

# Efficient Global 2D-3D Matching for Camera Localization in a Large-Scale 3D Map

Liu Liu, Hongdong Li, and Yuchao Dai

*In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017*

Presenter: B. Y. Huang

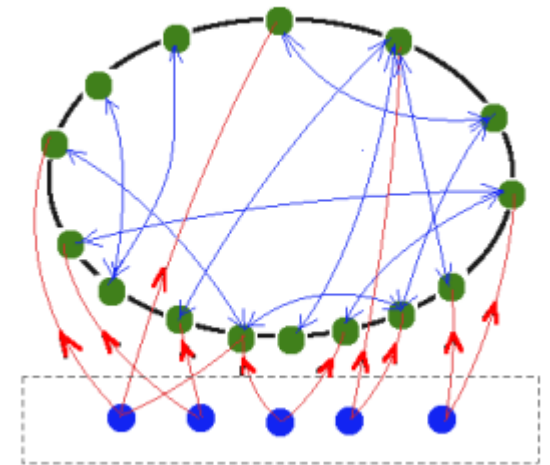
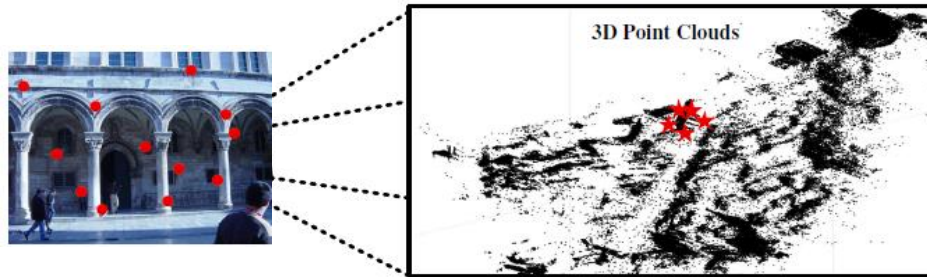
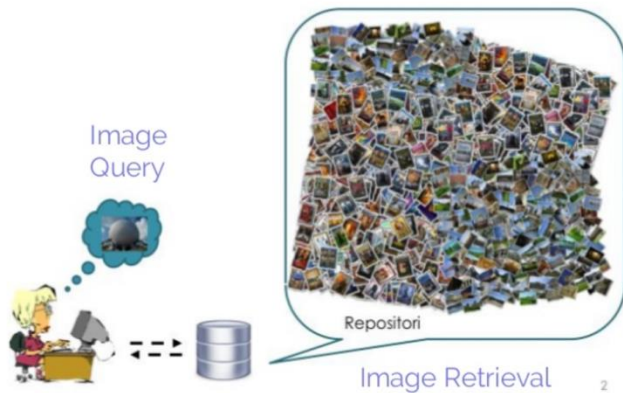
# Outline

---

- Research field
- Motivation
- Proposed scheme
- Experiment and comparison
- Conclusion

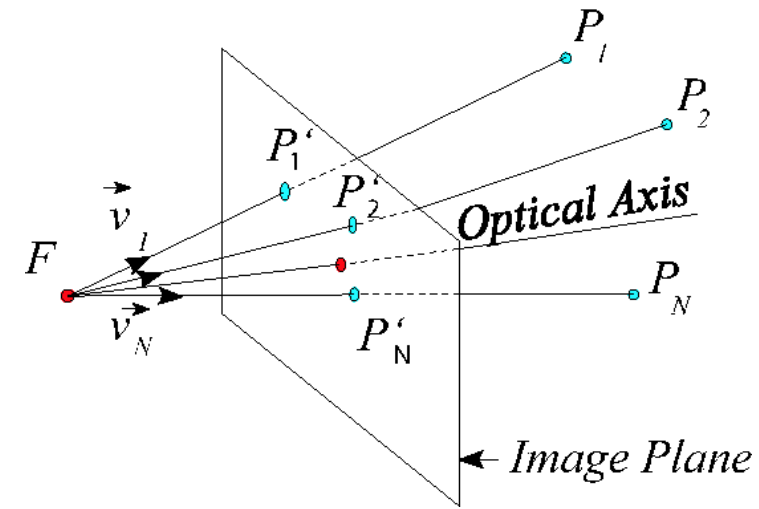
# What is IBL?

- **Image-based localization** → feature similarity
  - ▣ image retrieval: **without** building the global **map** in advance
  - direct 2D-3D matching: map is needed
  - co-visibility: consider the **relationship** among multiple global sub-models



