AALBORG UNIVERSITY

Topics in Geo-Spatial Data Management: Spatial Keyword Querying and Beyond

Christian S. Jensen

csj@cs.aau.dk

Big Data

Every day, we create 2.5 quintillion bytes of data — so much that 90% of the data in the world today has been created in the last two years alone. This data comes from everywhere: sensors used to gather climate information, posts to social media sites, digital pictures and videos, purchase transaction records, and cell phone GPS signals to name a few. This data is **big data.**

http://www-01.ibm.com/software/data/bigdata/

Big Data

The notion of big data refers to data management loads for which conventional techniques fall short and that call for new approaches.

Volume

- The amount of data

Velocity

- The update loads or query latency or throughput requirements

Variety

- The number and diversity of data sources

Outline

- A briefing on recent and ongoing research in three areas of geo-spatial data management.

- Keyword querying
- Eco-routing in spatial networks
- Managing high-velocity mobile location data

Outline: Keyword Querying

- Motivation
- Top-*k* spatial keyword queries
- Collective queries
- Group queries

The Mobile and Spatial Web

- A quickly evolving mobile Internet infrastructure
  - Mobile devices, e.g., smartphones, tablets, laptops, navigation devices, glasses
  - Communication networks and users with access

- Mobile is a mega trend.
  - Google went “mobile first” in 2010.
  - Mobile data traffic 2020 = 2010 x 1000.

- Increasingly sophisticated technologies enable the accurate geo-positioning of mobile users.
  - GPS-based technologies
  - Positioning based on Wi-Fi and other communication networks
  - New technologies are underway (e.g., GNSSs and indoor).

Spatial Web Querying

- Total web queries
  - Google: 2011 daily average: 4.7 billion
- Queries with local intent
  - ”cheap pizza” vs. ”pizza recipe”
  - Google: ~20% of desktop queries; Bing: 50+% of mobile queries
- Vision: Improve web querying by exploiting accurate user and content geo-location
  - Smartphone users issue keyword-based queries
  - The queries concern web content representing places (POIs)
- Support different use cases
  - Nearest relevant POI – ”I want a bottle of water”
  - Exploratory or browsing behavior – ”I want a pair of shoes”

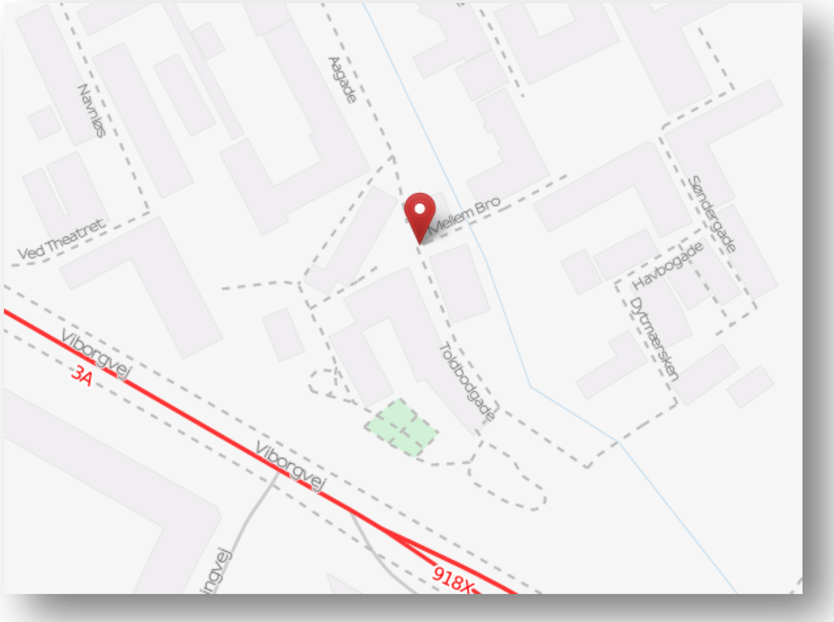
Spatial Web Objects

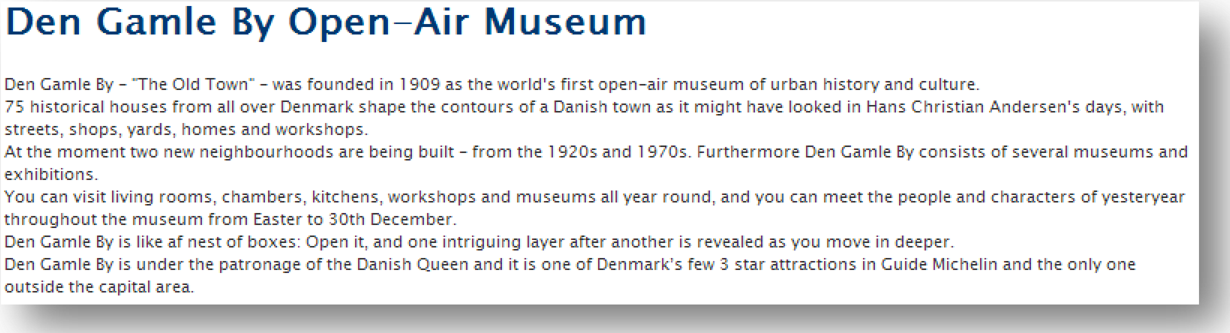
- Objects: $p = \langle \lambda, \psi \rangle$ (location, text description)

- Example:

λ = *(56.158889, 10.191667)*

ϕ = Den Gamle By Open-Air Museum
Den Gamle By - "The Old Town" – was founded in 1909 as the world's first open-air museum of urban history and culture…





