Vehicle localization without SLAM: Learning to find your camera's pose in an aerial image

Julian Kooij
Intelligent Vehicles group, TU Delft

ACM Multimedia Workshop UAVs in Multimedia
November 1, 2024

TU Delft

Julian Kooij

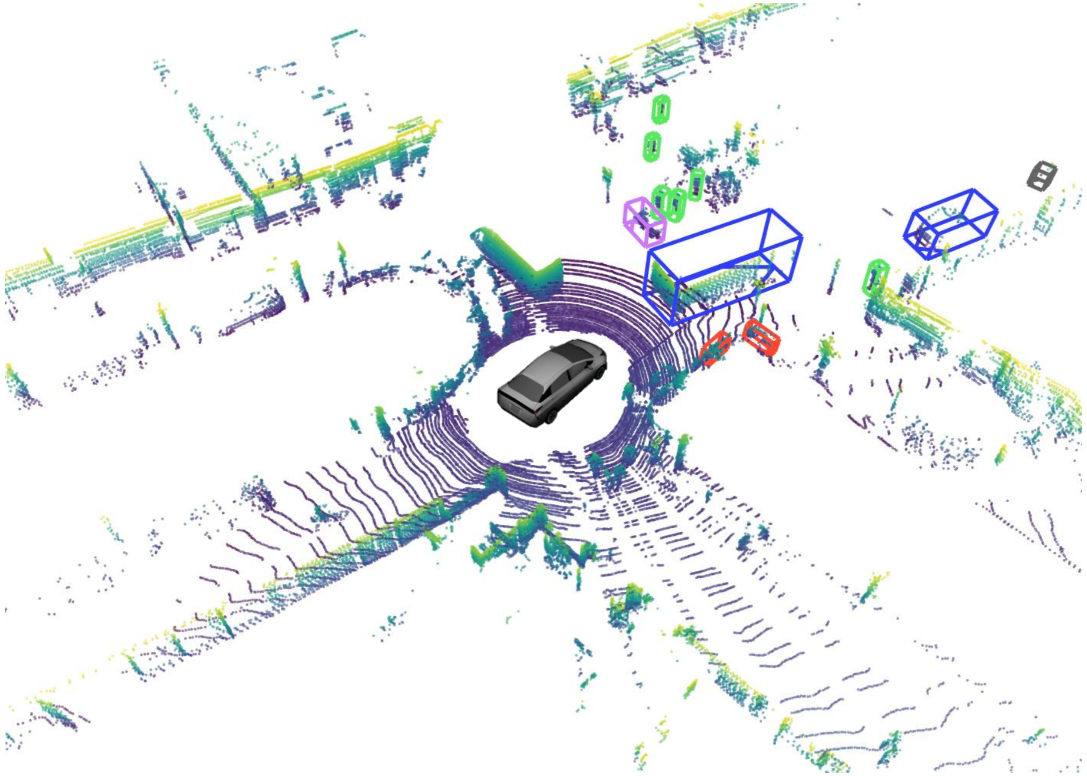
- **Associate Professor**
  Intelligent Vehicles group
  Cognitive Robotics department
  Mechanical Engineering Faculty, TU Delft

- **Director of 3DUU Delft AI lab**
  TUD AI Initiative Labs & Talent program
  Collaboration between ME and ABE faculties

- **PhD**
  2010-2014
  University of Amsterdam
  Daimler R&D (Mercedes), Ulm Germany

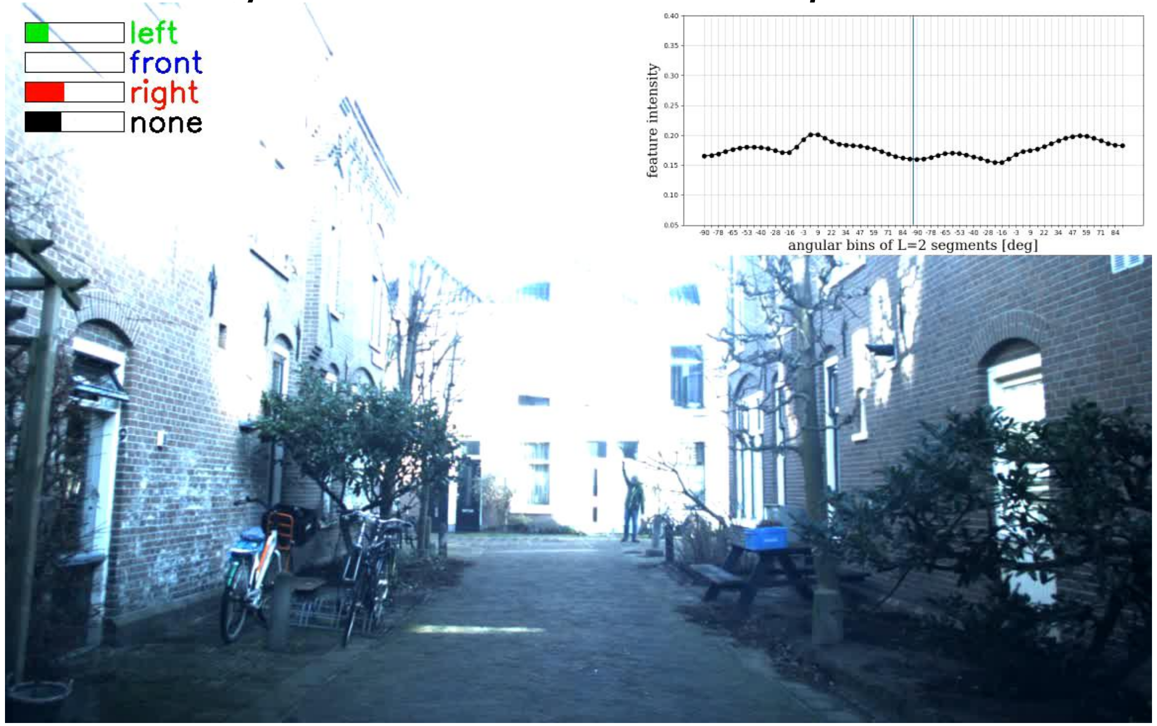
Research at IV group

Driving around Vulnerable Road Users,
use of 3+1D radar



Multi-class Road User Detection with 3+1D Radar in the View-of-Delft Dataset, A. Palffy, E.A.I. Pool, S. Baratam, J.F.P. Kooij, D.M. Gavrila, IEEE Robotics and Automation Letters, 2022, vol. 7(2), 4961-4968

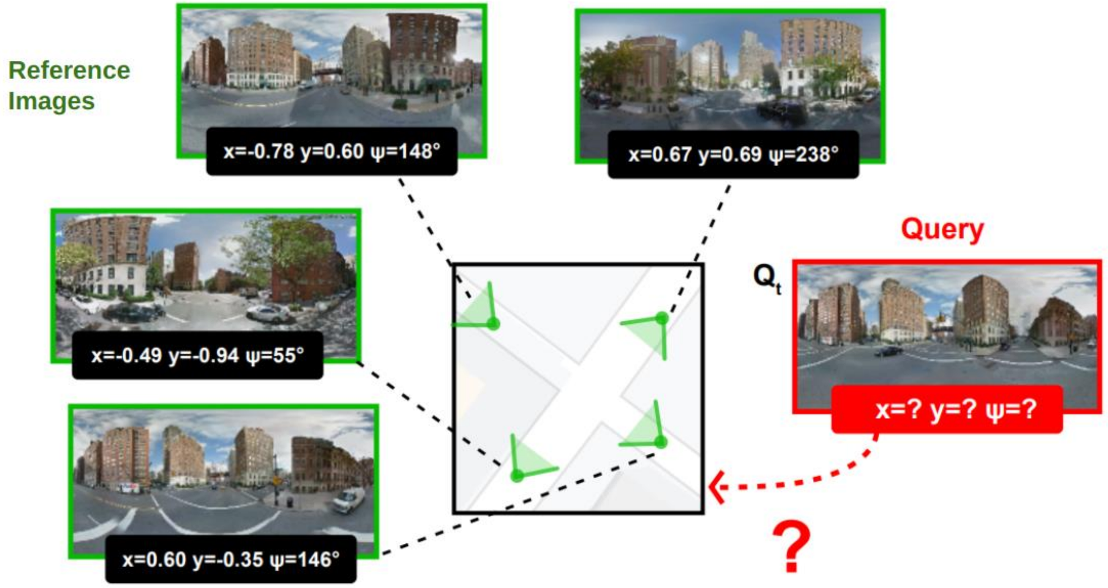
Predict approaching traffic
using vehicle-mounted microphones



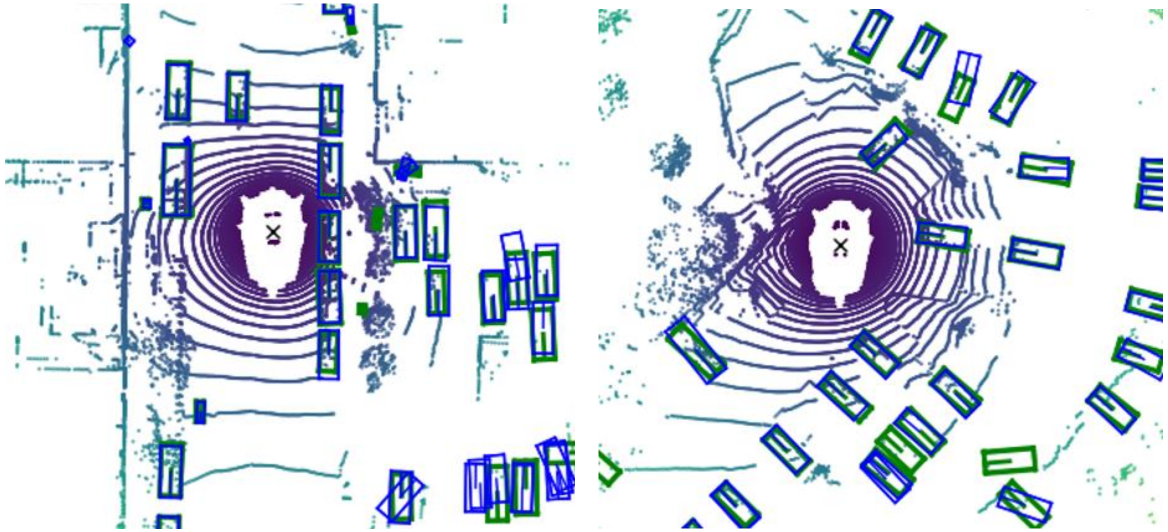
Hearing What You Cannot See: Acoustic Vehicle Detection Around Corners, Y. Schulz, A.K. Mattar, T.M. Hehn, J.F.P. Kooij, IEEE Robotics and Automation Letters (RA-L), 2021, vol. 6(2), 2587-2594

3D Urban Understanding (3DUU) AI lab

Localization: Where was this 2D image taken in 3D world?



Fusing 2D images and 3D point clouds for detection



Interpreting and segmenting pointclouds

Reconstructing & imagining

TU Delft Intelligent Vehicles group

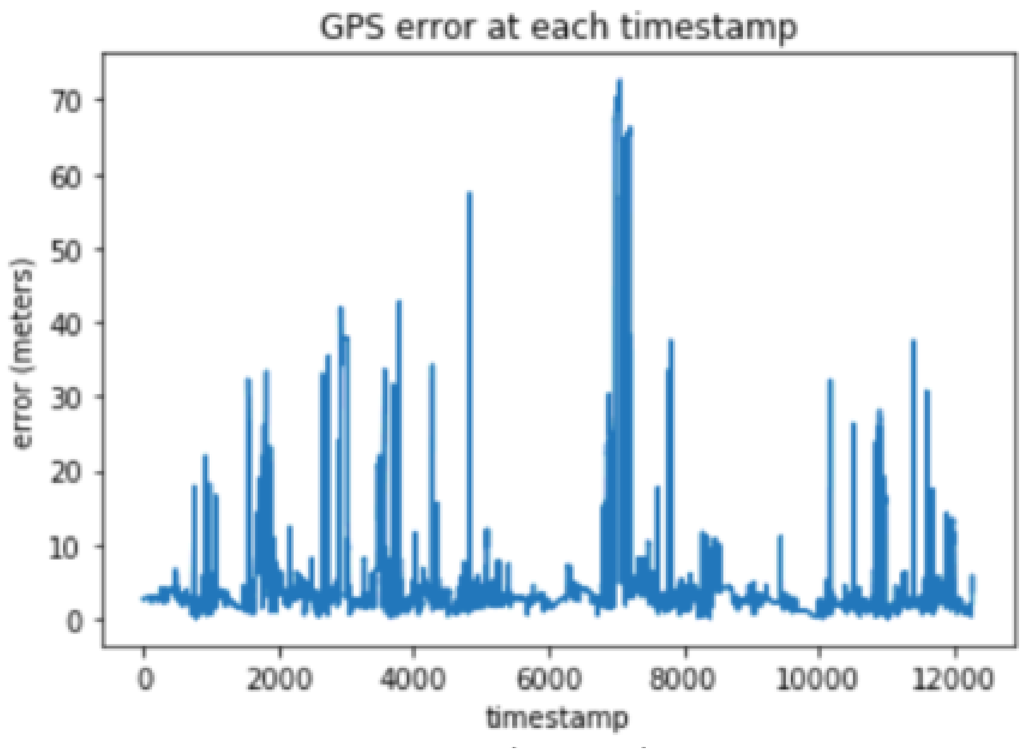
https://intelligent-vehicles.org/

TU Delft

The problem of GPS

- GNSS/GPS → can reach tens of meters[1] in urban areas

1. Ben-Moshe, Boaz, et al. "Improving Accuracy of GNSS Devices in Urban Canyons." CCCG. 2011

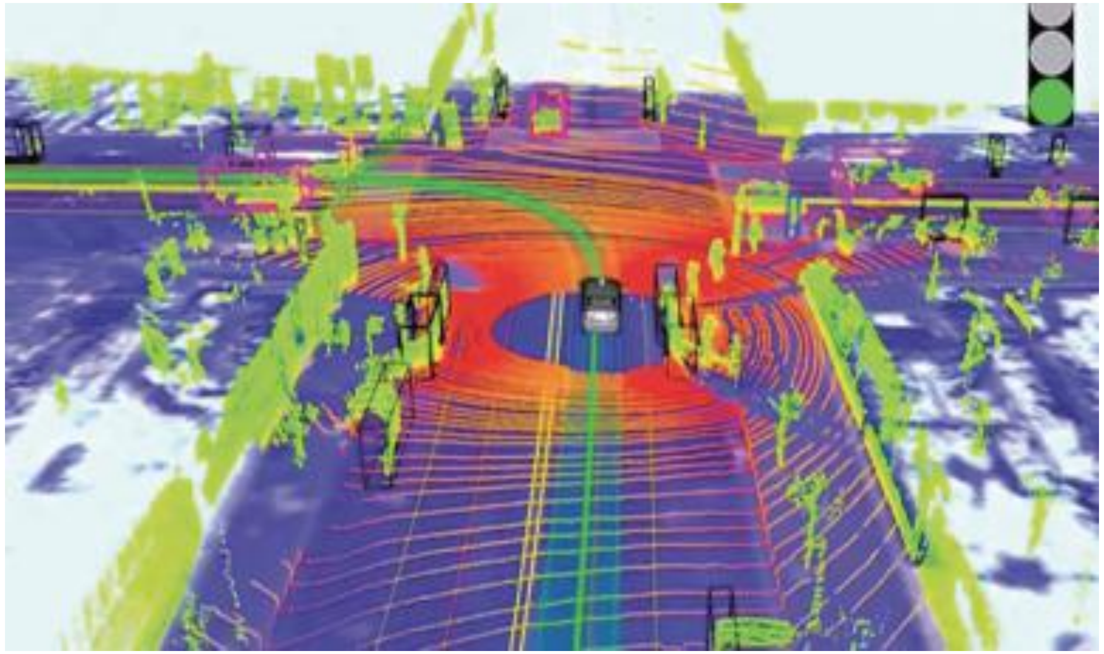


One traversal from Oxford RobotCar

- Red: Raw GPS
- Green: Ground truth RTK (high precision GPS, post-processing)
- Here, worst-case ~70 meters

Model-driven visual localization

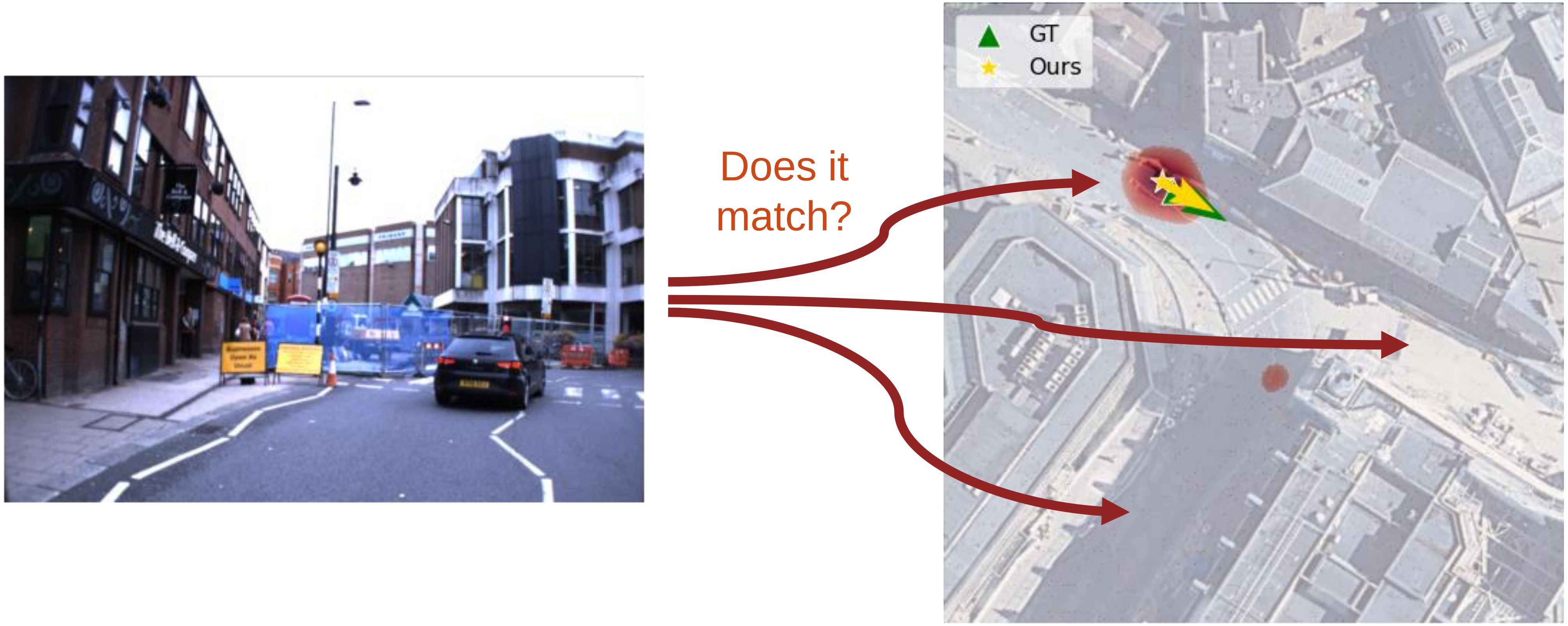
- Alternative: match sensor data to detailed 3D models,
  but implies huge data collection, processing, updating effort … hard to scale



Credit right image: "Autonomous driving in the iCity—HD maps as a key challenge of the automotive industry.“, Seif, Heiko G., and Xiaolong Hu, *Engineering* 2.2 (2016): 159-162.