# General solution for 2D-3D matching localization

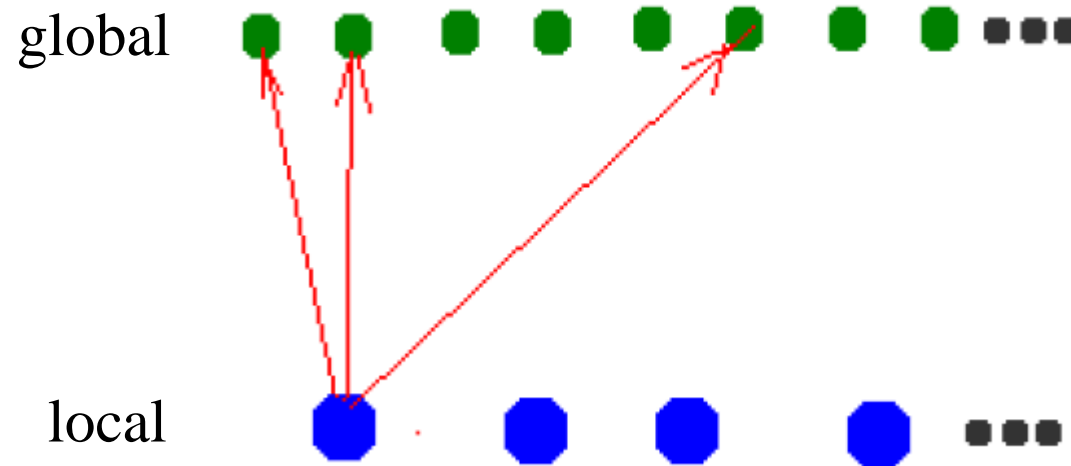
- Selection of 2D-3D matches
  - ▣ PnP + RANSAC
- Remove ambiguous matches
  - ▣ ratio test



<https://www.semanticscholar.org/paper/An-Analytical-Solution-to-the-Perspective-n-Point-Fabrizio-Devars/95cacfebed0cde6b111559342e9b29fff69c06e6/figure/0>

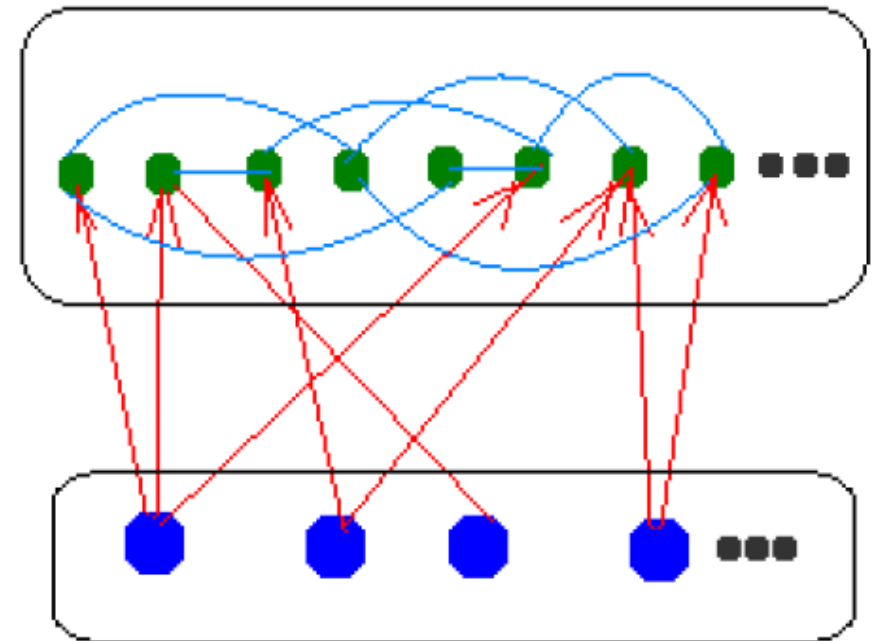
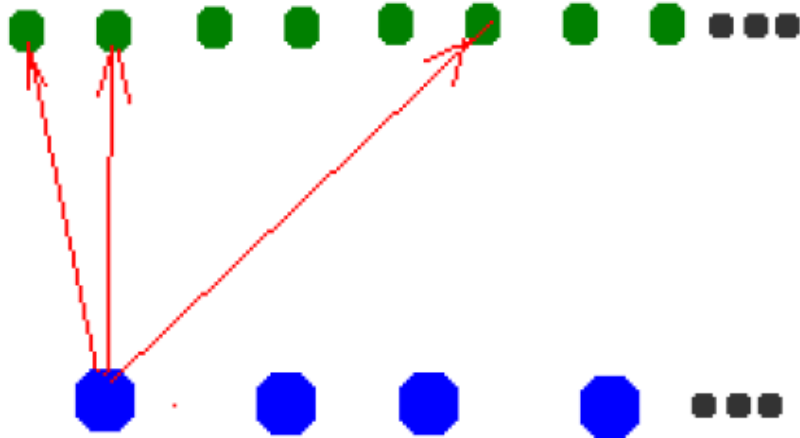
# Why?

