Robust State Estimation and Mapping in Challenging Environments

Sebastian Scherer
6/17/2024

Our Motivating Scenarios: State Estimation and Mapping is Safety-Critical and Requires High Accuracy for Autonomous Systems

DARPA SubT (2nd place in Urban, 1st place in Tunnel)

Offroad Driving by Learning from Demonstration

Wildfire Monitoring

Caves

Tunnel

Offroad

Smoke

DARPA
SUBTERRANEAN CHALLENGE

Team Explorer Aerial Autonomy

Carnegie Mellon University

explorer

Oregon State University

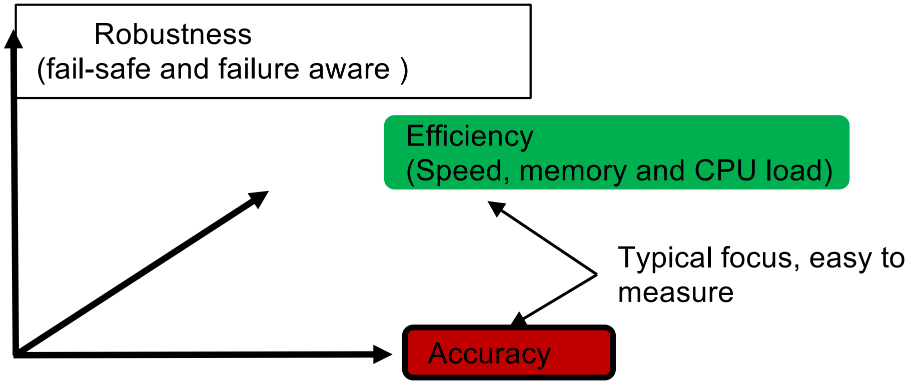
Vision

- A robust, real-time semantically and multi-agent aware way to understand where we are in the world.
- Unified inference between the different modules
- Transition to combine it with perception and dynamic modules.

However, robustness is still the greatest challenge for SLAM today!

Challenges:

- Appearance variation across time
- Methods sensitive to outliers
- Computational tradeoffs

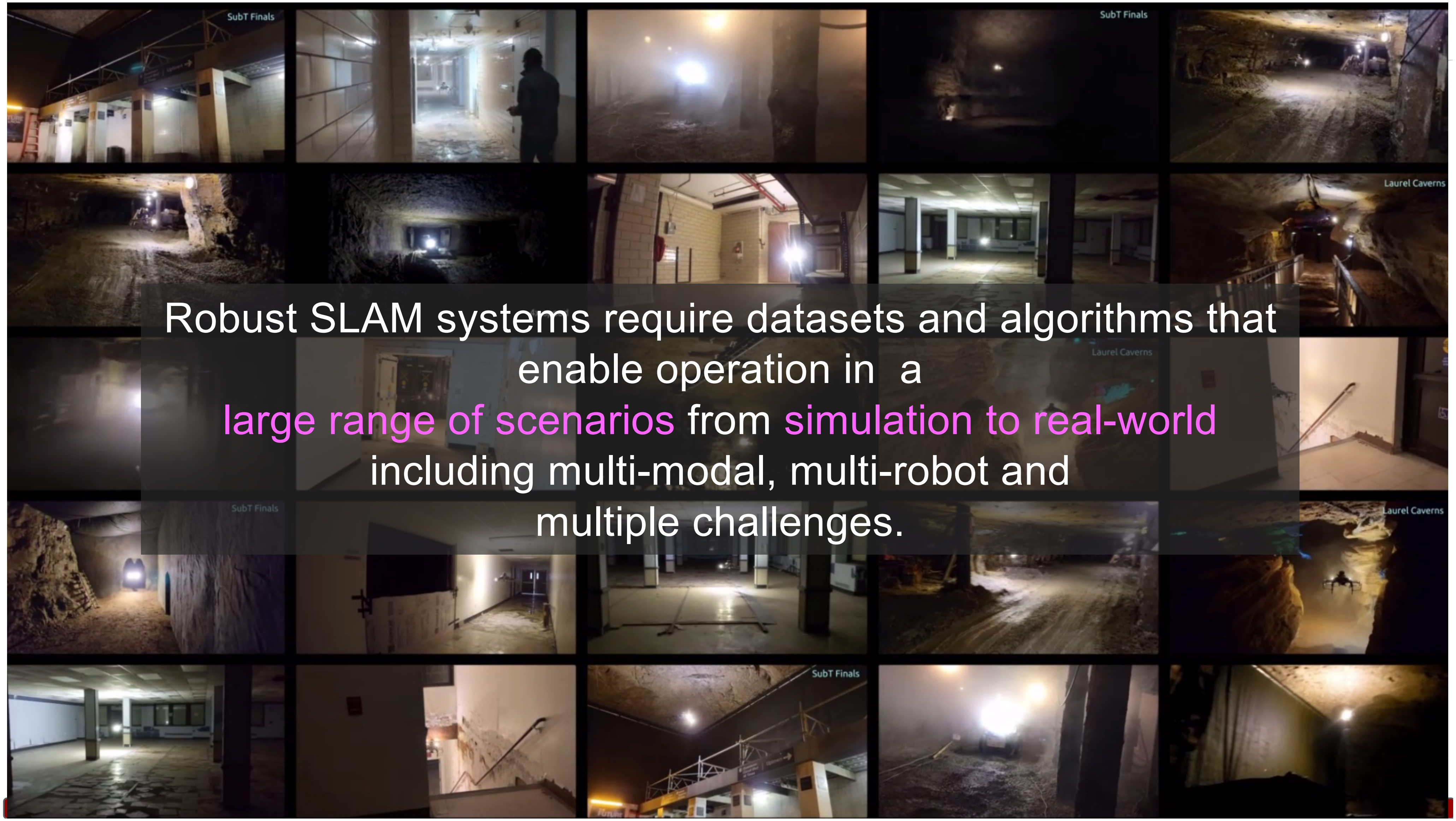


Robustness
(fail-safe and failure aware)

Efficiency
(Speed, memory and CPU load)

Typical focus, easy to measure

Accuracy

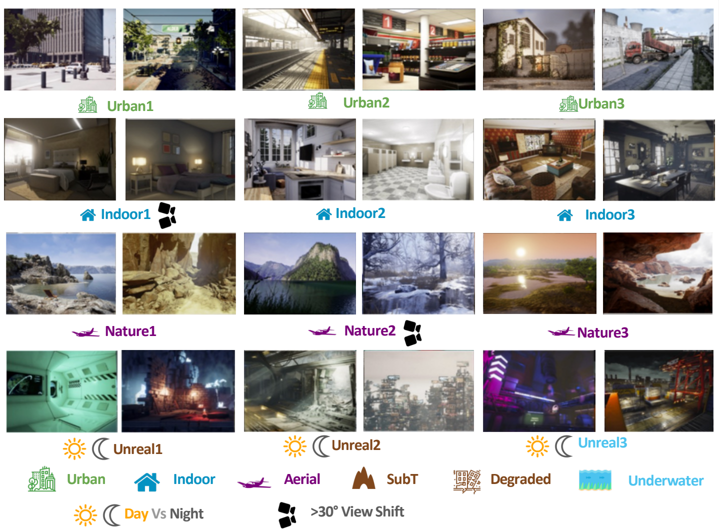


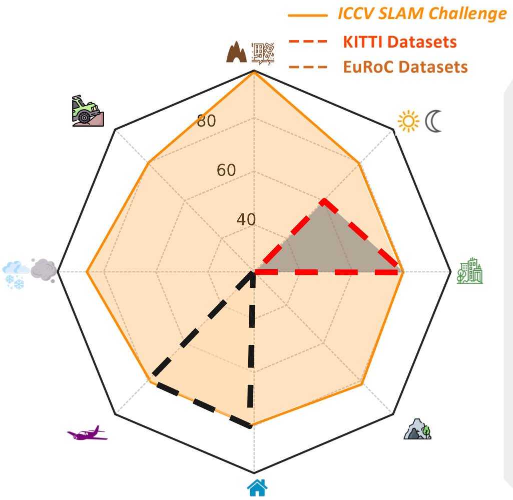
Robust SLAM systems require datasets and algorithms that enable operation in a large range of scenarios from simulation to real-world including multi-modal, multi-robot and multiple challenges.

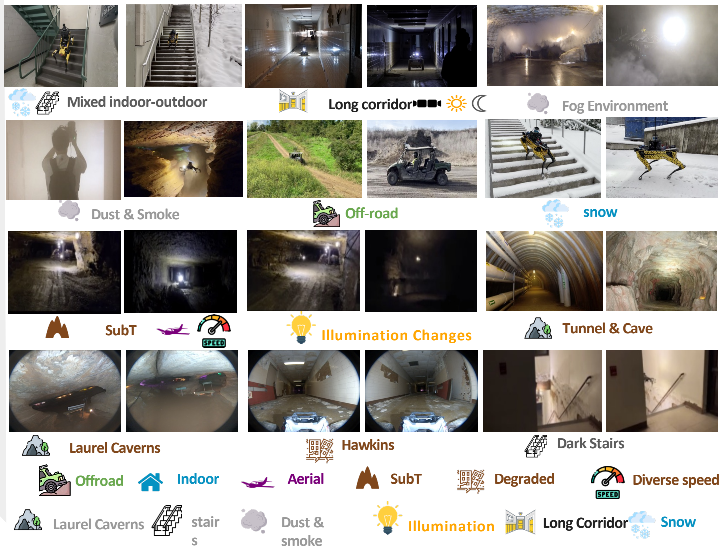
Carnegie Mellon University
The Robotics Institute

AIR LAB

SubT-MRS Datasets provides 8X More Diverse Data







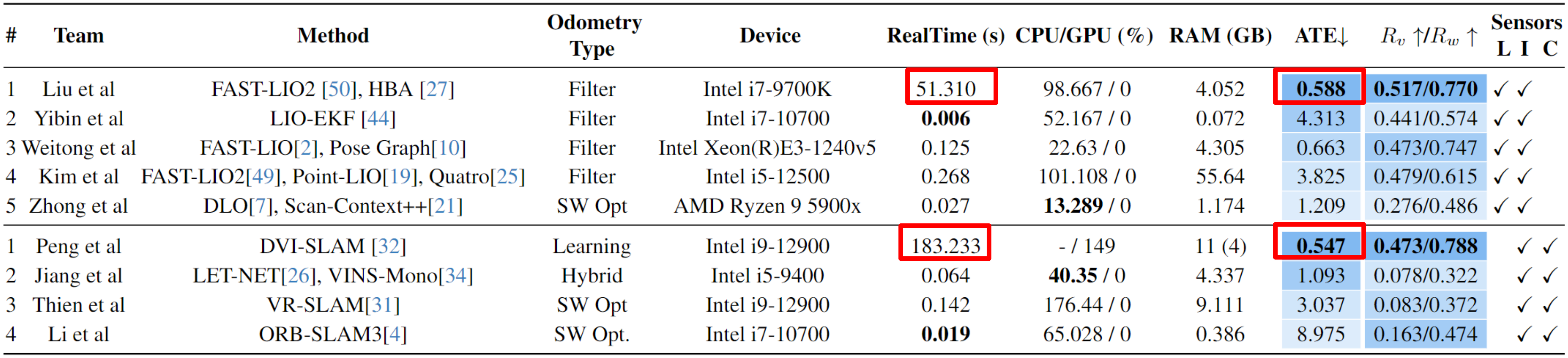
Tartan Air Datasets

SubT-MRS Datasets

Sim2Real: Digital World Meets the Physical World

ICCV 2023 SLAM Challenge Summary

Table 2. SLAM Challenge Results (Blue shadings indicate rankings)



#	Team	Method	Odometry Type	Device	RealTime (s)	CPU/GPU (%)	RAM (GB)	ATE↓	R_v ↑/R_w ↑	Sensors L	I	C
1	Liu et al	FAST-LIO2 [50], HBA [27]	Filter	Intel i7-9700K	51.310	98.667 / 0	4.052	**0.588**	**0.517/0.770**	✓	✓	
2	Yibin et al	LIO-EKF [44]	Filter	Intel i7-10700	**0.006**	52.167 / 0	0.072	4.313	0.441/0.574	✓	✓	
3	Weitong et al	FAST-LIO[2], Pose Graph[10]	Filter	Intel Xeon(R)E3-1240v5	0.125	22.63 / 0	4.305	0.663	0.473/0.747	✓	✓	
4	Kim et al	FAST-LIO2[49], Point-LIO[19], Quatro[25]	Filter	Intel i5-12500	0.268	101.108 / 0	55.64	3.825	0.479/0.615	✓	✓	
5	Zhong et al	DLO[7], Scan-Context++[21]	SW Opt	AMD Ryzen 9 5900x	0.027	**13.289** / 0	1.174	1.209	0.276/0.486	✓	✓	
1	Peng et al	DVI-SLAM [32]	Learning	Intel i9-12900	183.233	- / 149	11 (4)	**0.547**	**0.473/0.788**		✓	✓
2	Jiang et al	LET-NET[26], VINS-Mono[34]	Hybrid	Intel i5-9400	0.064	**40.35** / 0	4.337	1.093	0.078/0.322		✓	✓
3	Thien et al	VR-SLAM[31]	SW Opt	Intel i9-12900	0.142	176.44 / 0	9.111	3.037	0.083/0.372		✓	✓
4	Li et al	ORB-SLAM3[4]	SW Opt.	Intel i7-10700	**0.019**	65.028 / 0	0.386	8.975	0.163/0.474		✓	✓

There are no current solutions that can balance **high accuracy** and **real-time** performance in challenging environments.

In the sensor fusion track, which addresses **both visual and geometric** degradation, no submissions met the criteria for success.

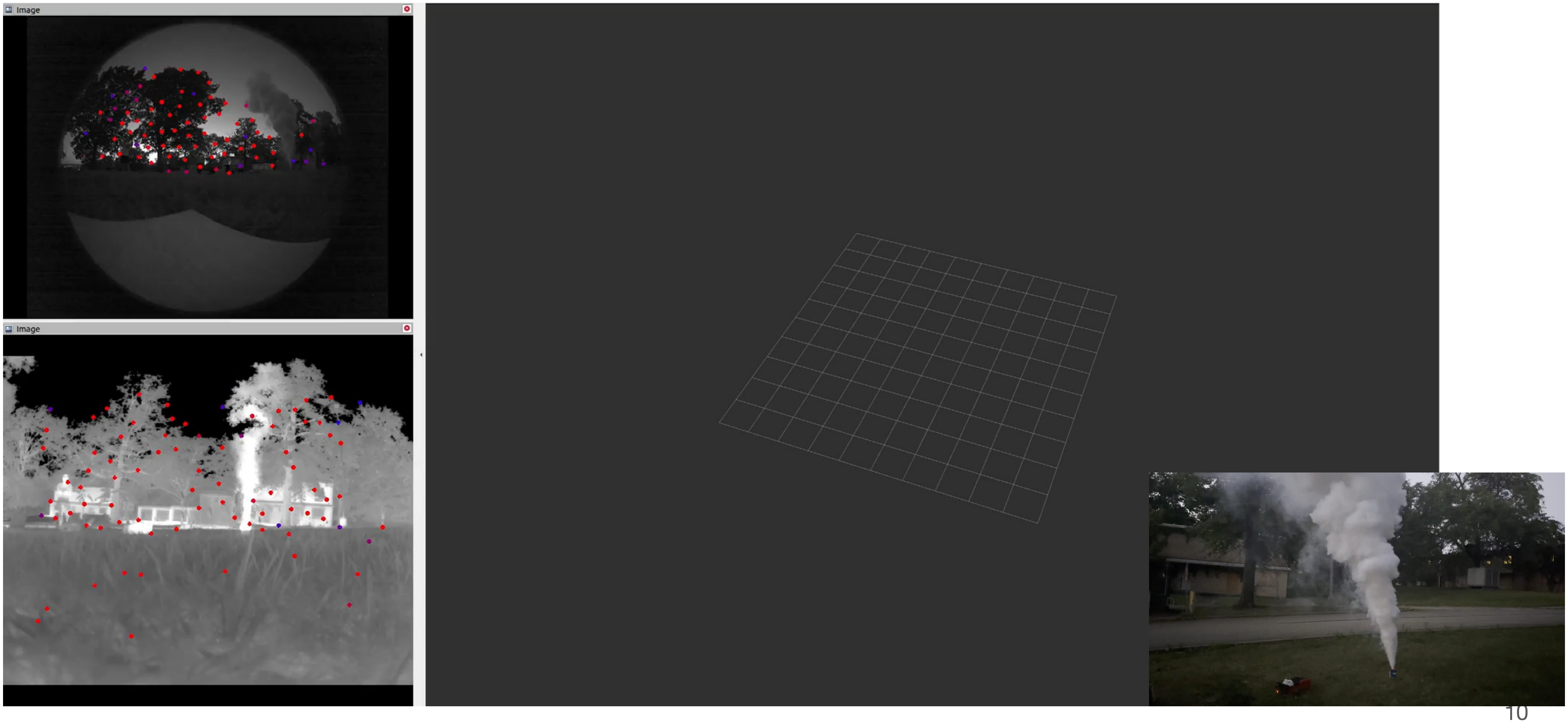
How do we achieve robustness for SLAM?

Carnegie Mellon University
The Robotics Institute

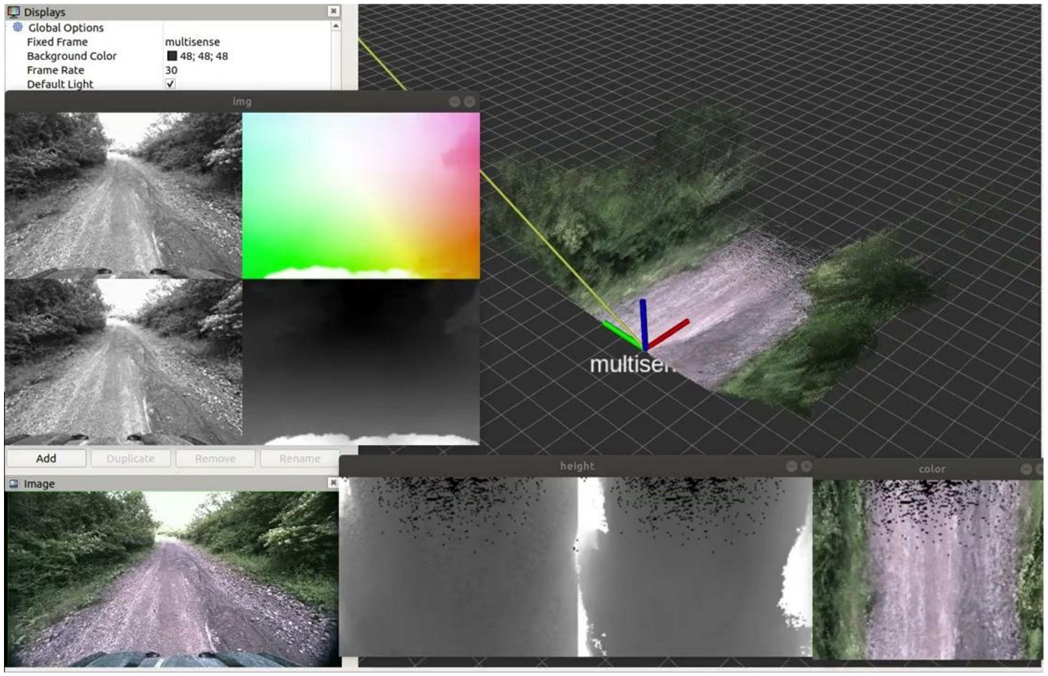
Super Odometry 2: Multi Spectral Odometry in Smoke Environments

Integrated the Multi Spectral Odometry into Super Odometry Pipeline

Visual + Thermal Visual Inertial Odometry



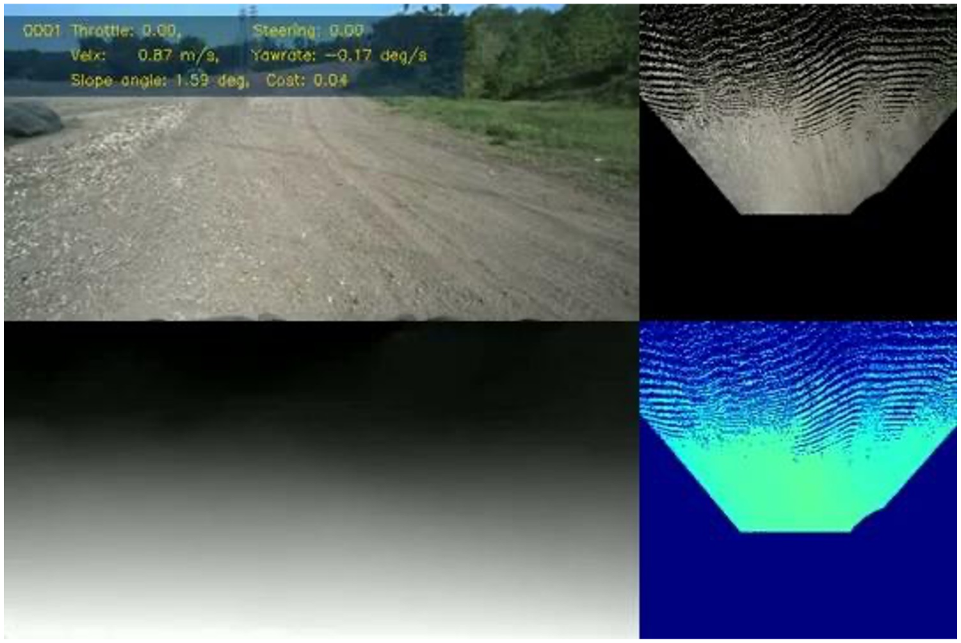
Visual Odometry - Learning-based Dense Stereo Mapping (TartanVO Stereo)



Displays

Global Options

Fixed Frame	multisense
Background Color	■ 48; 48; 48
Frame Rate	30
Default Light	☑

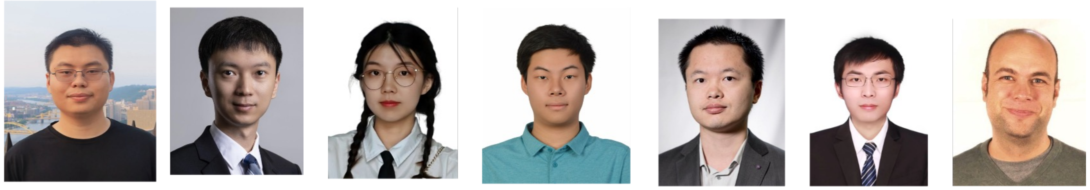


```
0001 Throttle: 0.00,          Steering: 0.00
     Velx:    0.87 m/s,       Yawrate: -0.17 deg/s
     Slope angle: 1.59 deg,   Cost: 0.04
```