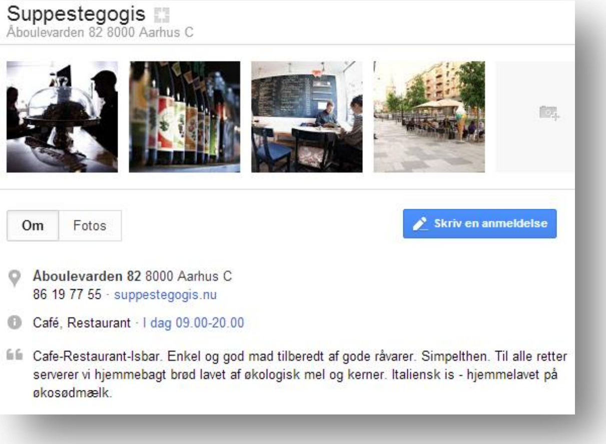
Den Gamle By Open–Air Museum

Den Gamle By – "The Old Town" – was founded in 1909 as the world's first open-air museum of urban history and culture.
75 historical houses from all over Denmark shape the contours of a Danish town as it might have looked in Hans Christian Andersen's days, with streets, shops, yards, homes and workshops.
At the moment two new neighbourhoods are being built – from the 1920s and 1970s. Furthermore Den Gamle By consists of several museums and exhibitions.
You can visit living rooms, chambers, kitchens, workshops and museums all year round, and you can meet the people and characters of yesteryear throughout the museum from Easter to 30th December.
Den Gamle By is like af nest of boxes: Open it, and one intriguing layer after another is revealed as you move in deeper.
Den Gamle By is under the patronage of the Danish Queen and it is one of Denmark's few 3 star attractions in Guide Michelin and the only one outside the capital area.

Spatial Web Objects – Sources

- Web pages with location

- Online business directories
  - Business name, location, categories, reviews, etc.
  - Example: Google Places



Suppestegogis

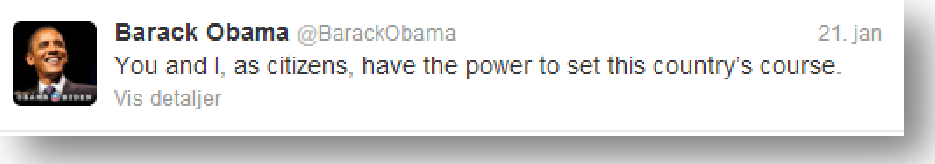
Åboulevarden 82 8000 Aarhus C

Åboulevarden 82 8000 Aarhus C
86 19 77 55 · suppestegogis.nu

Café, Restaurant · I dag 09.00-20.00

Cafe-Restaurant-Isbar. Enkel og god mad tilberedt af gode råvarer. Simpelthen. Til alle retter serverer vi hjemmebagt brød lavet af økologisk mel og kerner. Italiensk is - hjemmelavet på økosødmælk.

- Geocoded micro-blog posts
  - Example: Twitter
  - Messages with up to 140 characters



Barack Obama @BarackObama 21. jan
You and I, as citizens, have the power to set this country's course.
Vis detaljer

- Foursquare, Facebook Places, Navigation Devices

foursquare TOMTOM®

Facebook Places
Who. What. When. And now **where.**

GARMIN™
You'll be lost without it!

Top-*k* Spatial Keyword Query

- Objects: $p = \langle \lambda, \psi \rangle$ (location, text description)
- Query: $q = \langle \lambda, \psi, k \rangle$ (location, keywords, # of objects)

- Ranking function

$$rank_q(p) = \alpha \frac{\| q.\lambda, p.\lambda \|}{\max D} + (1-\alpha)(1 - \frac{tr_{q.\psi}(p.\psi)}{\max P}) \qquad 0 \le \alpha \le 1$$

  - Distance: $\| q.\lambda, p.\lambda \|$
  - Text relevancy: $tr_{q.\psi}(p.\psi)$
    - Probability of generating the keywords in the query from the language models of the documents

