Yandex

ETH zürich

Parameter-Free Discovery and Recommendation of Areas-of-Interest

Dmitry Laptev, Alexey Tikhonov,
Pavel Serdyukov, Gleb Gusev

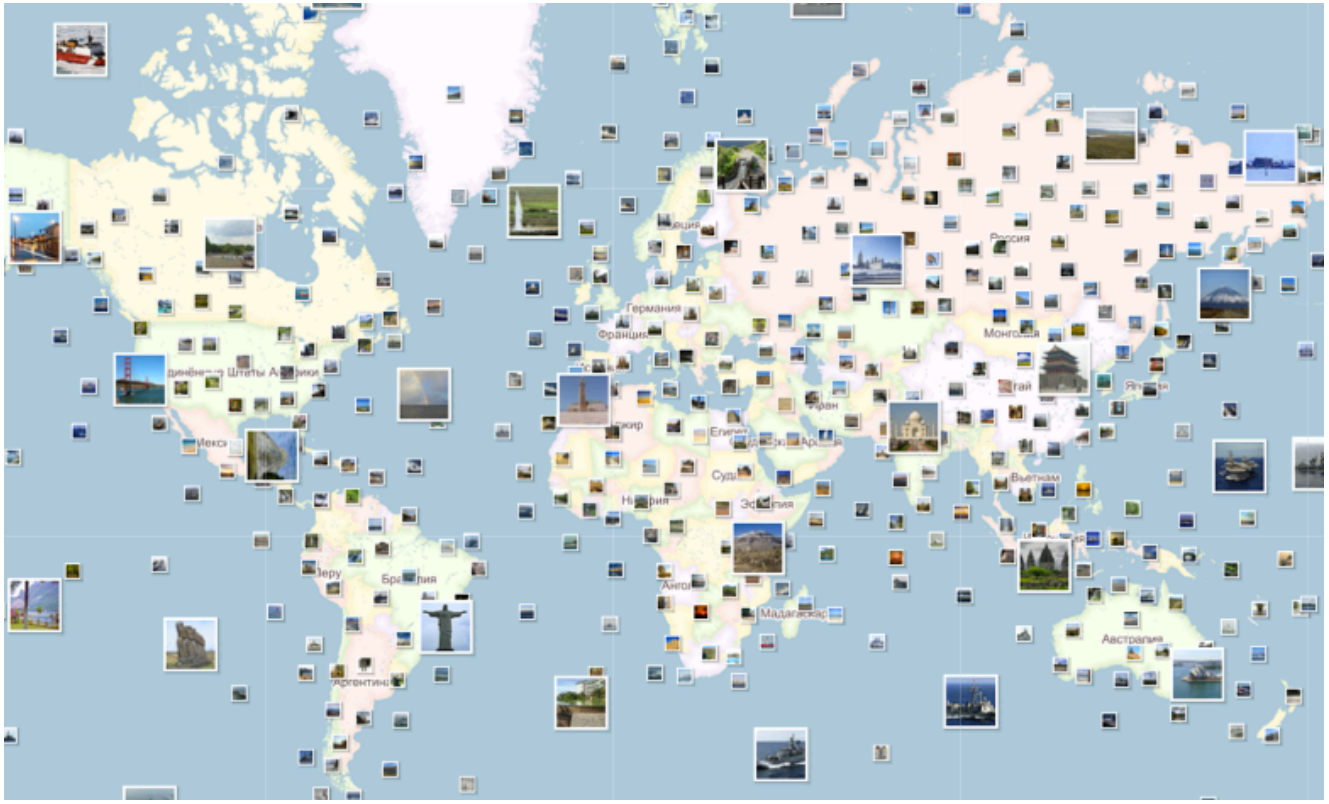
Outline

- Motivation
  - Why Areas-of-Interest?
  - Why parameter-free?
- Method description:
  - Density estimation
  - Watershed partitioning
  - Recommendations
- Results and conclusions

Motivation

- As a tourist you want to:
  - Visit attractions or spend some spare time
  - Discover monuments, squares, parks
  - Cover the most, but meet my time constraints
- Problems:
  - Guidebooks are not always available
  - Time-driven trip planning is hard

Motivation



Geo-tagged photos cover the most attractive places.
Can we discover these places automatically?