Carnegie Mellon University
The Robotics Institute

AirIMU: Learning Uncertainty Propagation for Inertial Odometry

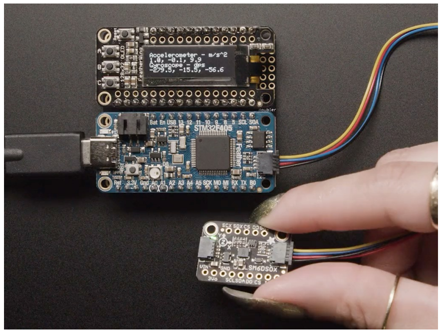
Yuheng Qiu, Chen Wang, Can Xu, Yutian Chen, Xunfei Zhou, Youjie Xia and Sebastian Scherer



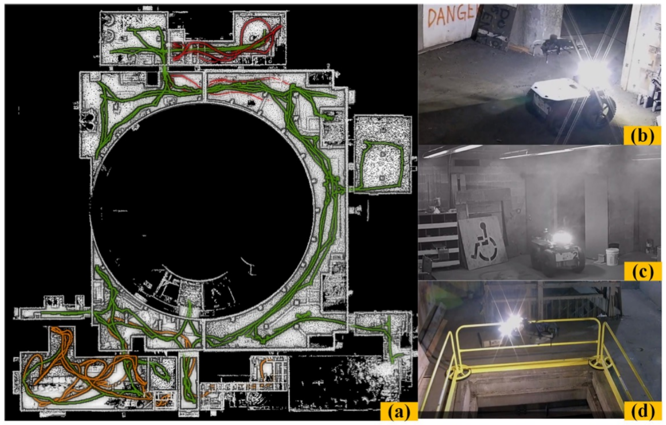
theairlab.org

Background: IMU determines your lower bound

- IMU (Inertial Measurement Unit)
    - **Fundamental**: acceleration & angular velocity
    - **Popular**: Almost in any smart device
    - **Low cost** (cheap IMU only cost 2$)



- **Robustness**
    - No outside references required



SuperOdometry [2]

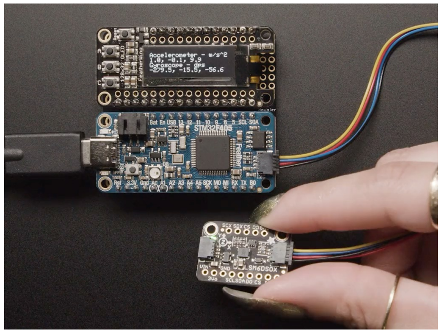
Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

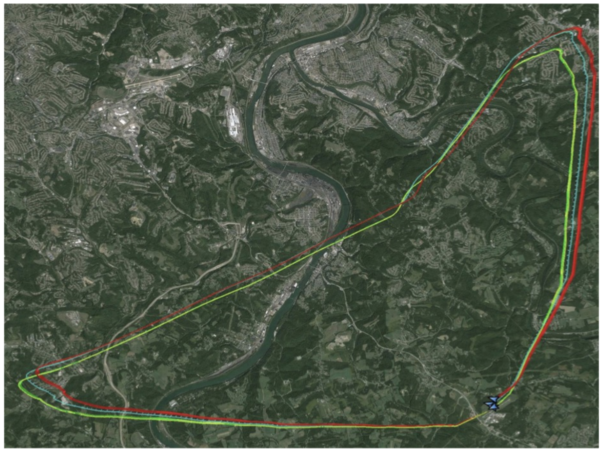
Background: IMU determines your upper bound

- IMU (Inertial Measurement Unit)
  - **Fundamental**: acceleration & angular velocity
  - **Popular**: Almost in any smart device
  - **Low cost** (cheap IMU only cost 2$)
  - **Robust Guaranteed** (Inertial only)
- **Robustness**
  - No outside references required

Lidar may be blocked, Camera may fail, But IMU will not.

- **Frequency and Accuracy**
  - High-frequency state estimation for **control**
  - High-accuracy local state estimation





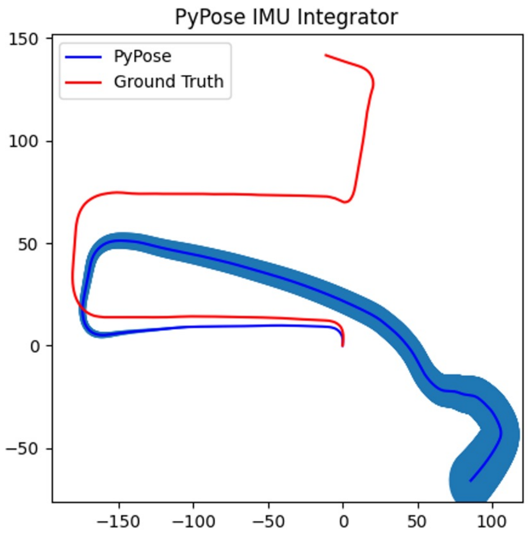
ALTO dataset [1]: IMU integration from helicopter flight data

Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

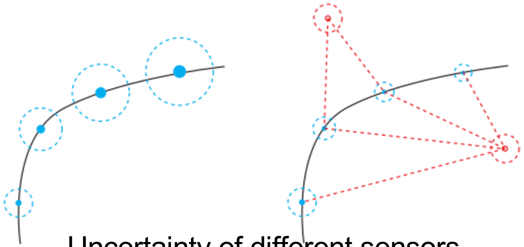
Problems: IMU Noise and Uncertainty

- Reducing Noise: Drift is unavoidable due to integration and IMU noise.

- Characterizing Uncertainty: Uncertainty determines how long and how well you can trust the IMU



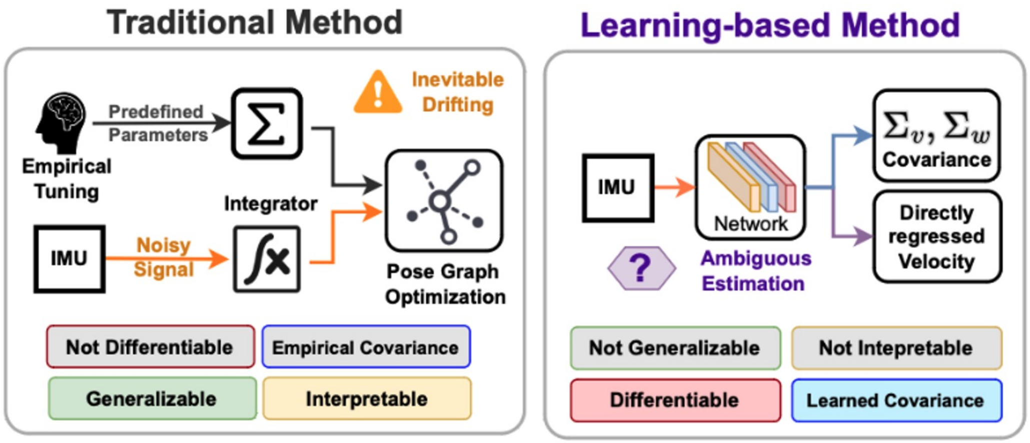
IMU integration on KITTI dataset

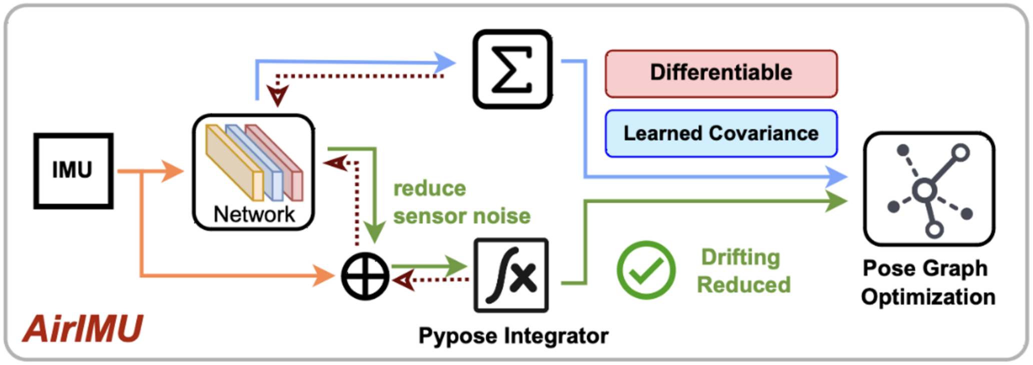


Uncertainty of different sensors in BA optimization

Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

AirIMU: Model and Approach





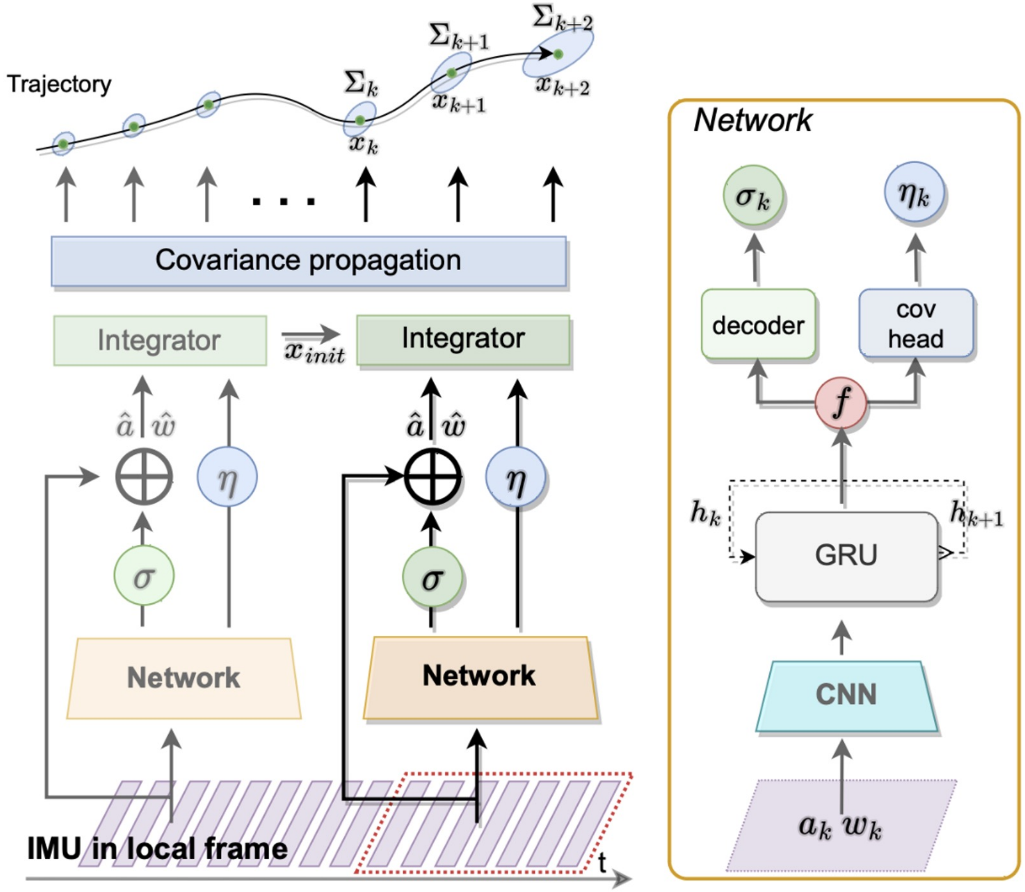
Benefit:

1. Differentiable Integrator
2. Uncertainty-aware IMU model
3. Generalizable across modality

AirIMU serve dual purposes to correct noise and estimate the uncertainty

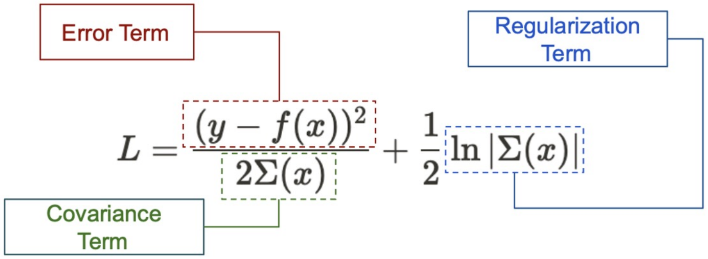
Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

AirIMU: Model and Approach



1. We design a shared CNN-GRU encoder to encode raw IMU data.

2. To supervise covariance model we build a differentiable covariance propagation method.


Error Term

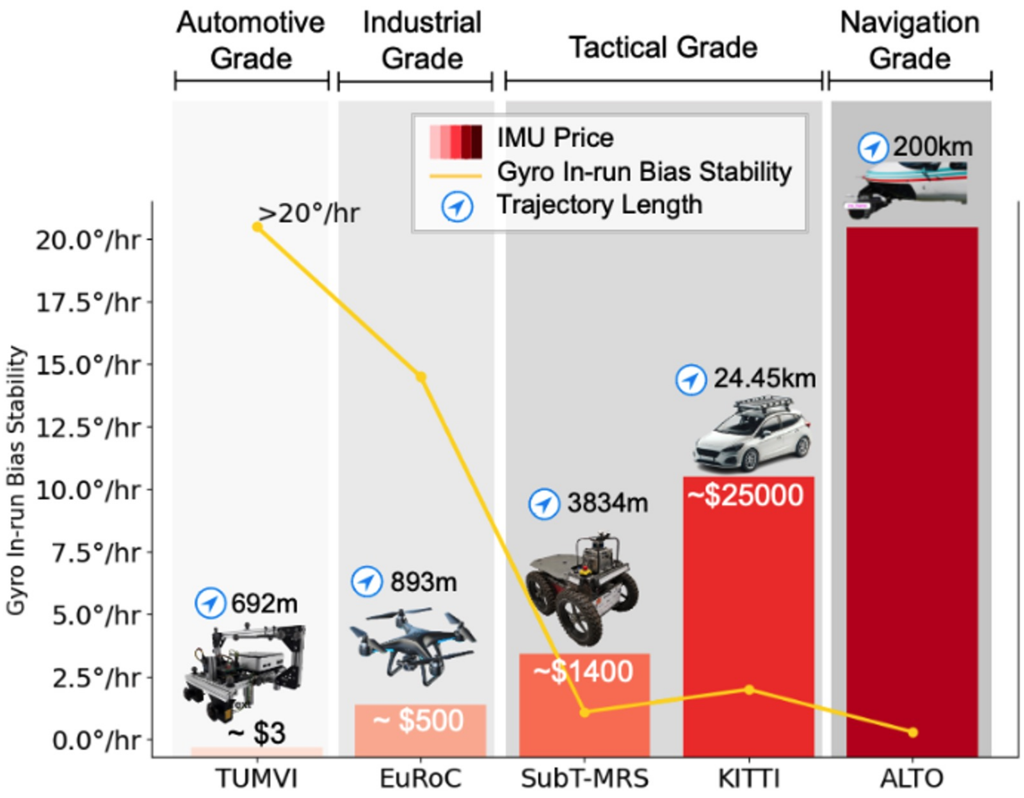
Regularization Term

$$L = \frac{(y - f(x))^2}{2\Sigma(x)} + \frac{1}{2}\ln|\Sigma(x)|$$

Covariance Term

Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

Datasets and Benchmarks: Learning-based methods



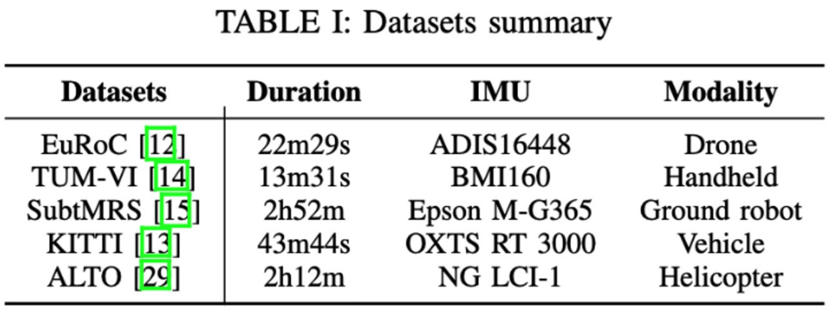


TABLE I: Datasets summary

Datasets	Duration	IMU	Modality
EuRoC [12]	22m29s	ADIS16448	Drone
TUM-VI [14]	13m31s	BMI160	Handheld
SubtMRS [15]	2h52m	Epson M-G365	Ground robot
KITTI [13]	43m44s	OXTS RT 3000	Vehicle
ALTO [29]	2h12m	NG LCI-1	Helicopter

Integration Accuracy: TUMVI, EuRoC

Learning inertial odometry: KITTI

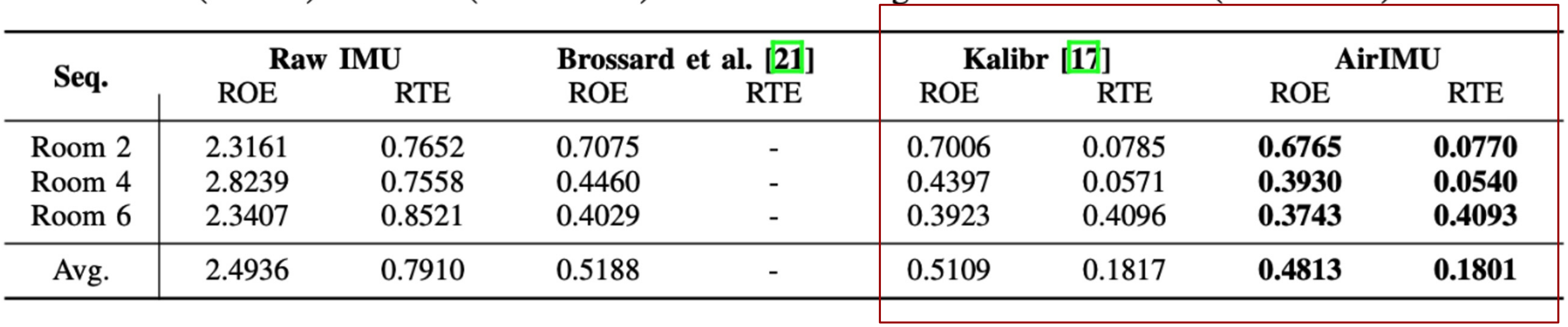
Ablation Study: Subt-MRS

GPS-denied Navigation: ALTO

Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

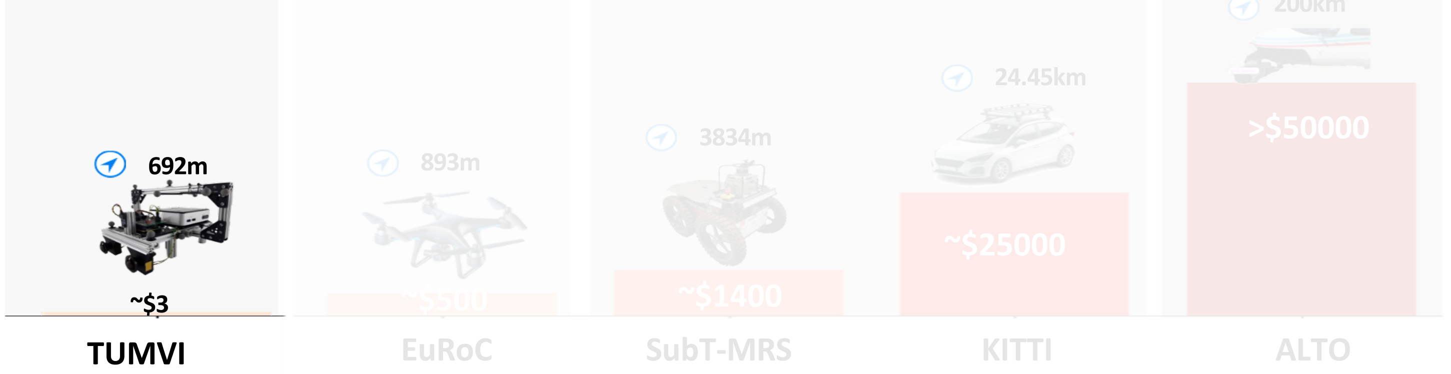
TUMVI: Automotive-Grade

TABLE II: The ROE (Unit: °) and RTE (Unit: meter) of IMU Pre-integration over 1 second (200 frames) on TUMVI dataset.



Seq.	Raw IMU		Brossard et al. [21]		Kalibr [17]		AirIMU	
	ROE	RTE	ROE	RTE	ROE	RTE	ROE	RTE
Room 2	2.3161	0.7652	0.7075	-	0.7006	0.0785	**0.6765**	**0.0770**
Room 4	2.8239	0.7558	0.4460	-	0.4397	0.0571	**0.3930**	**0.0540**
Room 6	2.3407	0.8521	0.4029	-	0.3923	0.4096	**0.3743**	**0.4093**
Avg.	2.4936	0.7910	0.5188	-	0.5109	0.1817	**0.4813**	**0.1801**

AirIMU can further improve based on the Kalibr



Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

EuRoC: Industrial-Grade IMU

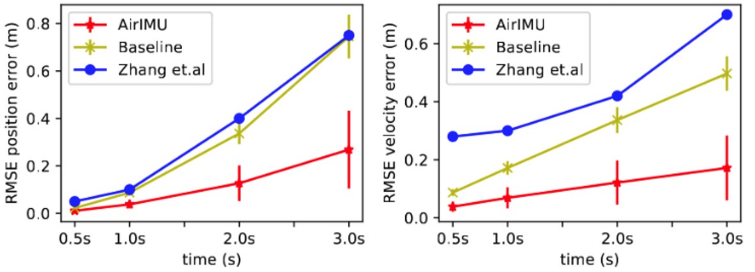
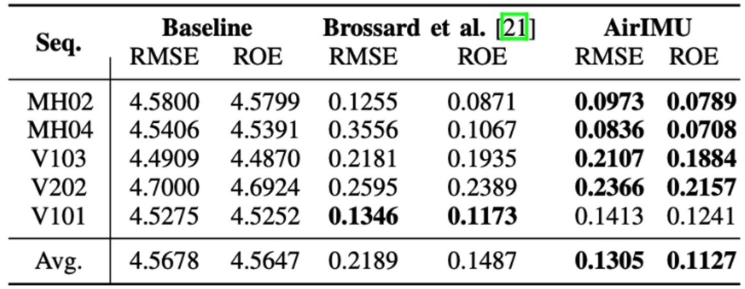


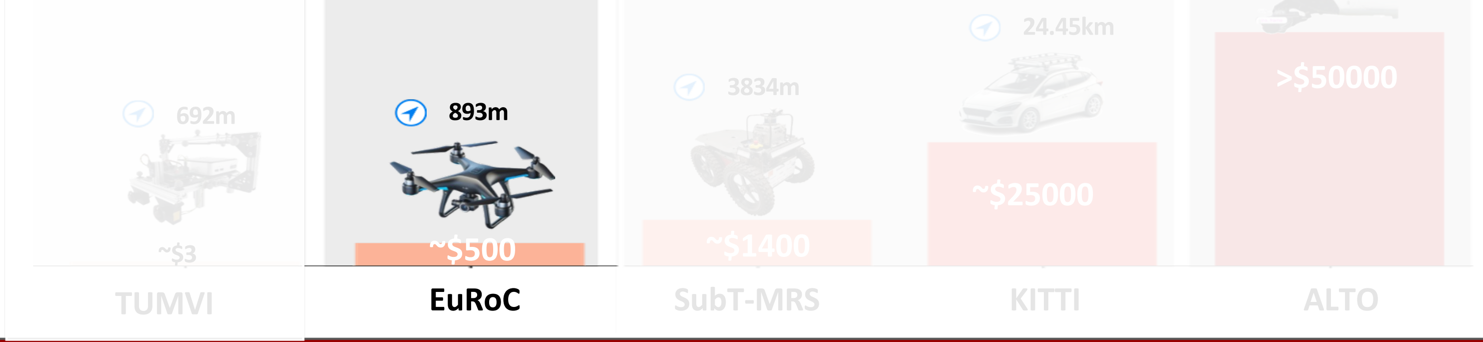
Fig. 6: We present the RMSE error of both translation and velocity over intervals of 0.5s, 1s, 2s, and 3s. The results illustrate the accumulation of error throughout the integration, where the AirIMU exhibiting a reduced error after integration.