- **Large-scale problem → ambiguity**
  - 3D points can be **visually similar** or even identical (repeated structure)
  - **ambiguous matches** are almost inevitable



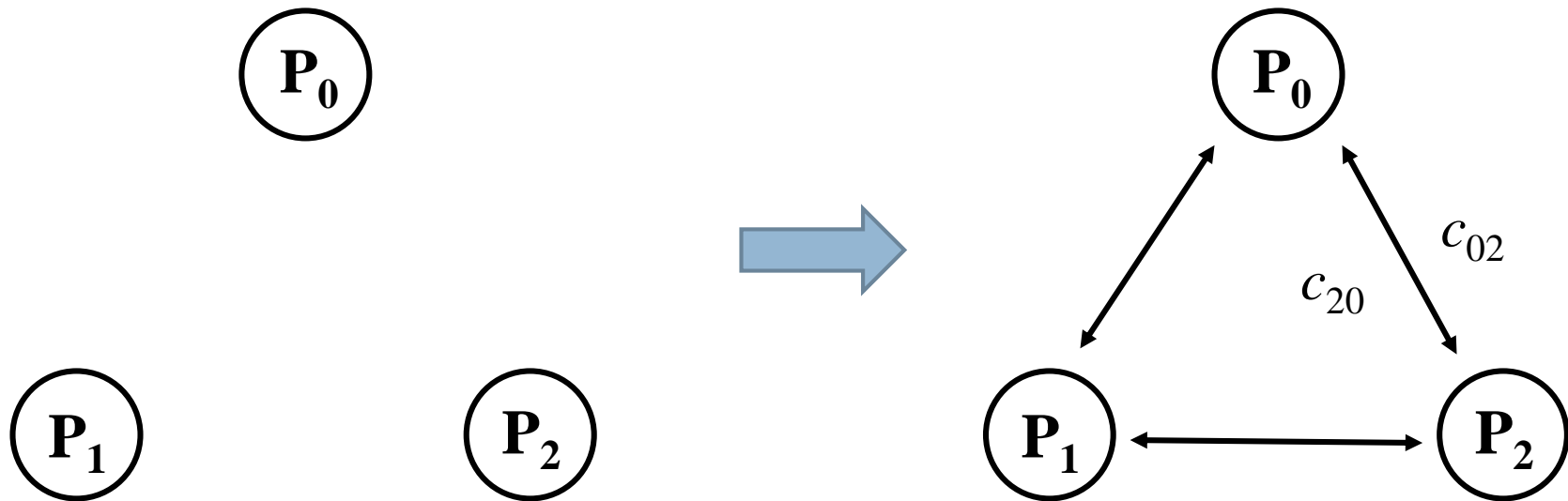
# Why?

- **Local search → sub-optimal solution**
  - ▣ take account of similarities between 2D-3D matches
  - ▣ involves in **global compatibility** among all matching pairs? effective?



# How? – Step 1 (Build a map graph)

- Traditionally, 3D map is in the form of **unordered point clouds**
- Transform into **weighted** and **bi-directed** map graph
  - ▣ **covisibility**  $c_{ij}$  – if  $P_j$  is seen by an image set, how likely  $P_i$  can also be seen from the same image set



$$c_{ij} = \frac{|A_i \cap A_j|}{|A_j|}$$

# How? – Step 1 (Build a map graph)

- Collect all  $c_{ij}$  into a square matrix  $C = [c_{ij}]$  of size  $N \times N$
- **Normalize each column** unit norm to form a left stochastic matrix
  - ▣ with each column summing to 1
- Also call  $C$  as **state transition matrix**

$$c_{ij} = \frac{|A_i \cap A_j|}{|A_j|}$$

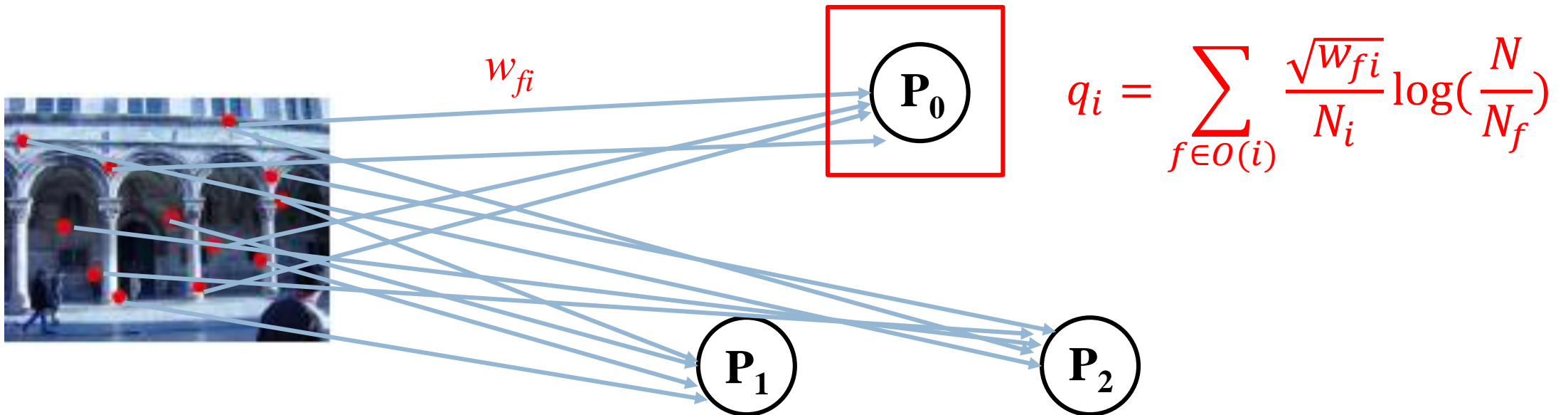


$$\begin{bmatrix} c_{00} & \cdots & c_{0N} \\ \vdots & \ddots & \vdots \\ c_{N0} & \cdots & c_{NN} \end{bmatrix}$$