Motivation

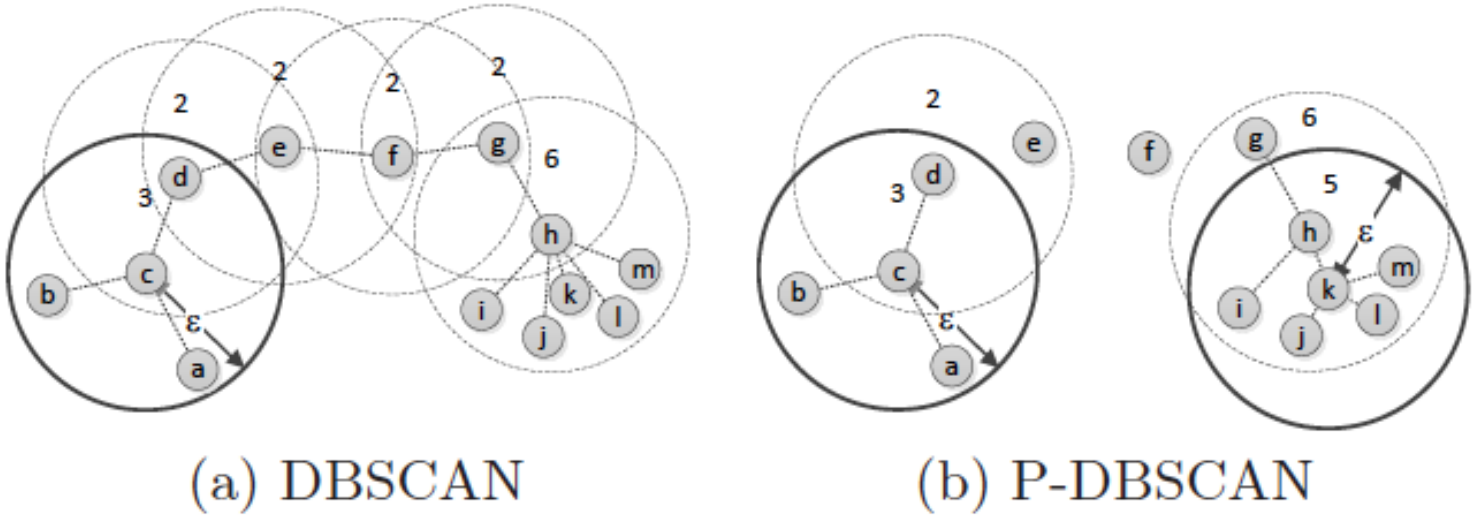
- The proposed method:
  - Data: a set of geo-tagged photos
  - Result: attractive areas recommendations
- Novelty:
  - Areas-of-Interest, not just Points
  - Non-parametric algorithm
  - Provides better recommendations

Why Areas-of-Interest?

- Points-of-interest (POI)
  - + Perfect for monuments, buildings, etc.
  - – Does not discover spatially distributed objects: parks, streets, river banks, squares
  - – Planning is hard: is it better to visit three points close to each other, or one point away?
  - – Points are more subjective than areas
- Solved by Areas-of-Interest (AOI)