TABLE IV: Gyroscope integration on EuRoC Dataset, we show the R-RMSE and the ROE (Unit: °).



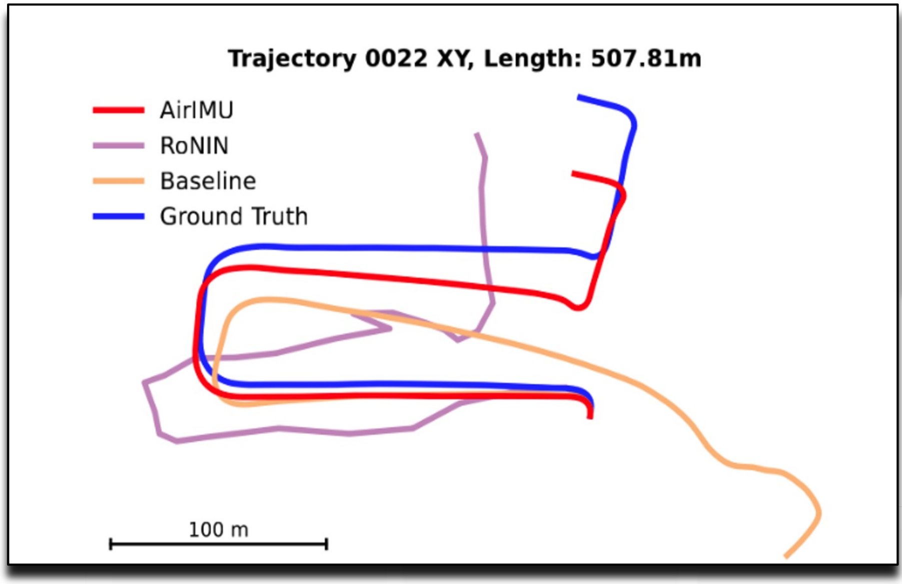
Seq.	Baseline		Brossard et al. [21]		AirIMU	
	RMSE	ROE	RMSE	ROE	RMSE	ROE
MH02	4.5800	4.5799	0.1255	0.0871	**0.0973**	**0.0789**
MH04	4.5406	4.5391	0.3556	0.1067	**0.0836**	**0.0708**
V103	4.4909	4.4870	0.2181	0.1935	**0.2107**	**0.1884**
V202	4.7000	4.6924	0.2595	0.2389	**0.2366**	**0.2157**
V101	4.5275	4.5252	**0.1346**	**0.1173**	0.1413	0.1241
Avg.	4.5678	4.5647	0.2189	0.1487	**0.1305**	**0.1127**

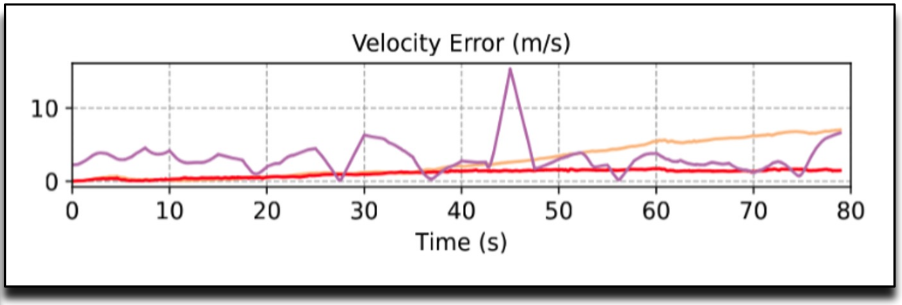
Compared to learned method 40% improvement in RMSE.



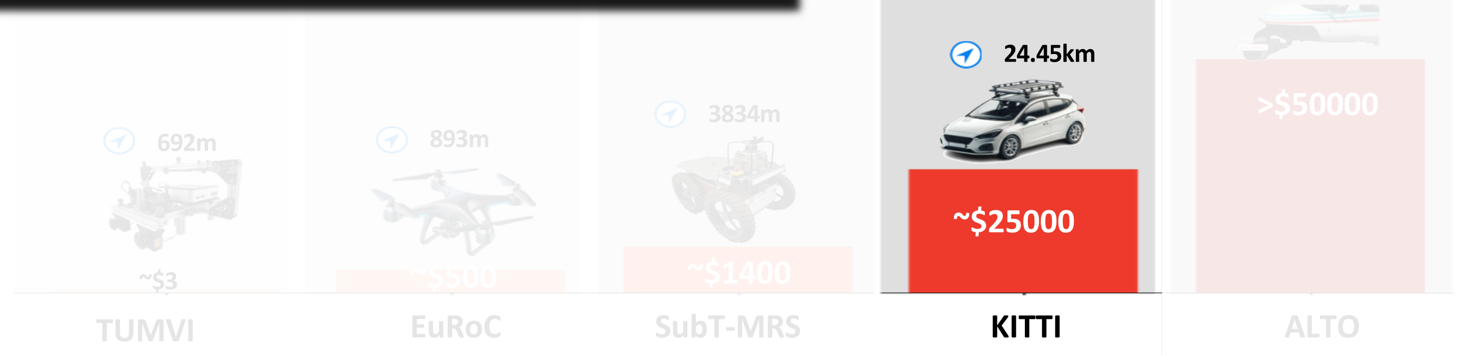
Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

KITTI: Tactical-Grade IMU



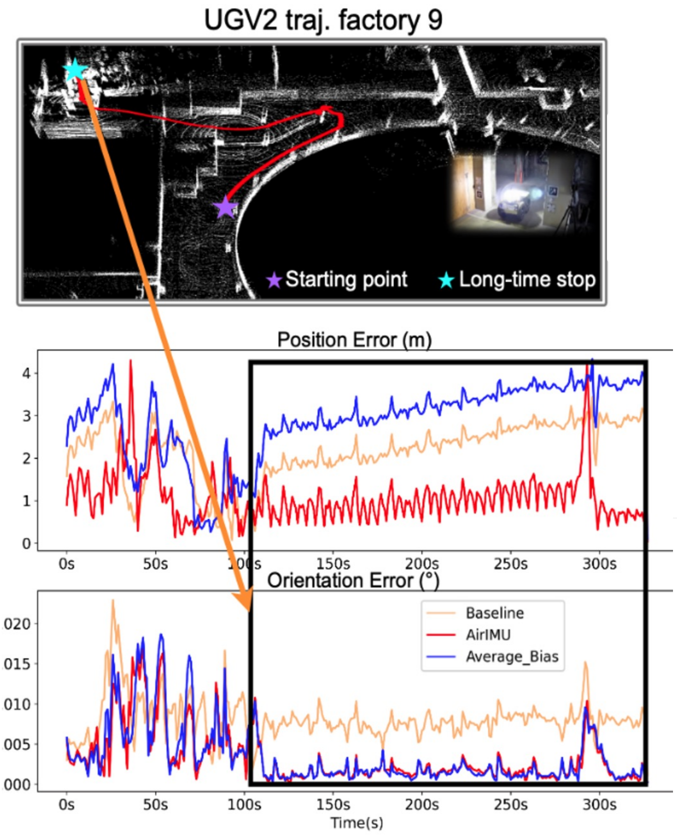


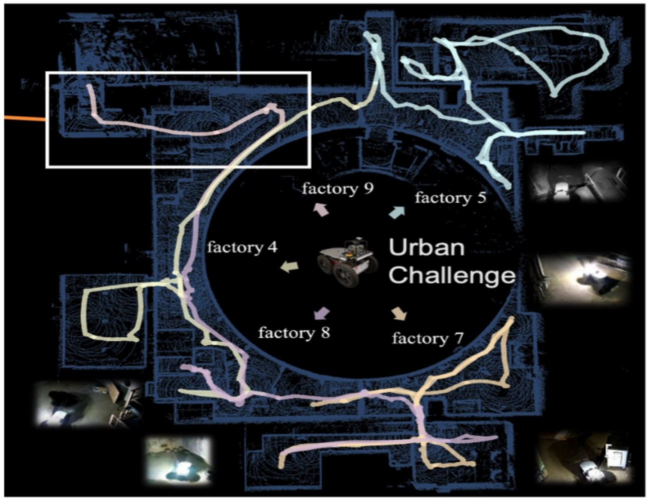
The accumulated error on the velocity is stable and small

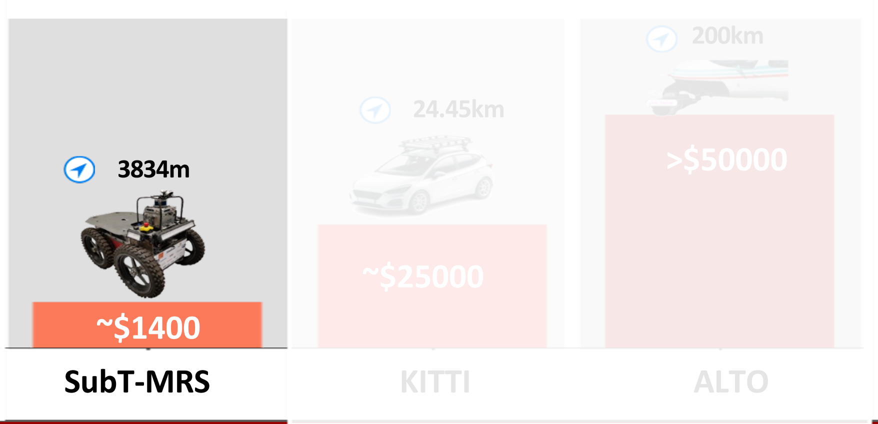


Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

Subt-MRS: Tactical-Grade IMU

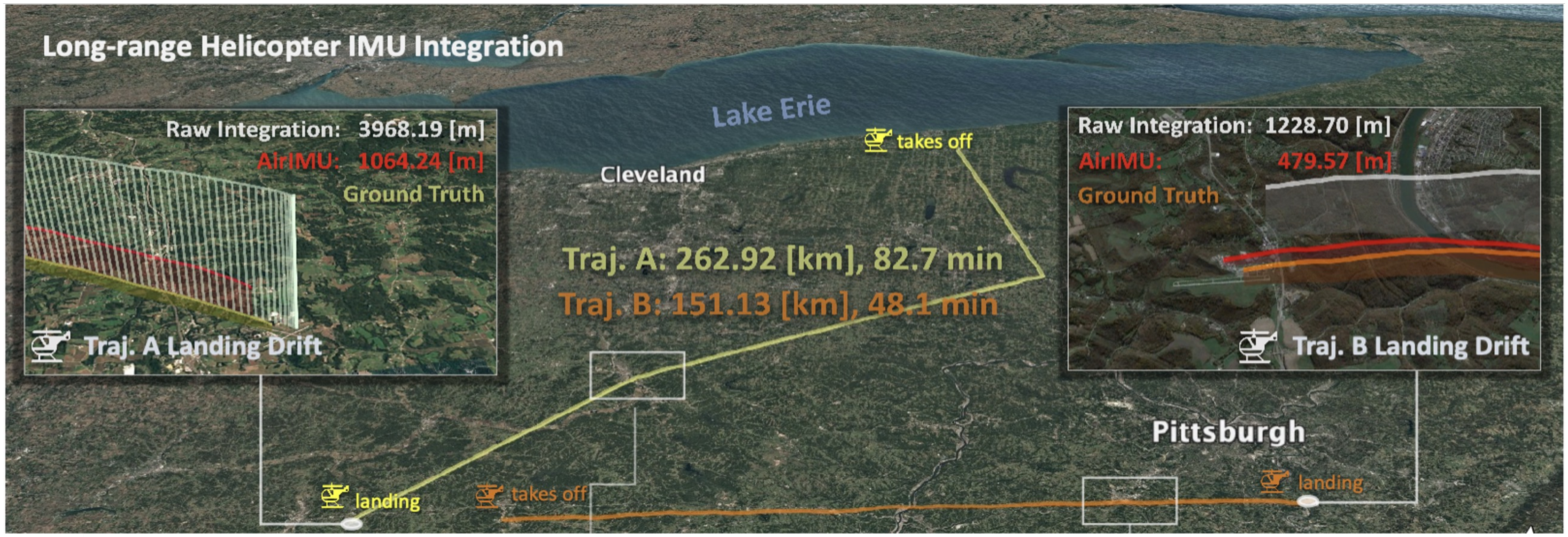


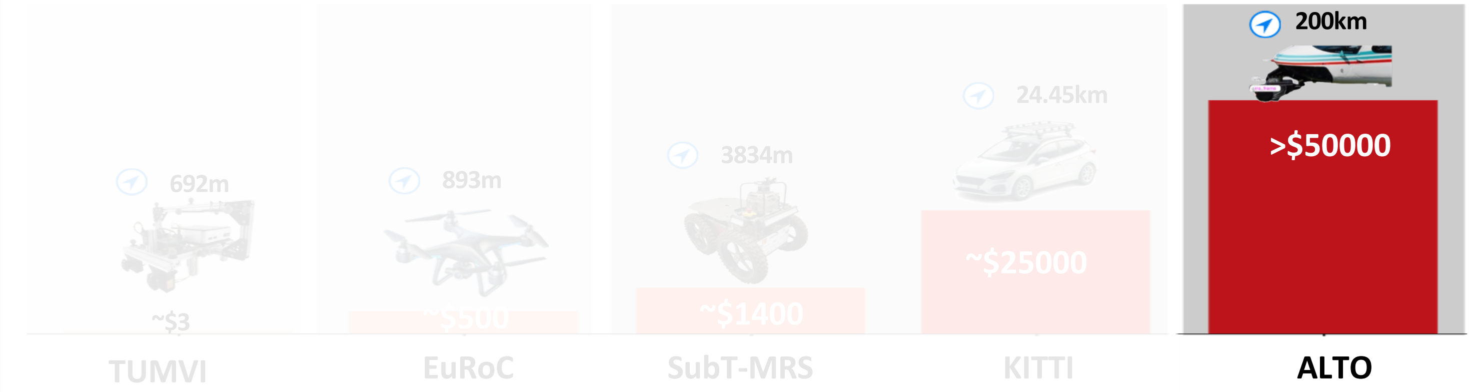




Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

ALTO: Navigation-Grade IMU

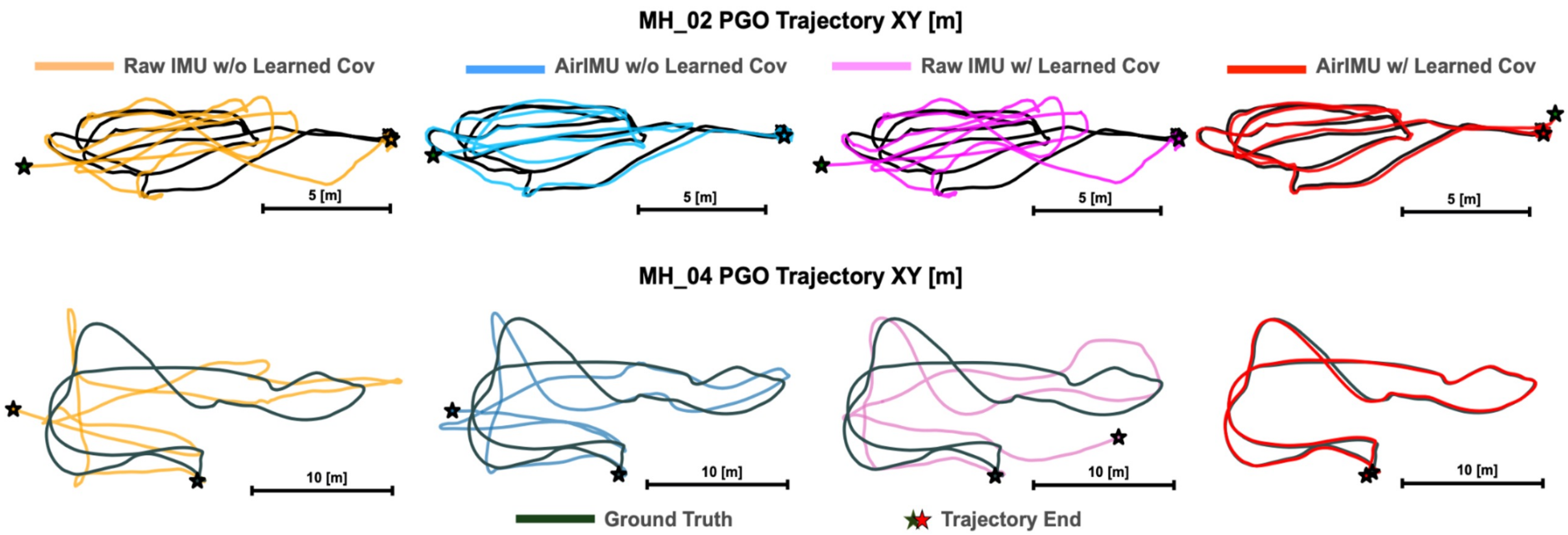




Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

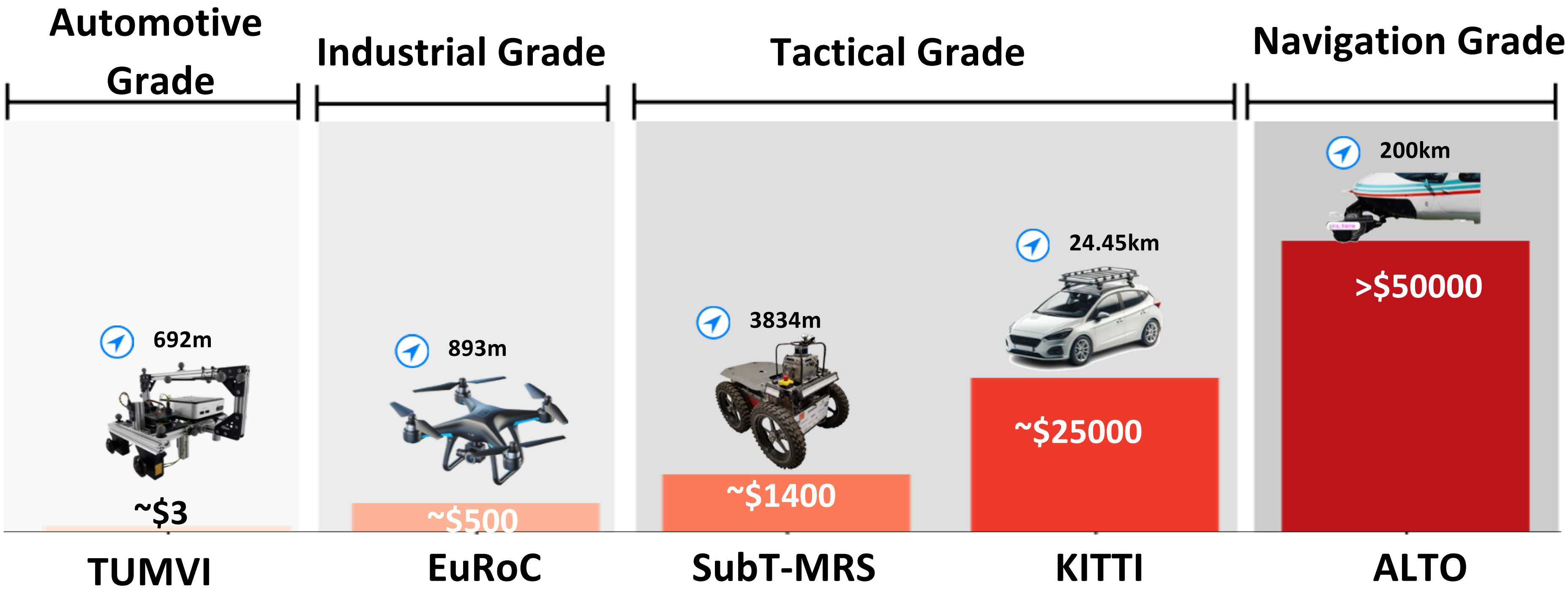
IMU-centric PGO

IMU-centric GPS Graph optimization performed at 0.1 Hz.



Qiu, Yuheng, et al. "AirIMU: Learning Uncertainty Propagation for Inertial Odometry." *arXiv preprint* (2023).

