tune also the competing algorithms
    - do not twist the experimental setup to handicap the others
    - **include realtime cost** (as discussed earlier)

addressing Q2

Thank you for your attention!