Data-driven visual localization

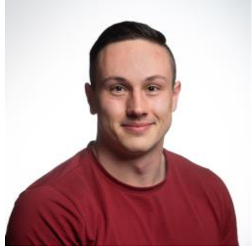
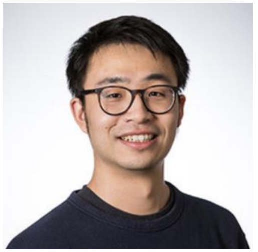
- Can we not use aerial images ("Google Earth") as easily obtained map?
- Match learned representations of vehicle's camera image to map's aerial image



JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation

CVPR 2023

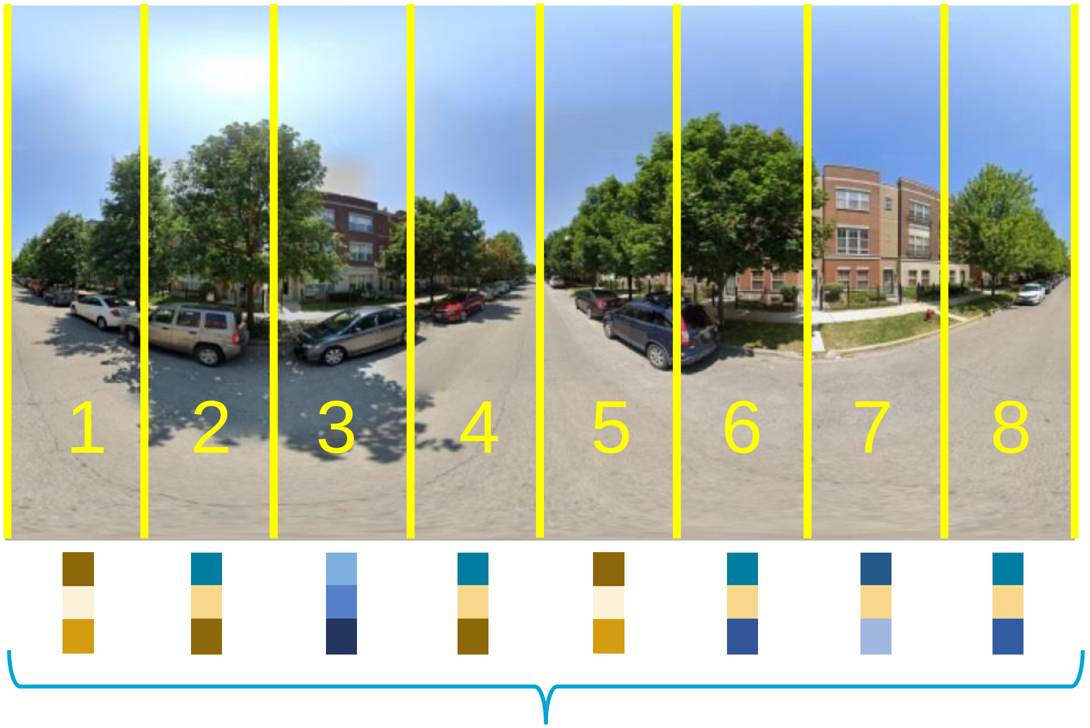



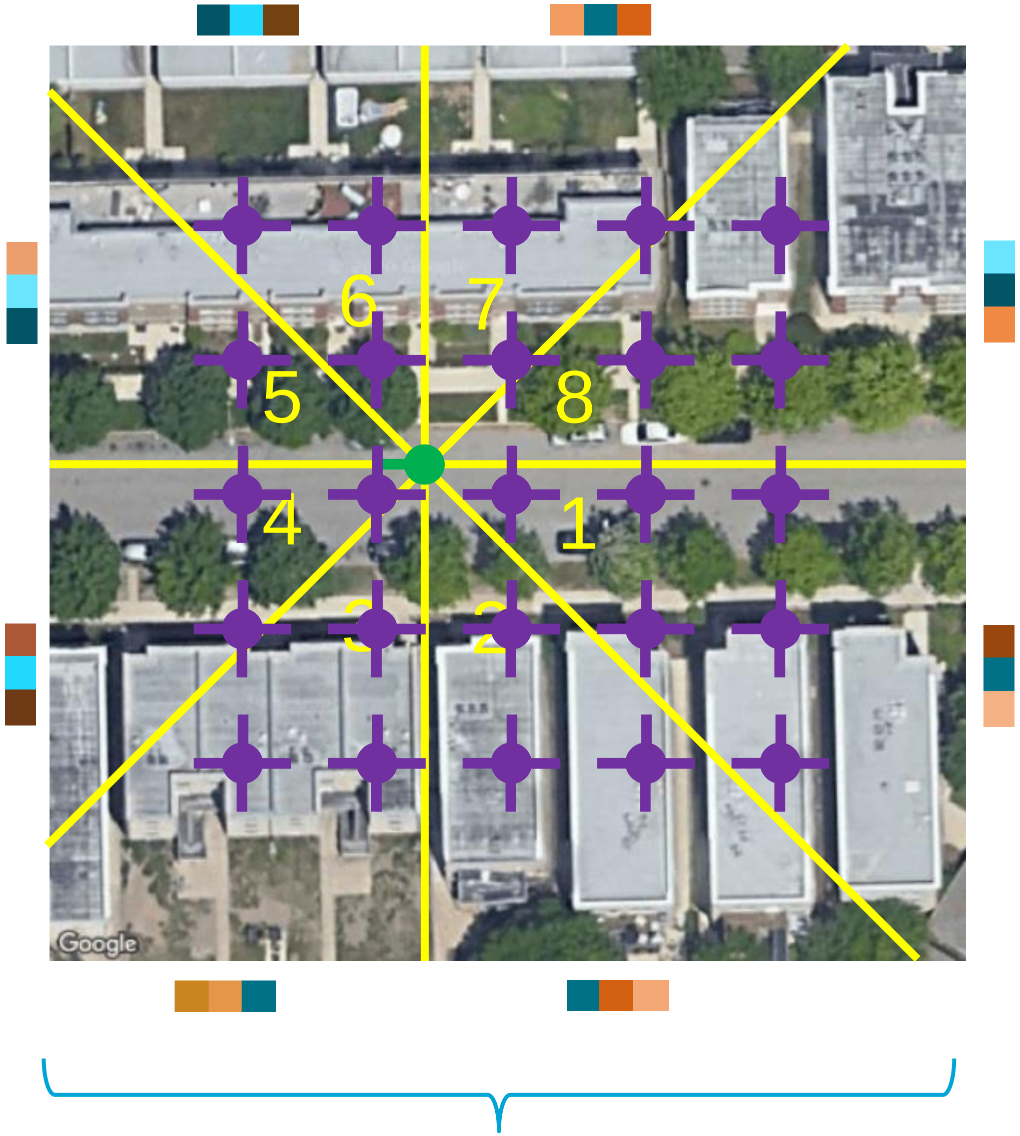
Ted Lentsch[*] Zimin Xia[*] Holger Caesar Julian F. P. Kooij

Intelligent Vehicles Group, Delft University of Technology, The Netherlands
[*]Equal contribution

TU Delft

What is a slice?

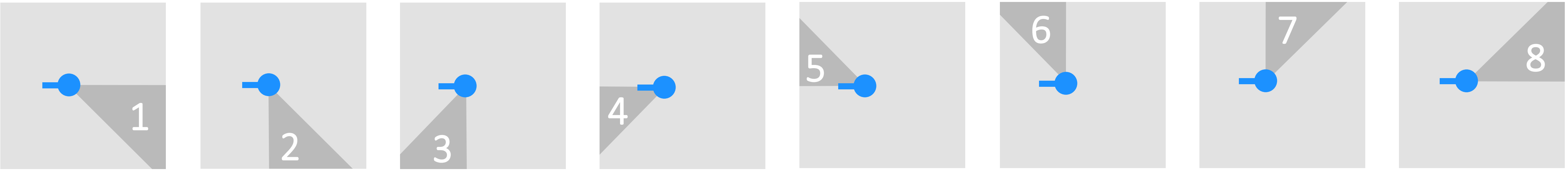




Cosine Similarity

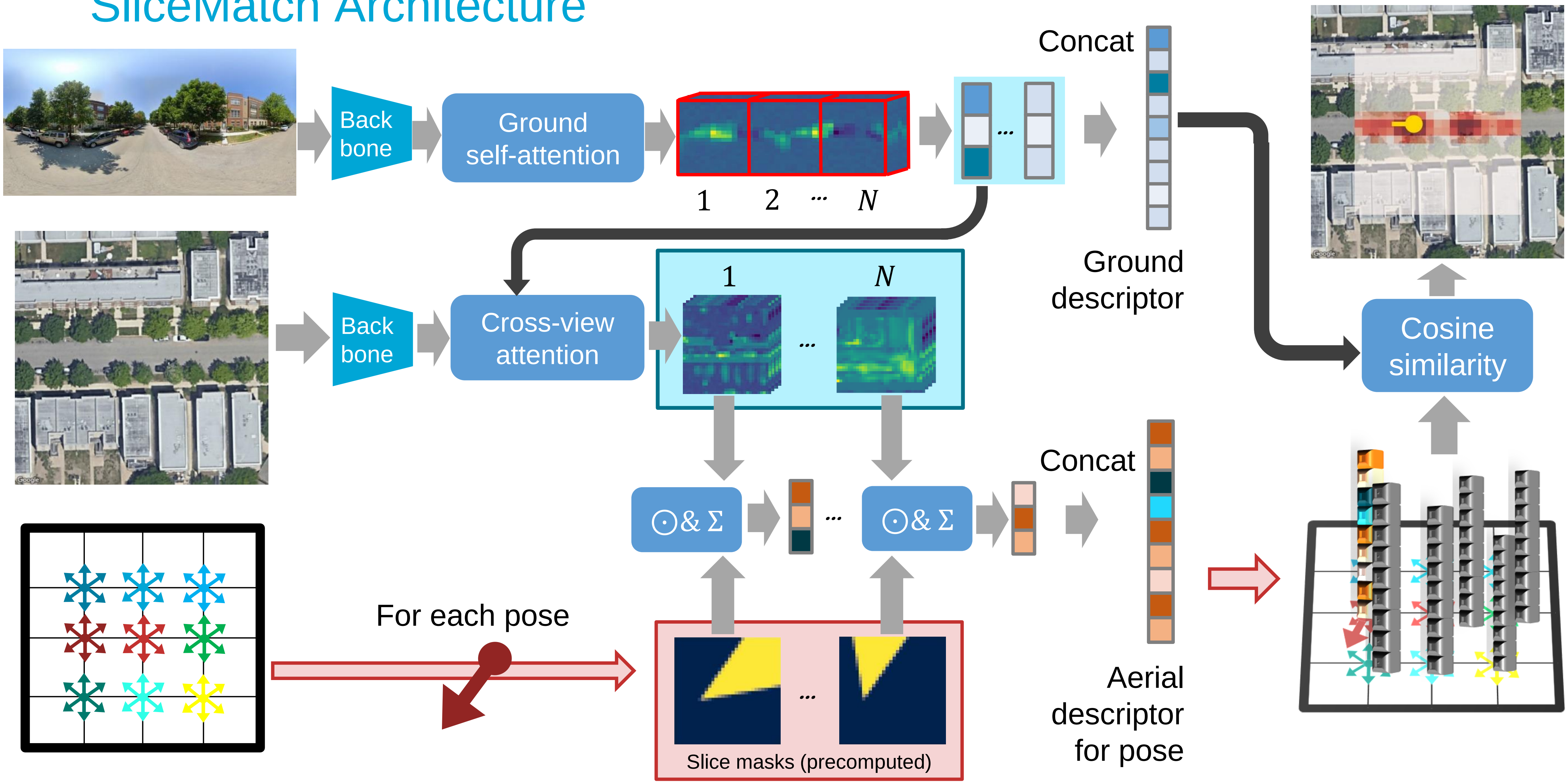
Geometry-guided feature aggregation

- **Goal**: Achieve accuracy by testing many pose hypotheses
- **Idea**: Minimize computations per pose hypothesis → only do feature aggregation per pose
- **“Slice masks”:** Per pose, pre-compute the aerial region for each slice



- Use contrastive learning to obtain pos + orientation discriminative features

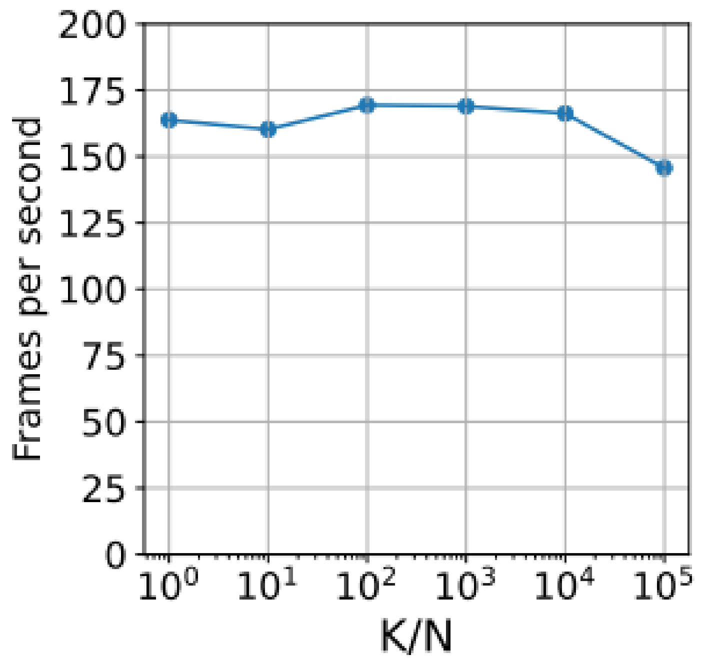
SliceMatch Architecture



Efficient computation

- Feature extraction is done only once
- Per pose hypotheses operations are minimized:
  - Feature aggregation module uses pre-computed slice masks
  - Apply slice masks to feature maps: just multiply and sum
  - Highly parallelizable tensor operations

- Inference speed at over 150 FPS (on V100 GPU)
- Significantly faster than previous SOTA method
- Did not observe run time increase when testing more poses



K = number of poses
N = 16 slices

Quantitative evaluation

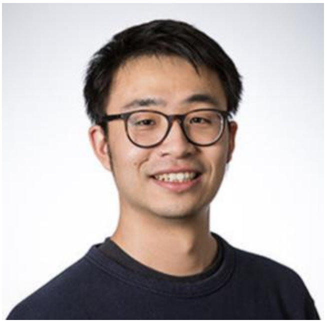
Model	Backbone	Aligned Images	Same-Area ↓ Location (m) Mean	Same-Area ↓ Location (m) Median	Same-Area ↓ Orientation (°) Mean	Same-Area ↓ Orientation (°) Median	Cross-Area ↓ Location (m) Mean	Cross-Area ↓ Location (m) Median	Cross-Area ↓ Orientation (°) Mean	Cross-Area ↓ Orientation (°) Median
CVR [55]	VGG16	✓	8.99	7.81	-	-	8.89	7.73	-	-
MCC [50]	VGG16	✓	6.94	3.64	-	-	9.05	5.14	-	-
SliceMatch (ours)	VGG16	✓	**5.18**	**2.58**	-	-	**5.53**	**2.55**	-	-
MCC [50]	VGG16	X	9.87	6.25	56.86	16.02	12.66	9.55	72.13	29.97
SliceMatch (ours)	VGG16	X	**8.41**	**5.07**	**28.43**	**5.15**	**8.48**	**5.64**	**26.20**	**5.18**
SliceMatch (ours)	ResNet50	X	**6.49**	**3.13**	**25.46**	**4.71**	**7.22**	**3.31**	**25.97**	**4.51**

CVR: Zhu, et al, CVPR 2021
MCC: Xia, et al, ECCV 2022

- SliceMatch outperforms SOTA baselines (at the time) in all use cases
- With a more advanced backbone (ResNet50 vs VGG16), accuracy further improves
- *(N.B.: we improved the GT localization annotations for the VIGOR datasets)*

Convolutional Cross-View Pose Estimation

IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)



Zimin Xia

Olaf Booij

Julian F. P. Kooij

Intelligent Vehicles Group, TU Delft, The Netherlands

TomTom, Amsterdam, The Netherlands

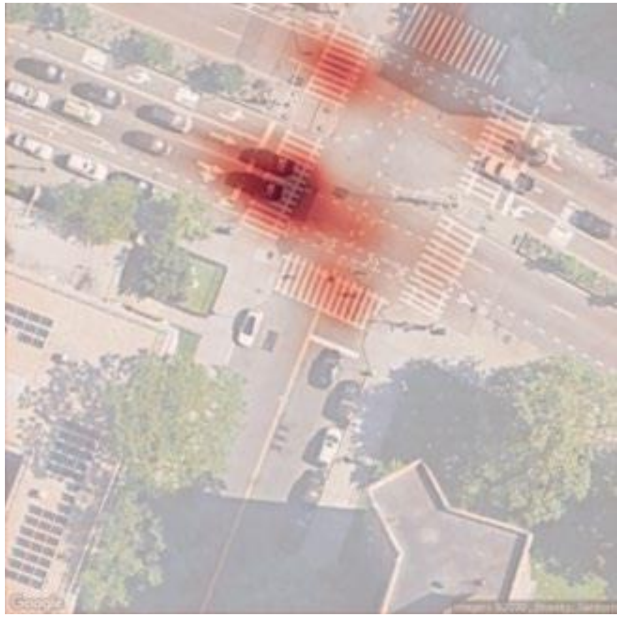
Cross-view localization

- Pinpoint the location of the ground camera on a local aerial image
- We formulate localization as dense multi-class classification