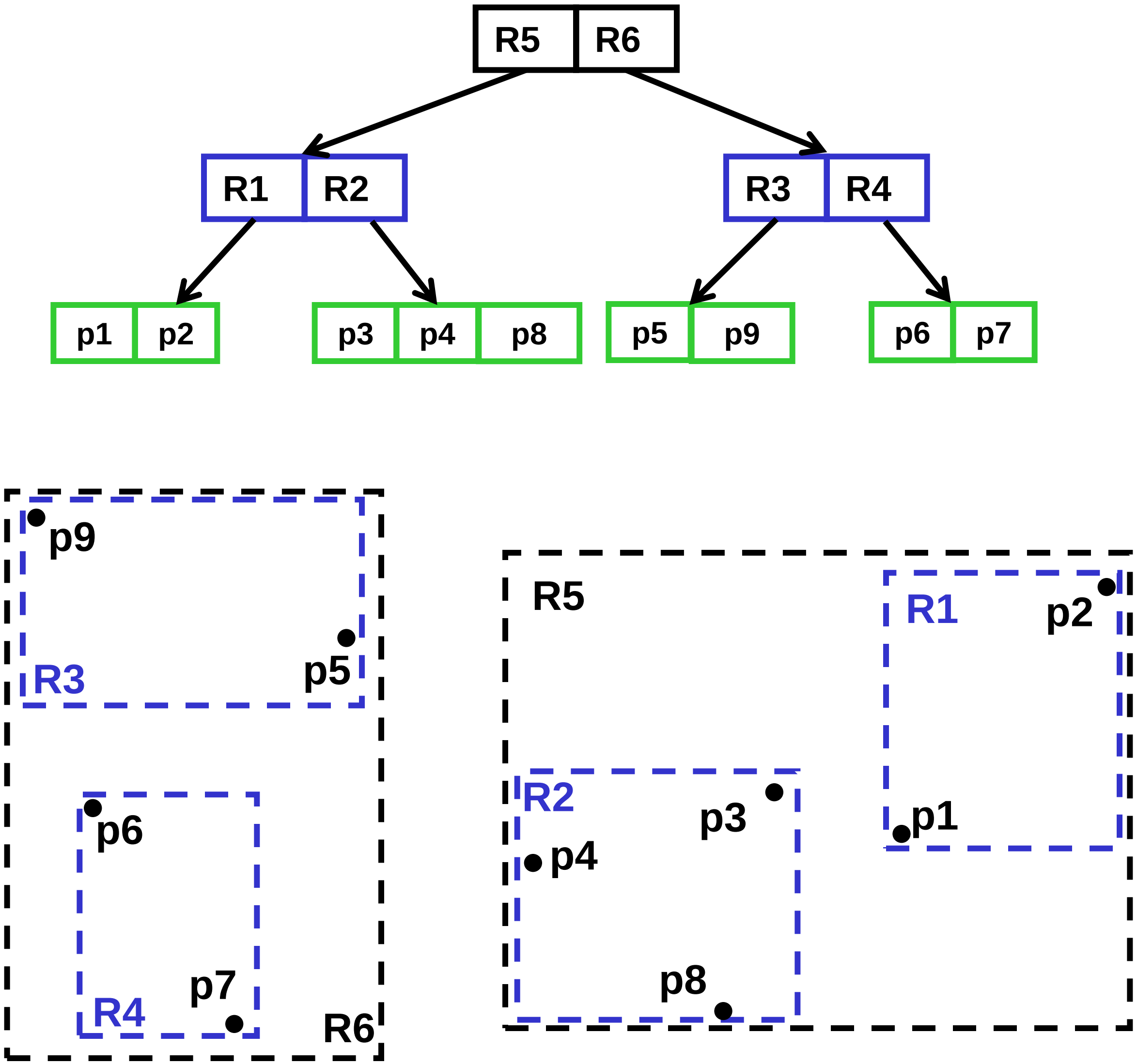
- Generalizes the *k*NN query and text retrieval

Spatial Keyword Query Processing

- How do we process spatial keyword queries efficiently?

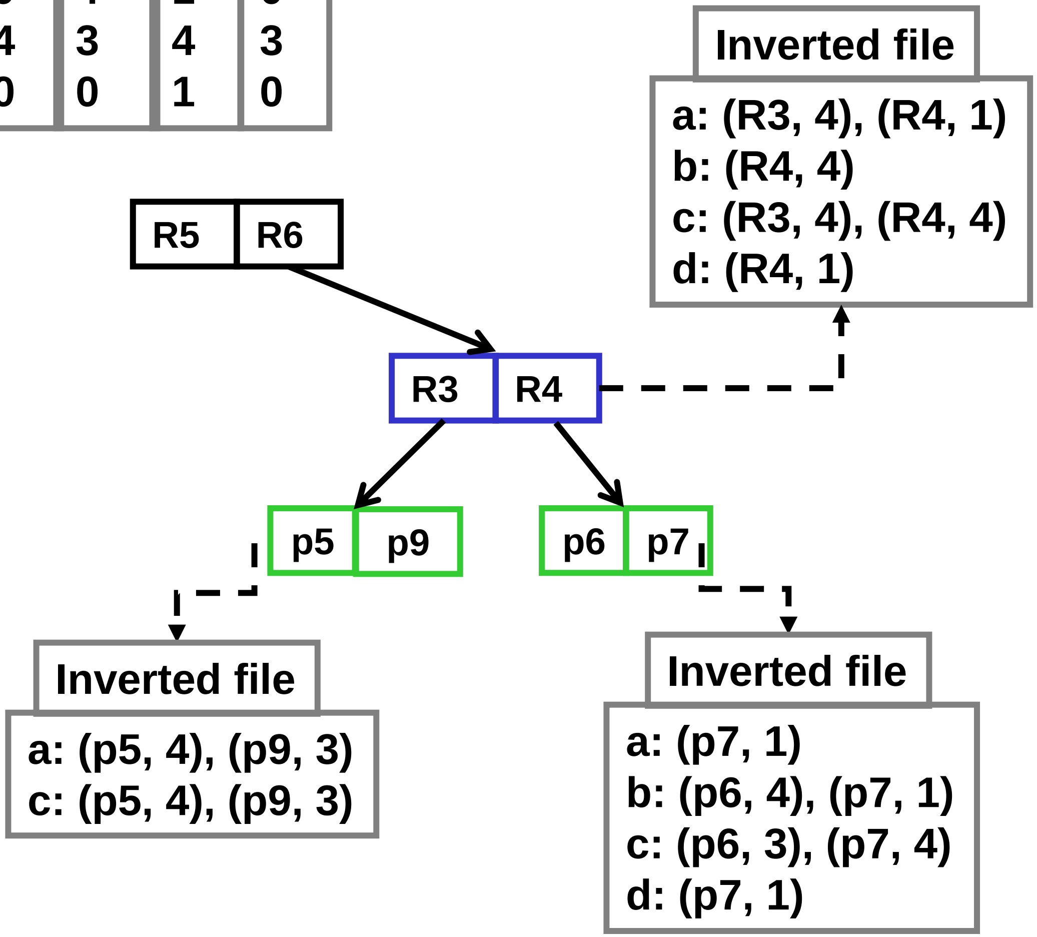
- Proposal
  - Prune both spatially and textually in an integrated fashion
  - Invent indexing to accomplish this