Areas-of-Interest baselines

- Basically any 2d-clustering techniques

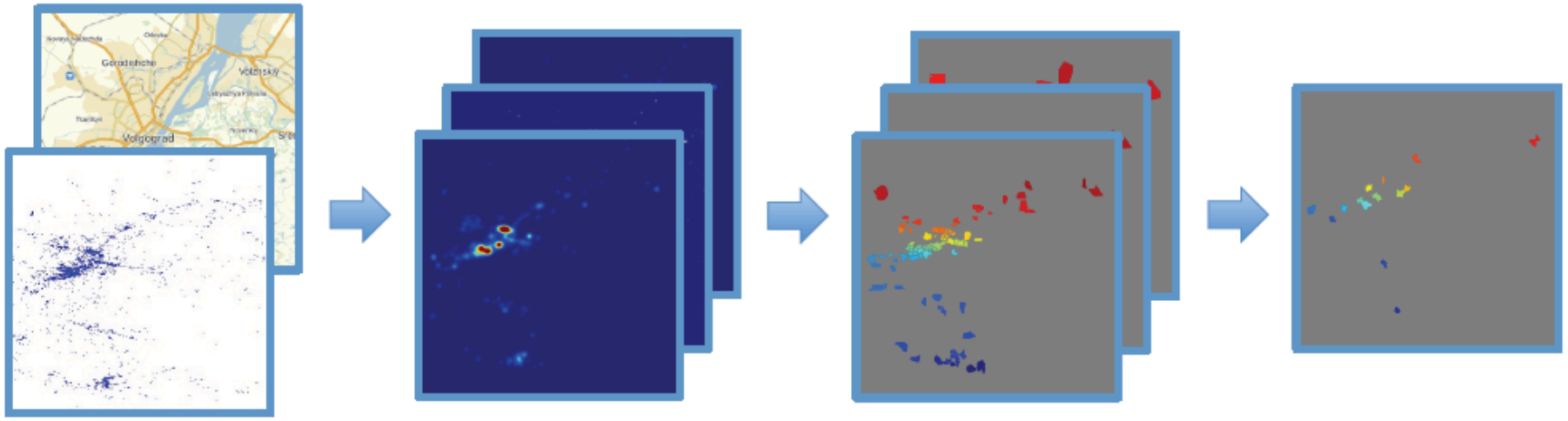


(a) M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise.
(b) S. Kisilevich, F. Mansmann, and D. Keim. P-DBSCAN: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos.

Why parameter-free?

- Cities are very different:
  - City area and population
  - Number of geo-tagged photos
  - Number of attractions
- Algorithm parameters should be different:
  - Tuning is hard and sometimes subjective
  - Idea: walking time is a universal constraint

Method description



Photos are projected to the map grid

Multiple density hypothesis are generated

AOI candidates are extracted through density partitioning

One candidate set of AOIs is selected, AOIs are ranked and recommended

Density estimation

- Gaussian kernel density estimation

$G_{p,q}$ – the number of photos in a cell *(p,q)* of a map grid (*K* x *K*)

$$D_{i,j}(h) = \frac{1}{N} \sum_{p=1}^{K} \sum_{q=1}^{K} \frac{G_{p,q}}{2\pi h^2} \exp\left(-\frac{(i-p)^2 + (j-q)^2}{2h^2} \right)$$

$D_{i,j}(h)$ – estimated density in a grid cell *(i,j)*

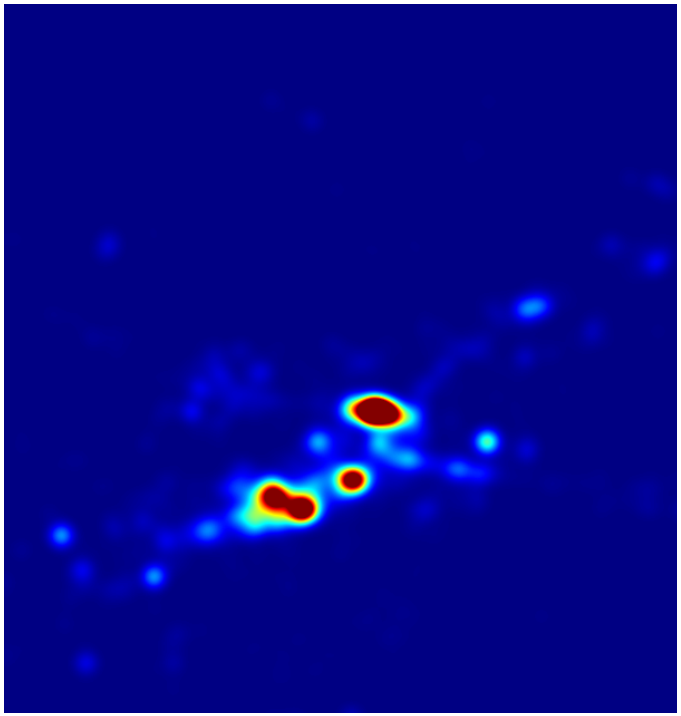
h – kernel bandwidth (temporary parameter