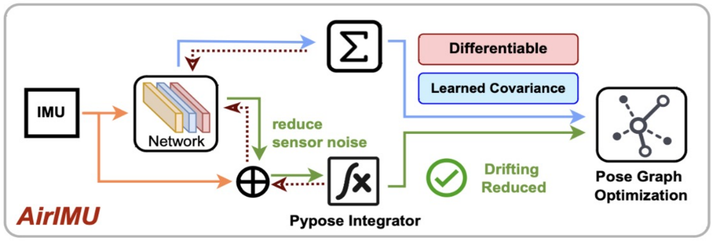
Experiment Summary



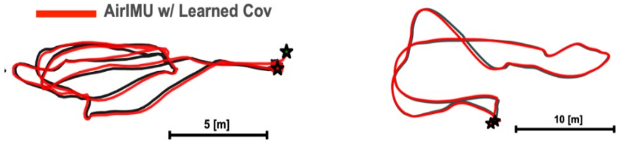
Improvement Compared to Baseline (Raw Data):

ROE(1s): 80.7% RTE (1s): 77.2%	R-RMSE(1s): 97.1% P-RMSE(1s): 73.8%	ROE(5s): 72.6% RTE(5s): 42.1% ATE: 14.7%	Accum Velo Err: 54.9% Landing Drift: 73.2%

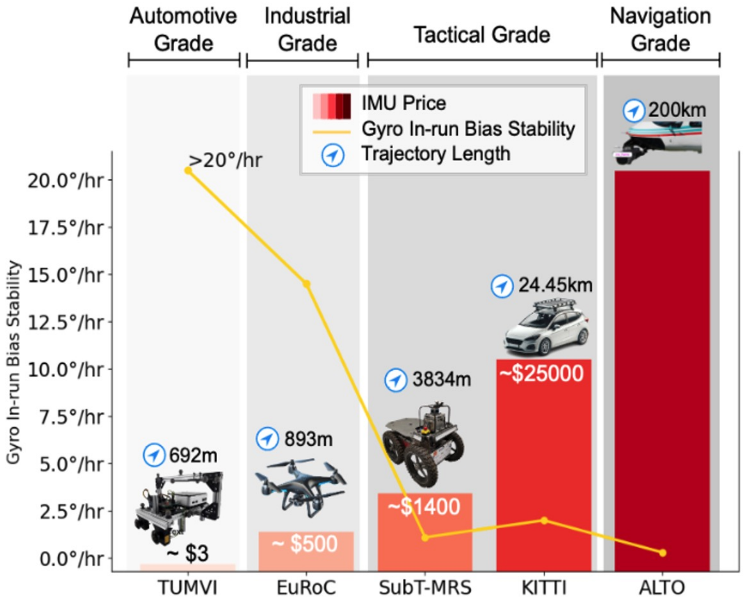
Conclusion



AirIMU servers dual purposes to correct noise and estimate the uncertainty



Better uncertainty improves pose graph optimization and sensor fusion



Testing on a range of IMU types showcases the effectiveness of the method

Carnegie Mellon University
The Robotics Institute

AIR LAB

AnyLoc: Towards Universal Visual Place Recognition

Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula,
Sebastian Scherer, Madhava Krishna, Sourav Garg

Fundamental Question of “Where Am I”?

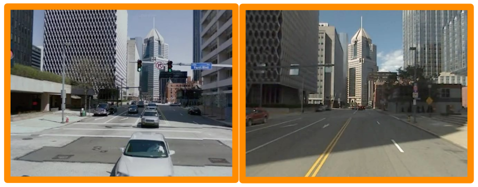
Humans & Robots alike need to know where they are for Scene Understanding & Navigation

How can we achieve this?

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

Oxford RobotCar

St Lucia



Pitts-30k

Gardens Point

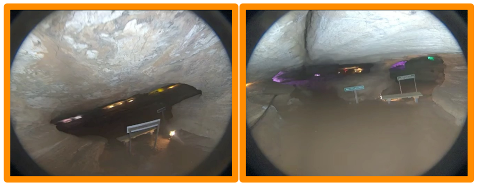
17 Places

Baidu Mall

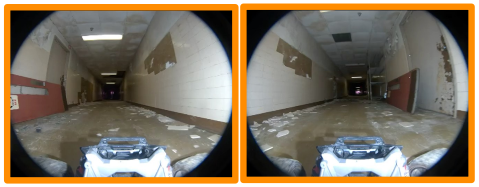
Nardo-Air

Nardo-Air R

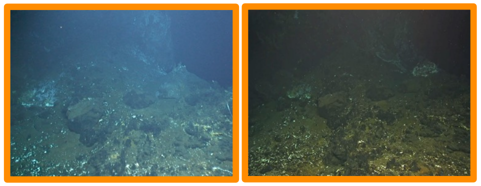
VP-Air



Laurel Caverns



Hawkins



Mid-Atlantic Ridge

Anywhere

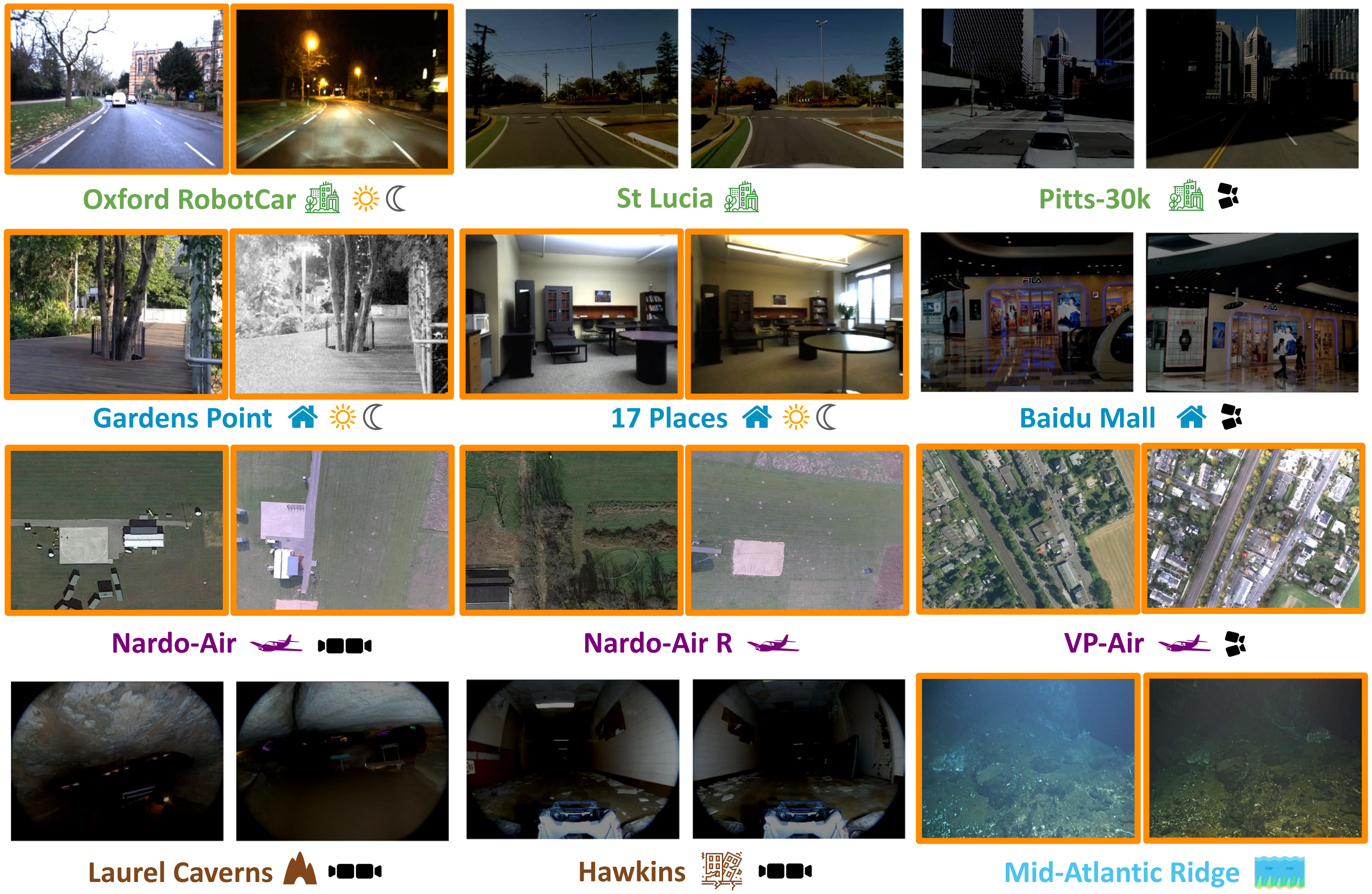
Urban Indoor Aerial SubT Degraded Underwater

☀ ☾ Day Vs Night

< 90° View Shift

> 90° View Shift

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024



Urban Indoor Aerial SubT Degraded Underwater

☼ ☾ Day Vs Night

< 90° View Shift

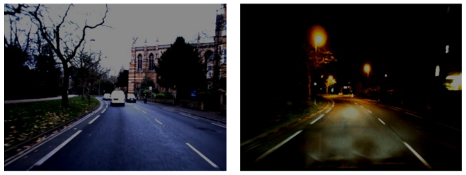
> 90° View Shift

Anywhere AIR LAB

Anytime

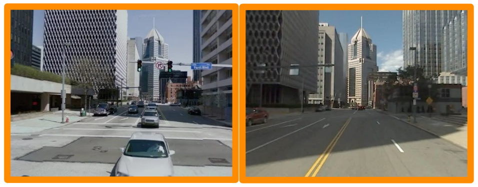
☼ ☾ 🕒

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024



Oxford RobotCar 🏙 ☀ ☾

St Lucia 🏙



Pitts-30k 🏙 ⧉

Gardens Point 🏠 ☀ ☾

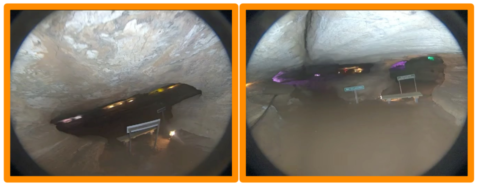
17 Places 🏠 ☀ ☾

Baidu Mall 🏠 ⧉

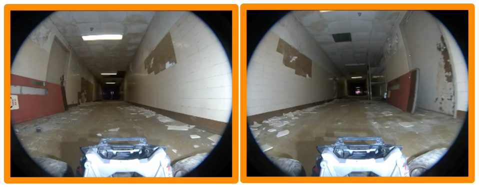
Nardo-Air ✈ ▮■■▮

Nardo-Air R ✈

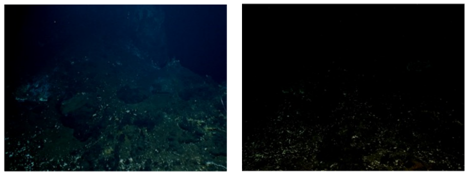
VP-Air ✈ ⧉



Laurel Caverns ⛰ ▮■■▮



Hawkins ▦ ▮■■▮



Mid-Atlantic Ridge 🌊

🏙 Urban 🏠 Indoor ✈ Aerial ⛰ SubT ▦ Degraded 🌊 Underwater

☀ ☾ Day Vs Night ⧉ < 90° View Shift ▮■■▮ > 90° View Shift

Anywhere

🏙 🏠 ✈ ⛰ ▦ 🌊

Anytime

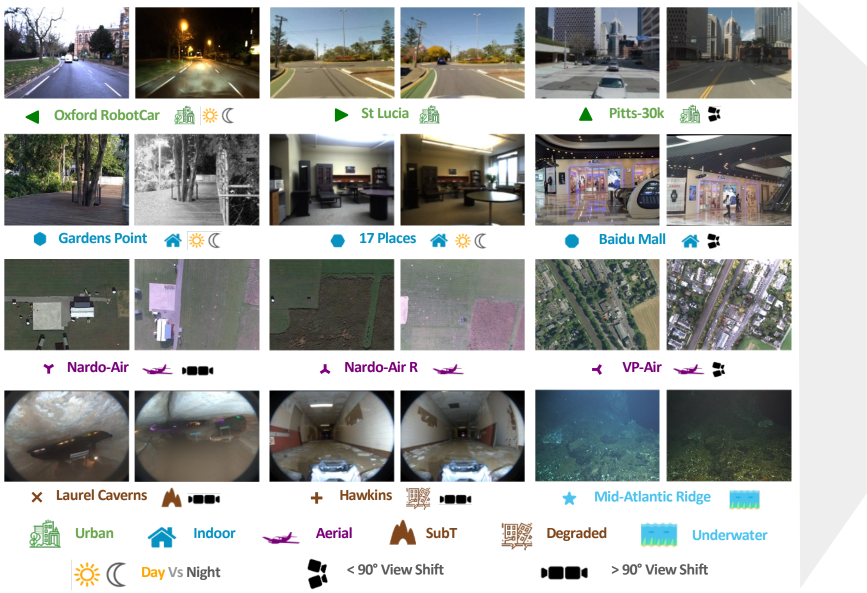
☀ ☾ 🕒

Anyview

⧉ ▮■■▮

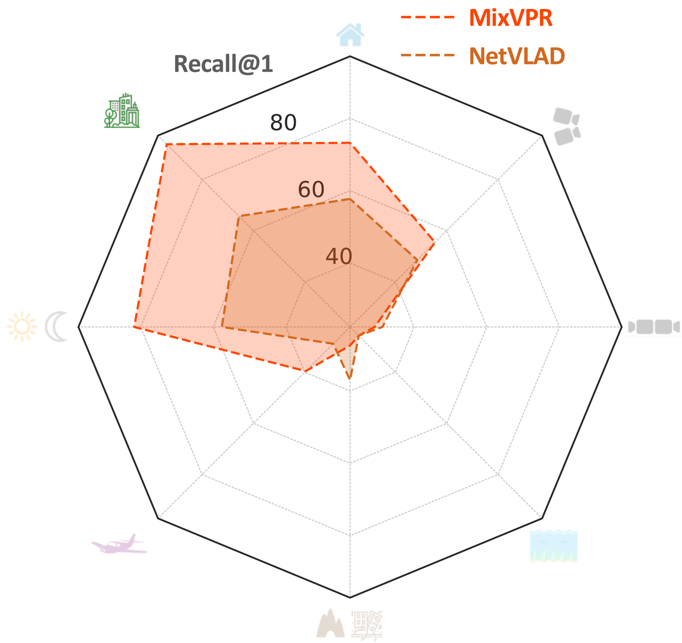
Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

Current State-of-the-art (SOTA) …



Large-Scale VPR Training

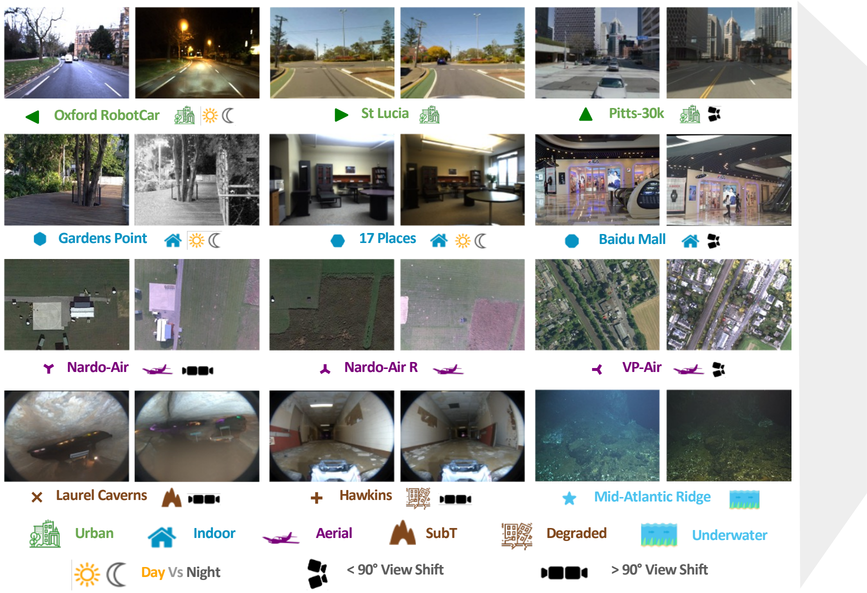
Supervised SOTA VPR Baselines

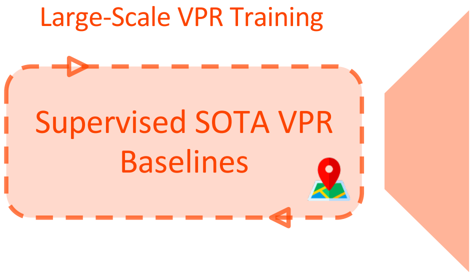


Perform well in Training Distribution (Urban)

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

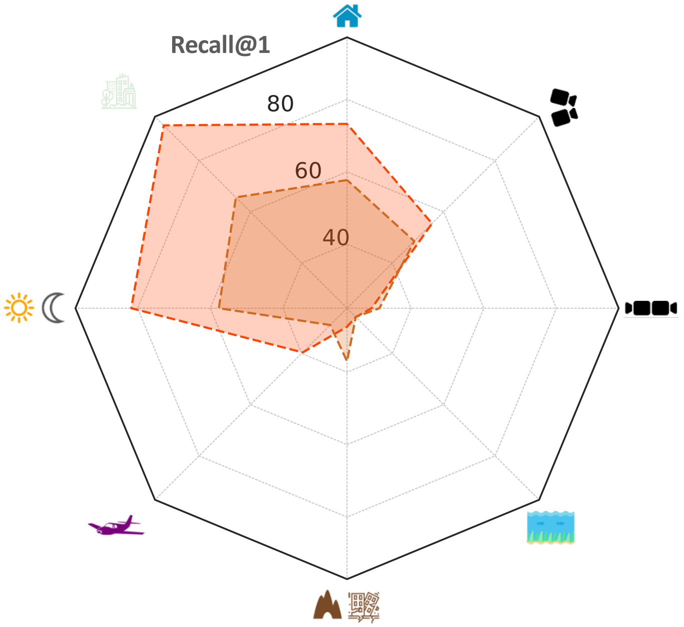
Do not generalize to diverse conditions





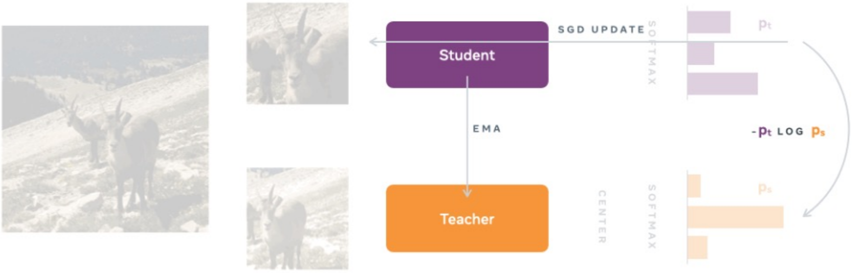
Large-Scale VPR Training

Supervised SOTA VPR Baselines

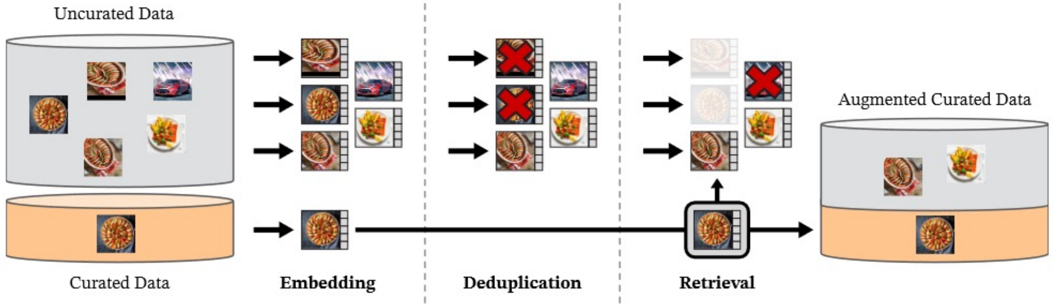


Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

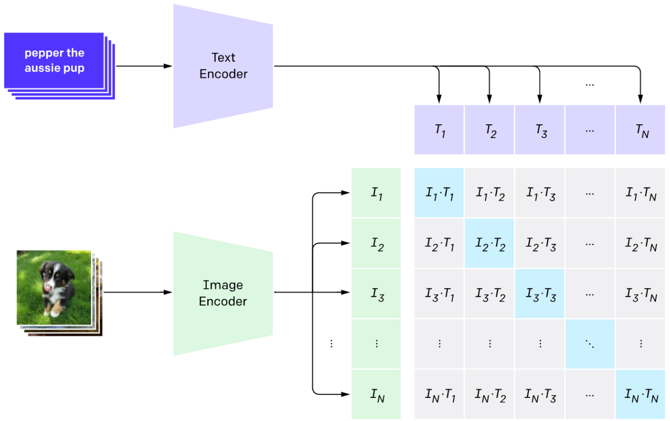
Self-Supervised Foundation Models for Generalization



DINO



DINOv2

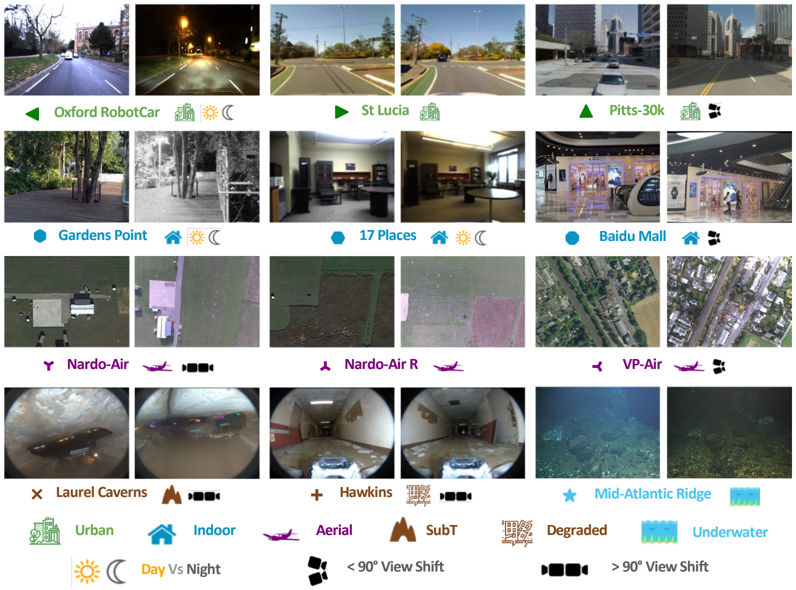


	T_1	T_2	T_3	...	T_N
I_1	$I_1 \cdot T_1$	$I_1 \cdot T_2$	$I_1 \cdot T_3$	...	$I_1 \cdot T_N$
I_2	$I_2 \cdot T_1$	$I_2 \cdot T_2$	$I_2 \cdot T_3$	...	$I_2 \cdot T_N$
I_3	$I_3 \cdot T_1$	$I_3 \cdot T_2$	$I_3 \cdot T_3$	...	$I_3 \cdot T_N$
⋮	⋮	⋮	⋮	⋱	⋮
I_N	$I_N \cdot T_1$	$I_N \cdot T_2$	$I_N \cdot T_3$	...	$I_N \cdot T_N$

CLIP

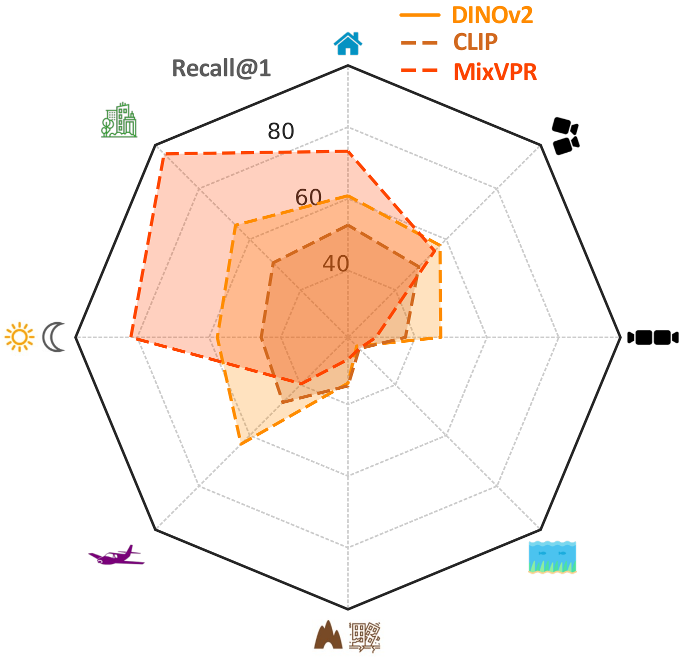
Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