# How? – Step 2 (query vector)

- query image → a set of 2D feature points
- for every 2D feature points, find a set of tentative matches from 3D graph nodes, by comparing their descriptor similarity via **Bag-of-words vocabulary tree search**



# How? – Step 2 (query vector)

- Collect all  $q_i$  into a vector  $q$
- **Normalize  $q$  to have unit norm**  $q_i \leftarrow \frac{q_i}{\sum_{i=1}^N q_i}$
- $q$  can also be interpreted as a **probability**
  - ▣ measures the probability of point  $i$  belongs to the optimal sub-set of 3D points that can be matched to the set of 2D query features

# How? – Step 3 (random walk)

- **Map graph → Markov network (aka. Markov Random Field)**
  - ▣ for **global match** between 2D query image and 3D map
  - ▣  $p_v(t)$  is the probability of finding random walker **at node  $v$  at time  $t$**
  - ▣  $p_v(\infty)$  gives the probability that the random walker **eventually ends at node  $v$**
  - ▣ when random walkers **converge**, they reach **steady state**

# How? – Step 3 (random walk)

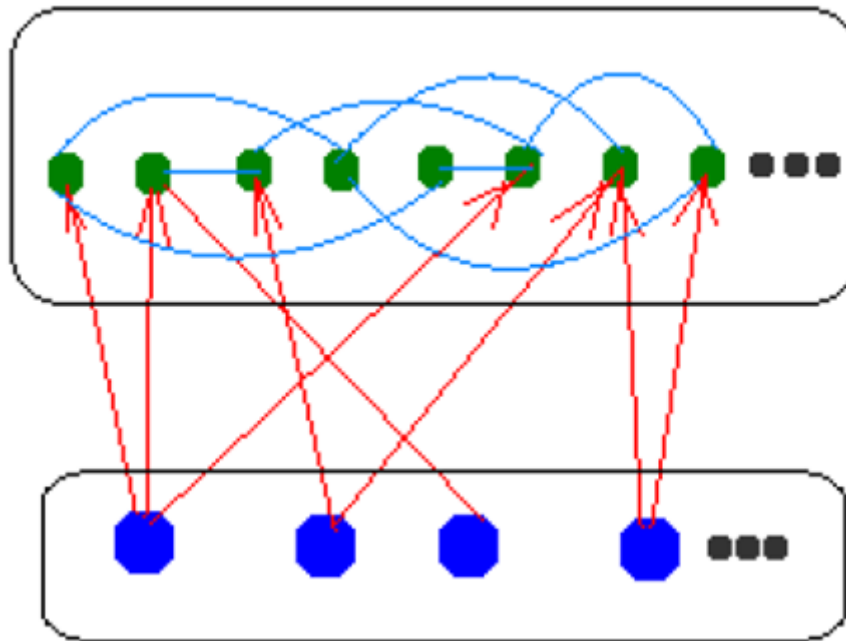
## □ Random Walk with Restart (RWR)

- $p(t+1) = \alpha C p(t) + (1 - \alpha) q$
- $\alpha$  is chosen empirically between 0.8–0.9
- $P(0)$  is initialized by  $1/N$
- $P(\infty)$  stands for the final **matchability** of every 3D point to the set of 2D query features
- Sort the elements in  $p(\infty)$  in descending order