- Can be done very efficiently with Fast Fourier Transform

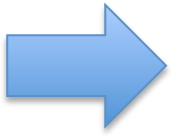
Watershed partitioning

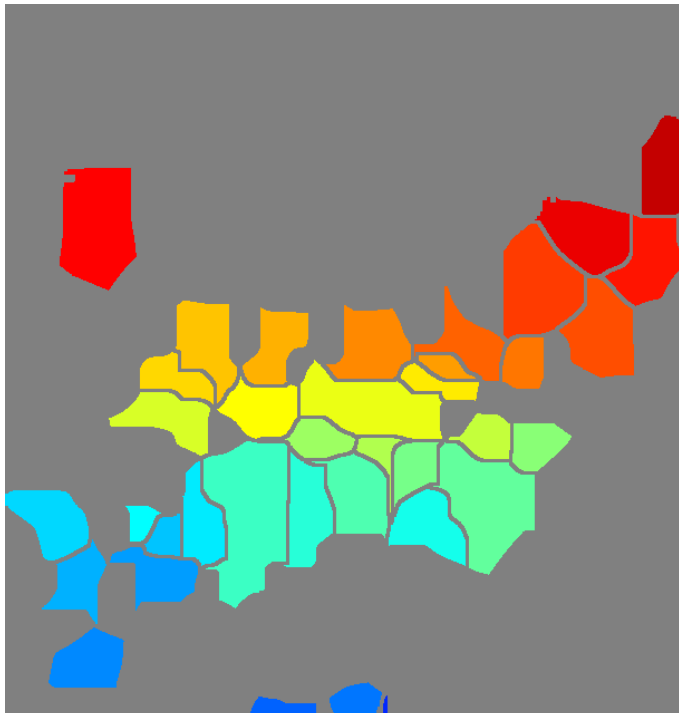
- Density peaks already show POIs
- To get AOIs, we need partitioning / clustering
- Watershed algorithm:
  - starts with density peaks,