- The IR-tree [Cong et al. 2009 ; Li et al. 2011; Wu et al. 2012]
  - Combines the R-tree with inverted files
  - R-tree: good for spatial
  - Inverted files: good for text


R5
R6
R1
R2
R3
R4
p1
p2
p3
p4
p8
p5
p9
p6
p7
p9
p5
R3
p6
p7
R4
R6
R5
R1
p2
R2
p3
p4
p1
p8

Object descriptions

	p5	p6	p7	p9
a	4	0	1	3
b	0	4	1	0
c	4	3	4	3
d	0	0	1	0



Inverted file

a: (R3, 4), (R4, 1)
b: (R4, 4)
c: (R3, 4), (R4, 4)
d: (R4, 1)

Inverted file

a: (p5, 4), (p9, 3)
c: (p5, 4), (p9, 3)

Inverted file

a: (p7, 1)
b: (p6, 4), (p7, 1)
c: (p6, 3), (p7, 4)
d: (p7, 1)

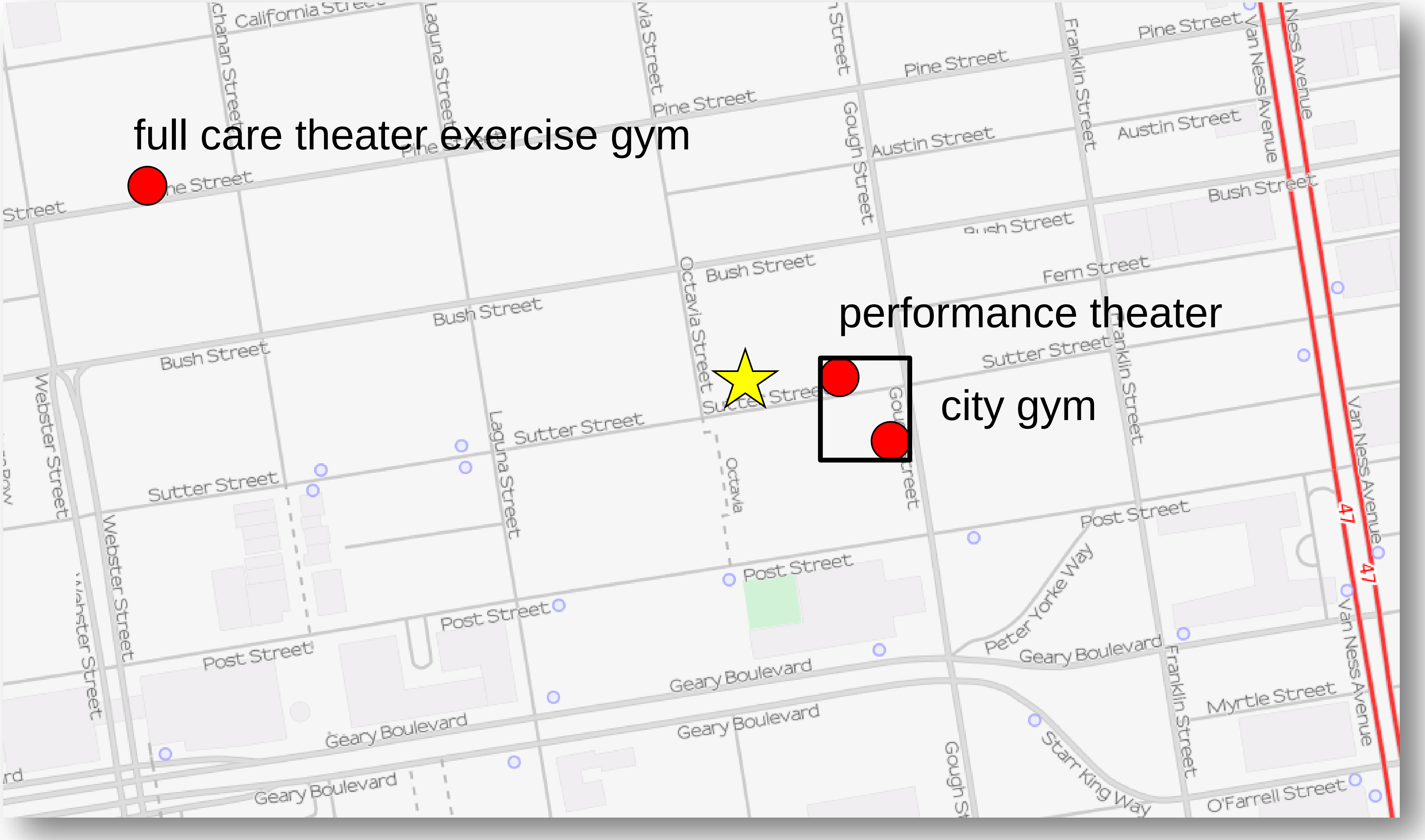
Collective Spatial Keyword Querying

- So far, the granularity of a result has been a single object

- We may want to return *sets* of objects that collectively satisfy a query.

The Collective Spatial Keyword Query

- Query location: ★ (Kenmore Hotel, SF)
- Query keywords: theater, gym



The Collective Spatial Keyword Query

- Objects: $o = \langle \lambda, \psi \rangle$ (location and text description)
- Query: $Q = \langle \lambda, \psi \rangle$ (location and keywords)

- The result is a group of objects χ satisfying two conditions.
  - $Q.\psi \subseteq \bigcup_{o \in \chi} o.\psi$
  - *Cost(Q, χ)* is minimized.