G

A

$$p(D|G,A)$$



D

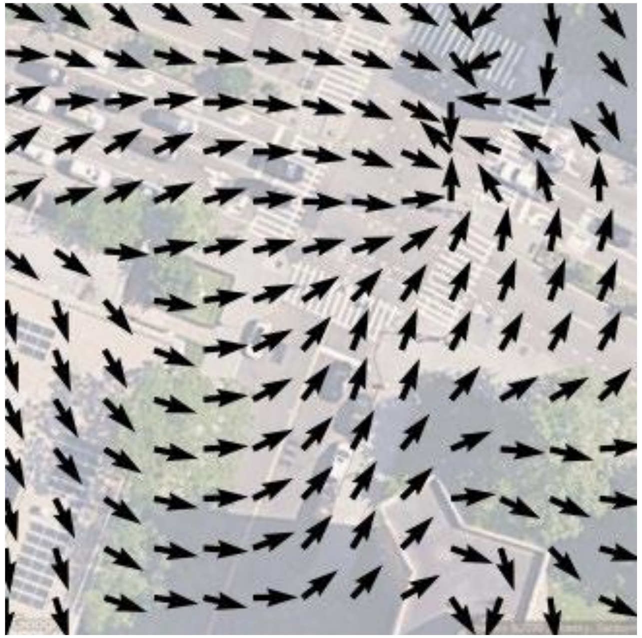
Cross-view pose estimation

- Estimate location and *orientation*
- Location and orientation are <u>related</u>
- Solution: also predict a vector field

G

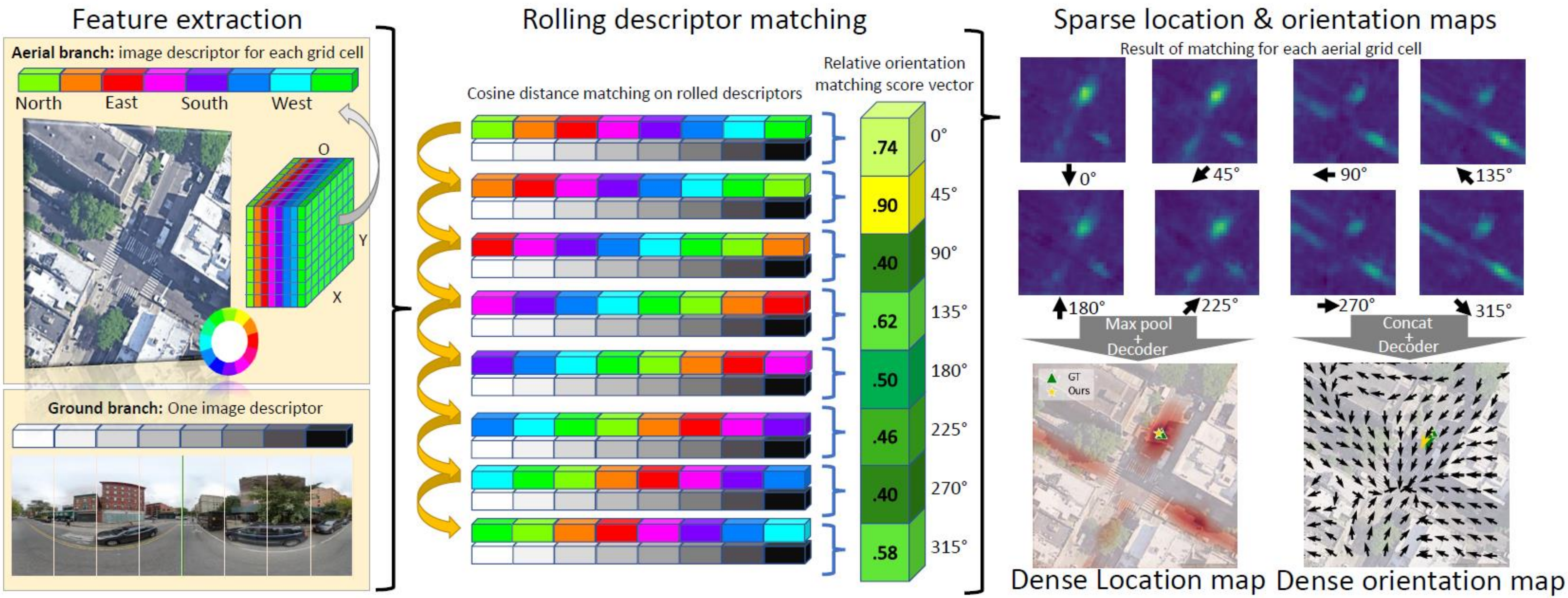
A

$$\mathrm{f}(G, A) \rightarrow O$$



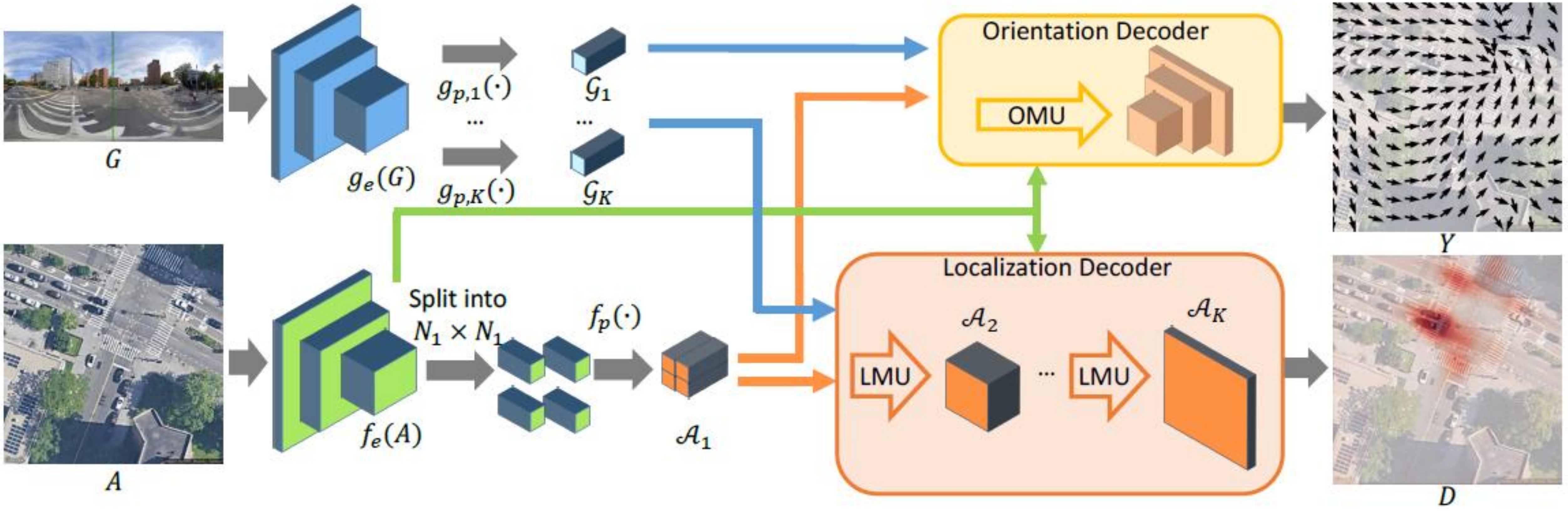
O

Architecture

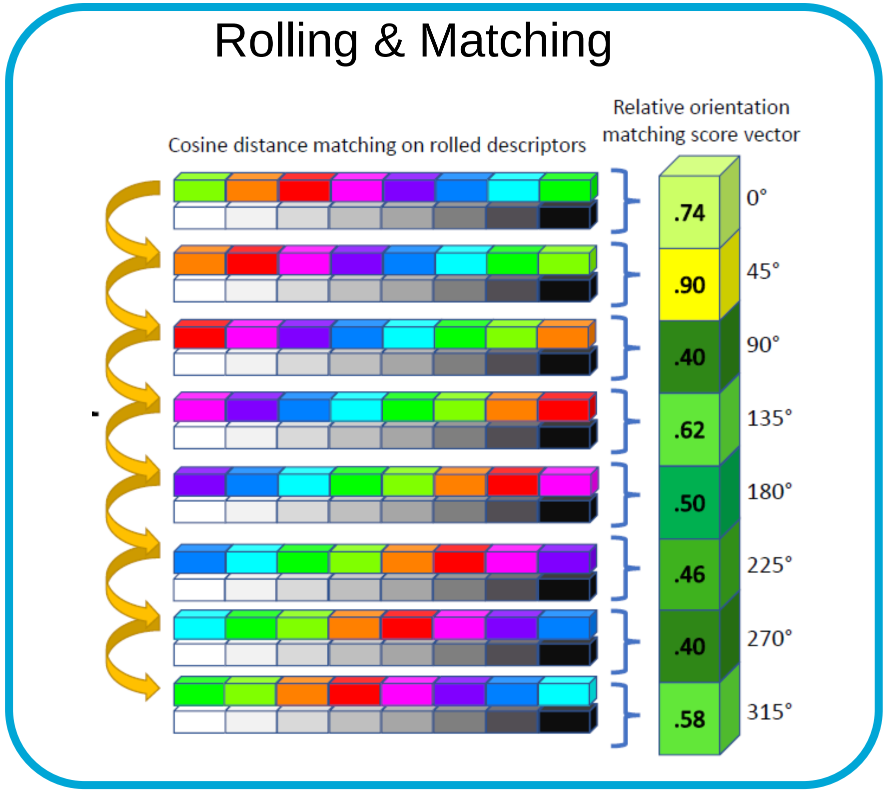


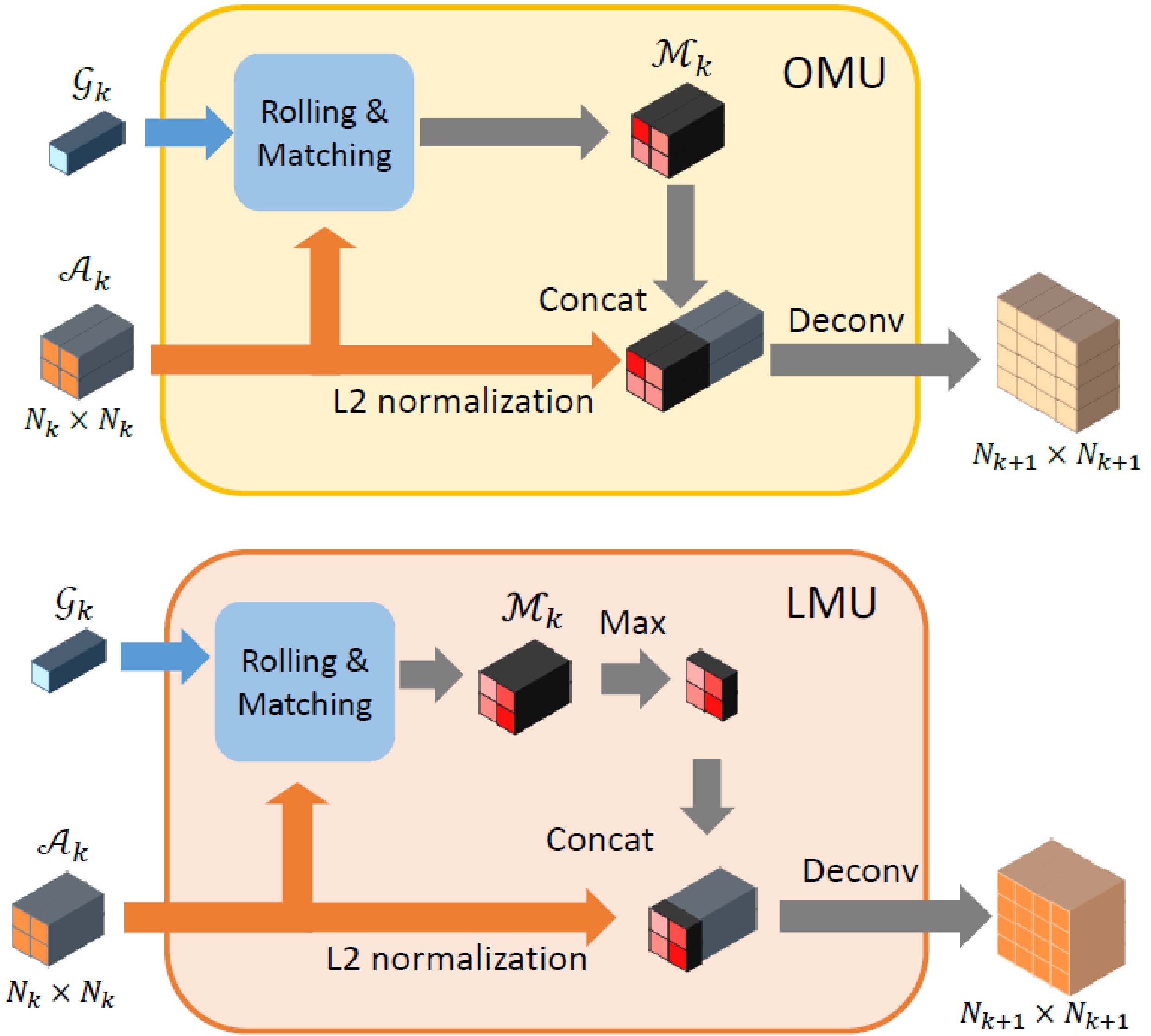
Architecture

- Siamese-like network, with two predictor heads
- Rolling & Matching happens in the Decoders