  - propagates it to spatial clusters
  - non-parametric algorithm
- Label matrix $L(h) \in \{0, \ldots, R(h)\}^{K \times K}$
- Where $R(h)$ is the number of clusters

Watershed partitioning



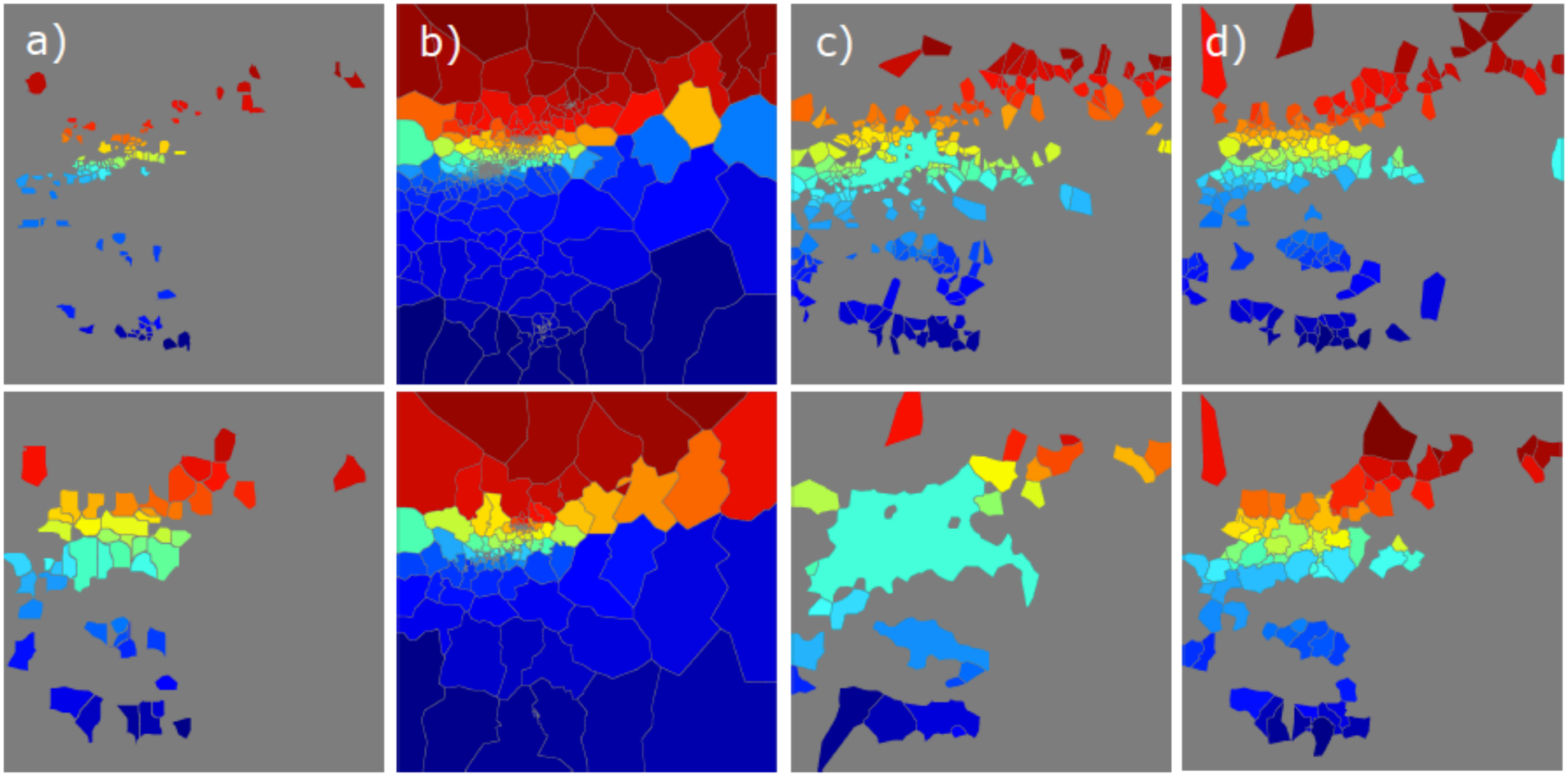
Estimated density
(with a given bandwidth)