- $Cost(Q, \chi) = \alpha C_1(Q, \chi) + (1 - \alpha) C_2(\chi)$
  - C_1(.,.) depends on the distances of the objects in χ to Q.
  - C_2(.) characterizes the inter-object distances among objects in χ.
  - α balances the weights of the two components.

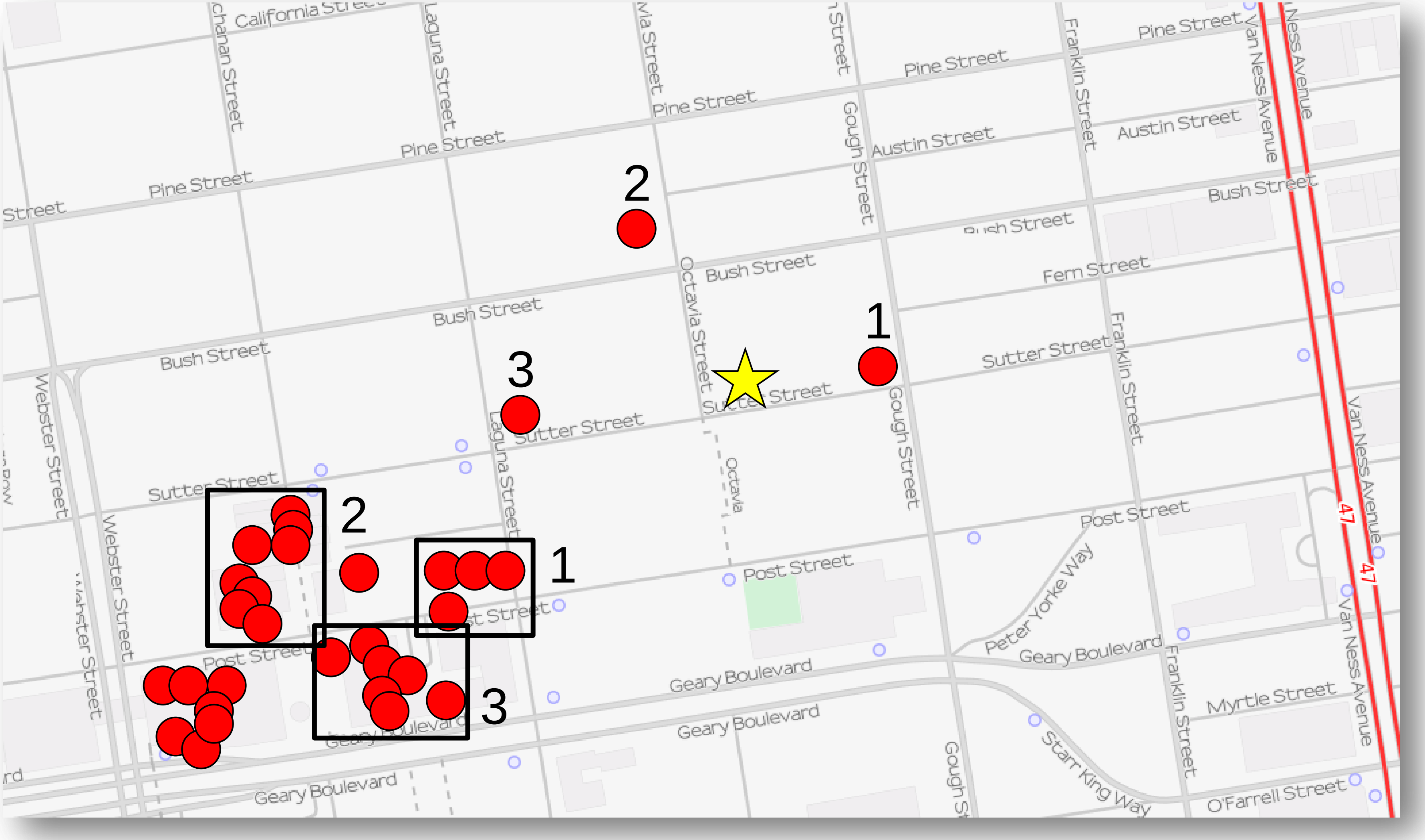
Collective Query Variants

- Cost function: $Cost(Q,\chi) = \sum_{o \in \chi} Dist(o,Q)$
- Application scenario
  - The user wishes to visit the places one by one while returning to the query location in-between.
  - Go to the hotel between the museum visit and the jazz concert
  - NP-hard: proof by reduction from the Weighted Set Cover problem

- Cost function: $Cost(Q,\chi) = \max_{o \in \chi} Dist(o,Q) + \max_{o_i, o_j \in \chi} Dist(o_i, o_j)$
- Application scenario
  - Visit places without returning to the query location in-between
  - E.g., go to a movie and then dinner
  - NP-hard: proof from reduction from the 3-SAT problem

Top-*k* Groups Query Illustration

- Query location: ★ (Kenmore Hotel, SF)
- Query keyword: Restaurant



Top-*k* Groups Query

- Objects: $p = \langle \lambda, \psi \rangle$ (location, text description)
- Query: $q = \langle \lambda, \psi, k \rangle$ (location, keywords, # of objects)