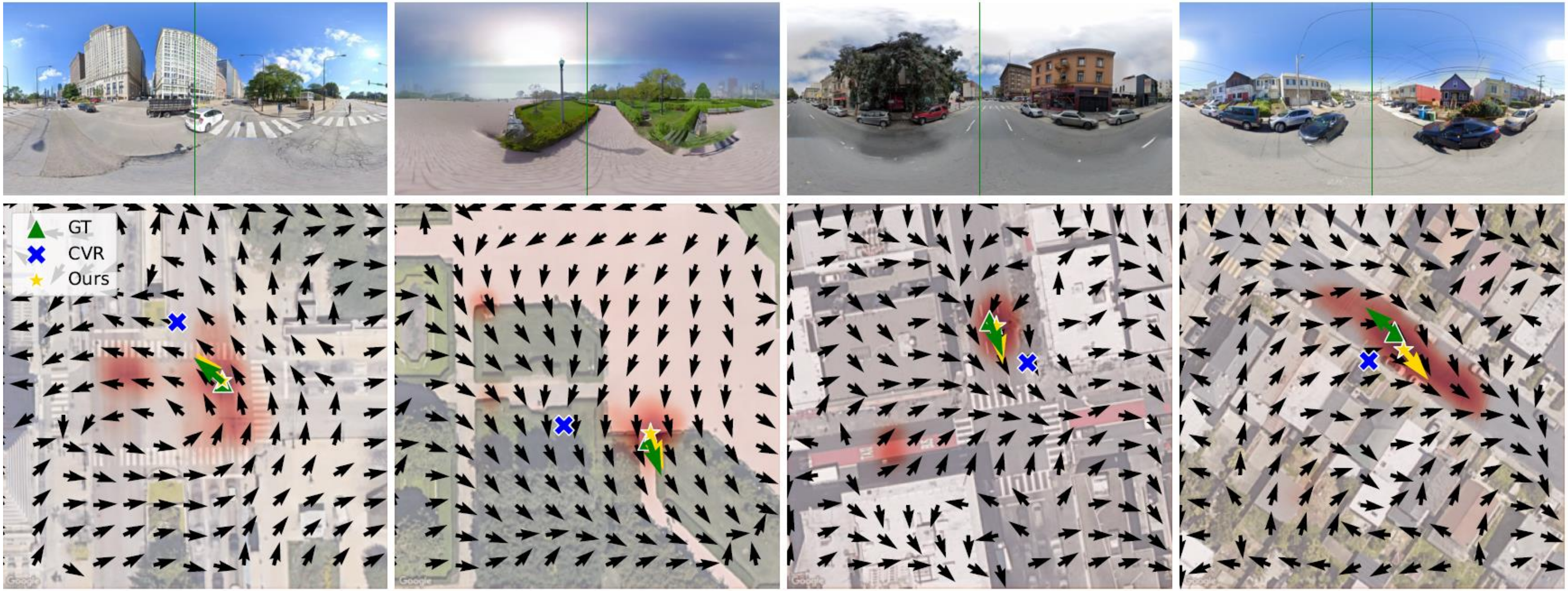
Architecture



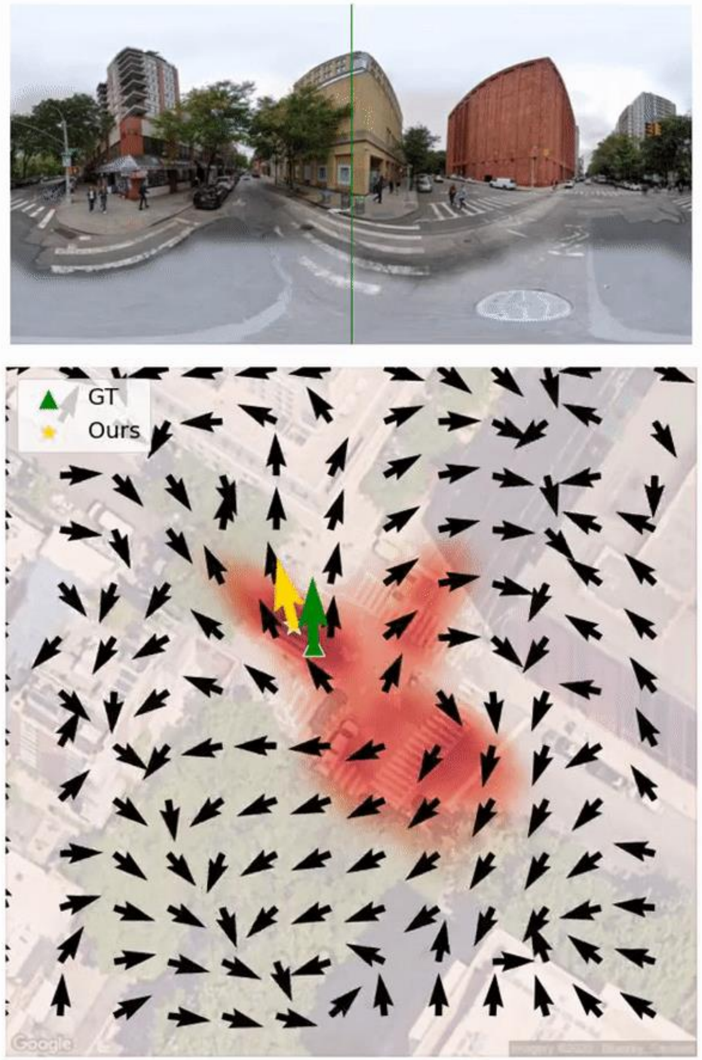


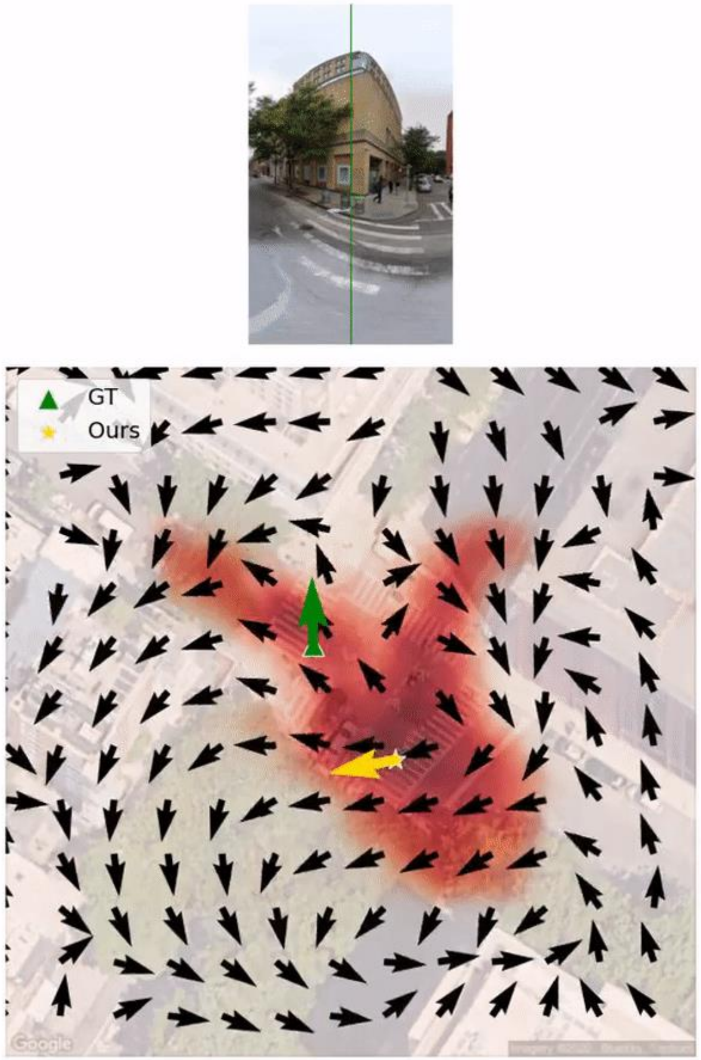
- OMU = Orientation Matching Upsampling
- LMU = Localization Matching Upsamping

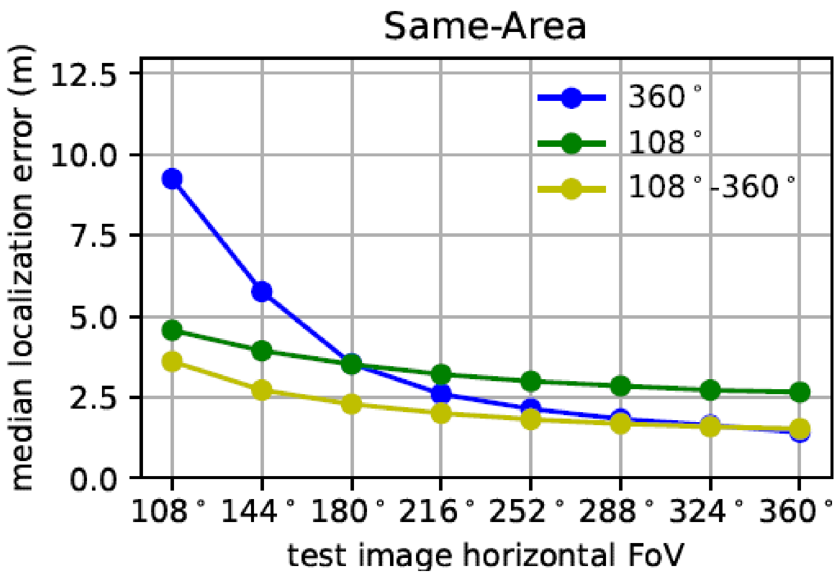
Qualitative results



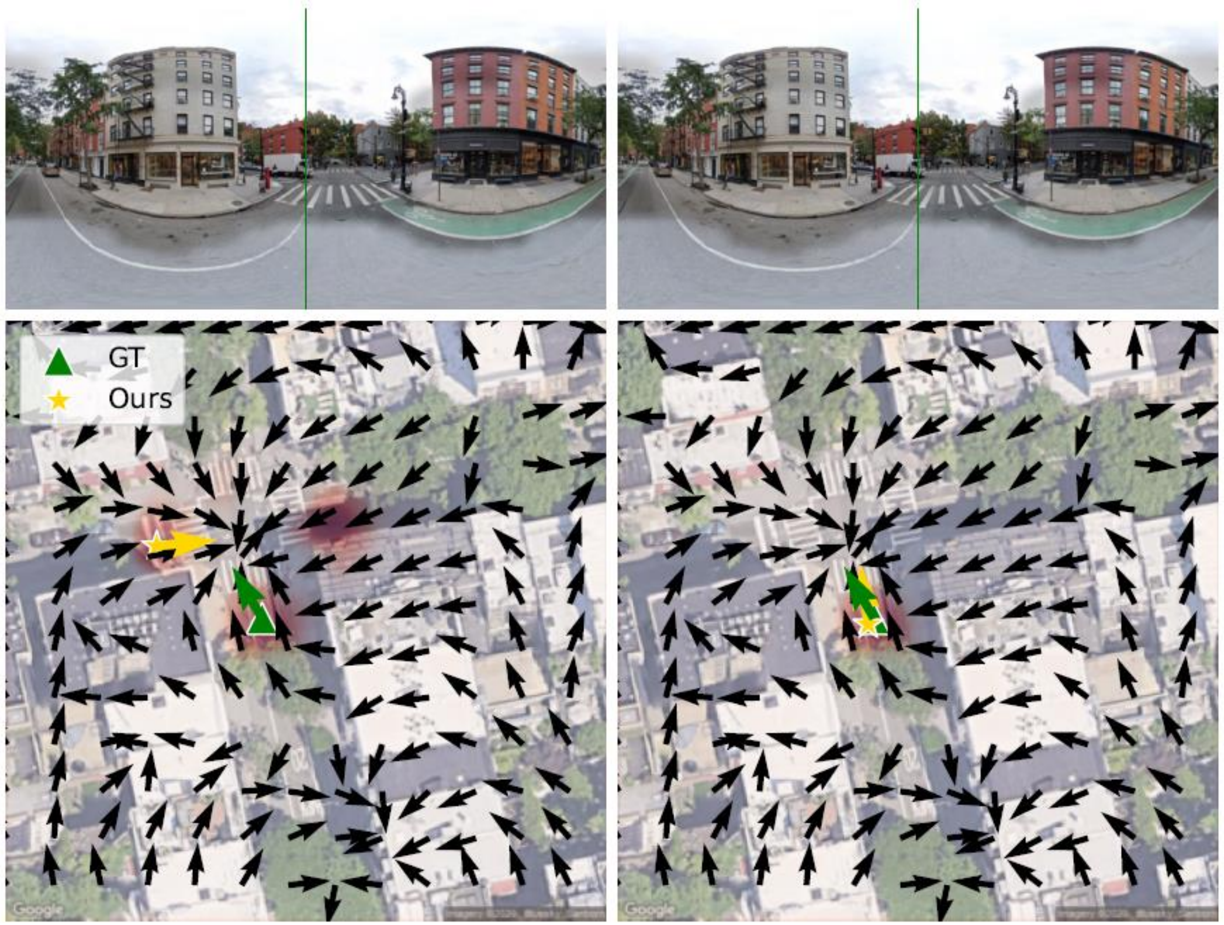
Qualitative results, varying FoV



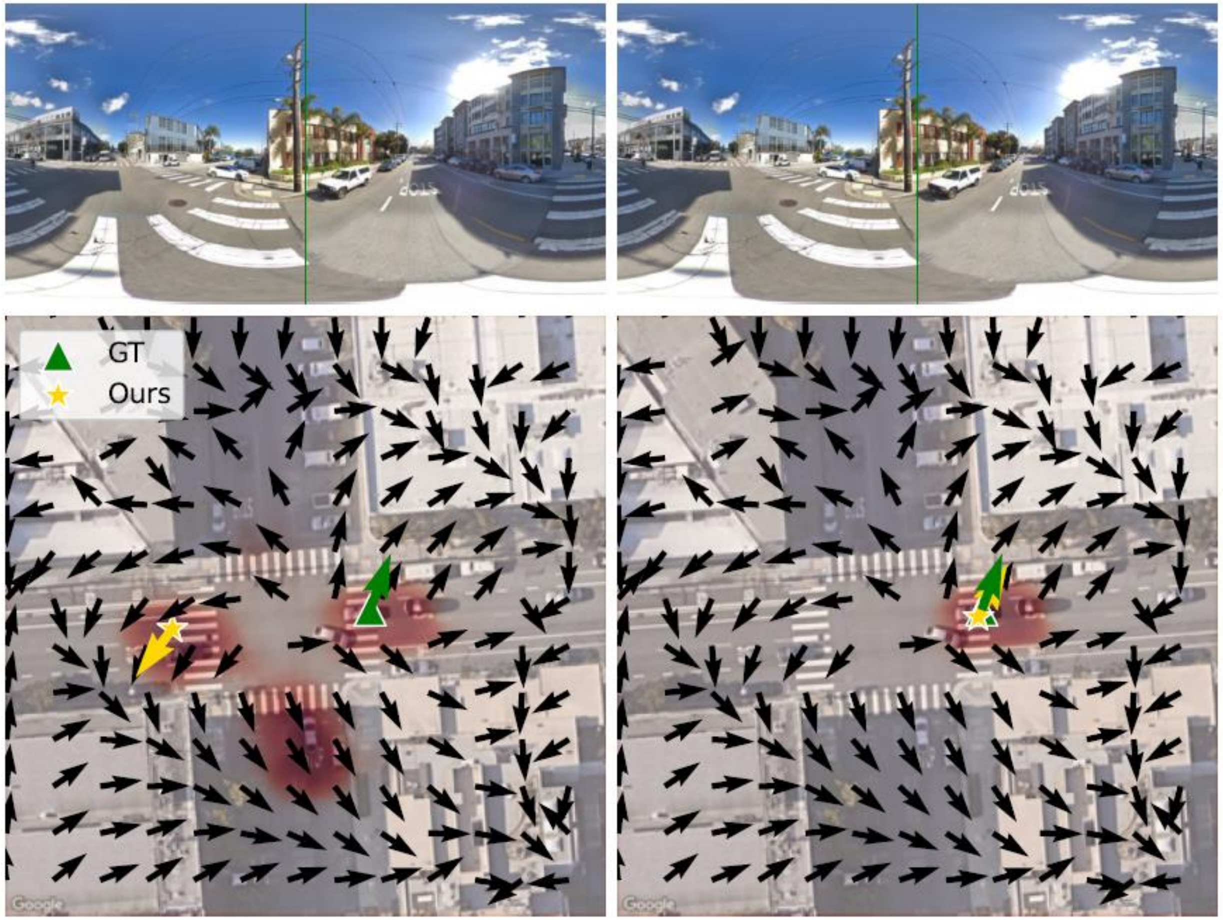




Adding orientation prior



no orientation prior | known orientation



no orientation prior | known orientation

Quantitative results

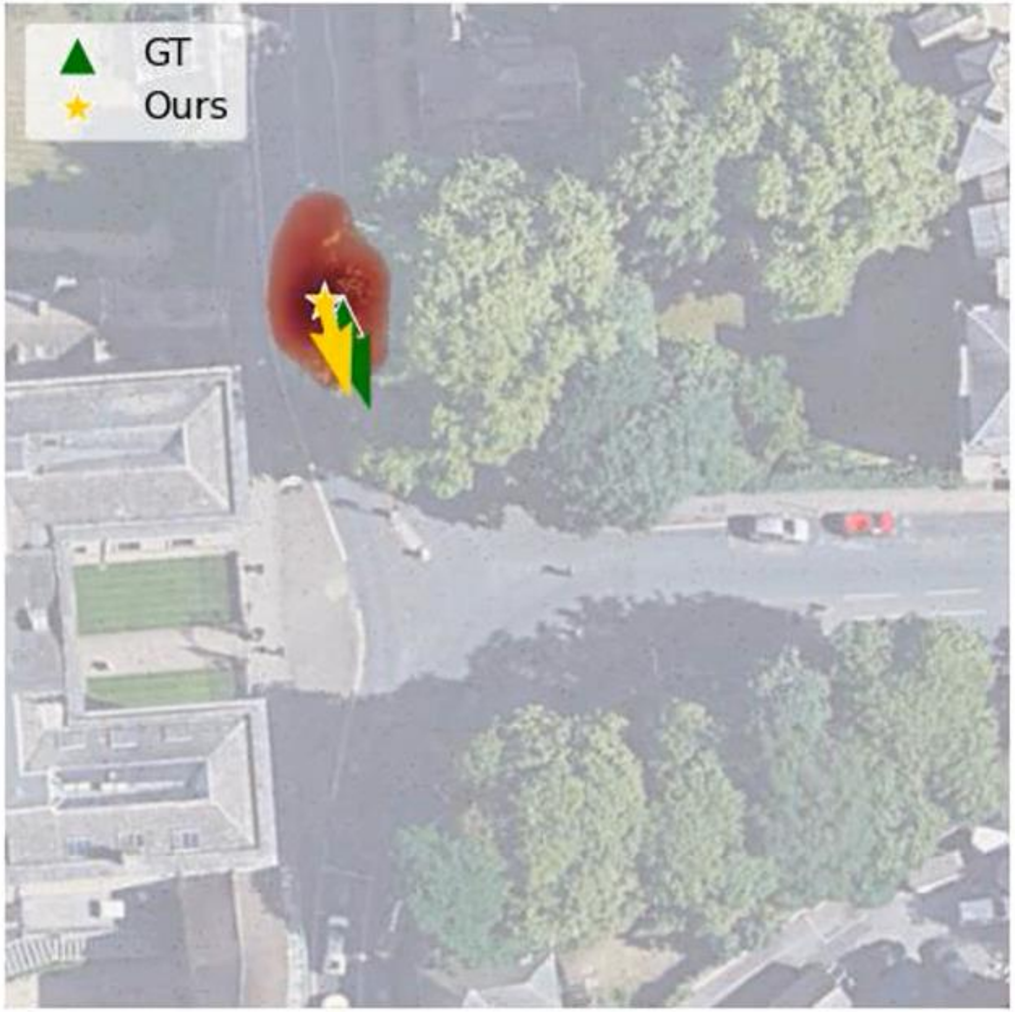
- **VIGOR**
  Large scale dataset, 4 cities in USA
  Same-area: train 2 cities, test same cities
  Cross-area: train 2 cities, test other cities

VIGOR test		Same-Area						Cross-Area					
		↓ Localization (m)		↓ Orientation (°)		↑ P@GT (10^{-3})		↓ Localization (m)		↓ Orientation (°)		↑ P@GT (10^{-3})	
		mean	median	mean	median	mean	median	mean	median	mean	median	mean	median
0°	CVR [22]	8.82	7.68	-	-	0.02	0.02	9.45	8.33	-	-	0.02	0.02
	Eff-CVR	7.89	6.25	-	-	0.02	0.03	8.27	6.60	-	-	0.02	0.03
	SliceMatch [32]	5.18	2.58	-	-	0.06	0.05	5.53	2.55	-	-	0.06	0.06
	CCVPE (ours)	**3.60**	**1.36**	-	-	**1.60**	**1.12**	**4.97**	**1.68**	-	-	**1.08**	**0.71**
360°	SliceMatch [32]	8.41	5.07	28.43	**5.15**	0.02	0.02	8.48	5.64	**26.20**	**5.18**	0.02	0.02
	CCVPE (ours)	**3.74**	**1.42**	**12.83**	6.62	**1.47**	**1.00**	**5.41**	**1.89**	27.78	13.58	**0.93**	**0.58**

- **KITTI**

Same-Area		↓ Localization (m)		↑ Lateral (%)			↑ Longitudinal (%)			↓ Orientation (°)		↑ Orientation (%)		
		mean	median	1m	3m	5m	1m	3m	5m	mean	median	1°	3°	5°
retrieval	CVM-Net [16]	-	-	5.83	17.41	28.78	3.47	11.18	18.42	-	-	-	-	-
	CVFT [19]	-	-	7.71	22.37	36.28	3.82	11.48	18.63	-	-	-	-	-
	SAFA [15]	-	-	9.49	29.31	46.44	4.35	12.46	21.10	-	-	-	-	-
	Polar-SAFA [15]	-	-	9.57	30.08	45.83	4.56	13.01	21.12	-	-	-	-	-
	DSM [47]	-	-	10.12	30.67	48.24	4.08	12.01	20.14	-	-	3.58	13.81	24.44
±10°	LM [39]	12.08	11.42	35.54	70.77	80.36	5.22	15.88	26.13	3.72	2.83	19.64	51.76	71.72
	SliceMatch [32]	7.96	4.39	49.09	91.76	98.52	15.19	49.99	57.35	4.12	3.65	13.41	42.62	64.17
	CCVPE (ours)	**1.22**	**0.62**	**97.35**	**98.65**	**99.71**	**77.13**	**96.08**	**97.16**	**0.67**	**0.54**	**77.39**	**99.47**	**99.95**
360°	LM [39]	15.51	15.97	5.17	15.13	25.44	4.66	15.00	25.39	89.91	90.75	0.61	1.88	2.89
	SliceMatch [32]	9.39	5.41	39.73	**80.56**	**87.92**	13.63	40.75	49.22	**8.71**	**4.42**	**11.35**	**36.23**	**55.82**
	CCVPE (ours)	**6.88**	**3.47**	**53.30**	77.63	85.13	**25.84**	**55.05**	**68.49**	15.01	6.12	8.96	26.48	42.75

Cross-view pose estimation on Oxford RobotCar



N.B.: Single frame results,
no temporal filtering, no sensor fusion!