Watershed partitioning:
one color shows one cluster

Different algorithms AOIs



Our algorithm K-Means DBSCAN P-DBSCAN

Parameter selection

- Idea: walking time corresponds to the average area of the AOI given a bandwidth parameter

$$\mathbb{E}_h(\text{area}) = \frac{1}{R(h)} C_{\text{long}} C_{\text{lat}} \delta_{\text{grid}} \sum_{r \in \{1,\ldots,R(h)\}} |\{(i,j) : L(h)_{i,j} = r\}|$$

- Select AOIs that take 10-15 minutes to walk around (*time as a constraint, not a bandwidth*):

$$h_{\text{opt}} = \max_h \{h : \mathbb{E}_h(\text{area}) \leq 0.1\}$$

Recommendation

- Once the bandwidth is selected, just rank all the Areas-of-Interest:

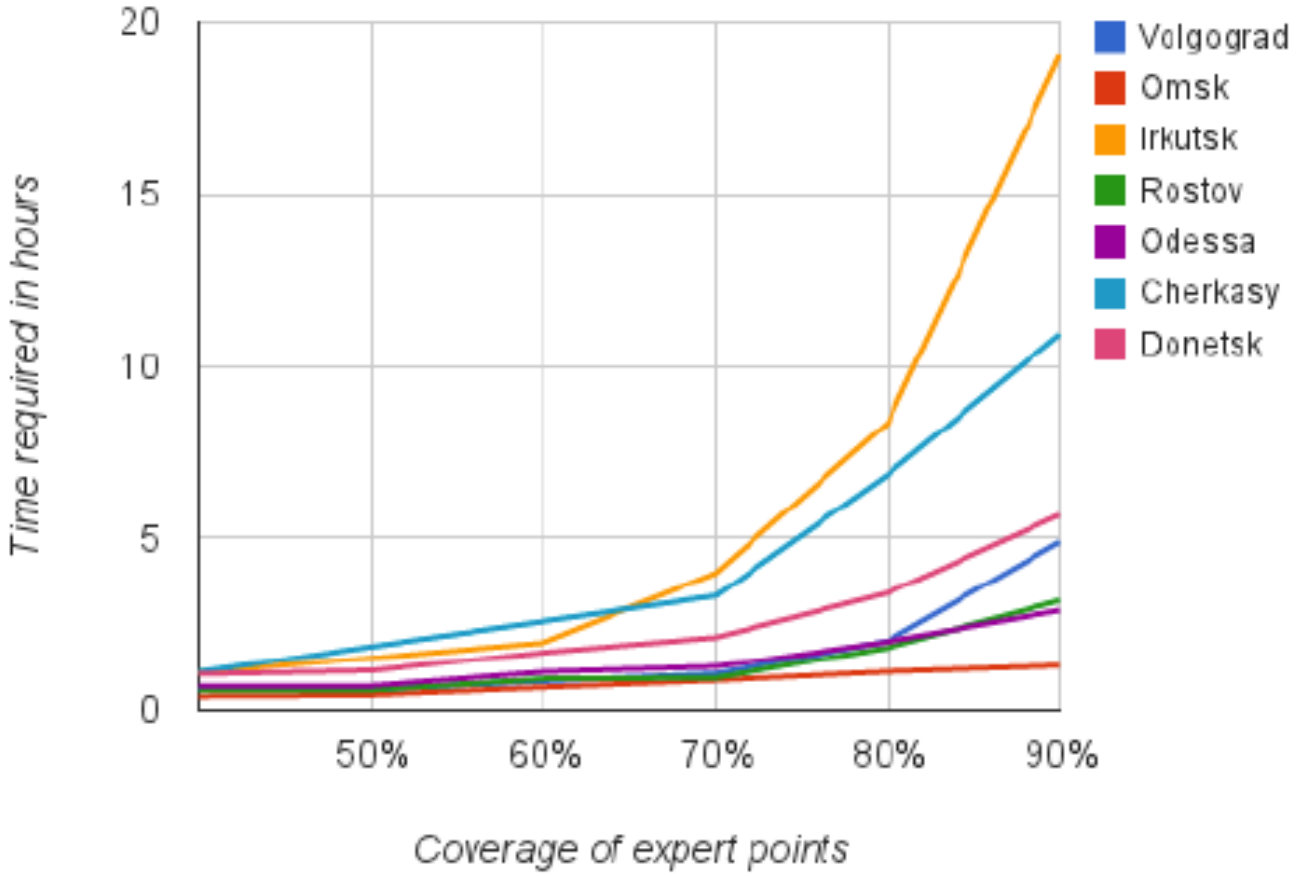
$$\mathrm{rank}(r) = \sum_{(i,j):\ L_{i,j}(h_{\mathrm{opt}})=r} D_{i,j}(h_{\mathrm{opt}})$$

- And recommend the number of AOIs that would fit tourist time constraints
  - If I have two hours, I will get ~10 AOIs