Suboptimal when used as-is



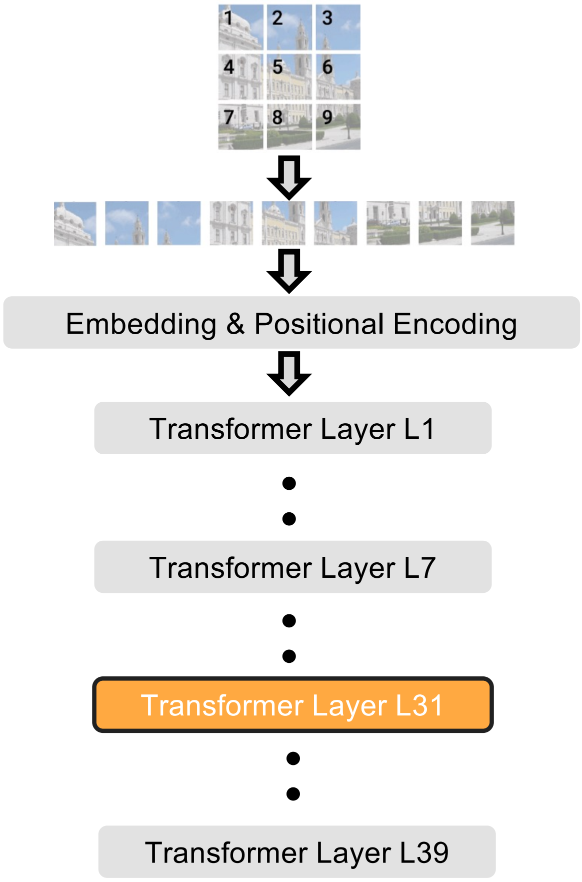
Large-Scale Task-Agnostic Pretraining ⇨ Freeze

Foundation Models



Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

AnyLoc: Use Intermediate Features from Self-Supervised ViT



Embedding & Positional Encoding

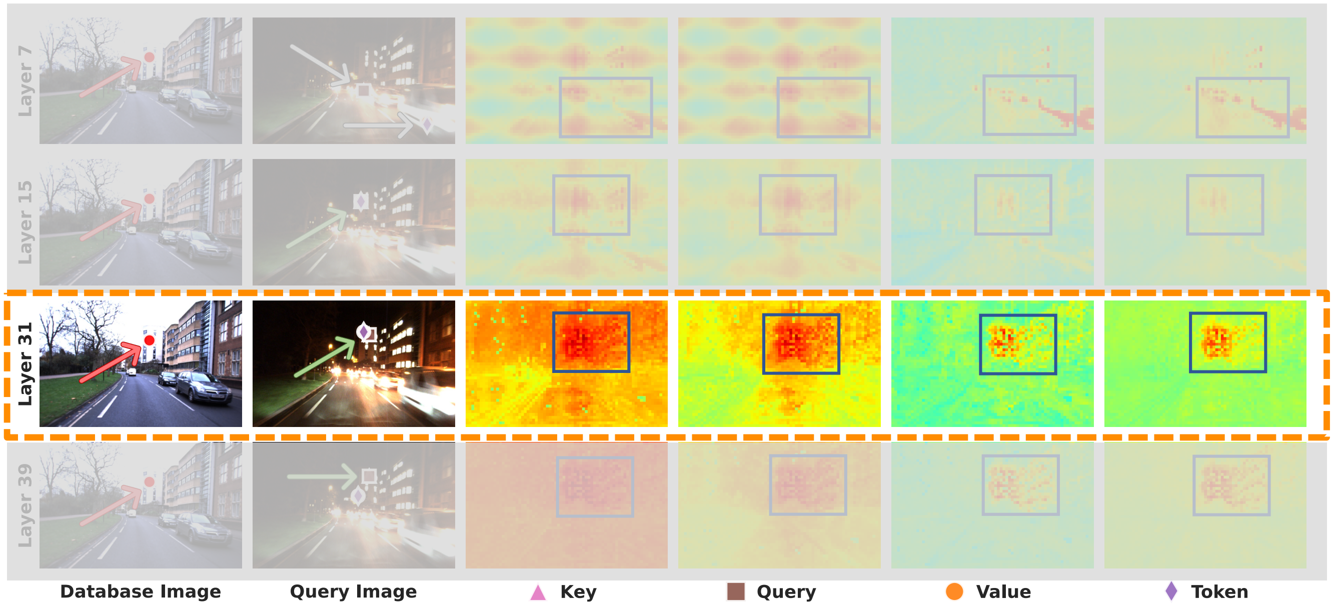
Transformer Layer L1

Transformer Layer L7

Transformer Layer L31

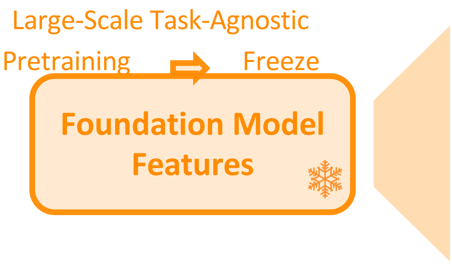
Transformer Layer L39

Layer 31 Value has the best contrast.



Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

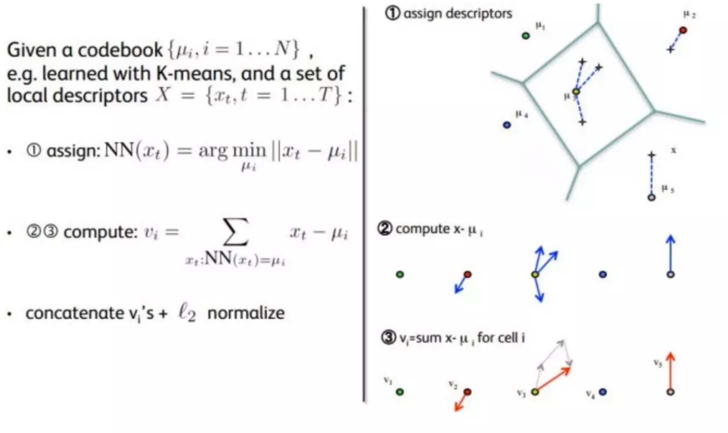
AnyLoc: Unsupervised Local Feature Aggregation



Large-Scale Task-Agnostic Pretraining ⇨ Freeze

Foundation Model Features

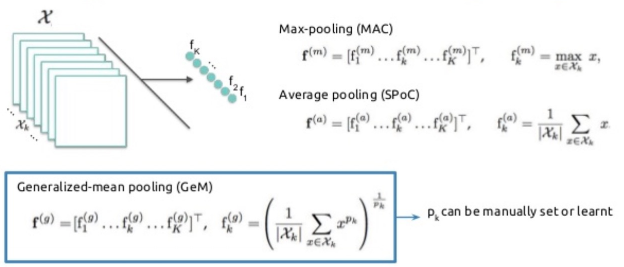
VLAD: Vector of Locally Aggregated Descriptors



Given a codebook $\{\mu_i, i = 1 \dots N\}$, e.g. learned with K-means, and a set of local descriptors $X = \{x_t, t = 1 \dots T\}$:

- ① assign: $\mathrm{NN}(x_t) = \arg\min_{\mu_i} ||x_t - \mu_i||$
- ②③ compute: $v_i = \sum_{x_t : \mathrm{NN}(x_t) = \mu_i} x_t - \mu_i$
- concatenate v_i's + ℓ_2 normalize

GeM: Generalized Mean Pooling



Max-pooling (MAC)

$$\mathbf{f}^{(m)} = [\mathrm{f}_1^{(m)} \dots \mathrm{f}_k^{(m)} \dots \mathrm{f}_K^{(m)}]^\top, \quad \mathrm{f}_k^{(m)} = \max_{x \in \mathcal{X}_k} x,$$

Average pooling (SPoC)

$$\mathbf{f}^{(a)} = [\mathrm{f}_1^{(a)} \dots \mathrm{f}_k^{(a)} \dots \mathrm{f}_K^{(a)}]^\top, \quad \mathrm{f}_k^{(a)} = \frac{1}{|\mathcal{X}_k|} \sum_{x \in \mathcal{X}_k} x$$

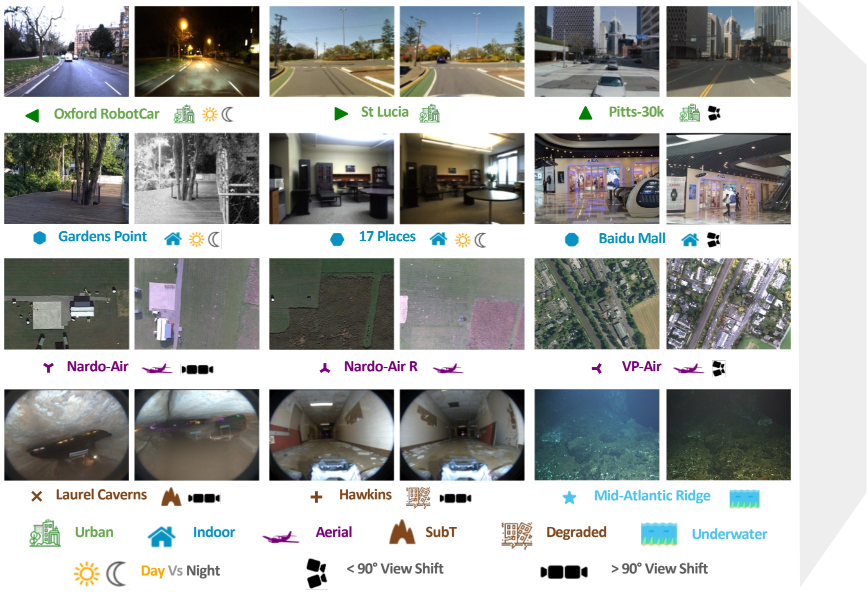
Generalized-mean pooling (GeM)

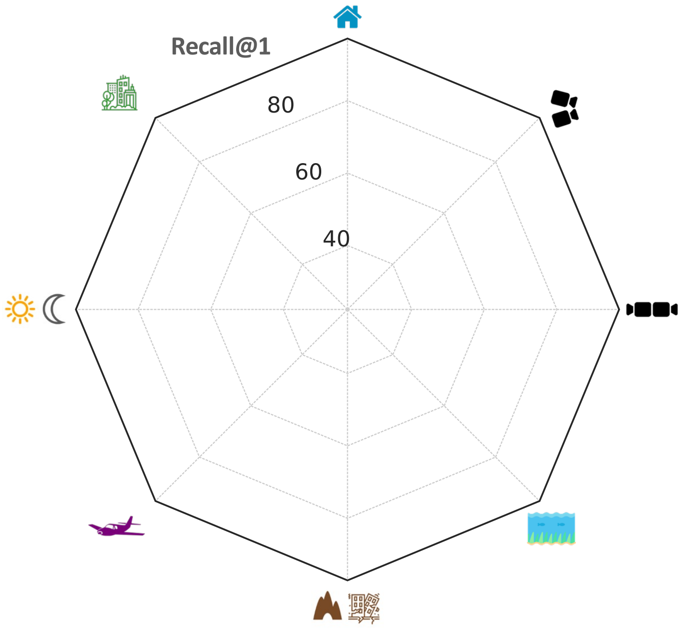
$$\mathbf{f}^{(g)} = [\mathrm{f}_1^{(g)} \dots \mathrm{f}_k^{(g)} \dots \mathrm{f}_K^{(g)}]^\top, \quad \mathrm{f}_k^{(g)} = \left(\frac{1}{|\mathcal{X}_k|} \sum_{x \in \mathcal{X}_k} x^{p_k} \right)^{\frac{1}{p_k}}$$

p_k can be manually set or learnt

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

Diverse Testbed consisting of 9 Datasets

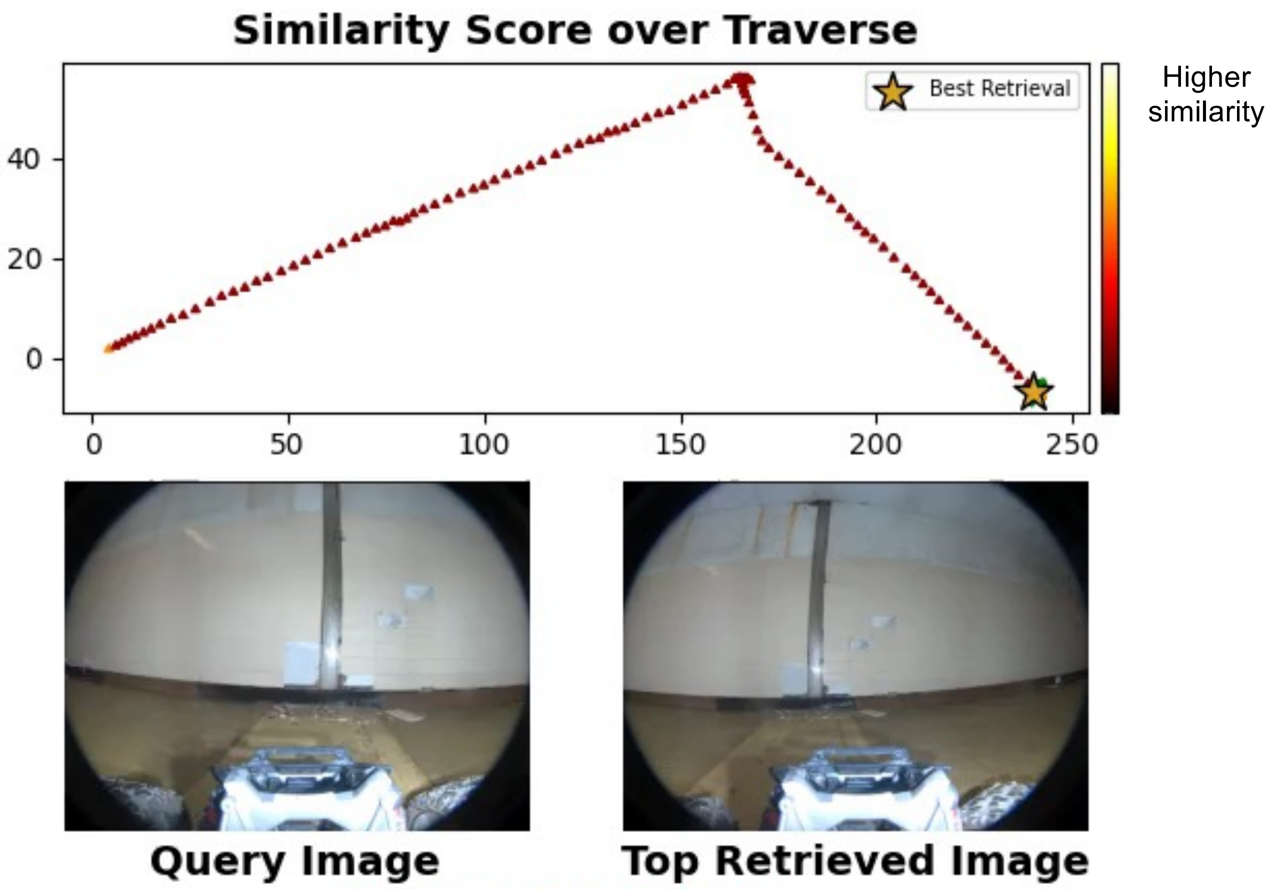




Metric is Recall@K, i.e., % Accuracy using the Top K Retrievals

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

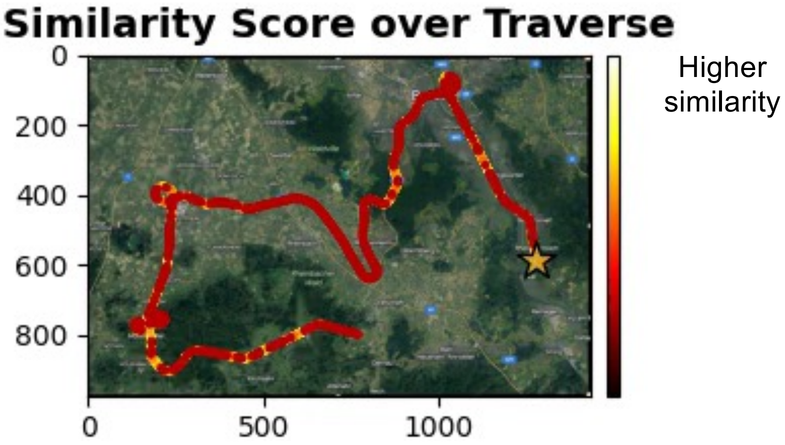
AnyLoc on a Visually Degraded Environment (Hawkins)



No Temporal Information used

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

AnyLoc on a 500 Km Aerial Dataset (VP-Air)


Similarity Score over Traverse

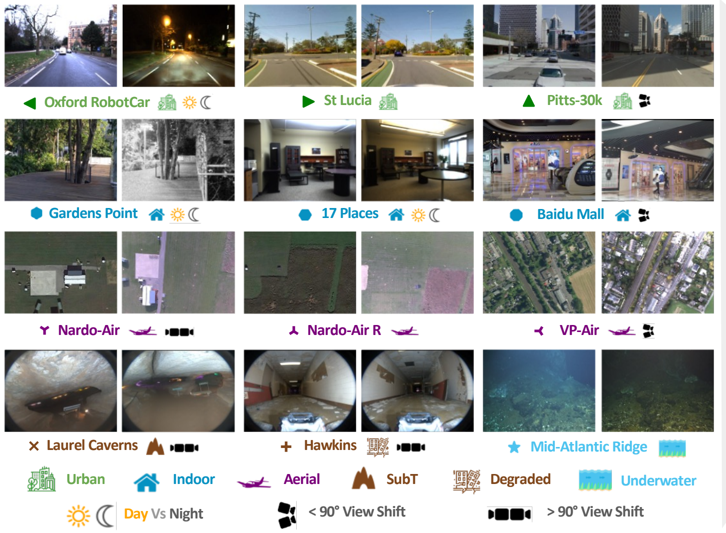
Query Image

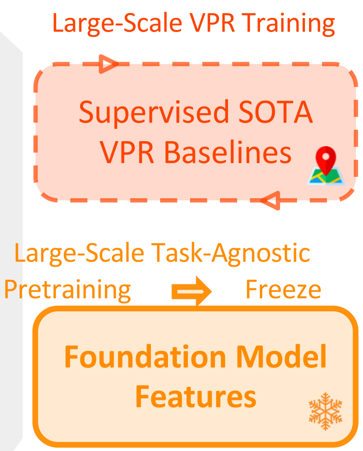
Top Retrieved Image

No Temporal Information Used

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

AnyLoc achieves up to 4X wider performance



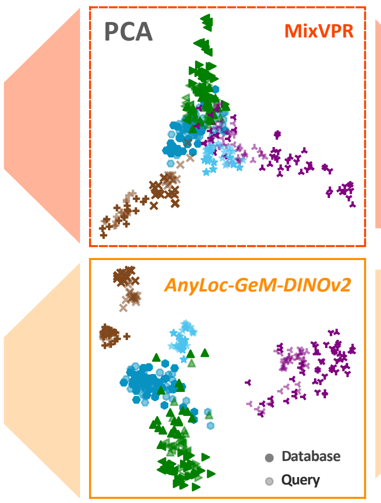


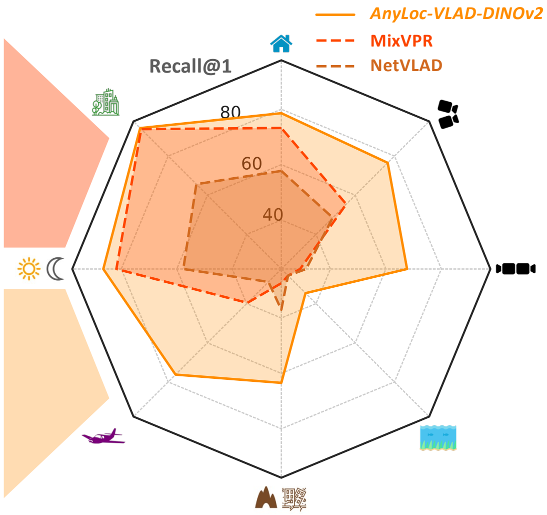
Large-Scale VPR Training

Supervised SOTA VPR Baselines

Large-Scale Task-Agnostic Pretraining ⇨ Freeze

Foundation Model Features





Emergence of Distinct Domains in the Latent Space

Key Takeaway: Self-Supervised Visual Features enable Universal Generalization

Next Step: Precise 6-DoF Pose Estimation using Fine Pixel-level Features

Keetha et.al, AnyLoc: Towards Universal Visual Place Recognition, RA-L 2023 & ICRA 2024

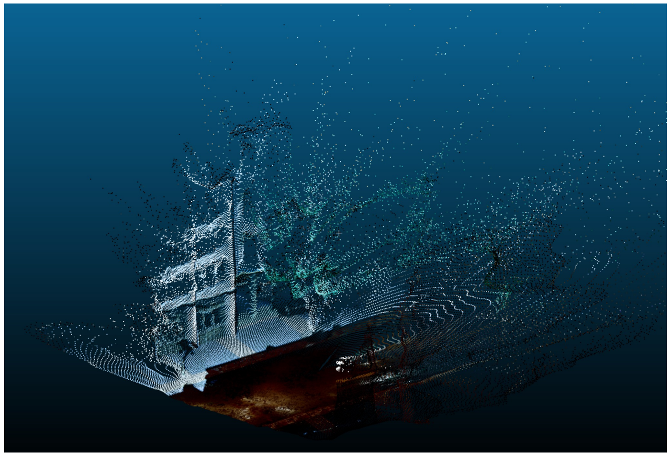
Carnegie Mellon University
The Robotics Institute

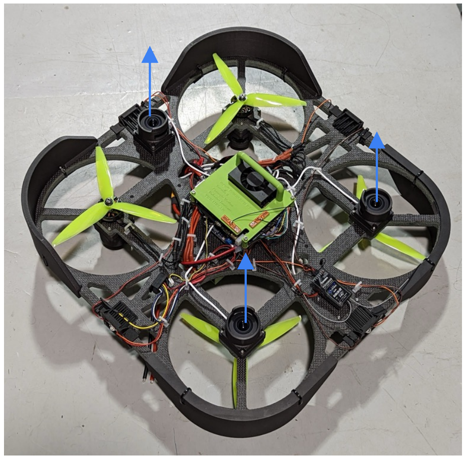
Geometry-Informed Distance Candidate Selection for Omnidirectional Stereo Vision with Fisheye Images