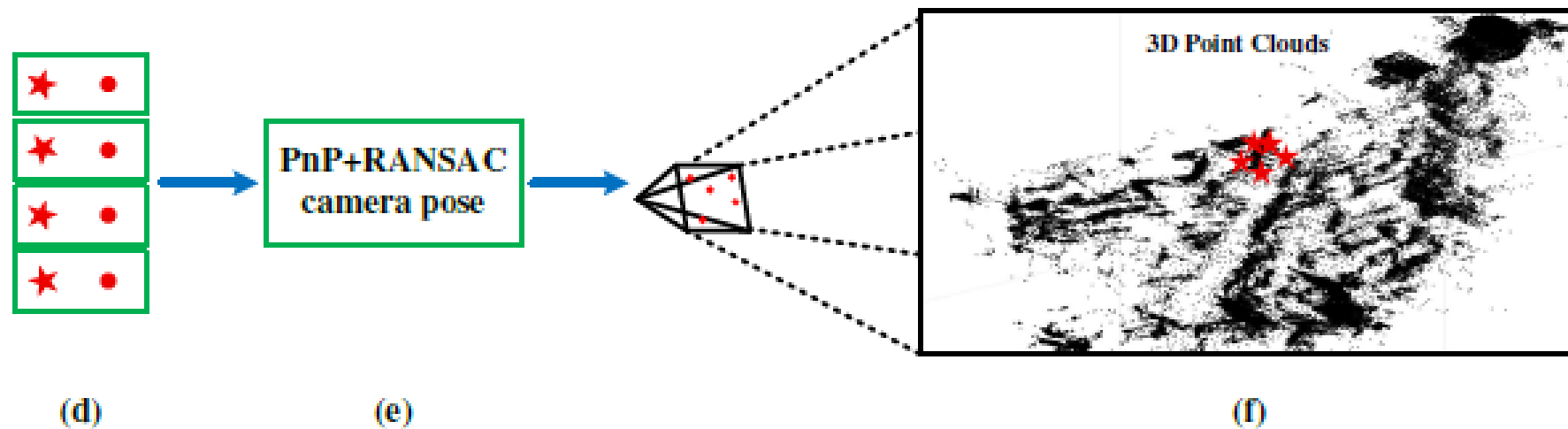
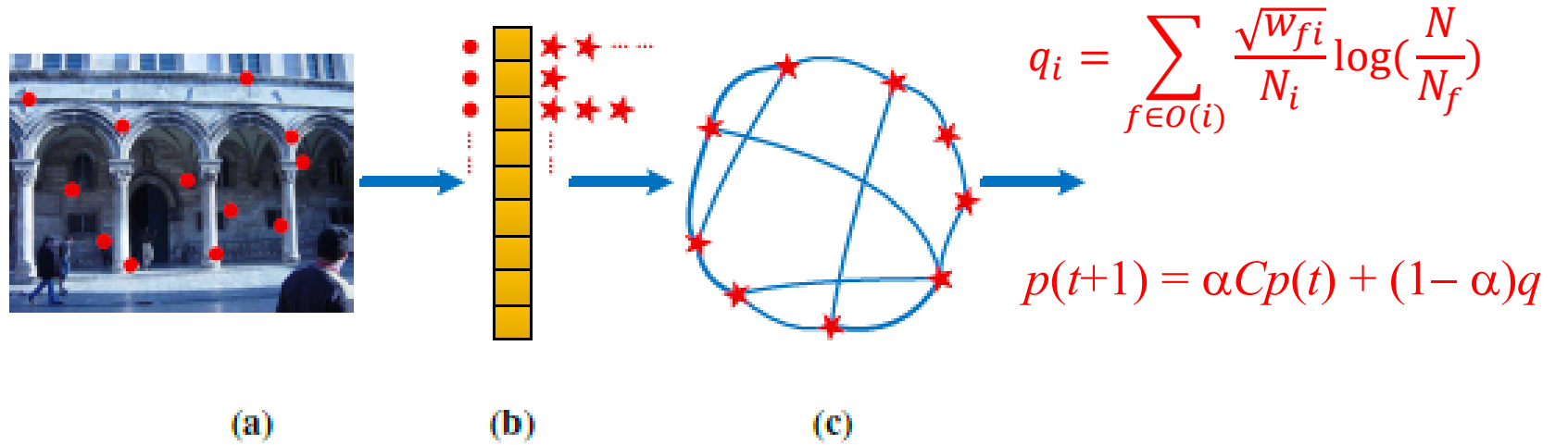
$$C = \begin{bmatrix} c_{00} & \cdots & c_{0N} \\ \vdots & C_{ij} & \vdots \\ c_{N0} & \cdots & c_{NN} \end{bmatrix}$$

# How? – Step 4 (camera pose computation)

- **Recover one-to-one correspondences**
  - ▣ do **ratio test** to retrieve one-to-one matches
  - ▣ fed into **PnP-RANSAC** to find camera position and orientation