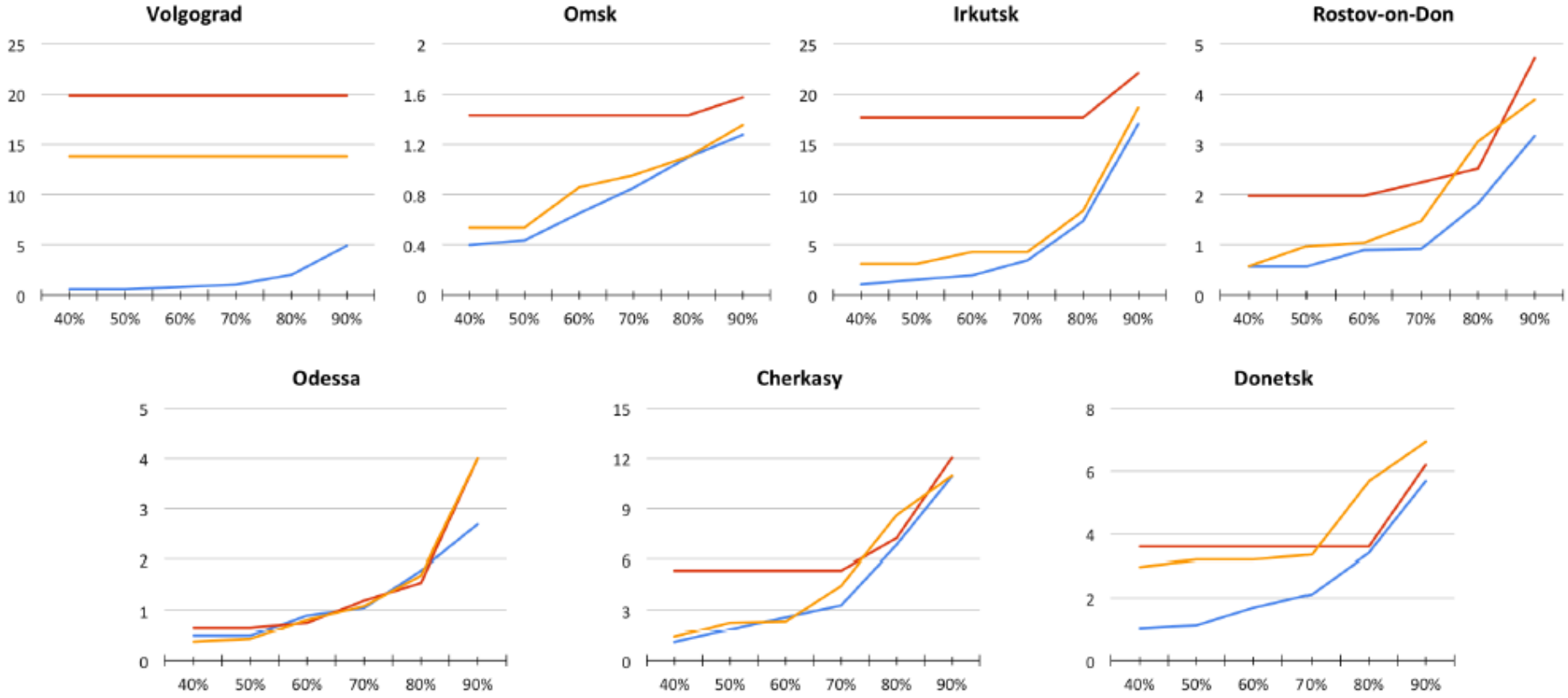
Results

- Dataset from Yandex.Photos
  - Volgograd, Omsk, Irkutsk, Rostov-on-Don, Odessa, Cherkasy, Donetsk (very different cities)
- Baselines
  - K-Means, DBSCAN, P-DBSCAN
- Metric
  - How long does it take to cover 40-90% of the selected POIs given the recommended AOIs?

Results: metric



Results



Red line: DBSCAN coverage, orange line: P-DBSCAN coverage, blue line: ours.
The lower – the better (less time required)

Results: ours vs. DBSCAN

City	60% coverge			80% coverage		
	DBSCAN	Ours	Gain	DBSCAN	Ours	Gain
Volgograd	19.9	0.8	2309%	19.9	2	888%
Omsk	1.4	0.7	120%	1.4	1.1	30%
Irkutsk	17.7	2	801%	17.7	7.4	140%
Rostov-on-Don	2	0.9	122%	2.5	1.8	38%
Odessa	0.8	0.9	-11%	1.6	1.8	-11%
Cherkasy	5.4	2.6	108%	7.2	6.9	6%
Donetsl	3.6	1.7	113%	3.6	3.4	5%