Convolutional Cross-View Pose Estimation,
Z. Xia, O. Booij, J.F.P. Kooij, T-PAMI 2023, https://arxiv.org/abs/2303.05915

Conclusions

Cross-view pose estimation is a rapidly developing field

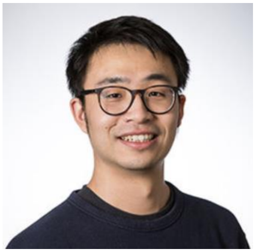
- Current focus is on fine-grained localization, relying on rough location prior
- Originally only localization, now also orientation estimation

Use Multi-modal distributions

  - Represent localization ambiguity, supports confidence estimation
  - Can condition orientation on localization estimate

- On Oxford RobotCar, we start to reach < 1m performance. VIGOR dataset is more difficult.
- Evaluation is single frame only, temporal and sensor fusion should highly improve accuracy

Adapting Fine-Grained Cross-View Localization to Areas without Fine Ground Truth


Zimin Xia[1]

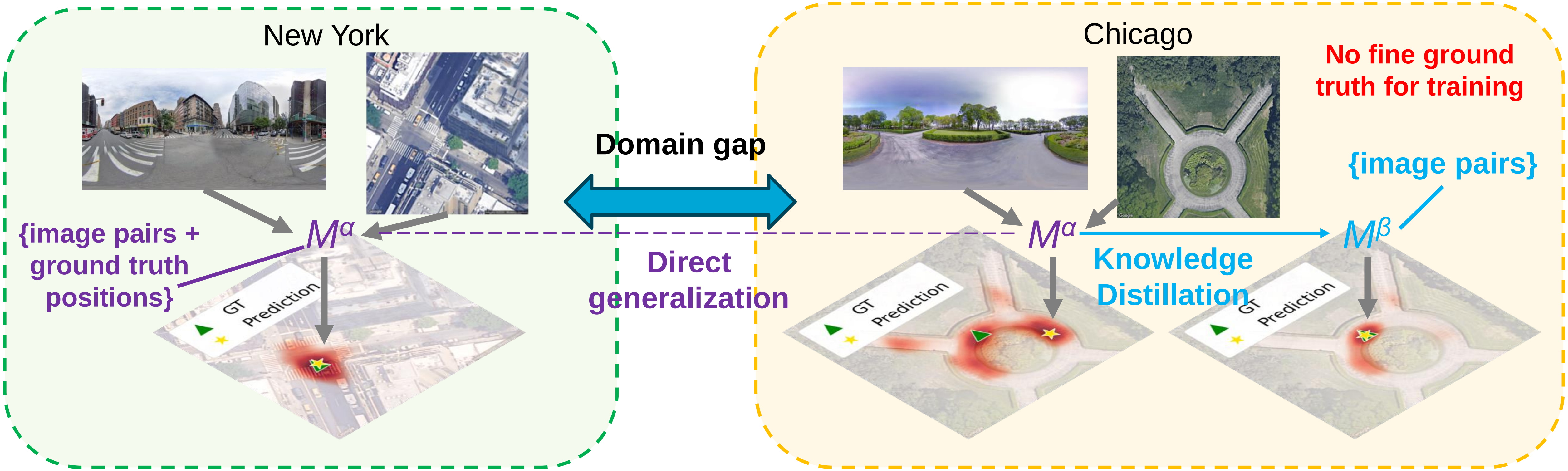

Yujiao Shi[2]


Hongdong Li[3]


Julian F. P. Kooij[4]

École Polytechnique Fédérale de Lausanne (EPFL)
ShanghaiTech University
Australian National University
Delft University of Technology

Does CV pose estimation generalize to new unseen areas?



- Collecting fine ground truth in a new target city can be expensive or infeasible
- Collecting images with noisy location data is easy, e.g. by phone-grade GPS

TU Delft

Can we use images of the new area without fine-grained location annotations to adapt our model?

Leveraging noisy location data to adapt the models to new areas

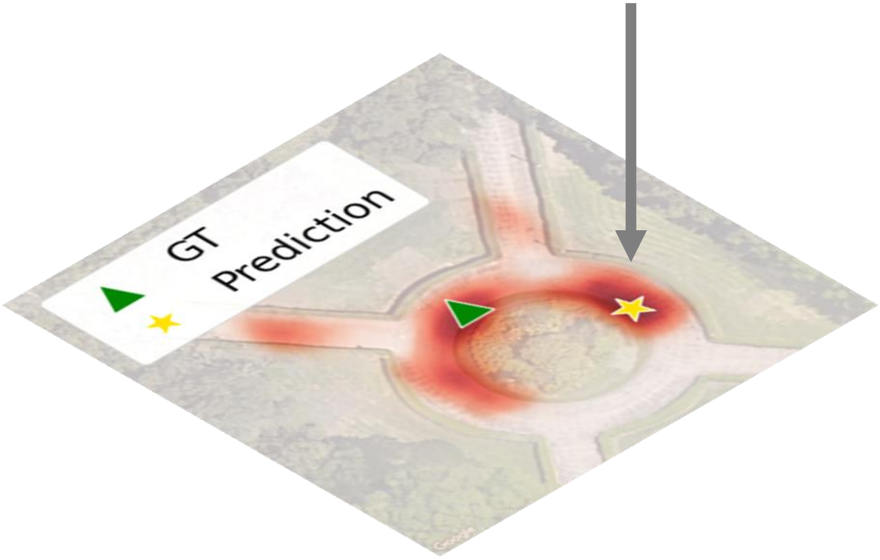
Observations:

- Direct generalization leads to uncertain localization:
  - high uncertainty
  - small positional noise
  - large outliers

- SOTA[1,2] predicts heat maps for localization

- SOTA[1,2] has coarse-to-fine outputs

Use these insights to design a suitable knowledge distillation strategy



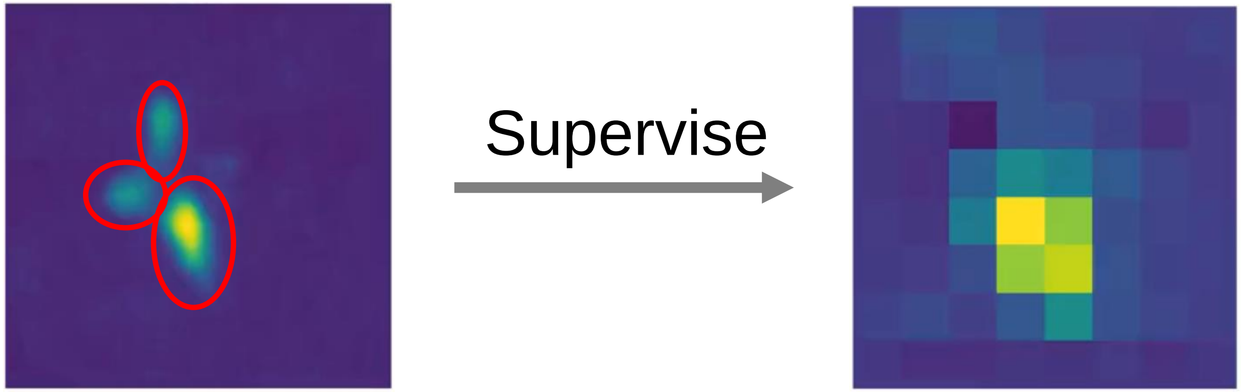
TUDelft

1. Xia et al. "Convolutional cross-view pose estimation." TPAMI 2023
2. Shi et al. "Boosting 3-dof ground-to-satellite camera localization accuracy via geometry-guided cross-view transformer." ICCV 2023.

Leveraging noisy location data to adapt the models to new areas

- Use the pretrained model as a <u>teacher</u> to supervise a copy (<u>student</u>) of itself
- But how can this really improve the model?

1. Use teacher's high-res output to supervise student's lower-resolution heat maps (remember: SotA models use coarse-to-fine strategy)
2. Take teacher's multi-modal output and select single mode as "pseudo GT" *"pretend the teacher's best guess is correct"*



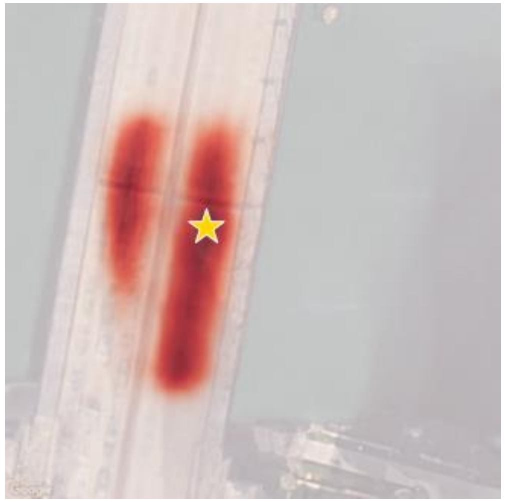
Now we can train student model on fine-grain pseudo GT for the images at the target area. But is it good?

Leveraging noisy location data to adapt the models to new areas

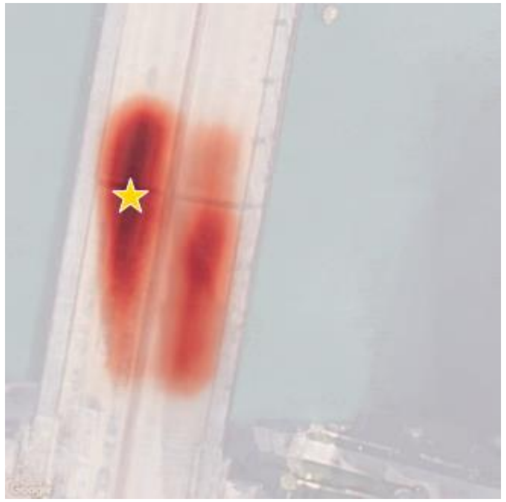
What if there are multiple equally likely modes?

Train an auxiliary student model for outlier detection:

- Measure difference between the teacher's and the student's predictions
- If the difference is large, exclude this sample in final student model training
- Keep top X% more consistent samples



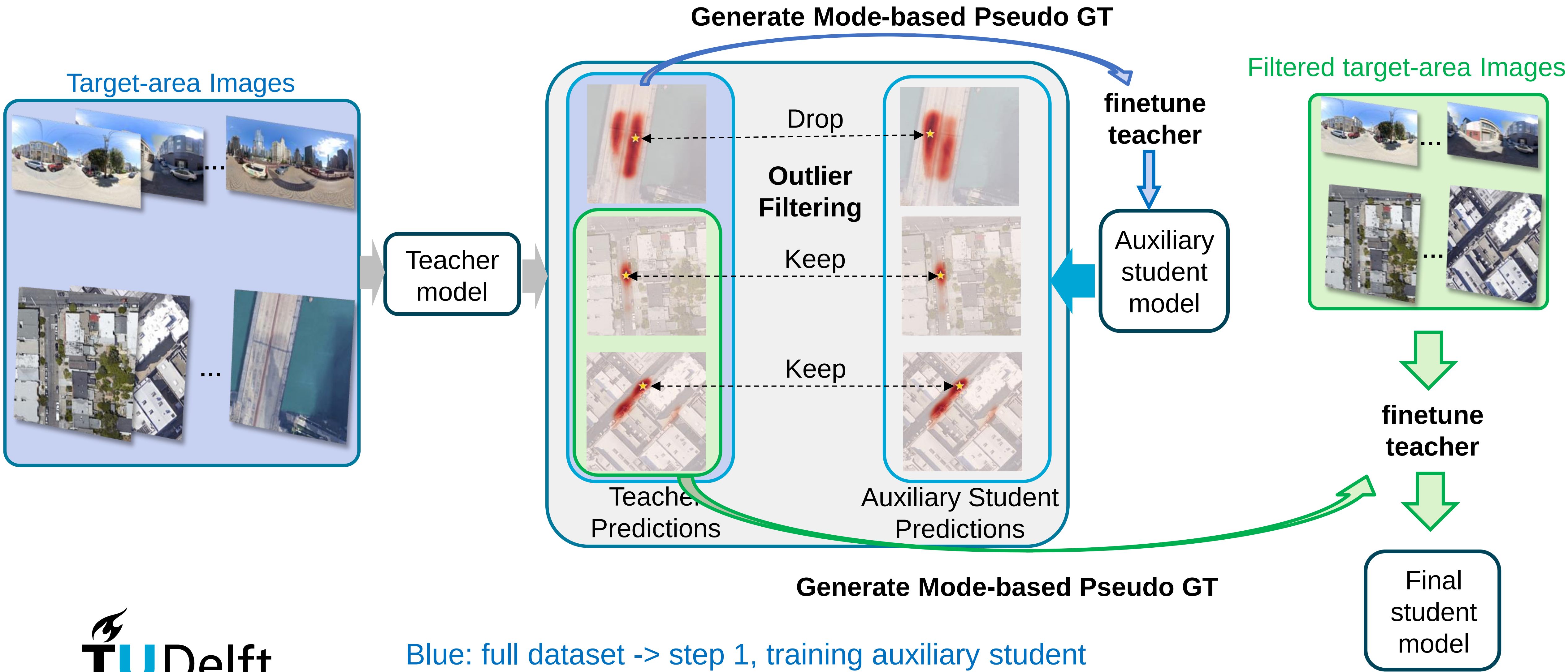
Teacher's prediction



Auxiliary student's prediction

TU Delft

Leveraging noisy location data to adapt the models to new areas



Blue: full dataset -> step 1, training auxiliary student

Green: outlier filtered dataset -> step 2, training final student

Experimental results

VIGOR, cross-area test	Known orientation		Unknown orientation	
	Mean (m)	Median (m)	Mean (m)	Median (m)
CCVPE [56]	4.38	1.76	5.35	1.97
CCVPE student (ours)	**3.85** (↓ 12%)	**1.57** (↓ 11%)	**4.27** (↓ 20%)	**1.67** (↓ 15%)
GGCVT [39]	5.19	1.39	-	-
GGCVT student (ours)	**4.34** (↓ 16%)	**1.32** (↓ 5%)	-	-

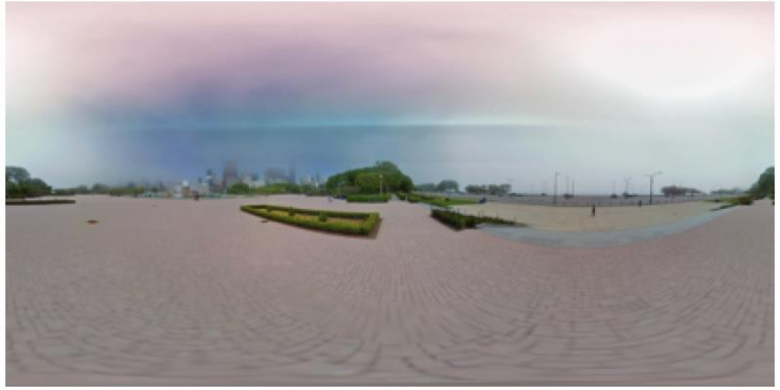
KITTI, cross-area test	Longitudinal error		Lateral error	
	Mean (m)	Median (m)	Mean (m)	Median (m)
CCVPE [56]	6.55	2.55	1.82	**0.98**
CCVPE student (ours)	**6.18** (↓ 6%)	**2.35** (↓ 8%)	**1.76** (↓ 3%)	**0.98** (↓ 0%)
GGCVT [39]	9.27	4.66	2.19	0.85
GGCVT student (ours)	**8.56** (↓ 8%)	**4.35** (↓ 7%)	**1.90** (↓ 13%)	**0.79** (↓ 7%)

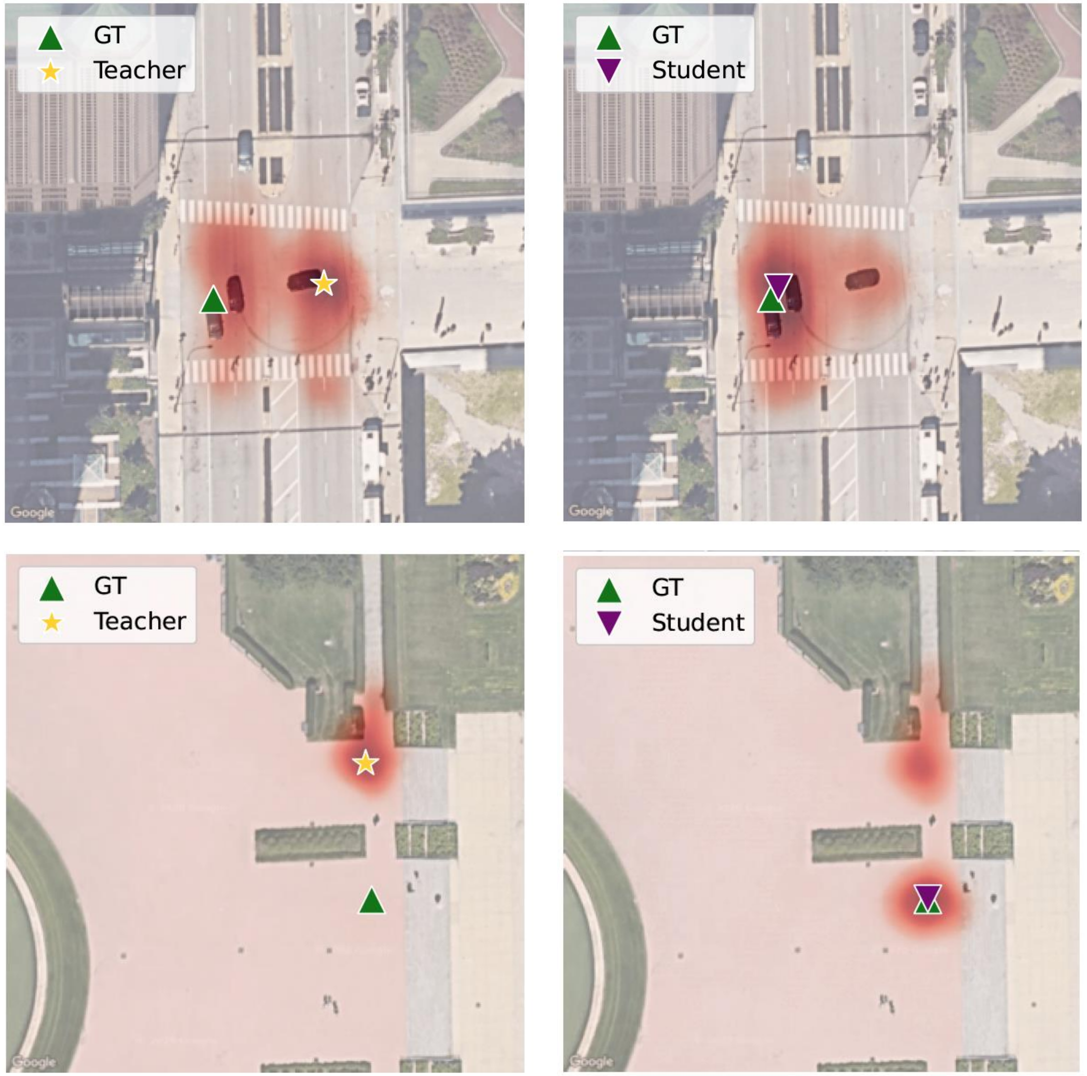
- Our approach shows consistent and considerable improvement for two SOTA methods on two benchmarks
- We also tried generic Domain Adaptation using Entropy Minimization, but that does not work: it just makes the heat maps sharper, but does not resolve wrong modes!