# How? – Summary



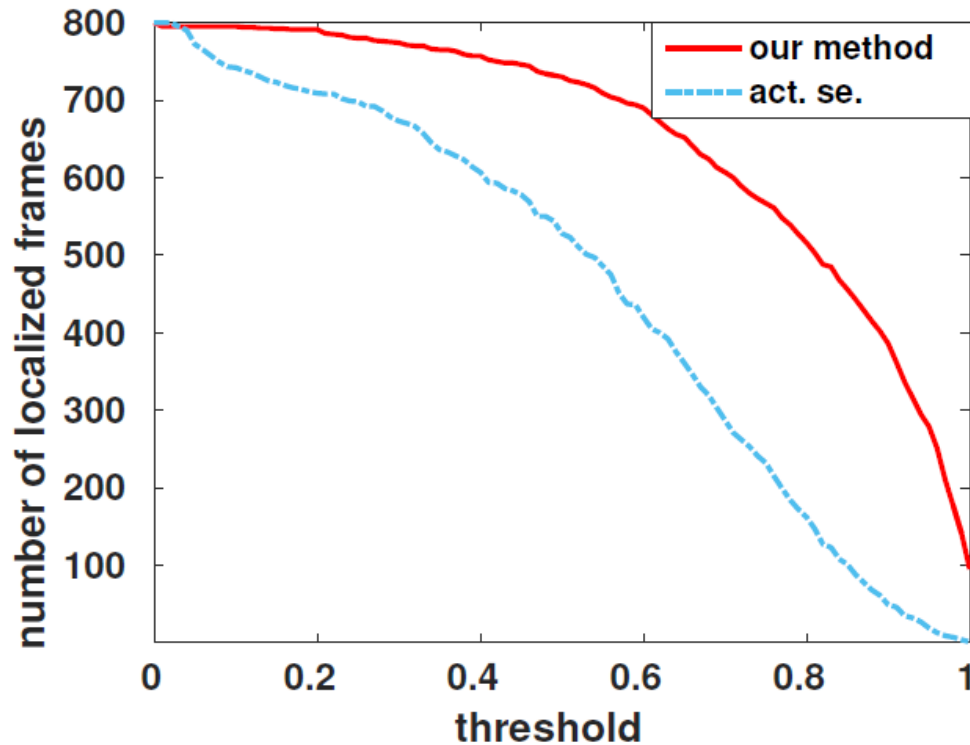
# Experiment – dataset selection

- Four standard publicly available benchmark datasets for city-scale localization  
(1) Dubrovnik, (2) Rome, (3) Vienna, (4) San Francisco (SF-0)

Dataset	#(images)	#(points)	#(query images)
Dubrovnik [32]	6,044	1,975,263	800
Rome [32]	15,179	4,067,119	1,000
Vienna [23]	1,324	1,123,028	266
SF-0 [12]	610,773	30,342,328	803

# Experiment

- In term of **recall-rate** (# images have been successfully localized)



Method	Inlier thresholds						
	0.2	0.3	0.4	0.5	0.6	0.7	0.8
Active Search [43, 44]	709	673	607	528	420	287	162
<b>Our method</b>	<b>791</b>	<b>774</b>	<b>757</b>	<b>730</b>	<b>690</b>	<b>607</b>	<b>516</b>

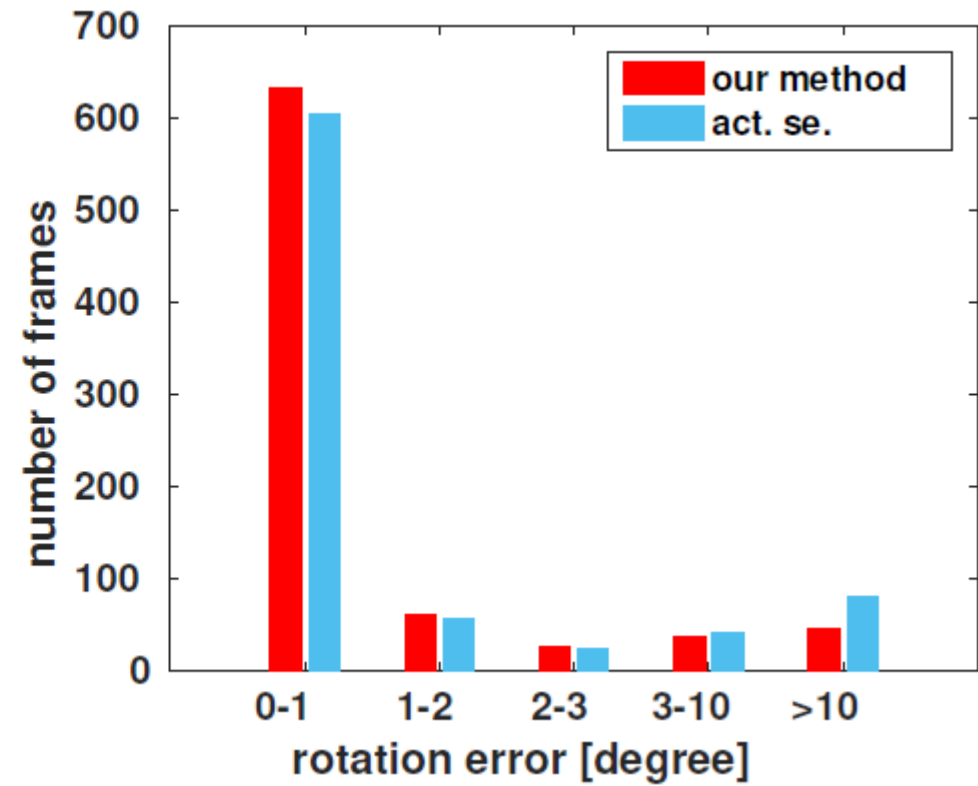
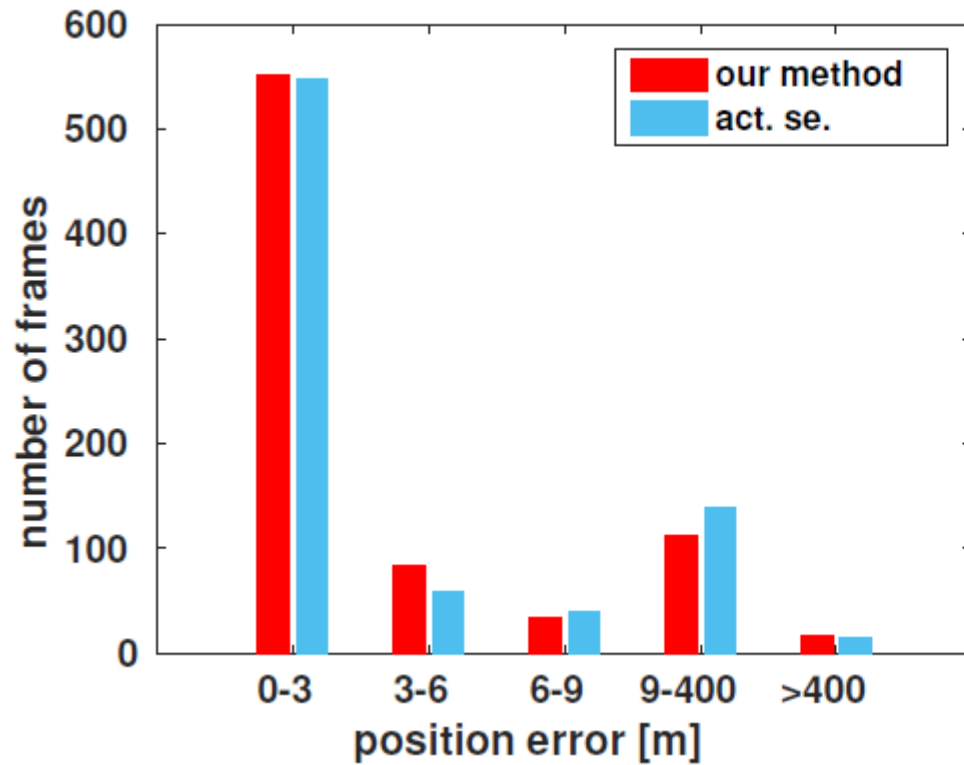
[43] T. Sattler, B. Leibe, and L. Kobbelt, “Improving image-based localization by active correspondence search,” ECCV’12