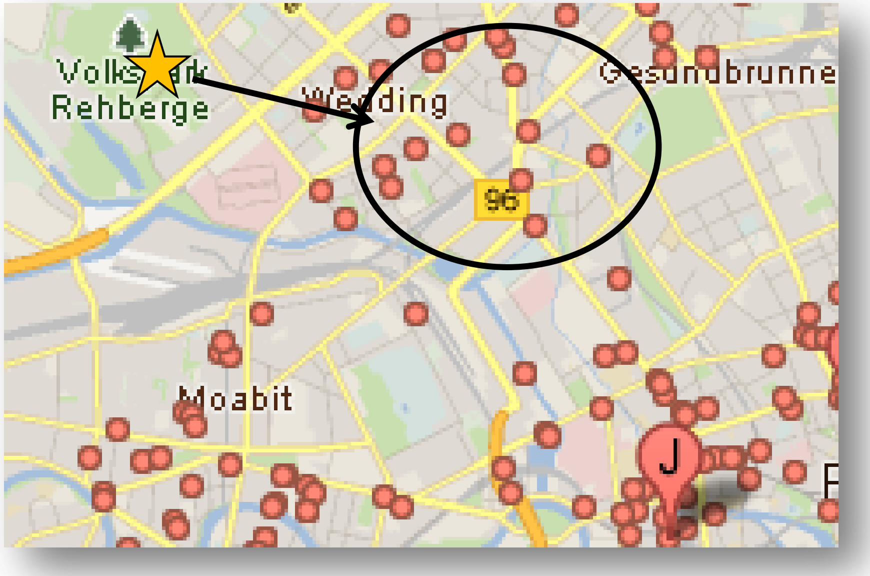
- Ranking function

$$rank_q(G) = \alpha \frac{\beta dist(q.\lambda, G) + (1-\beta) diam(G)}{\max D} + (1-\alpha) TR_G(q.\psi, G)$$

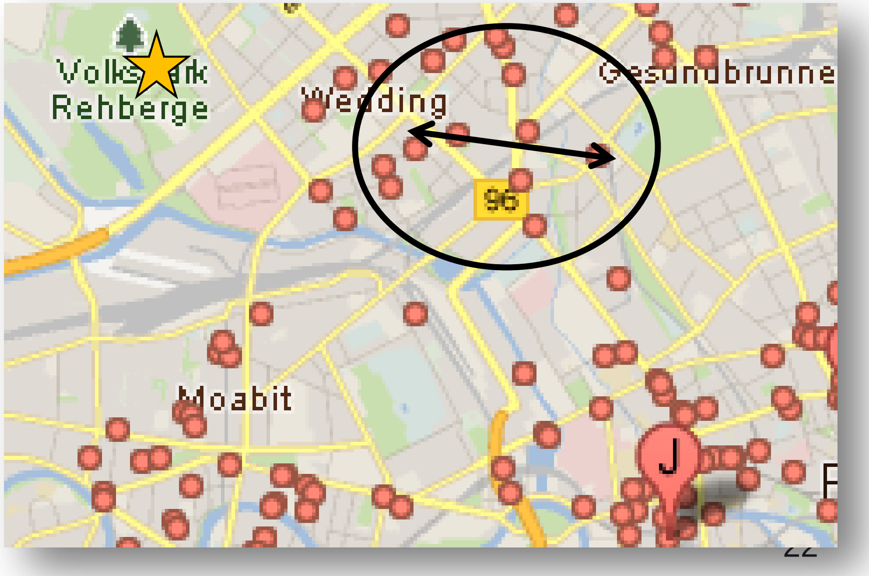
  - $0 \le \alpha \le 1$ and $0 \le \beta \le 1$
  - Distance: $dist(q.\lambda, G) = \min_{o \in G} \| q.\lambda, o.\lambda \|$
  - Diameter: $diam(G) = \max_{o_1, o_2 \in G} \| o_1.\lambda, o_2.\lambda \|$
  - The text relevance function favors large groups and groups where the query keywords are distributed evenly among group objects.
  - Groups are disjoint

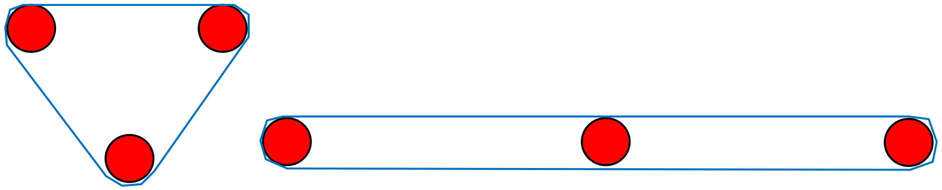
Problem Definition

- Distance to the group
  - Distance to the nearest object



- Group diameter
  - Maximum distance between two objects
  - Better than, e.g., area of the convex hull





Challenges

- Structured queries and Amazon-style and social queries
  - Ample opportunities for much more customization of results
- Build in feedback mechanisms
  - "Figuring out how to build databases that get better the more people use them is actually the secret source of every Web 2.0 company" –Tim O'Reilly
- Tractability versus utility
  - The area is prone to NP-hardness
- Avoid parameter overload
  - Problem vs. solution parameters
  - Hard-to-set, impossible-to-set parameters – relevance decreases exponentially with the number of such parameters
- User evaluation
  - If you can't measure it, you can't improve it.
  - Challenging – particularly for someone who used to study joins.

Outline: Spatial Networks

- Motivation
- Setting
- Challenges

Motivation – Eco-Routing

- The reduction of greenhouse gas (GHG) emissions from transportation is essential to combat global climate change.
  - EU: reduce GHG emissions by 30% by 2020.
  - G8: a 50% GHG reduction by 2050.
  - China: a 17% GHG reduction by 2015.
- Eco-routing can reduce vehicular impact by up to 20%.