TU Delft

CCVPE: Xia et al. "Convolutional cross-view pose estimation." TPAMI 2023
GGCVT: Shi et al. "Boosting 3-dof ground-to-satellite camera localization accuracy via geometry-guided cross-view transformer." ICCV 2023.

Experimental results





The student learns to better disambiguate modes and explore new modes

Thank you! Questions?

SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation
Lentsch et al., CVPR 2023

[Github] [arXiv]

Convolutional Cross-View Pose Estimation
Xia et al., T-PAMI 2023

[Github] [arXiv]

Adapting Fine-Grained Cross-View Localization to Areas without Fine Ground Truth
Xia et al., ECCV 2024

[Github] [arXiv]

SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation

Ted Lentsch* Zimin Xia* Holger Caesar Julian F. P. Kooij
Intelligent Vehicles Group, Delft University of Technology, The Netherlands
{T.deVriesLentsch,Z.Xia,H.Caesar,J.F.P.Kooij}@tudelft.nl

Convolutional Cross-View Pose Estimation

Zimin Xia, Olaf Booij, and Julian F. P. Kooij, *Member, IEEE*

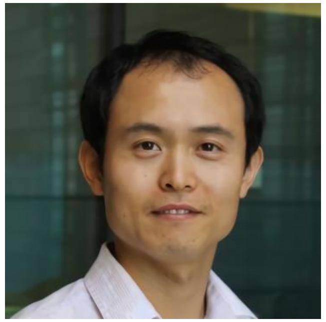
Adapting Fine-Grained Cross-View Localization to Areas without Fine Ground Truth

Zimin Xia[1], Yujiao Shi[2], Hongdong Li[3], and Julian F. P. Kooij[4]
[1] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
zimin.xia@epfl.ch
[2] ShanghaiTech University, China
[3] Australian National University, Australia
[4] Delft University of Technology, The Netherlands

CVPR
JUNE 17-21, 2024
SEATTLE, WA

On the Estimation of Image-matching Uncertainty in Visual Place Recognition

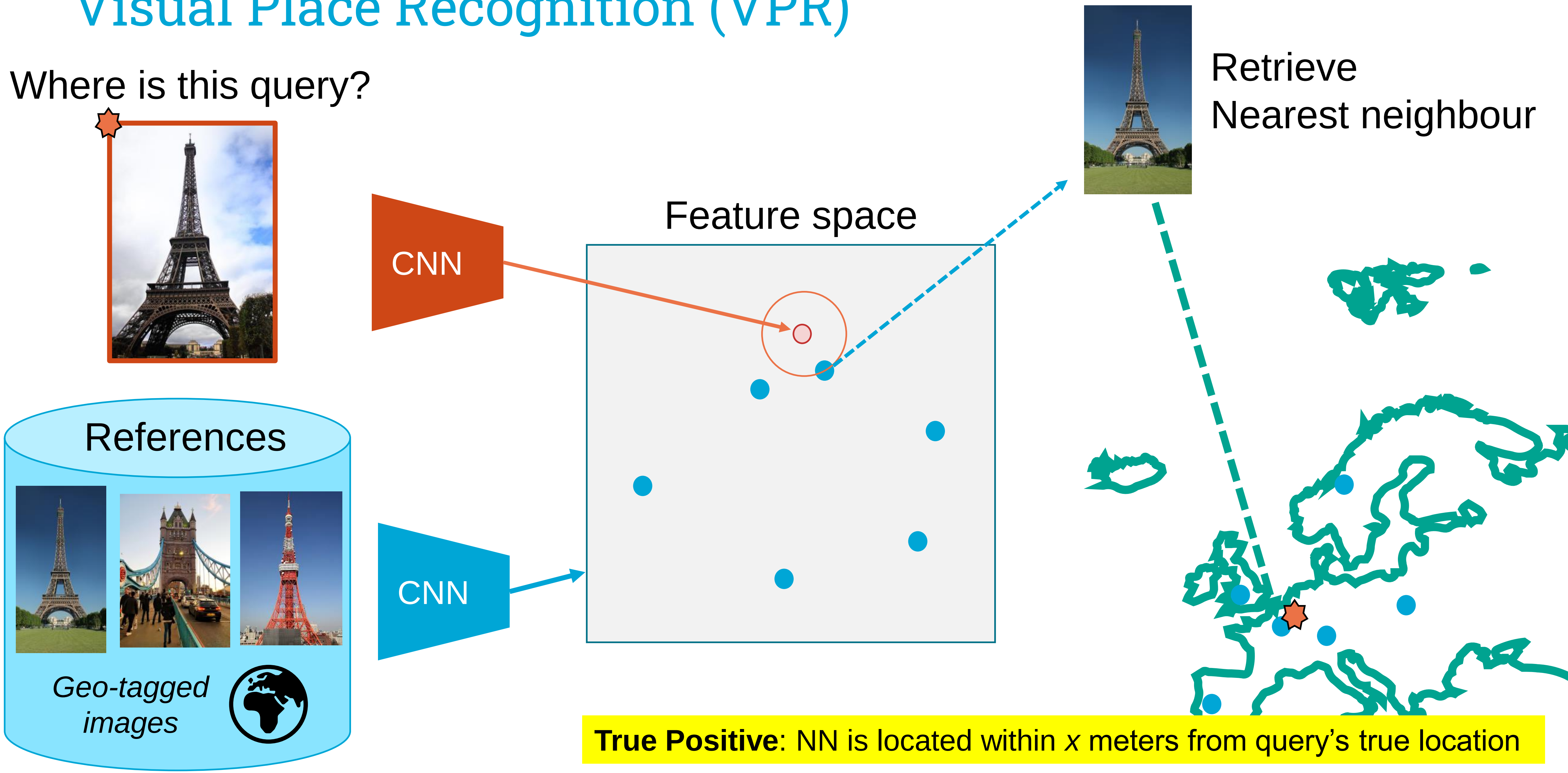
CVPR 2024 (*Poster Highlight*)


Mubariz Zaffar   Liangliang Nan   Julian F. P. Kooij

Intelligent Vehicles Group / 3D Geoinformation Group
TU Delft, The Netherlands

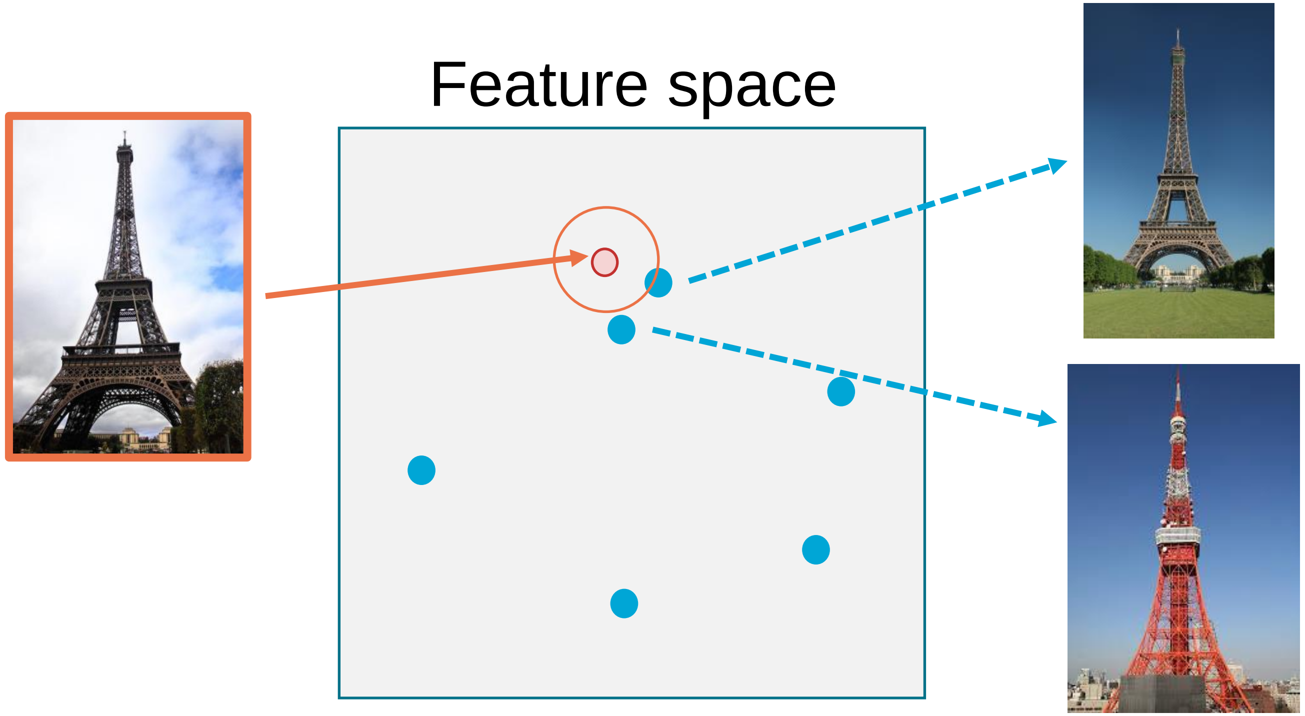
Visual Place Recognition (VPR)



True Positive: NN is located within *x* meters from query's true location

VPR Matching Uncertainty

- How do we know if retrieved VPR result is reliable?
- "Visual aliasing": some locations just look similar (patch of sky, white wall, …)



- Various procedures to estimate VPR Matching Uncertainty exist …

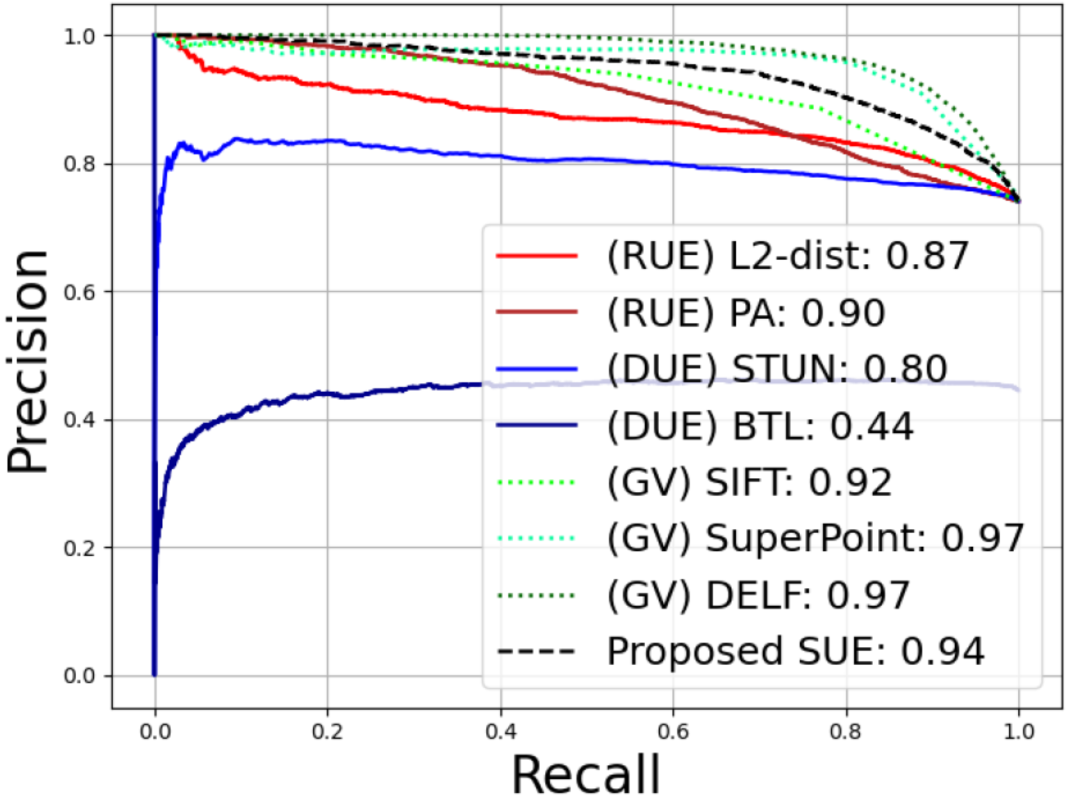
VPR Matching Uncertainty

We categorize current VPR uncertainty estimation

- RUE: Retrieval-based uncertainty estimation
- DUE: Data-driven uncertainty estimation
- GV: Geometric verification

Propose a new simple approach

- **SUE**: Spatial Uncertainty Estimation

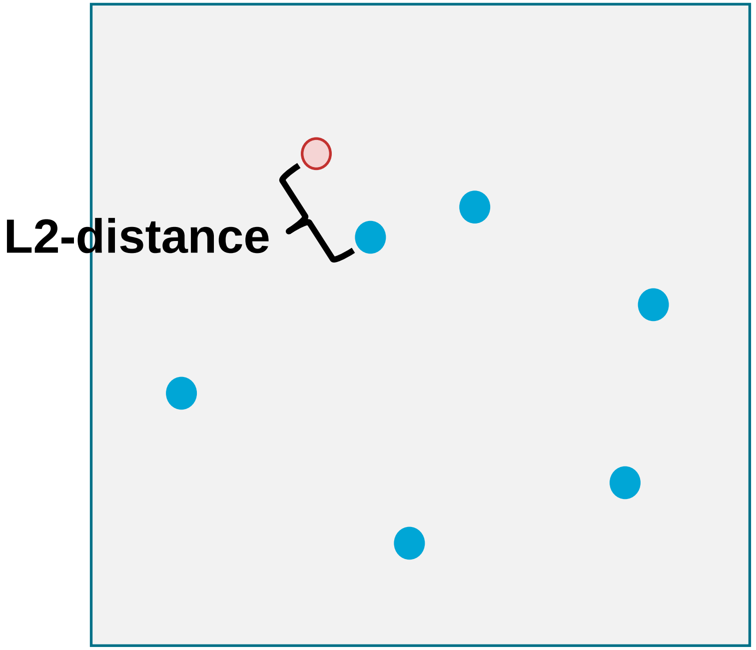


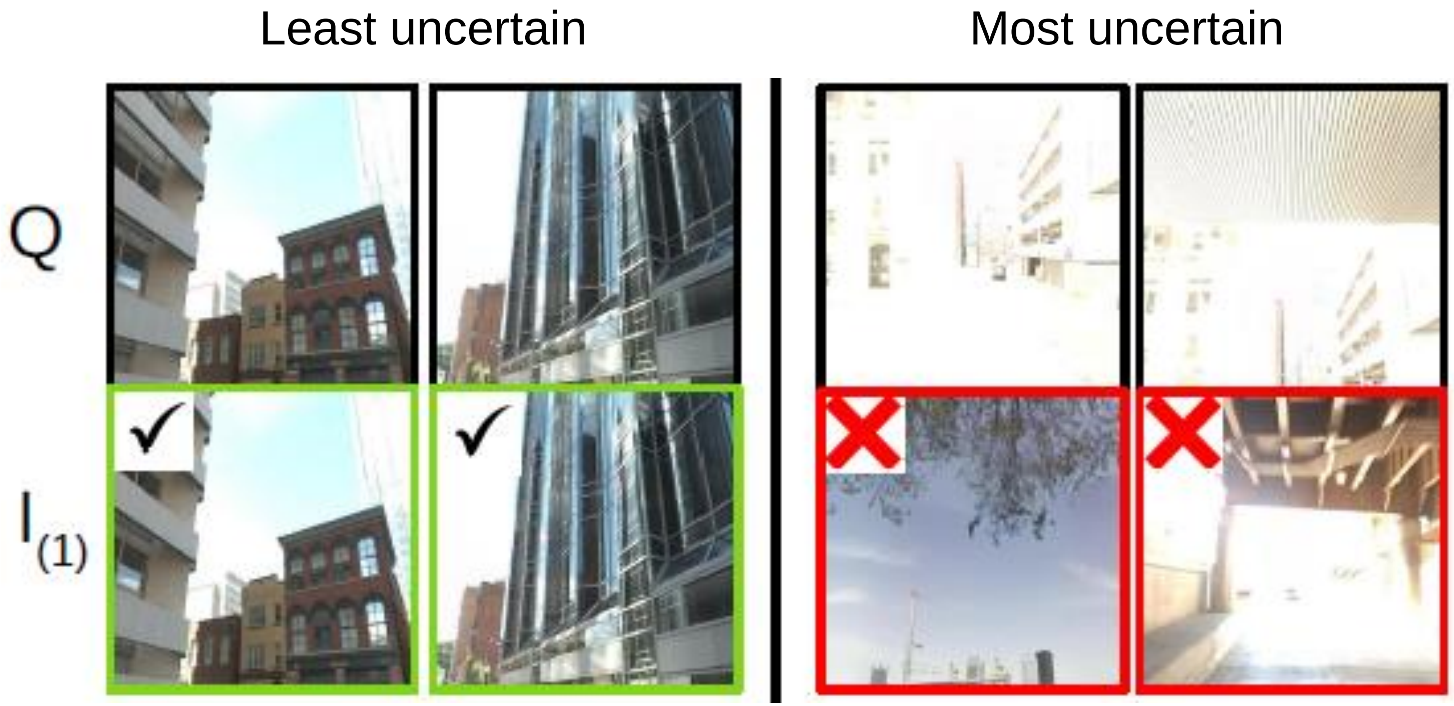
RUE: Retrieval-based uncertainty estimation