[44] T. Sattler, B. Leibe, and L. Kobbelt, “Efficient effective prioritized matching for large-scale image-based localization,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016.



# Experiment

- In term of **precision** (camera localization error)



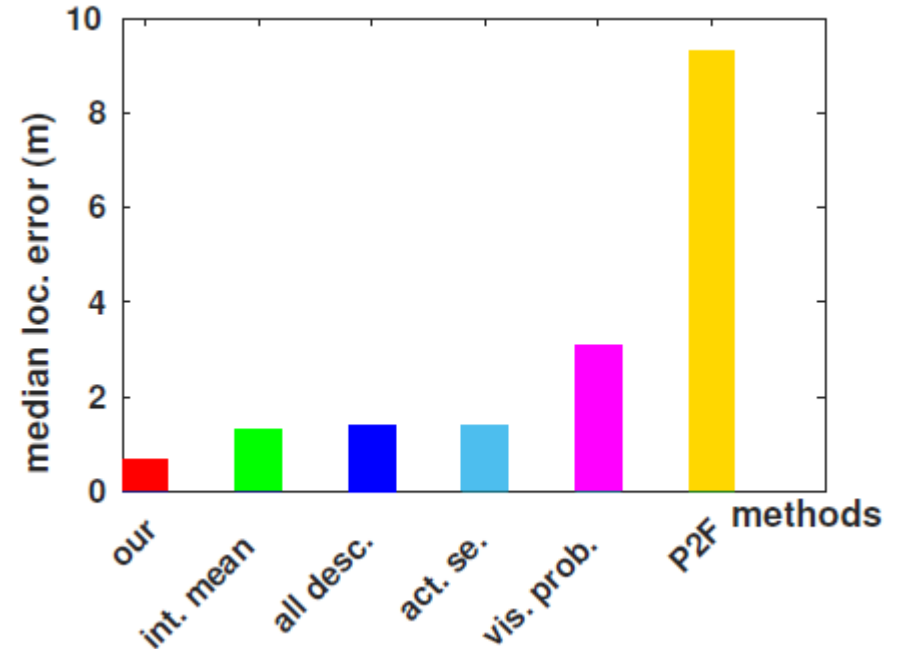
[43] T. Sattler, B. Leibe, and L. Kobbelt, "Improving image-based localization by active correspondence search," ECCV'12.

[44] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient effective prioritized matching for large-scale image-based localization," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016.

# Experiment

- In term of **precision** (camera localization error)

Method	quartile errors (m)			num. of images		
	1st	median	3rd	<18.3m	>400m	#(reg.)
<b>our method</b>	<b>0.24</b>	<b>0.70</b>	<b>2.67</b>	<b>743</b>	<b>7</b>	794
act. se. [43,44]	0.40	1.40	5.30	704	9	<b>795</b>
all desc. [42]	0.40	1.40	5.90	685	16	783
int. mean [42]	0.50	1.30	5.10	675	13	782
P2F [32]	7.50	9.30	13.40	655	-	753
vis. prob. [13]	0.88	3.10	11.83	-	-	788



[42] T. Sattler, B. Leibe, and L. Kobbelt, “Fast image-based localization using direct 2d-to-3d matching,” ICCV’11.