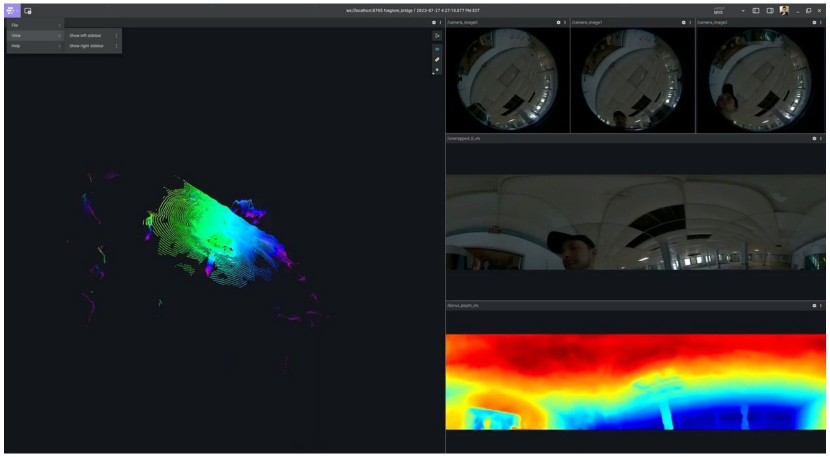
Conner Pulling, Je Hon Tan, Yaoyu Hu, Sebastian Scherer

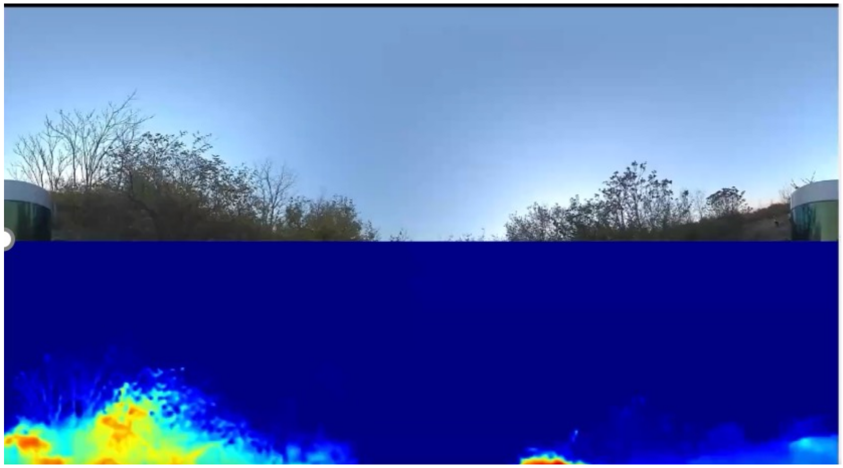
theairlab.org

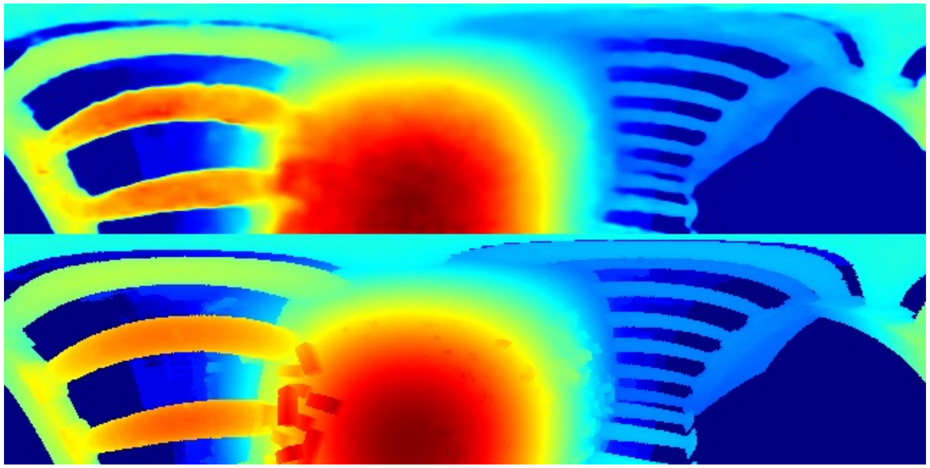
Omnidirectional Depth is an important downstream task for UAVs!





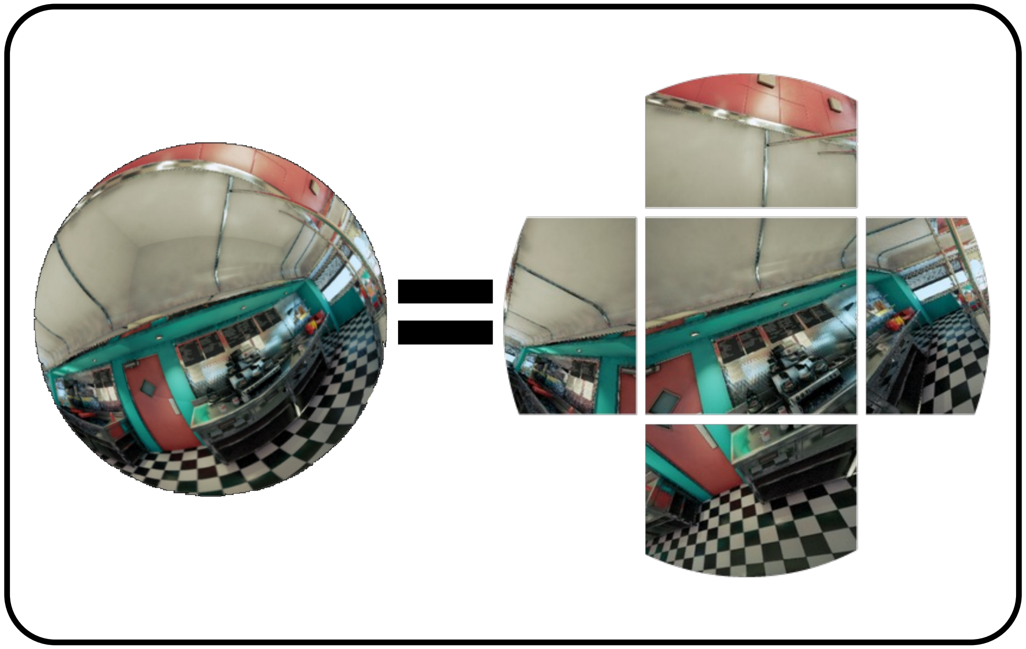


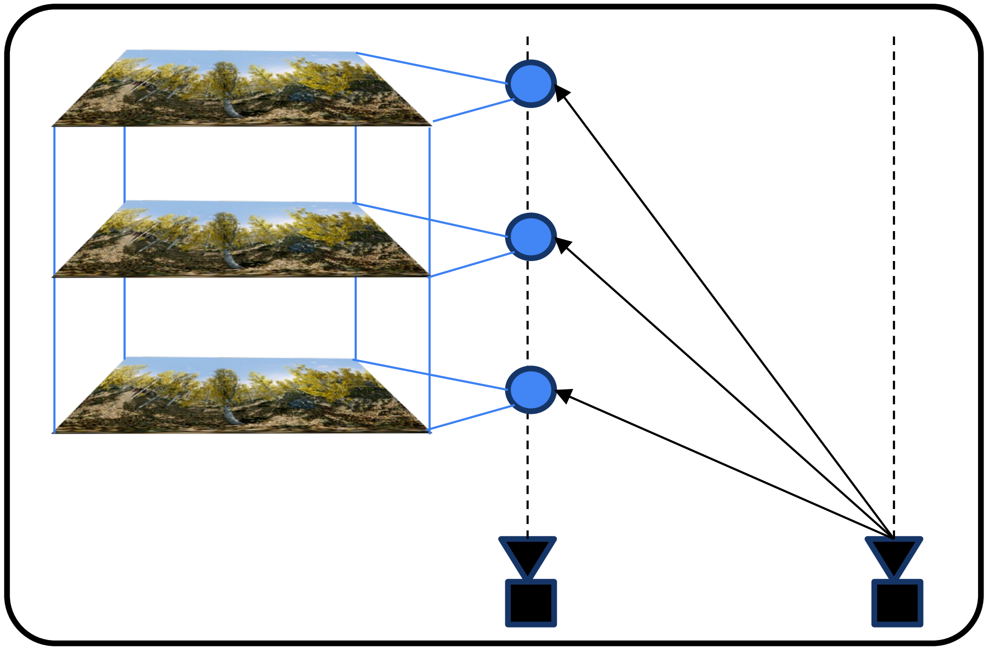




Using fisheye images increases coverage but also increases problem complexity.

Increased Field-of-View (FOV) But… More Complexity & Cost Volume

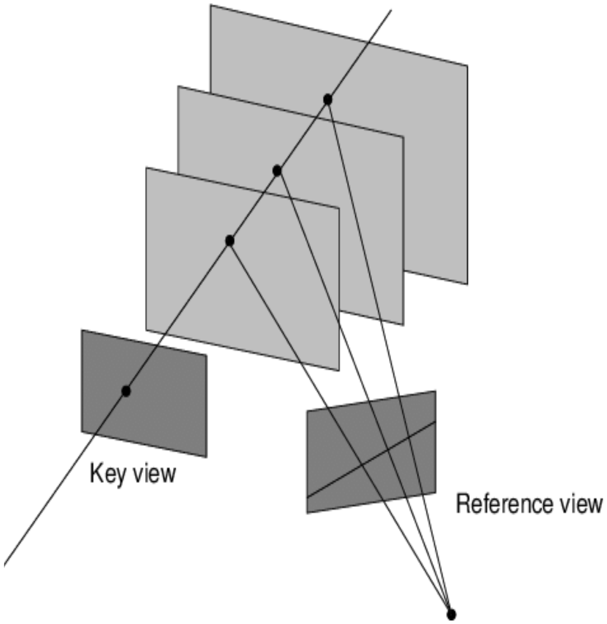




Fisheye images cover a larger field-of-view (FOV) but stereo correspondences lie on epipolar **curves** instead of epipolar **lines**. Possible correspondences are found through **warping images** with depth guesses, called **depth candidates**, with a **cost volume**.

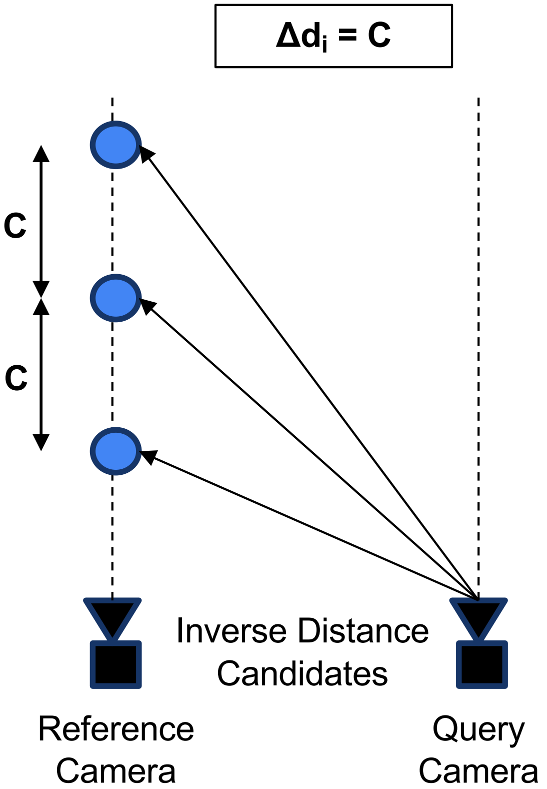
How does one choose depth candidates to build the cost volume?

With Pinhole Cameras…



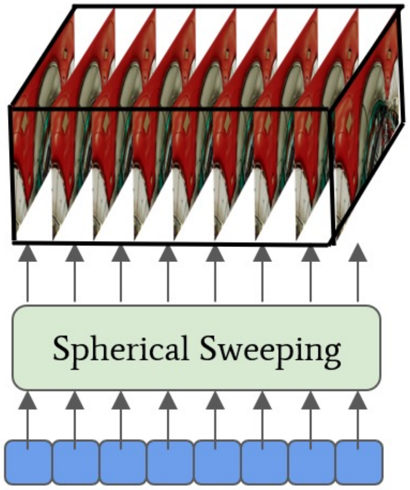
Plane-Sweeping

In Previous Work…



Warping w/ Candidates

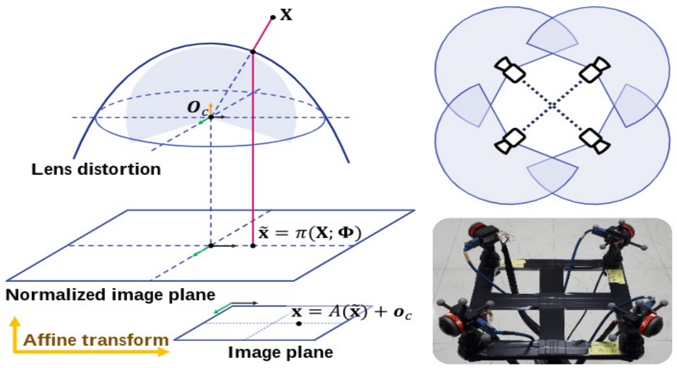
$$W(I, T, d_i)$$



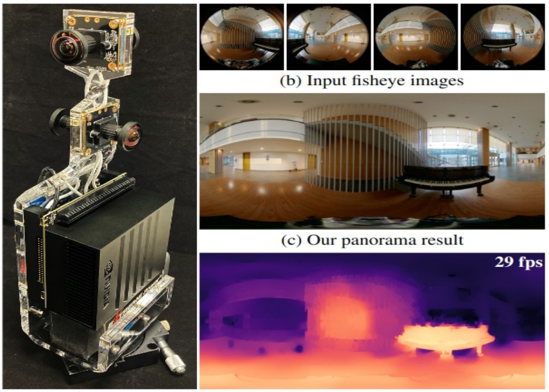
Sampling Grid Encodes Intrinsics & Extrinsics Too!

Prior Work

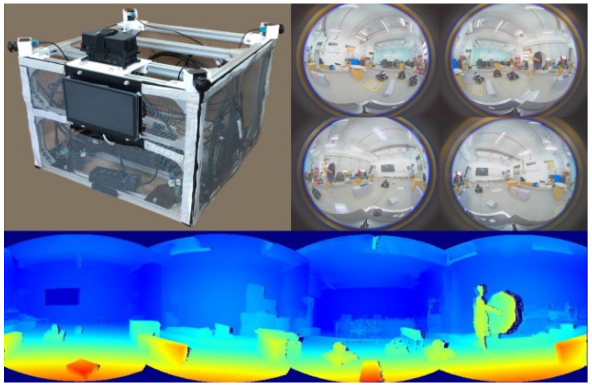
OmniMVS


X
O_c
Lens distortion
$\tilde{\mathbf{x}} = \pi(\mathbf{X}; \mathbf{\Phi})$
Normalized image plane
$\mathbf{x} = A(\tilde{\mathbf{x}}) + \boldsymbol{o}_c$
Affine transform
Image plane

RTSS

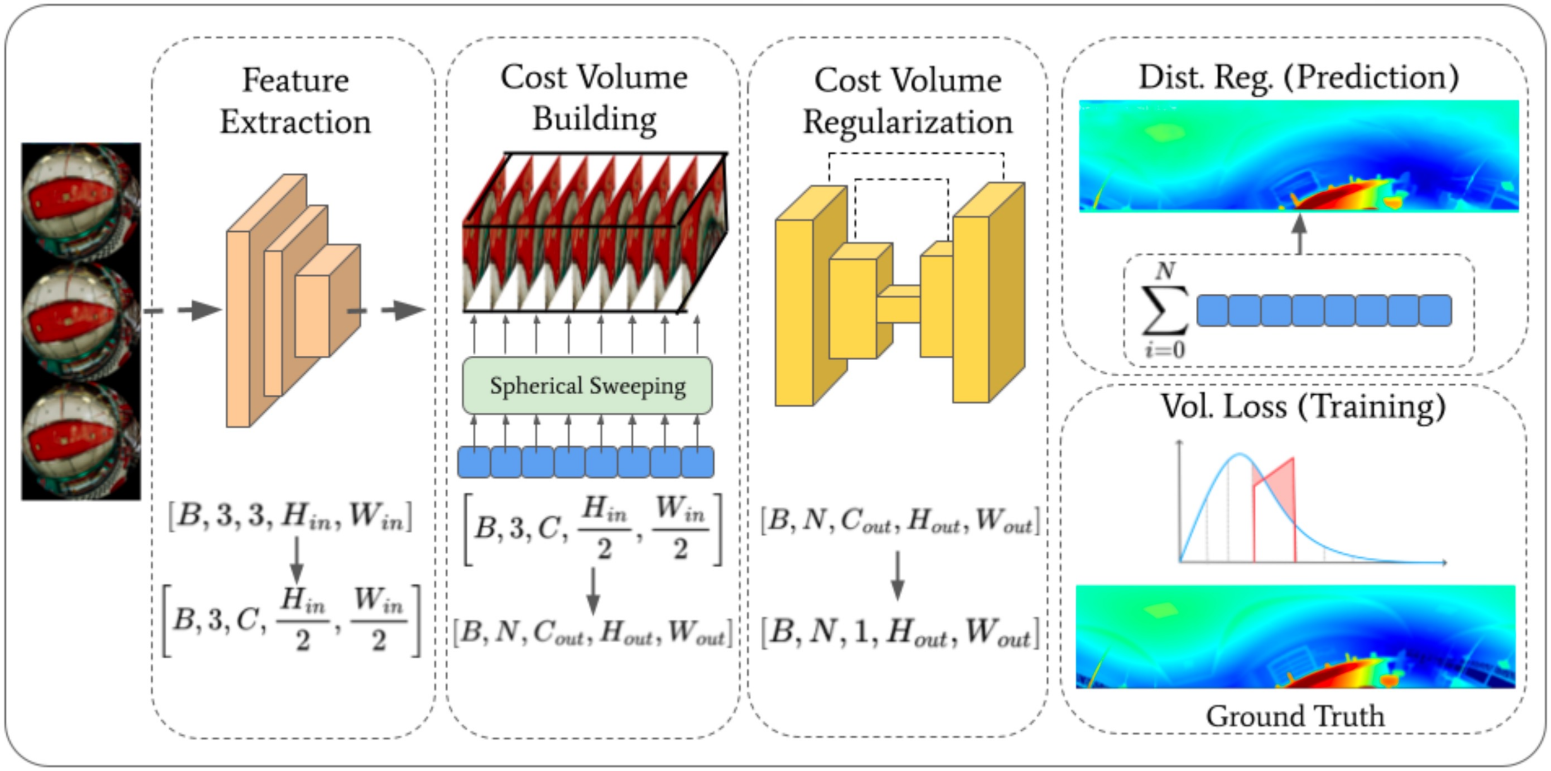


OmniVidar

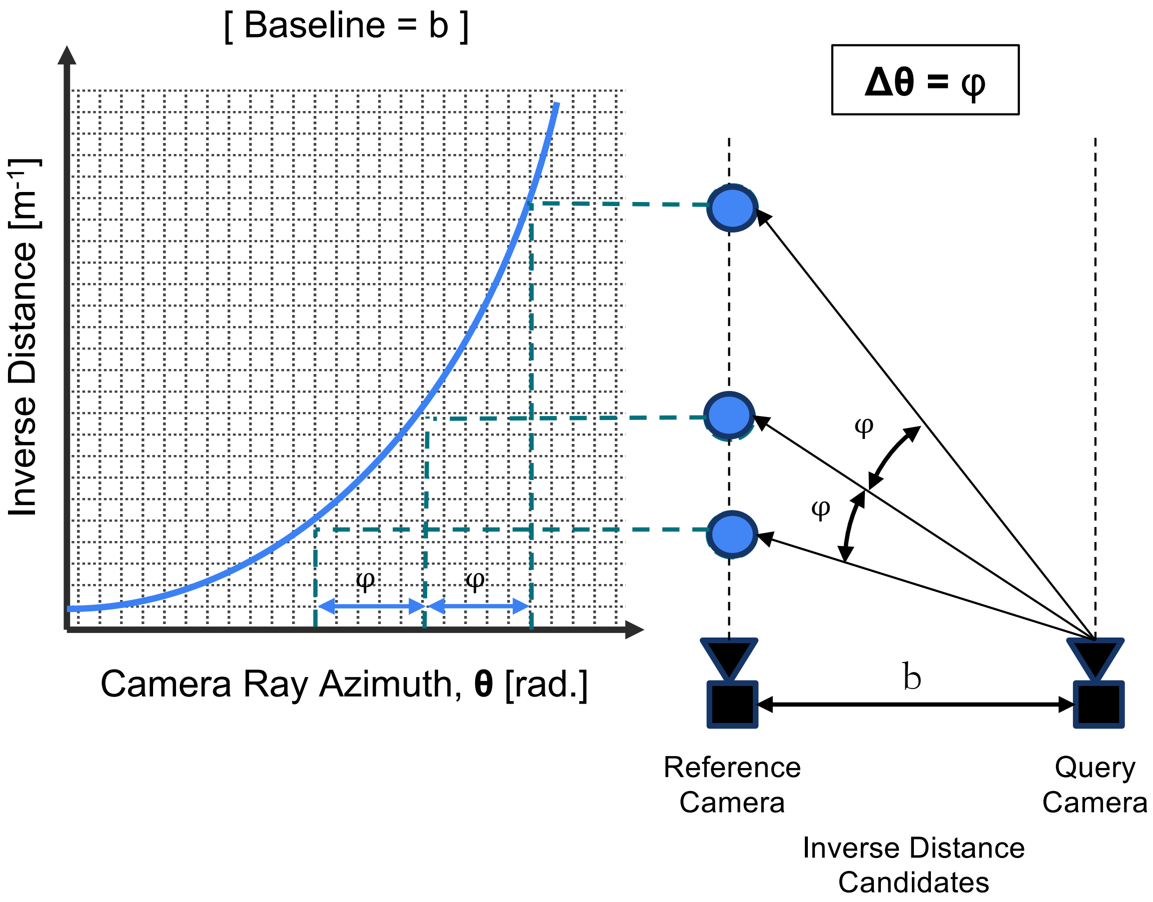


Method / Key Factors	OmniMVS	RTSS	OmniVidar	(Ours)
Learning-Based	✅	❌	✅	✅
Real-Time Inference	❌	✅	✅	✅
Full Field-of-View (FOV)	❌	✅	❌	✅
Reconfigurable w/o Finetuning	❌	✅	❌	✅
Self-Occlusion Masking (trinocular)	❌	❌	❌	✅