Common confidence score used in Visual SLAM for loop closure

- **L2-distance:** between query feature and best-matching feature
- **PA-Score:** ratio of L2-distance between 1st and 2nd nearest neighbour reference

Feature space





DUE: Data-driven uncertainty estimation

Estimate (aleatoric) uncertainty from image content

- **Bayesian Triplet Loss (BTL):** Warburg et al., ICCV 2021
- **STUN**: Cai et al., IROS 2022

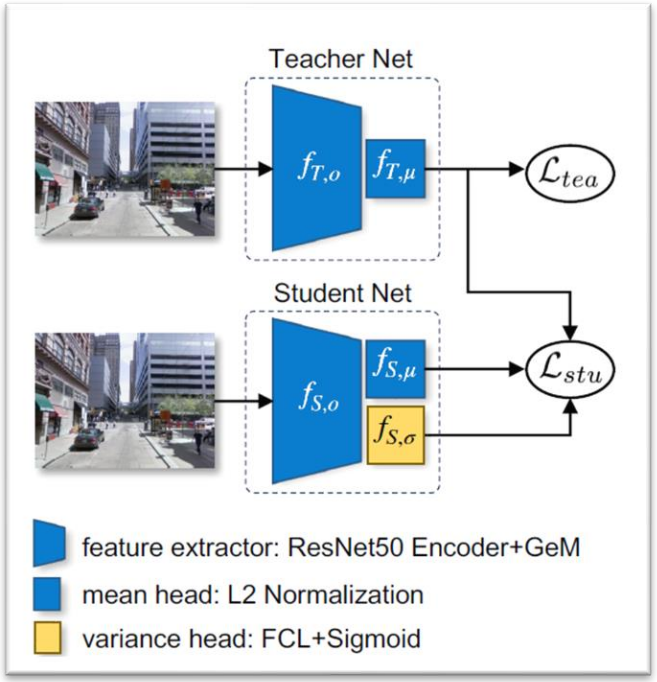
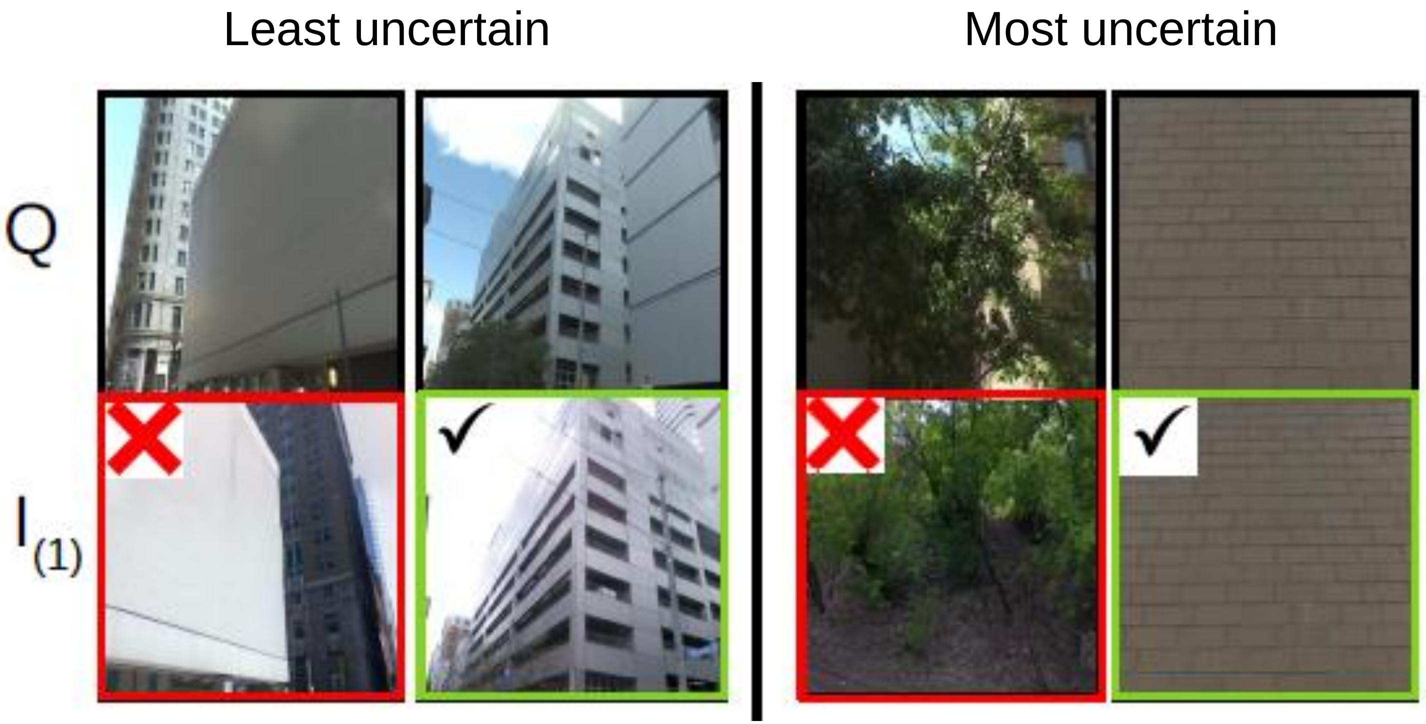


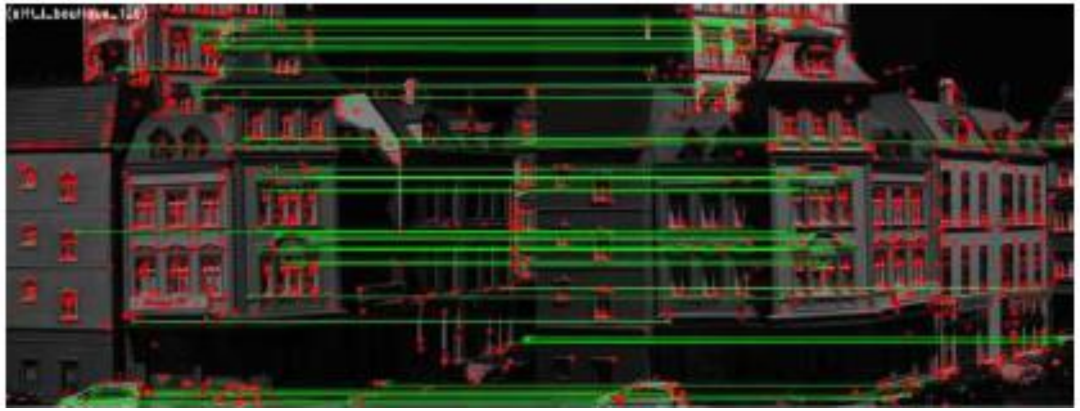
Image: [Cai et al., IROS 2022]



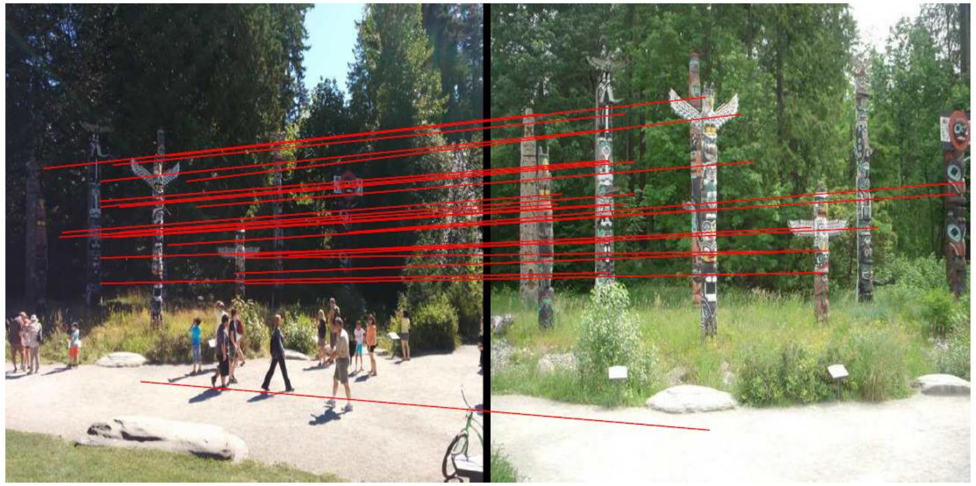
GV: Geometric Verification

Local feature matching + RANSAC
Slow but accurate, typically used to rerank top-K retrieved candidates

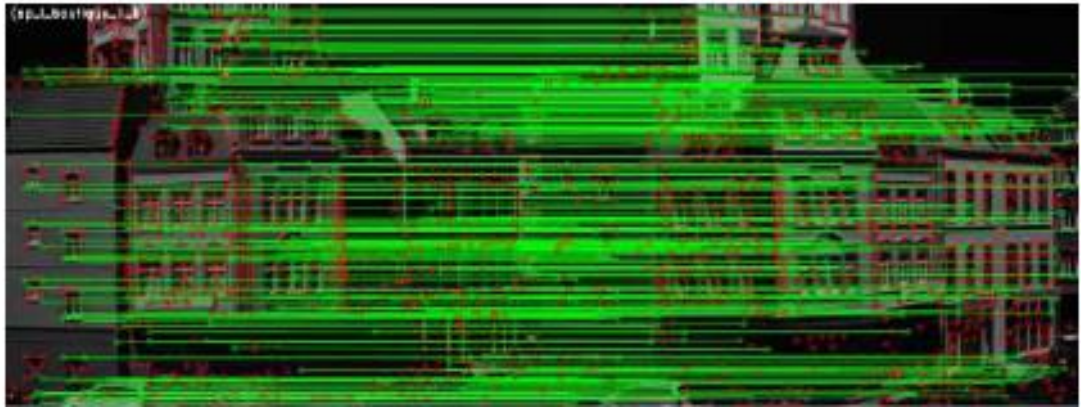
- **SIFT**: Lowe, IJCV 2004
- **DELF**: Noh et al., ICCV 2017
- **SuperPoint**: DeTone, CVPRW 2018



SIFT (image credit: [DeTone, CVPRW'18)



DELF (image credit: Noh, ICCV'17)



SuperPoint (image credit: DeTone, CVPRW'18)

SUE: Spatial Uncertainty Estimation

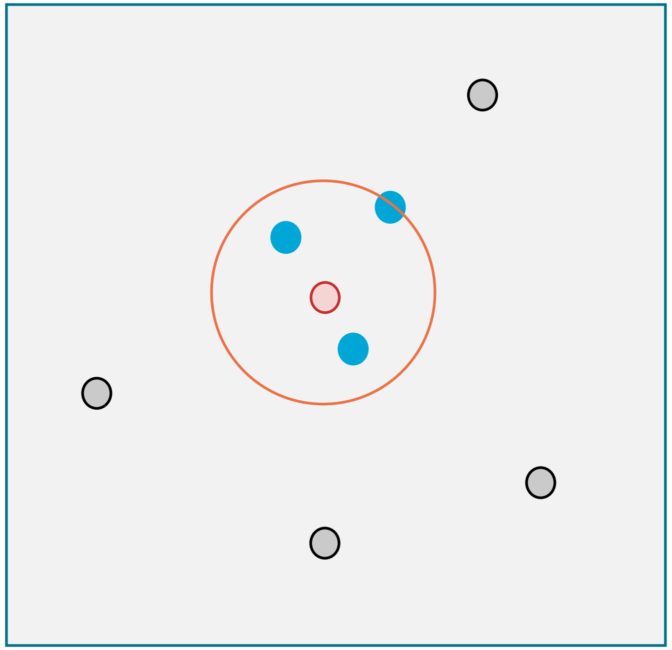
- Existing approaches do not consider geo-locations of the references!
- SUE estimates if visual aliasing occurs between distant map locations

- *SUE is ridiculously simple!*

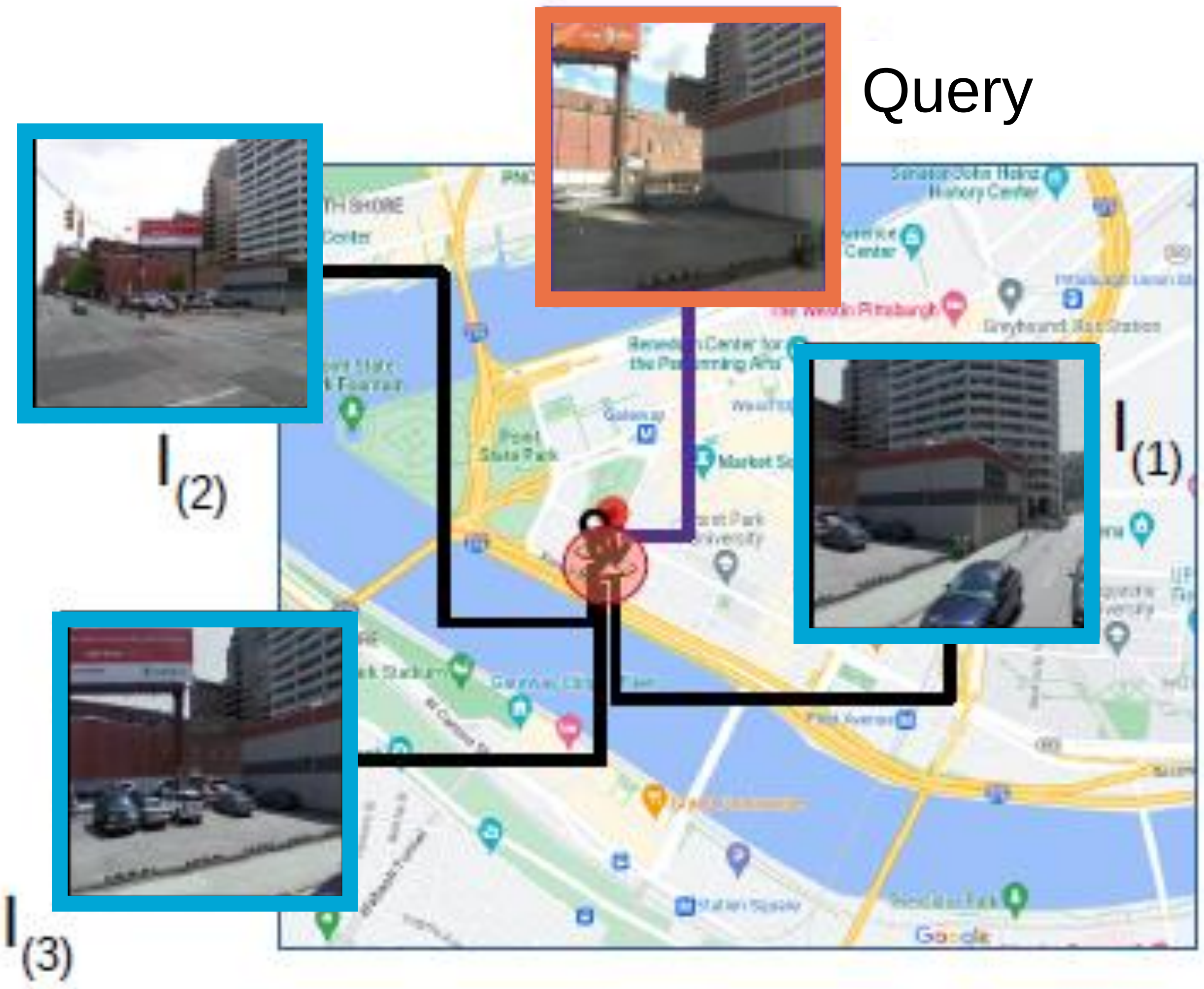
1. For a query, retrieve top-k best matching refences
2. Weigh top-k references inversely by L2-distance
3. Compute weighted variance of *spatial locations* of top-k references
4. Total variance is matching uncertainty score