Setting

- Use an existing road network model
  - E.g., OpenStreetMap
  - Germany: 10 million edges (rough estimate)
- Use tracking data from vehicles
  - E.g., GPS data
- Use fuel consumption data from vehicles

GPS Data

- Data warehouse statistics (as of November 28, 2013)
  - Number of data sources (and NDAs): 17
    - Daily data from 4 (~1.6 million per day); irregular batches from 4 (2 small and 2 big); finished projects: 9
  - Total number of fact table rows (before/after cleaning): 2,386,420,008/**2,372,212,609**
  - Number of vehicles: 23,660
  - Number of trajectories (trips): 2,015,109
  - Rows with fuel data: 118,945,566 (~140,000 per day)
  - Rows from EVs: 110,663,568
  - Number of rows per year: (2000, 209,356), (2001, 40,912,564), (2002, 17,077,612), (2003, 4), (2004, 464,068), (2005, 380), (2006, 33,491,547), (2007, 182,991,309), (2008, 161,383,000), (2009, 92,795,488), (2010, 172,246,375), (2011, 221,223,905), (2012, 674,314,285), (2013, 664,797,761)

Setting

- The setting may be modeled as a system of streams, one per edge
  - Spatial
  - Spatio-temporally correlated
  - Sparse
- Real, unlike envisioned smart dust applications!
- Infer eco weights of edges from the GPS data and a ”lifted” 3D spatial network using vehicular environmental impact models.

Deterministic vs. Uncertain Weights

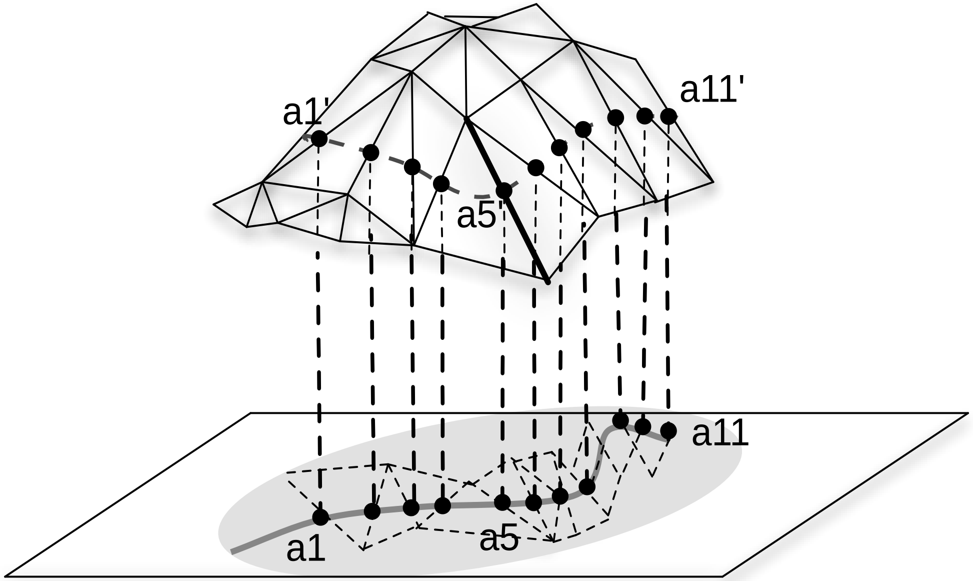
- Deterministic: Each interval has a deterministic weight.
  - E.g.: (0:00, 7:00]: 10 mg; (7:00, 9:00]: 18 mg; (9:00, 15:00]: 12 mg;…

- Uncertain: Each interval has a random variable that is modeled by, e.g., a normal distribution or a histogram
  - E.g.: (0:00, 7:00]: 8 to 12 mg, *N*(9, 10); (7:00, 9:00]: 13 to 23 mg, *N*(18, 10); (9:00, 15:00]: 8 to 12 mg, *N*(10, 20); …

	Coverage	**Temporal Granularity**	**Accuracy**
Deterministic Weights	High (all edges)	Coarse (e.g., pre-defined PEAK)	Good (better than speed limits)
Uncertain Weights	Low (only “hot” edges)	Fine (e.g., 15-min intervals)	High (capture the travel cost distributions)

Challenge: Spatial Network Lifting

- Build a 3D road network from 3D laser scan data and a 2D road network.
- Step 1: Create a Triangulated Irregular Network (TIN) from a 3D laser scan.
- Step 2: Project 2D polylines to the TIN to obtain 3D polylines and thus a 3D road network.



- Note: Big laser scan data...

Challenges: Assigning Eco-Weights

- Deterministic, static weights, all edges
  - Challenges: infer weights for “cold” edges – edges not covered by GPS data.
  - Graph weight annotation
    - Quantify edge similarities based on traffic flows derived from the topology of the road network.
    - Propagate the weights on hot edges onto similar cold edges.
- Uncertain, static weights, hot edges
  - Capturing the weights of hot edges using time dependent histograms.
  - Challenges: compact representations of the histograms, histograms for routes.
- Dynamic weights, hot edges
  - Inferring near-future weights of hot edges as GPS data streams in.
  - Challenges: correlated, sparse, heterogeneous.

Challenges: Routing

- Stochastic skyline route planning under time-varying uncertainty
  - Given a source, a destination, and a trip starting time, identify pareto-optimal routes considering multiple travel costs, e.g., travel times, GHG emissions, distances.
  - Account for probabilities
- Continuous routing that supports real-time weight updates
  - Maintain up-to-date routes for vehicles according to the up-to-date weights and the current vehicle locations.

Next Steps

- Automated trade-off between weight level of detail and available data.
- Stochastic routing at 40 milliseconds.
- Route-based weights instead of segment-based weights.
- Modeling spatio-temporal congestion from data.
- Detect “black spots” before they occur.