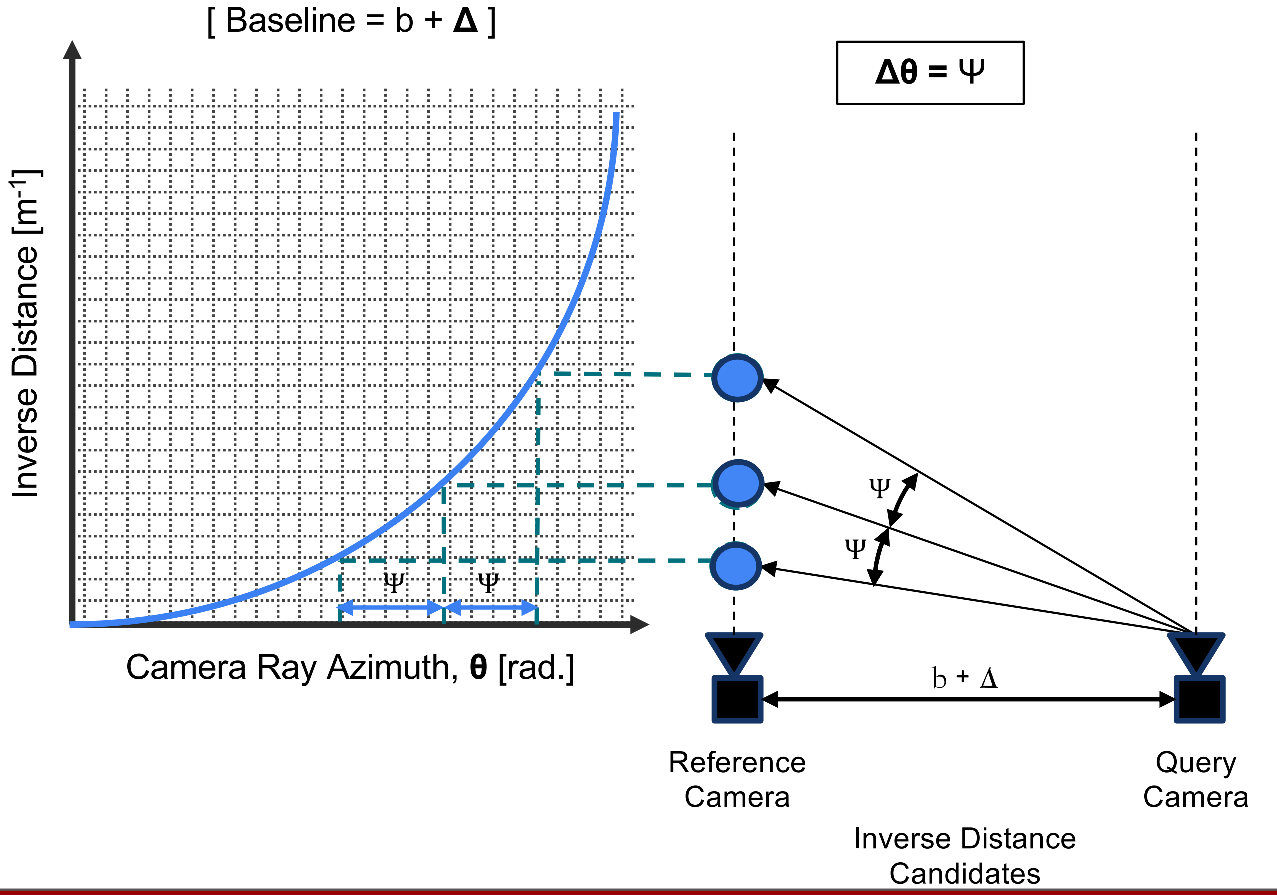
Approach


Feature Extraction
$[B, 3, 3, H_{in}, W_{in}]$
$\left[B, 3, C, \frac{H_{in}}{2}, \frac{W_{in}}{2}\right]$
Cost Volume Building
Spherical Sweeping
$\left[B, 3, C, \frac{H_{in}}{2}, \frac{W_{in}}{2}\right]$
$[B, N, C_{out}, H_{out}, W_{out}]$
Cost Volume Regularization
$[B, N, C_{out}, H_{out}, W_{out}]$
$[B, N, 1, H_{out}, W_{out}]$
Dist. Reg. (Prediction)
$\sum_{i=0}^{N}$
Vol. Loss (Training)
Ground Truth

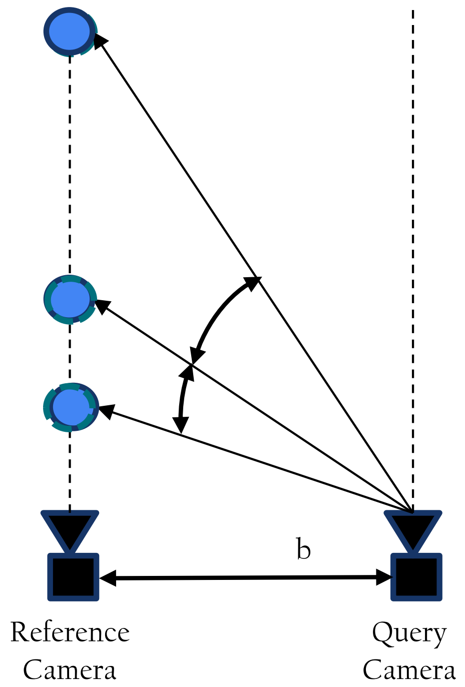
Candidates can be sampled such that the angle between camera rays in the reference space is constant.


[Baseline = b]
Inverse Distance [m^{-1}]
Camera Ray Azimuth, **θ** [rad.]
φ
φ
Δθ = φ
φ
φ
b
Reference Camera
Query Camera
Inverse Distance Candidates

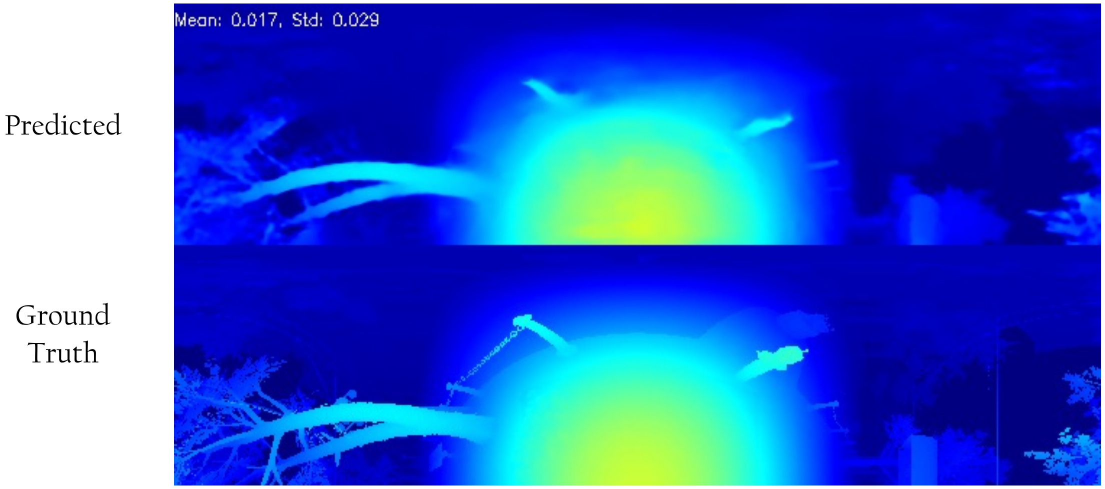
Note that this distribution changes when the baseline distance between cameras changes!


[Baseline = b + **Δ**]
Δθ = Ψ
Inverse Distance [m^{-1}]
Camera Ray Azimuth, **θ** [rad.]
Ψ
Ψ
Ψ
Ψ
b + Δ
Reference Camera
Query Camera
Inverse Distance Candidates

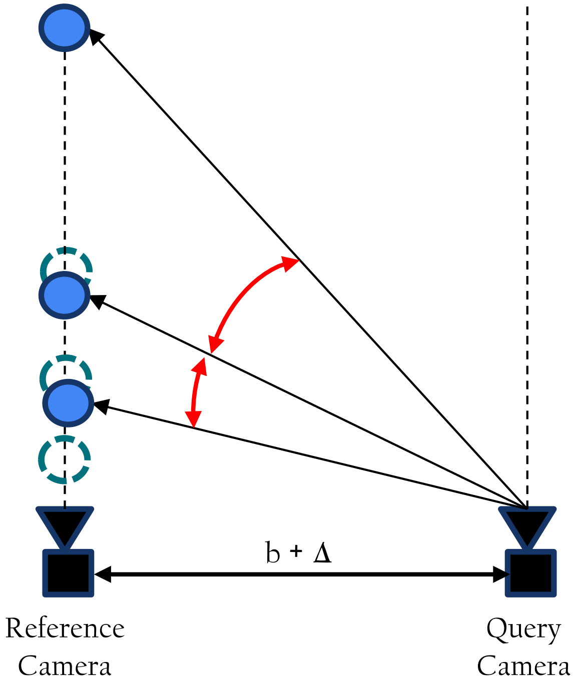
Geometry Informed (GI) Candidates Improve Performance when Baseline Changes.



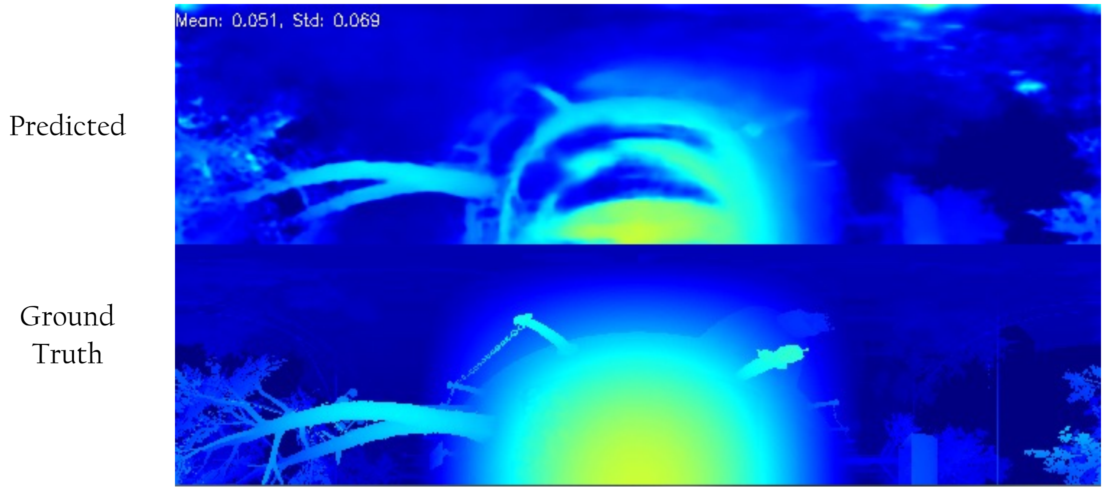
Inverse Distance Candidates



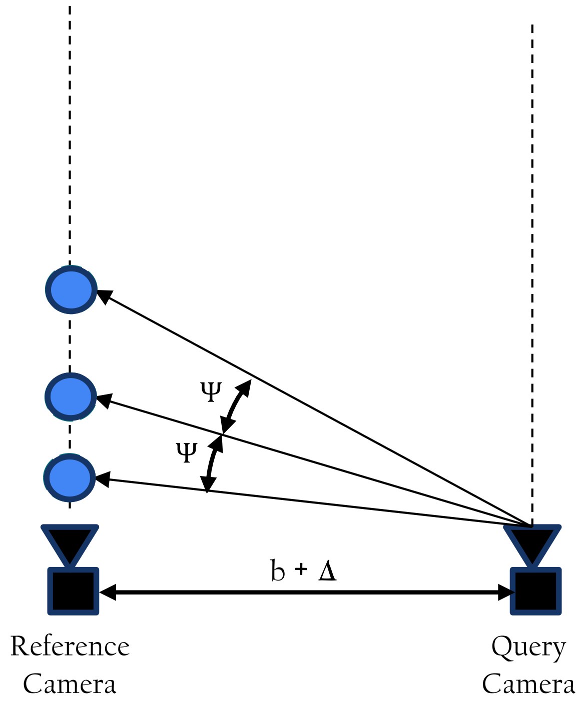
Changing the baseline distance after training degrades performance.



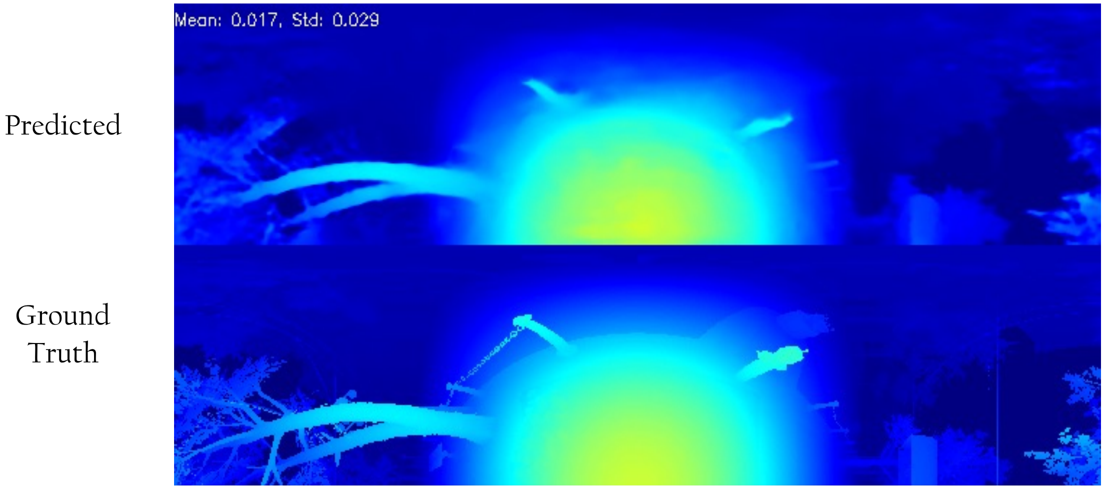
Inverse Distance Candidates



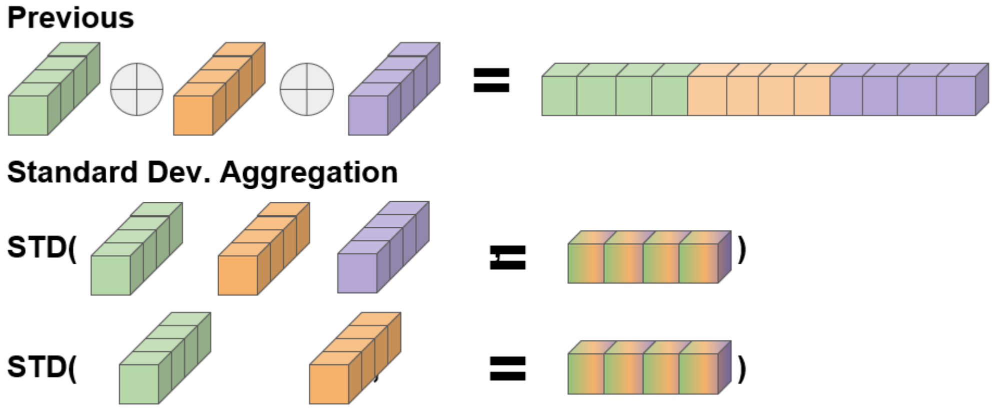
Correcting the Geomtry Informed (GI) candidate distribution after training restores performance without finetuning.



Inverse Distance Candidates



Self-Occlusion Masking using Data Augmentation and Novel Cost Volume Aggregation

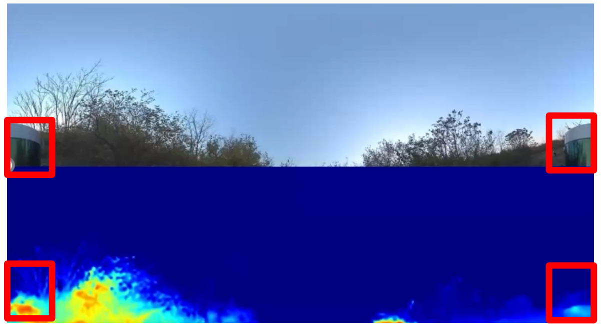


Previous

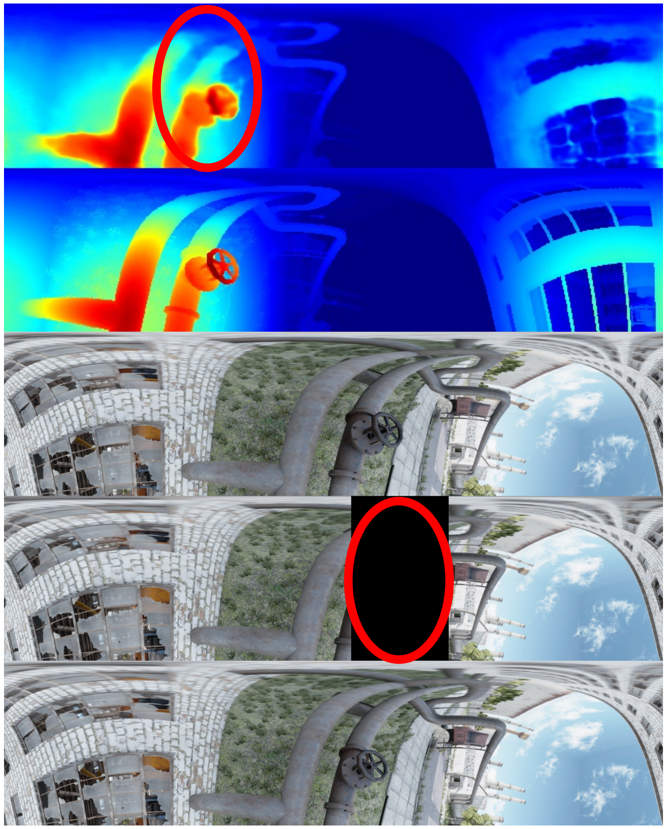
Standard Dev. Aggregation

STD(=)

STD(=)

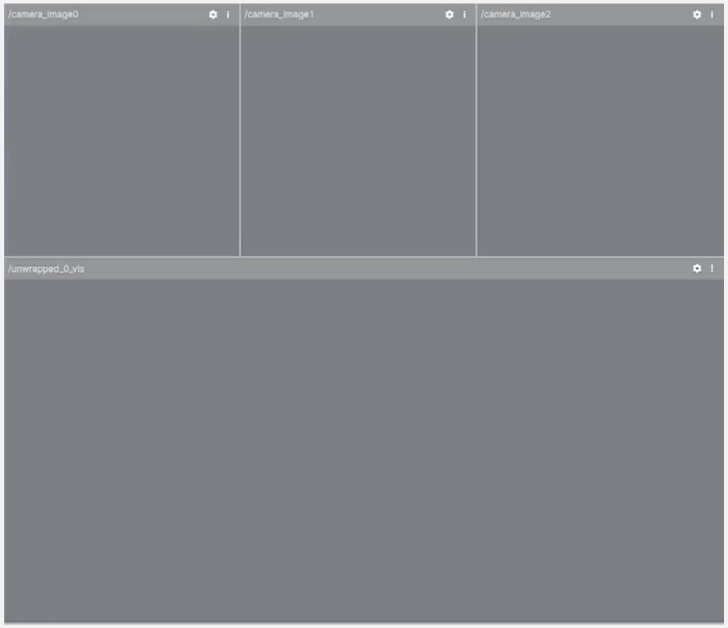


Per-Pixel Standard Deviation of all valid views allows masking.