SUE: Spatial Uncertainty Estimation

Feature space

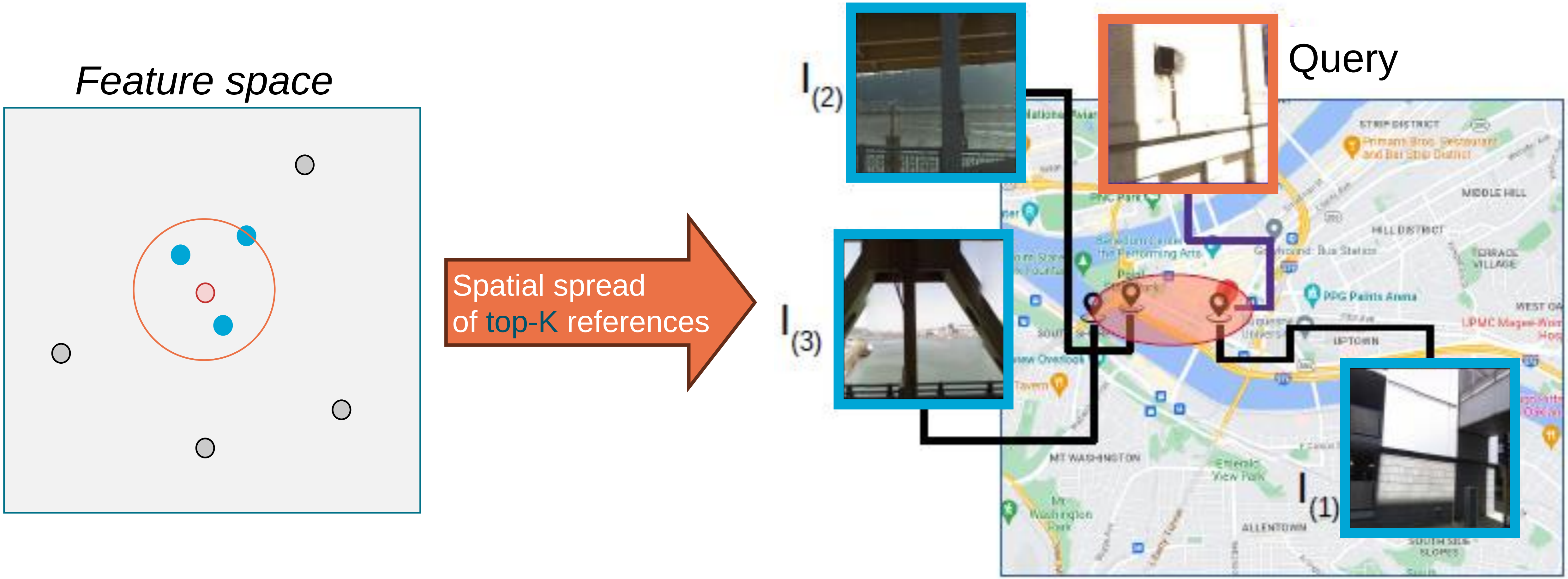


Spatial spread of top-K references



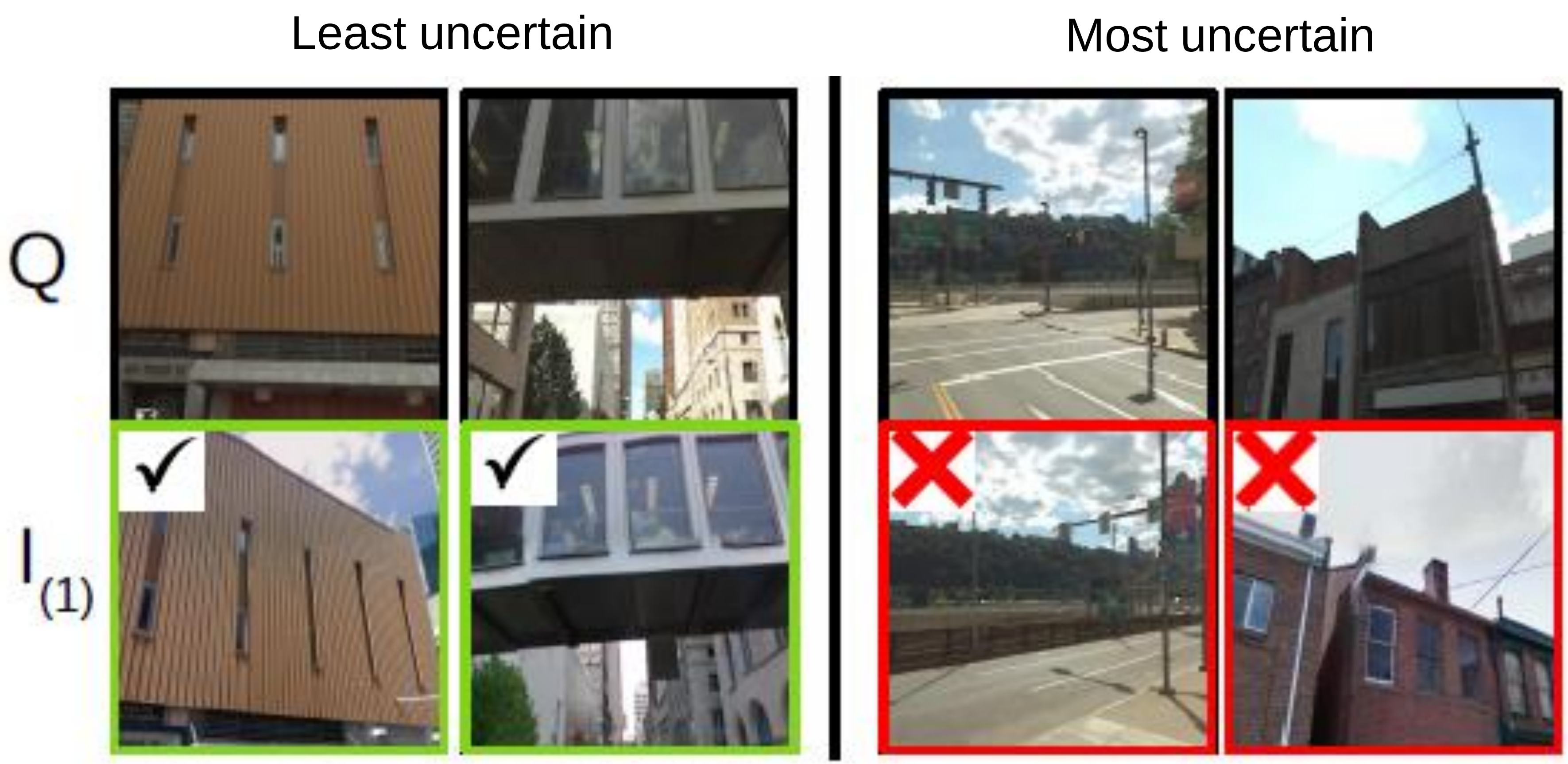
Low spatial spread → low SUE uncertainty

SUE: Spatial Uncertainty Estimation



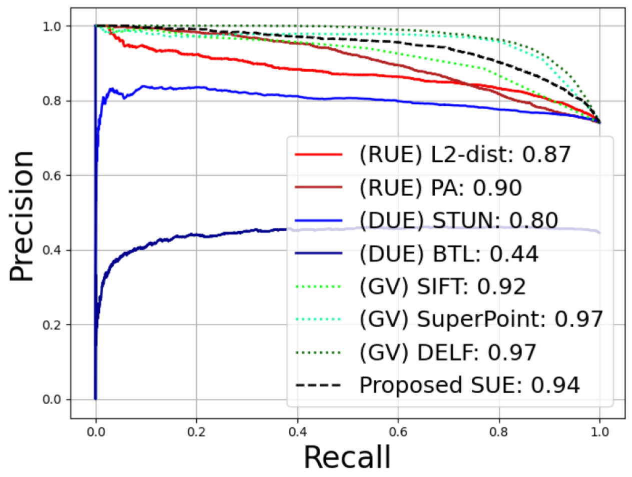
High spatial spread → high SUE uncertainty

SUE: Spatial Uncertainty Estimation



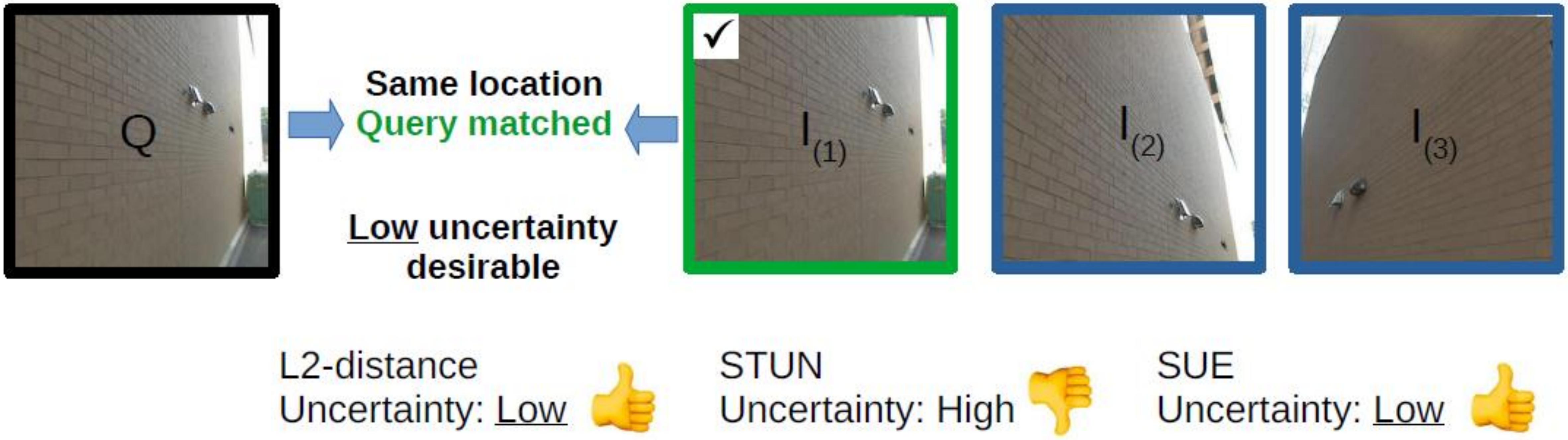
Results

- AUC-PR on various VPR benchmarks
- Processing time (milliseconds)

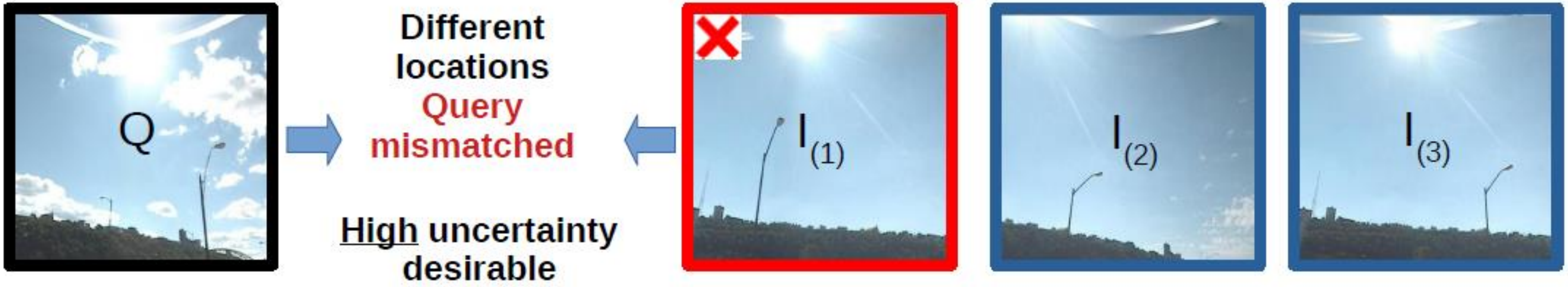


Method	↑ Pitts.	↑ Sanfr.	↑ Stluc.	↑ Eyn.	↑ MSLS	↑ Nordland	↑ Average	↓ Time
(RUE) L2-distance	0.87	0.76	0.79	0.87	0.64	0.18	0.69	**0.05**
(RUE) PA-Score [18]	0.90	0.65	0.77	0.88	0.68	0.21	0.68	**0.05**
(DUE) BTL [50]	0.44	0.17	0.34	0.45	0.21	0.07	0.28	0.20
(DUE) STUN [9]	0.79	0.57	0.66	0.71	0.44	0.05	0.54	0.10
SUE	**0.94**	**0.84**	**0.88**	**0.93**	**0.77**	**0.26**	**0.77**	1.08
(GV) SIFT-RANSAC [27]	0.92	0.89	0.93	**0.96**	0.70	0.15	0.76	**129**
(GV) DELF-RANSAC [33]	0.97	**0.92**	**0.97**	0.95	**0.95**	**0.84**	**0.93**	1587
(GV) Super-RANSAC [15]	0.95	**0.95**	**0.97**	**0.96**	0.87	0.50	0.87	848

Examples


Q
Same location
Query matched
Low uncertainty
desirable
✓
$I_{(1)}$
$I_{(2)}$
$I_{(3)}$
L2-distance
Uncertainty: Low
STUN
Uncertainty: High
SUE
Uncertainty: Low

Examples



L2-distance
Uncertainty: Low 👎

STUN
Uncertainty: <u>High</u> 👍

SUE
Uncertainty: <u>High</u> 👍

Conclusions

- Compared different approaches for estimating VPR uncertainty
- Recent work tends to ignore simple baselines!
- Simple L2-distance already outperforms Deep Learning-based

- We propose SUE, which also looks at reference locations
- SUE is best efficient method

- In paper we show SUE can complement computationally expensive Geometric Verification

Thank you! Questions?

SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation
Lentsch et al., CVPR 2023

[Github](#) [arXiv](#)

Convolutional Cross-View Pose Estimation
Xia et al., T-PAMI 2023

[Github](#) [arXiv](#)

On the Estimation of Image-matching Uncertainty in Visual Place Recognition
Zaffar et al., CVPR 2024

[arXiv](#)

SliceMatch: Geometry-guided Aggregation for Cross-View Pose Estimation

Ted Lentsch* Zimin Xia* Holger Caesar Julian F. P. Kooij
Intelligent Vehicles Group, Delft University of Technology, The Netherlands
{T.deVriesLentsch,Z.Xia,H.Caesar,J.F.P.Kooij}@tudelft.nl

Abstract

This work addresses cross-view camera pose estimation, i.e., determining the 3-Degrees-of-Freedom camera pose of a given ground-level image w.r.t. an aerial image of the local area. We propose SliceMatch, which consists of ground and aerial feature extractors, feature aggregators, and a pose predictor. The feature extractors extract dense features from the ground and aerial images. Given a set of candidate camera poses, the feature aggregators construct a single ground descriptor and a set of pose-dependent aerial descriptors. Notably, our novel aerial feature aggregator has a cross-view attention module for ground-view guided aerial feature selection and utilizes the geometric projection of the ground camera's viewing frustum on the aerial image to pool features. The efficient construction of aerial descriptors is achieved using precomputed masks. SliceMatch is trained using contrastive learning and pose estimation is formulated as a similarity comparison between the ground descriptor and the aerial descriptors. Compared to the state-of-the-art, SliceMatch achieves a 19% lower median localization error on the VIGOR benchmark using the same VGG16 backbone at 150 frames per second, and

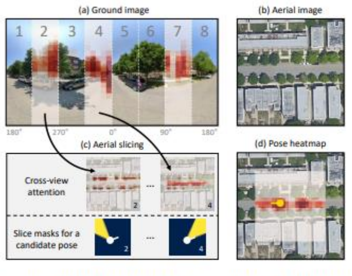


Figure 1. **SliceMatch identifies for a ground-level image (a) its camera's 3-DoF pose within a corresponding aerial image (b).** It divides the camera's Horizontal Field-of-View (HFoV) into 'slices', i.e., vertical regions in (a). After self-attention, our novel aggregation step (c) applies cross-view attention to create ground slice-specific aerial feature maps. To efficiently test many candidate poses, the slice features are aggregated using pose-dependent aerial slice masks that represent the camera's sliced HFoV at that pose. The slice masks for each pose are precomputed. All aerial

Convolutional Cross-View Pose Estimation

Zimin Xia, Olaf Booij, and Julian F. P. Kooij, *Member, IEEE*

Abstract—We propose a novel end-to-end method for cross-view pose estimation. Given a ground-level query image and an aerial image that covers the query's local neighborhood, the 3 Degrees-of-Freedom camera pose of the query is estimated by matching its image descriptor to descriptors of local regions within the aerial image. The orientation-aware descriptors are obtained by using a translational equivariant convolutional ground image encoder and contrastive learning. The Localization Decoder produces a dense probability distribution in a coarse-to-fine manner with a novel Localization Matching Upsampling module. A smaller Orientation Decoder produces a vector field to condition the orientation estimate on the localization. Our method is validated on the VIGOR and KITTI datasets, where it surpasses the state-of-the-art baseline by 72% and 36% in median localization error for comparable orientation estimation accuracy. The predicted probability distribution can represent localization ambiguity, and enables rejecting possible erroneous predictions. Without re-training, the model can infer on ground images with different field of views and utilize orientation priors if available. On the Oxford RobotCar dataset, our method can reliably estimate the ego-vehicle's pose over time, achieving a median localization error under 1 meter and a median orientation error of around 1 degree at 14 FPS.

Index Terms—Cross-view matching, camera pose estimation, aerial imagery, localization, orientation estimation.

arXiv:2303.05915v1 [cs.CV] 9 Mar 2023

1 Introduction

Localization, is a core task in autonomous driving and outdoor robotics [1]. In urban canyons [2], Global Navigation Satellite System (GNSS) such as GPS, often has positioning errors of up to tens of meters due to the multipath effect. Thus, other sensors [3], such as camera [4], [5], [6] and LiDAR [7], [8], [9], are used in combination with detailed HD maps [10], [11] to enhance the localization accuracy and robustness. In practice, most commercial vehicles are not equipped with expensive LiDAR sensors. Besides, maintaining an up-to-date HD map is laborious and expensive, especially for areas in fast development. Hence, exploring alternative map sources for camera-based methods is an important and practical task. One promising map source is aerial imagery as it provides rich semantic information with global coverage.

We consider the task of cross-view camera pose estimation, namely, estimating the camera's location and orientation from a given ground-level query image by matching it to geo-referenced aerial imagery. Previous deep learning-

to have large positioning errors. A few pioneer works [22], [28], [29], [30], [31], [32], [33] demonstrated the feasibility of pinpointing the 2D location, sometimes together with the orientation, of the ground camera within a known aerial image. Similar to [28], [29], [32], we are interested in the 3-Degrees-of-Freedom (3-DoF) camera pose, i.e. planar location and orientation (yaw), instead of the full 6-DoF pose, since the change in camera height, pitch, and roll are often very small in autonomous driving.

However, several gaps must be filled before large-scale real-world deployment of cross-view camera pose estimation methods is a realistic possibility for self-driving. So far, the localization accuracy of existing methods is not yet good enough for autonomous driving requirements, e.g. the lateral and longitudinal error should be below $0.29m$ [34]. Besides, many methods cannot be run in real-time, i.e. ~ 15 frames per second (FPS) in self-driving datasets [35], [36], [37], because of using expensive iterative optimization [28], [30] or computationally heavy Transformers [33]. We also

On the Estimation of Image-matching Uncertainty in Visual Place Recognition

Mubariz Zaffar
ME, TU Delft
The Netherlands
M.Zaffar@tudelft.nl

Liangliang Nan
ABE, TU Delft
The Netherlands
Liangliang.Nan@tudelft.nl

Julian F. P. Kooij
ME, TU Delft
The Netherlands
J.F.P.Kooij@tudelft.nl

Abstract

In Visual Place Recognition (VPR) the pose of a query image is estimated by comparing the image to a map of reference images with known reference poses. As is typical for image retrieval problems, a feature extractor maps the query and reference images to a feature space, where a nearest neighbor search is then performed. However, till recently little attention has been given to quantifying the confidence that a retrieved reference image is a correct match. Highly certain but incorrect retrieval can lead to catastrophic failure of VPR-based localization pipelines. This work compares for the first time the main approaches for estimating the image-matching uncertainty, including the traditional retrieval-based uncertainty estimation, more recent data-driven aleatoric uncertainty estimation, and the compute-intensive geometric verification. We further formulate a simple baseline method, "SUE", which unlike the other methods considers the freely-available poses of the reference images in the map. Our experiments reveal that

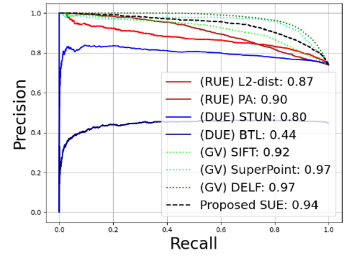


Figure 1. The Precision-Recall curves on the Pittsburgh dataset [4] for the three common categories of VPR uncertainty estimation methods (RUE, DUE, GV), and for our proposed baseline **SUE** which uniquely considers spatial locations of the top-K references. The global image descriptors [9] are fixed for all methods except