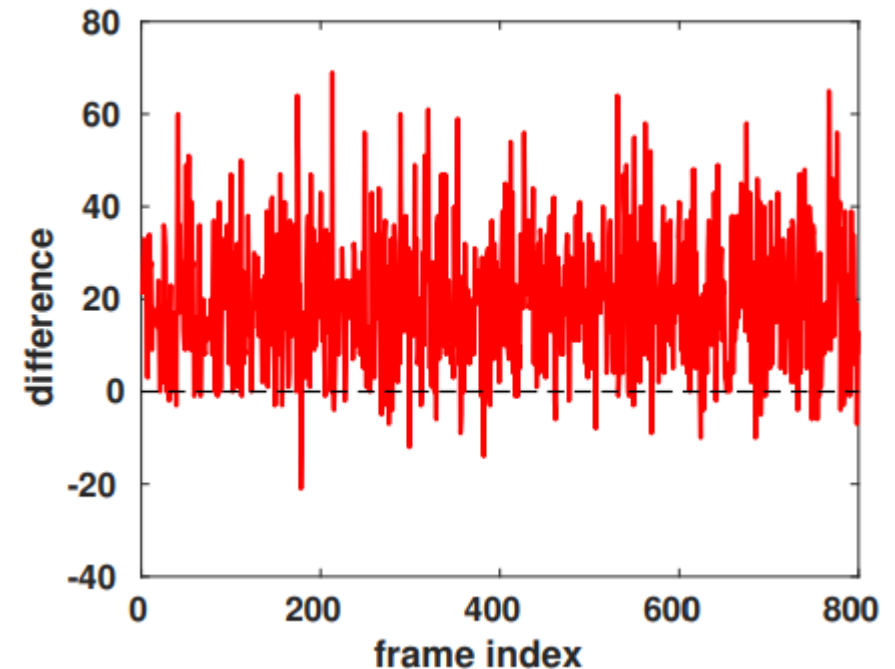
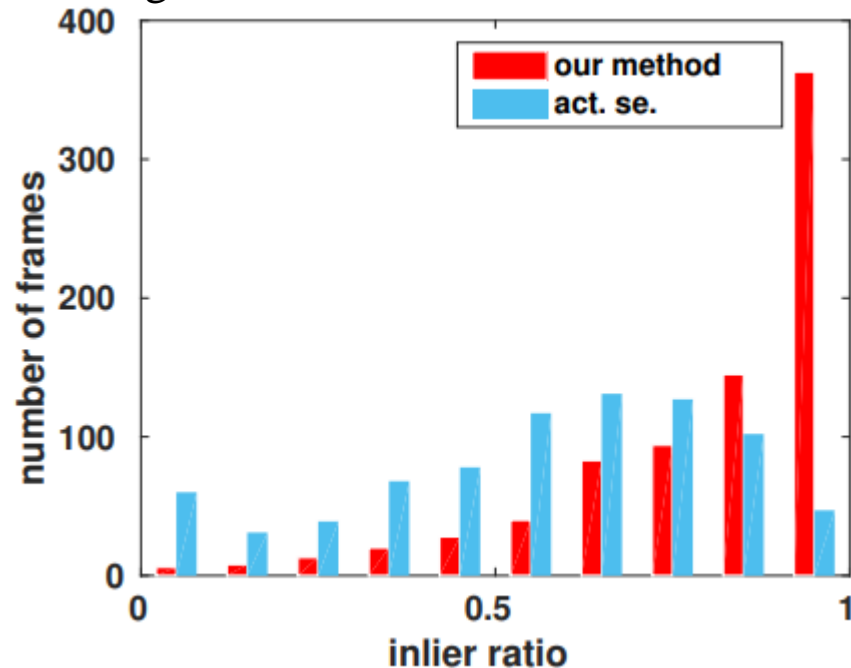
[13] S. Choudhary and P. Narayanan, “Visibility probability structure from sfm datasets and applications,” ECCV’12.

[32] Y. Li, N. Snavely, and D. P. Huttenlocher, “Location recognition using prioritized feature matching,” ECCV’10.

# Experiment

- In terms of **accuracy** (the inlier ratio in the final matched 2D-3D feature pairs after applying RANSAC)

average inlier ratio of our method: 81%  
average inlier ratio of active search: 57%



[43] T. Sattler, B. Leibe, and L. Kobbelt, "Improving image-based localization by active correspondence search," ECCV'12.

[44] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient effective prioritized matching for large-scale image-based localization," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016.

# Experiment

- In terms of **scalability** [use SF-0 dataset]

Dataset	#(images)	#(points)	#(query images)
SF-0 [12]	610,773	30,342,328	803

- our method localizes 652 images (out of totally 803 query images), and the average time spent per image is 0.30s
  - active search [43,44] was only able to localize 31 images
- 
- For Rome and Vienna datasets
    - has about 4-millions and 1-millions 3D map points
    - localized 990 (out of 1000) and 213 (out of 266) query images
    - the average query time by our unoptimized code was 2.35s and 1.67s

[43] T. Sattler, B. Leibe, and L. Kobbelt, “Improving image-based localization by active correspondence search,” ECCV’12.

[44] T. Sattler, B. Leibe, and L. Kobbelt, “Efficient effective prioritized matching for large-scale image-based localization,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016.

# Conclusion

- A global method, taking account of not only **individual feature match's visual similarity** but also the **global compatibilities** as measured by the pair-wise covisibility, to deal with scalability and ambiguity for localization.