High-Velocity Location Data: Outline

- Workloads
- Dual and single data structure approaches
- Experimental study and findings

Workloads

- Specific scenario assumed: country-wide vehicle tracking and intelligent transport system services
  - 10 million vehicles, 10 m/s speeds, 10 meter accuracy: 10 million updates/s
  - Plus queries, represented by range queries

- A main memory problem
- Exploit the parallelism offered by modern processors

- Fast single-object updates (nanoseconds, roughly)
- Relatively long-running queries (microseconds, roughly)
- Handle interference between queries and updates
  - An update waiting for a query is analogous to a traveler on an 8 hour trip being told that there is a slight delay so that the trip will take 8000 hours or ~1 year.

Dual Data Structure Approaches

- Idea: Isolate queries from updates using two copies of the data.

- A static, indexed copy is used for querying.
- A live copy is used for updates.
  - MOVIES: a log
  - TwinGrid: an up-to-date index

- Frequently refresh (called snapshotting) the static copy used for querying so that query results are reasonably fresh.

Single Data Structure Approach – PGrid

- The snapshotting solutions have problems!
- Stale query results
- Stop-the-world problem
- Waste of CPU cycles on frequent snapshotting
- The snapshotting frequency is difficult to set.

- Can we solve these problems?
- Yes: use one copy for both updates and queries.

- Allow insertions and deletions to happen concurrently with queries.
- Make sure that updates are atomic.
- Perform as little locking as possible.

Experimental Study

- Use a variety of multi-core platforms
- Considered half a dozen main-memory indexes
- Massive workloads
  - Simulated monitoring of up to 40M (10M default) moving objects in the road-network of Germany
- Findings
  - Throughput varies from about 10 to about 30 million operations per second (predominantly updates, but also range queries).
  - PGrid is generally best.

Challenges

- *k*NN queries, continuous queries, joins
- Integration with the handling of past states
  - These may not fit in memory.
  - Only the current states can be updated (partial persistence).
- (Arbitrarily) fast enough at lowest cost
  - Use only as much main memory as currently needed.
- Start thinking about self-driving vehicles
  - "…looking back and saying how ridiculous it was that humans were driving cars." [Sebastian Thrun, TED2011]
  - Machines don't make mistakes, human do.

Summary

- Briefing on ongoing research in three areas of geo-spatial data management

- More data is becoming available
- More devices are being networked
- Increasing needs

- Data driven research
- Tight integration with empirical studies

Acknowledgments

- Colleagues at Aalborg and Aarhus Universites and beyond.

- The EU ITN project, Geocrowd: www.geocrowd.eu
- The EU FP7 project, Reduction: www.reduction-project.eu
- The Obel Family Foundation: www.obel.com/en

Readings

- Bøgh, K. S., A. Skovsgaard, C. S. Jensen: GroupFinder: A New Approach to Top-K Point-of-Interest Group Retrieval , PVLDB 2013, demo
- Cao, X., L. Chen, G. Cong, C. S. Jensen, Q. Qu, A. Skovsgaard, D. Wu, M. L. Yiu: Spatial Keyword Querying. ER, pp. 16-29 (2012)
- Wu, D., M. L. Yiu, G. Cong, and C. S. Jensen: Joint Top-K Spatial Keyword Query Processing. TKDE, 24(1): 1889-1903 (2012)
- Cao, X., G. Cong, C. S. Jensen, J. J. Ng, B. C. Ooi, N.-T. Phan, D. Wu: SWORS: A System for the Efficient Retrieval of Relevant Spatial Web Objects. PVLDB, 5(12): 1914-1917 (2012), demo
- Wu, D., G. Cong, and C. S. Jensen: A Framework for Efficient Spatial Web Object Retrieval. VLDBJ, 21(6): 792-822 (2012)
- Cao, X., G. Cong, C. S. Jensen, B. C. Ooi: Collective Spatial Keyword Querying. SIGMOD, pp. 373-384 (2011)
- Cong, G., C. S. Jensen, D. Wu: Efficient Retrieval of the Top-k Most Relevant Spatial Web Objects. PVLDB 2(1): 337-348 (2009)

Readings

- Guo, C., Y. Ma, B. Yang, C. S. Jensen, M. Kaul: EcoMark: evaluating models of vehicular environmental impact. SIGSPATIAL/GIS 2012.
- Kaul, M., B. Yang, C. S. Jensen: Building Accurate 3D Spatial Networks to Enable Next Generation Intelligent Transportation Systems. MDM 2013.
- B. Yang, M. Kaul, C. S. Jensen: Using Incomplete Information for Complete Weight Annotation of Road Networks. IEEE TKDE, to appear.
- Yang, B., C. Guo, C. S. Jensen: Travel Cost Inference from Sparse, Spatio-Temporally Correlated Time Series Using Markov Models. PVLDB 2013.
- Yang, B., C. Guo, C. S. Jensen, M. Kaul, S. Shang: Stochastic Skyline Route Planning Under Time-Varying Uncertainty. ICDE 2014.

- Sidlauskas, D., S. Saltenis, C. S. Jensen: Parallel main-memory indexing for moving-object query and update workloads. SIGMOD 2012: 37-48
- Sidlauskas, D., K. A. Ross, C. S. Jensen, S. Saltenis: Thread-Level Parallel Indexing of Update Intensive Moving-Object Workloads. SSTD 2011: 186-204
- Sidlauskas, D., C. S. Jensen, S. Saltenis: A comparison of the use of virtual versus physical snapshots for supporting update-intensive workloads. DaMoN 2012: 1-8