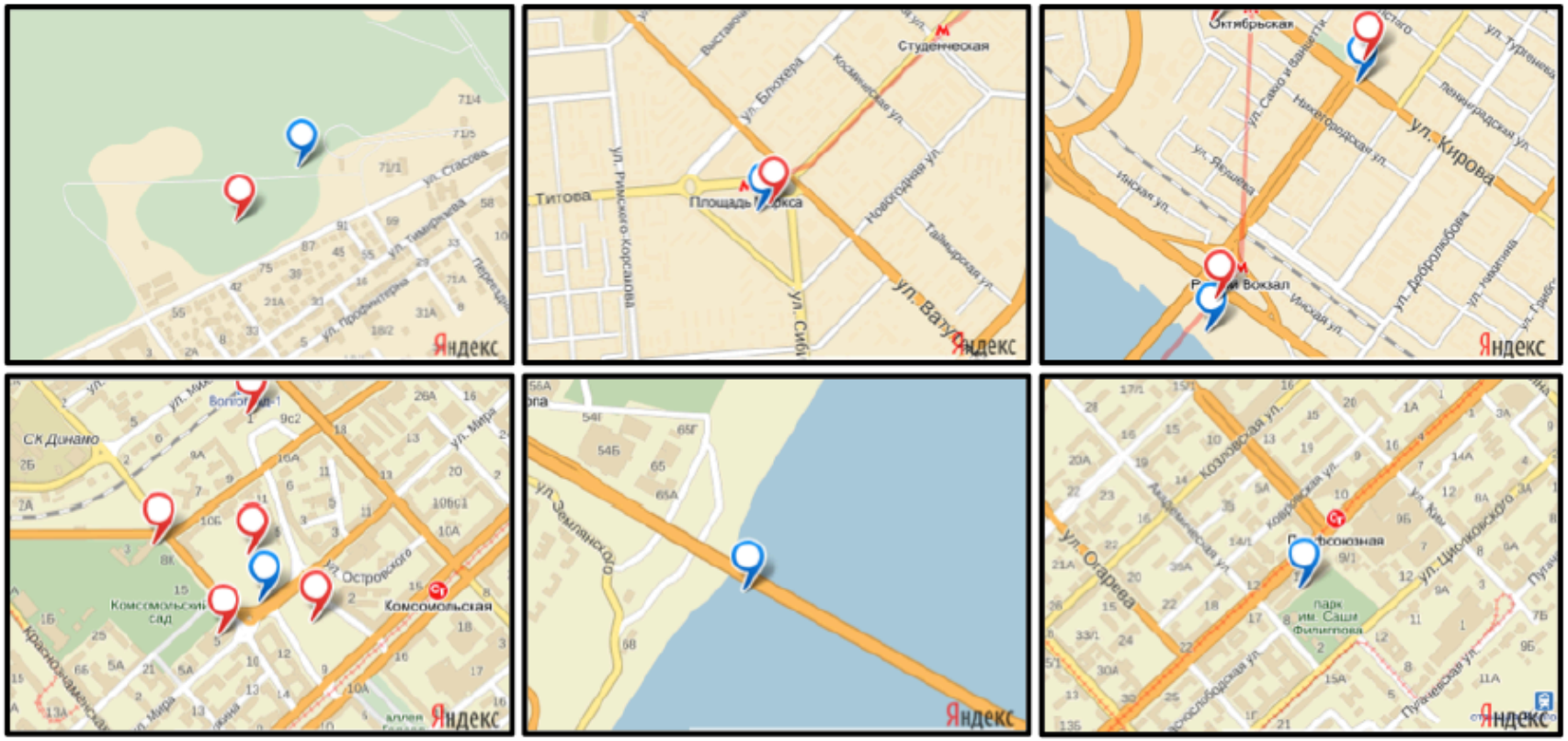
In most cities up to 2 times better.
Best case: 10 times faster exploration.
Worst case: only 15 minutes longer.

Results: ours vs. P-DBSCAN

City	60% coverge			80% coverage		
	P-DBSCAN	Ours	Gain	P-DBSCAN	Ours	Gain
Volgograd	13.8	0.8	1569%	13.8	2	584%
Omsk	0.9	0.7	32%	1.1	1.1	0%
Irkutsk	4.3	2	119%	8.4	7.4	14%
Rostov-on-Don	1	0.9	17%	3.1	1.8	68%
Odessa	0.8	0.9	-9%	1.7	1.8	-5%
Cherkasy	2.3	2.6	-9%	8.6	6.9	25%
Donetsl	3.2	1.7	89%	5.7	3.4	67%

In most cities up to 1.5 times better.
Best case: 5 times faster exploration.
Worst case: only 20 minutes longer.

Results



Red markers: POIs selected by experts. Blue markers: centers of AOI.
Some AOIs include many POIs, some only one,
some include none, but are still arguably relevant

Conclusions

- We propose a novel method
  - AOI discovery and recommendation
- Areas-of-Interest
  - better corresponds to tourist goals
- Non-parametric method
  - no tuning required, can be applied to every city
- Achieves consistently better results

Thanks for you attention

Questions & ideas are welcome

Contact me: dlaptev@inf.ethz.ch or http://dlaptev.org