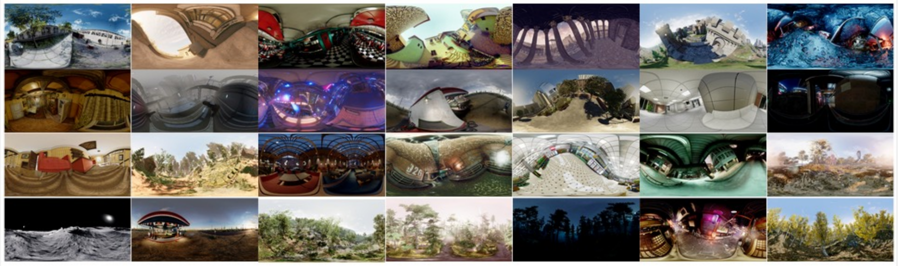


Random Mask Augmentation

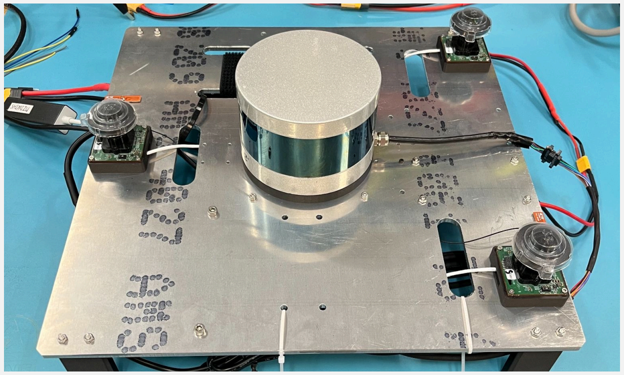
SOTA Dataset and Real-World Evaluation Setup



Real Evaluation Data Collected in Difficult Indoor & Outdoor Environments

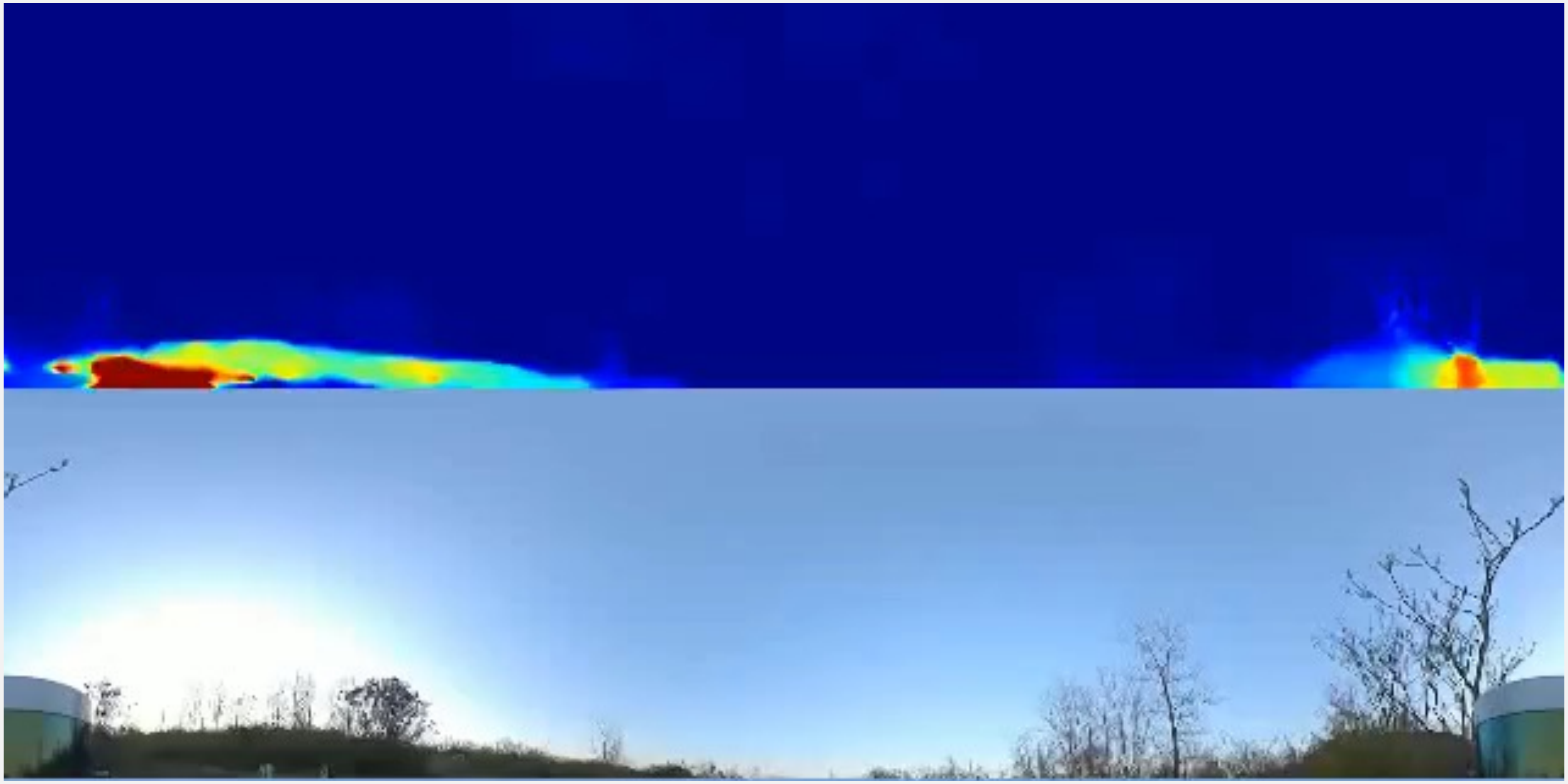


Synthetic Dataset with 100k+ samples, 70 Envs. (10x more samples than prev. work)



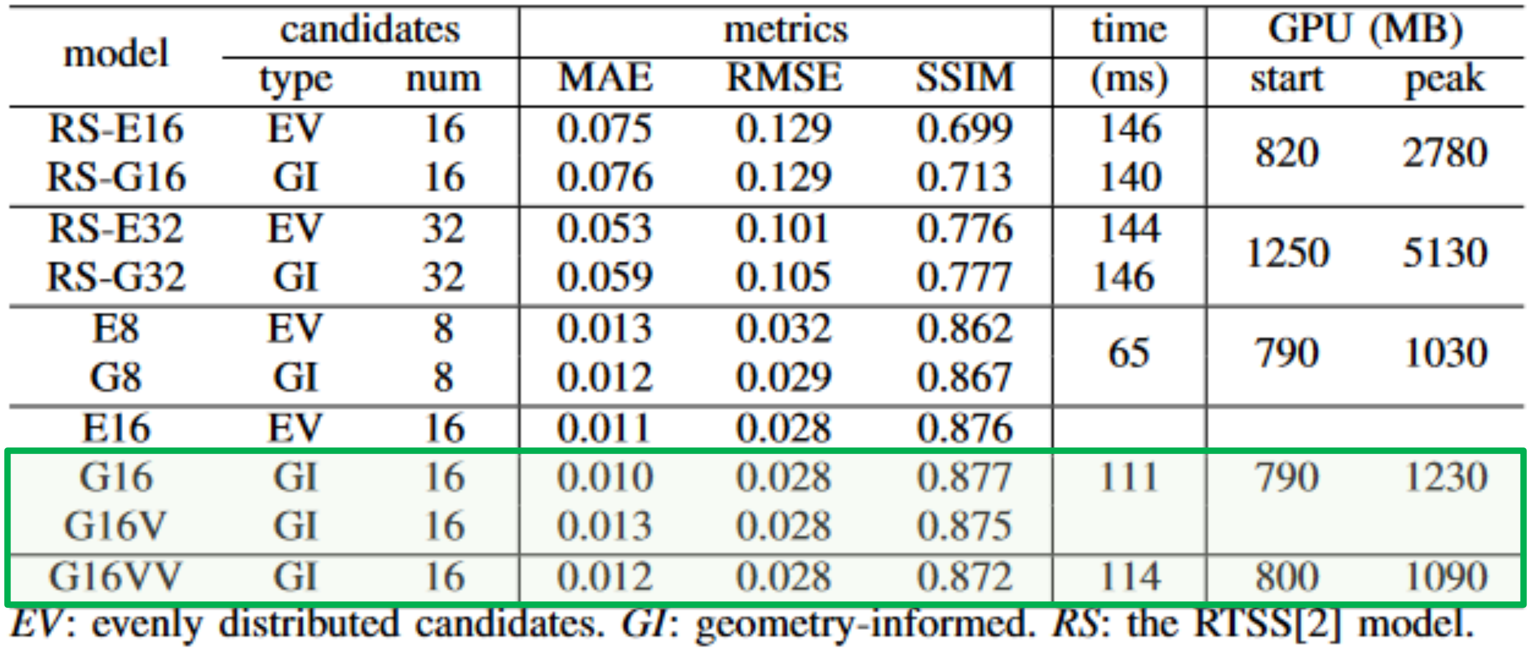
Models are compared using Mean Absolute Error (MAE), Root Mean-Squared Error (RMSE), and Structural Simularity Index Measure (SSIM) as in prior works.

Real World Inference In Outdoor Environments



Note that the self-occluding LiDar is masked out!

Quantitative Results



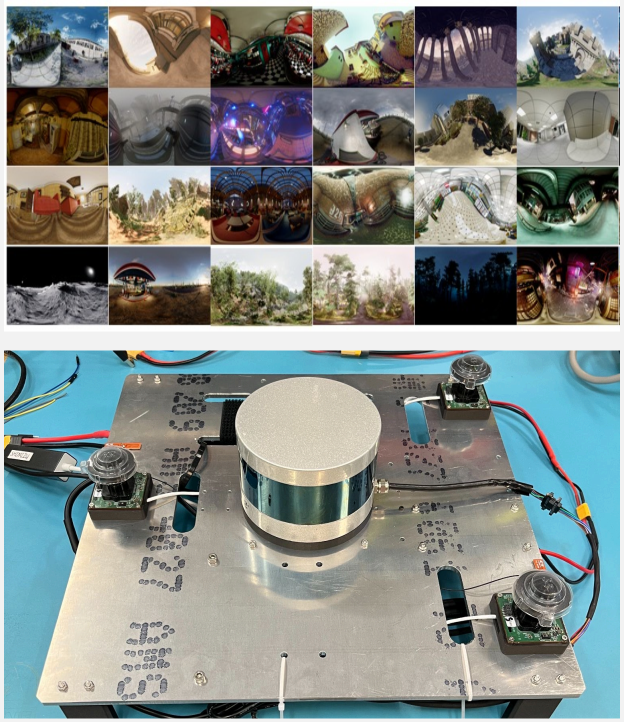
			↓	↓	↑	↓		
model	candidates		metrics			time	GPU (MB)	
	type	num	MAE	RMSE	SSIM	(ms)	start	peak
RS-E16	EV	16	0.075	0.129	0.699	146	820	2780
RS-G16	GI	16	0.076	0.129	0.713	140		
RS-E32	EV	32	0.053	0.101	0.776	144	1250	5130
RS-G32	GI	32	0.059	0.105	0.777	146		
E8	EV	8	0.013	0.032	0.862	65	790	1030
G8	GI	8	0.012	0.029	0.867			
E16	EV	16	0.011	0.028	0.876	111	790	1230
G16	GI	16	0.010	0.028	0.877			
G16V	GI	16	0.013	0.028	0.875			
G16VV	GI	16	0.012	0.028	0.872	114	800	1090

EV: evenly distributed candidates. *GI*: geometry-informed. *RS*: the RTSS[2] model.

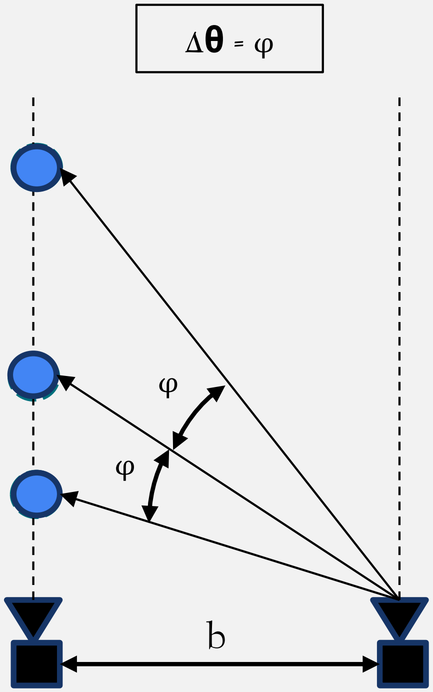
Better is... (↓ Lower/ ↑ Higher)

GI Candidates, Variance Cost Volume (G16V), and Self-Occlusion Masking (G16VV) are all **better/comparable to learning baselines but are more robust and adaptable.**

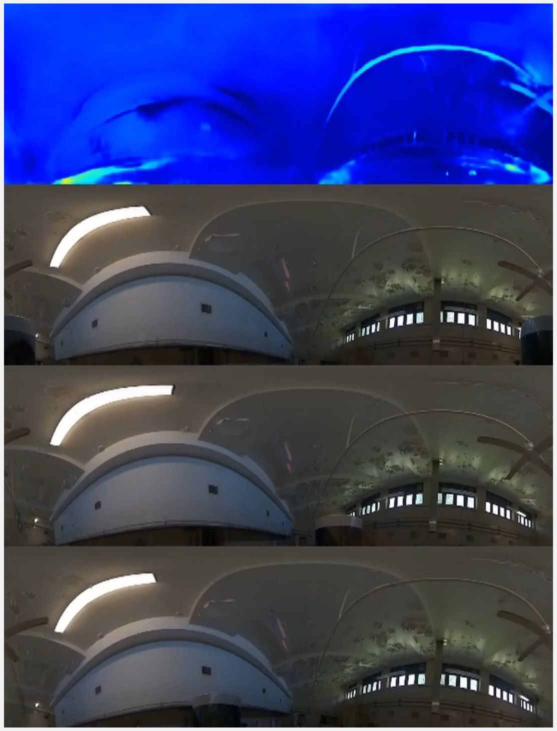
Contributions & Conclusions



State-of-the-Art Dataset with 10x Data and Real Data

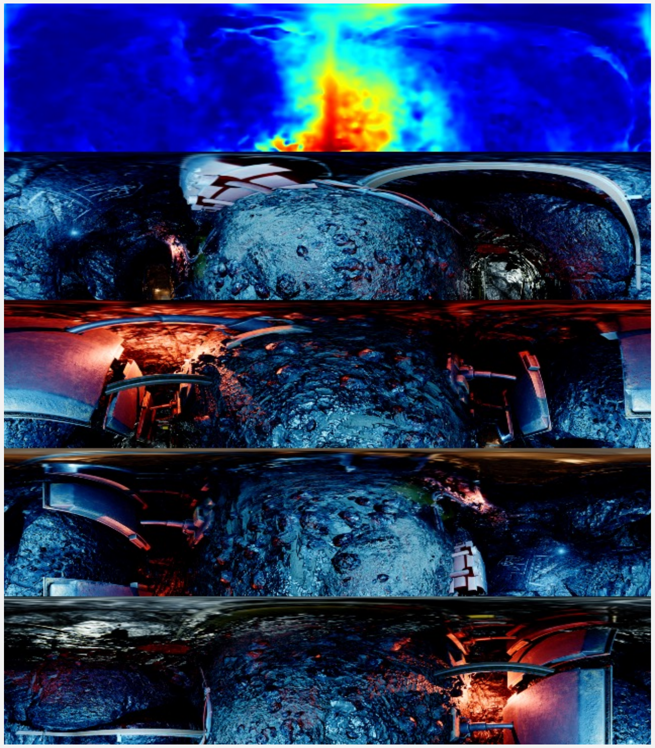


Geometry-Informed Candidates



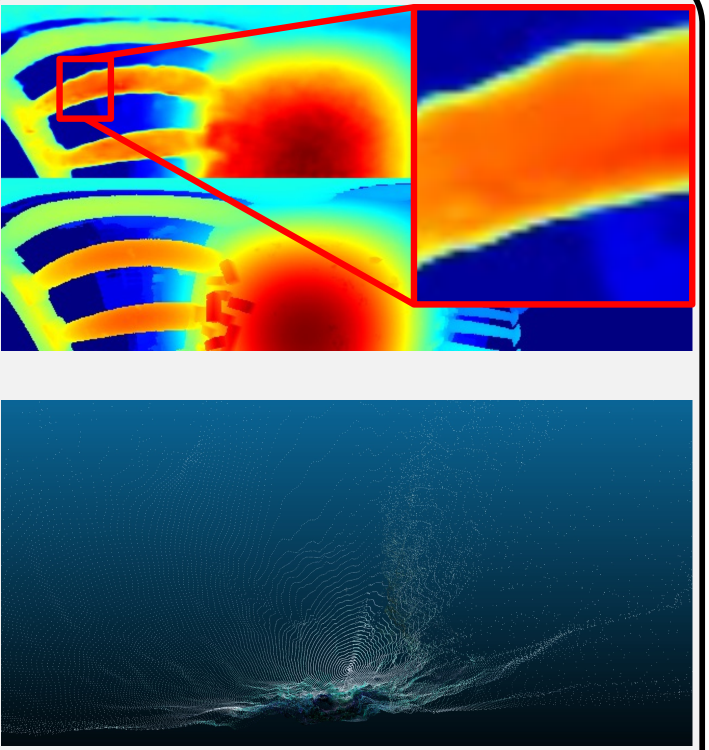
Pretrained, Reconfigurable, and Released Models

Current Limitations & Research Directions



Need for Rotation Invariant Features

Configuration-Agnostic Evaluation Techniques



Ghost Points

Summary

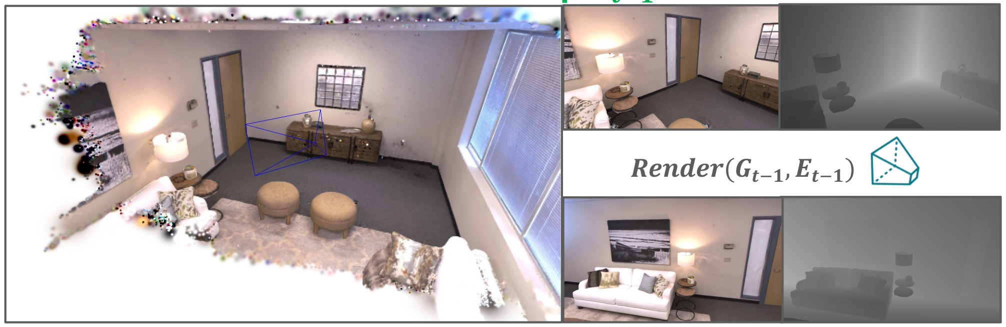
- We have created several challenging datasets for SLAM and place recognition that reflect real-world challenges for autonomous systems and might be useful for your research.

- There is still a significant progress required in all parts from odometry, mapping, to place recognition

- Robustness is more important in actual applications. What happens at the edge or beyond the “envelope” of your method?

Online Camera Tracking & Reconstruction

Gaussian Map G_{t-1}

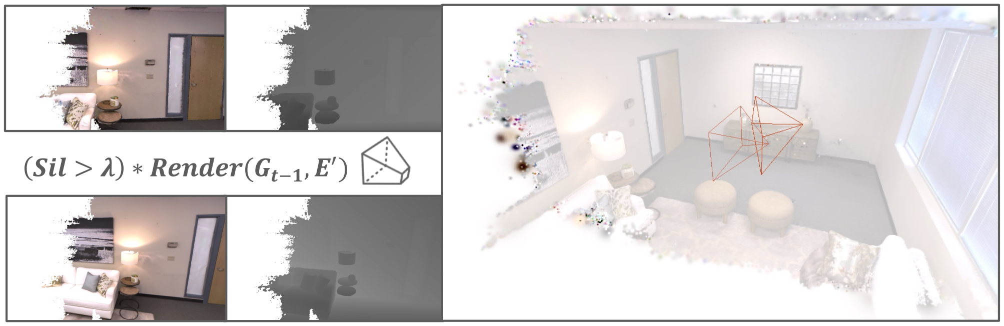


$Render(G_{t-1}, E_{t-1})$

Gaussian Splats

Incoming Frame F_t

(1) Camera Tracking $E_{t-1} \rightarrow E_t$

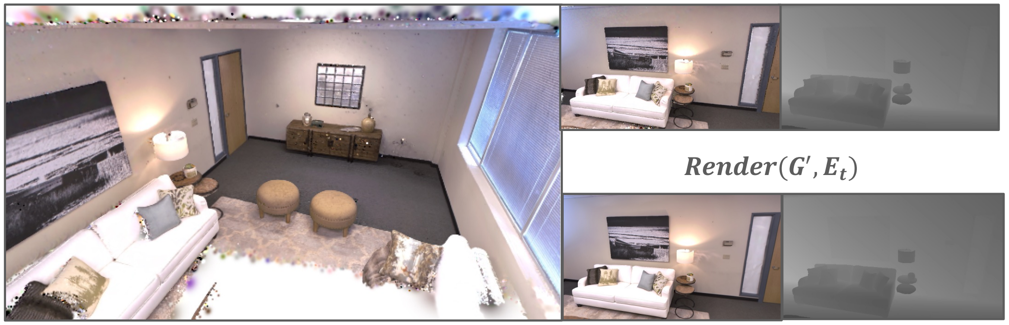


$(Sil > \lambda) * Render(G_{t-1}, E')$

$(Sil > \lambda) * F_t$

$E_t = \underset{E'}{\text{argmin}} \|(Sil > \lambda) * (Render(G_{t-1}, E') - F_t)\|_1$

(3) Map Update G_t

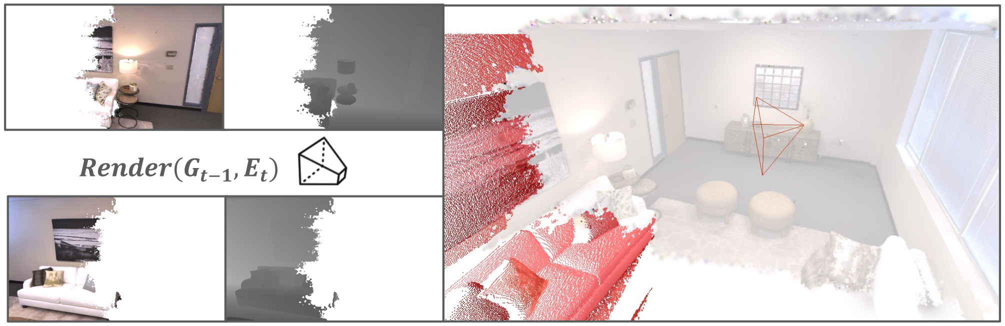


$Render(G', E_t)$

$G_t = \underset{G'}{\text{argmin}} \sum_{k=1}^{t} \|Render(G', E_k) - F_k\|_1$

Current Frame F_t

(2) Gaussian Densification G_t^d



$Render(G_{t-1}, E_t)$

$(Densify\ Mask) * F_t$

$G_t^d = Densify\,(G_{t-1}, F_t, E_t, Sil)$

Keetha et.al, SplaTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM, CVPR 2024

SplaTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM

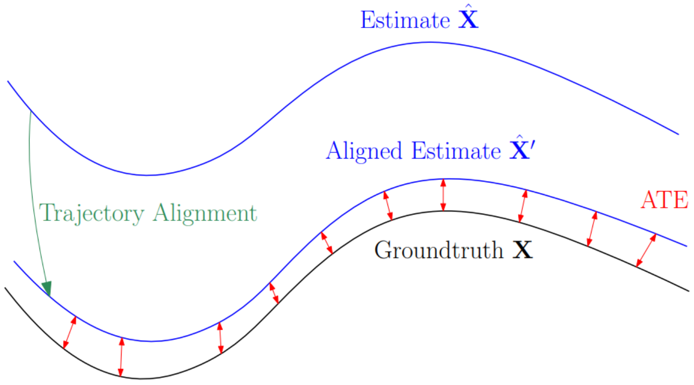
SLAM Visualization

Novel View Synthesis

Keetha et.al, SplaTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM, CVPR 2024

Rethinking SLAM Metrics for Robustness

Accuracy Metric:

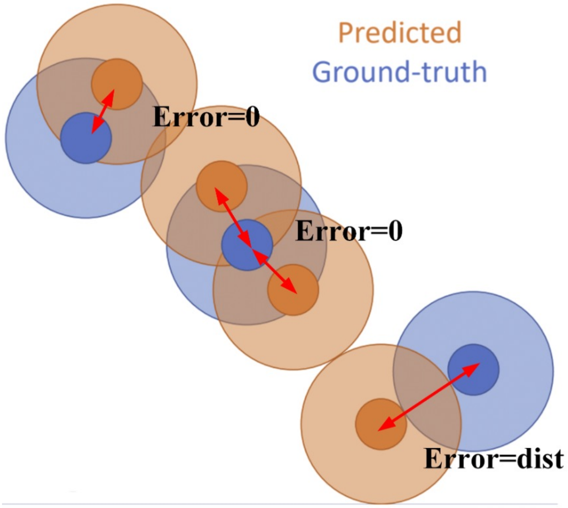

Estimate $\hat{\mathbf{X}}$
Aligned Estimate $\hat{\mathbf{X}}'$
Trajectory Alignment
ATE
Groundtruth $\mathbf{X}$

$$\mathrm{ATE}_{\mathrm{rot}} = (\frac{1}{N}\sum_{i=0}^{N-1} \|\angle(\Delta \mathrm{R}_i)\|^2)^{\frac{1}{2}},$$

$$\mathrm{ATE}_{\mathrm{pos}} = (\frac{1}{N}\sum_{i=0}^{N-1} \|\Delta \mathbf{p}_i\|^2)^{\frac{1}{2}},$$

Does not consider impact of local bad measurements

Robustness Metric:



$$F_1(e) = \frac{2P(e<T)R(e<T)}{P(e<T)+R(e<T)},$$

Considers both **Accuracy and Completeness**

Example Robustness Metric Evaluation from ICCV 2023 SLAM Challenge

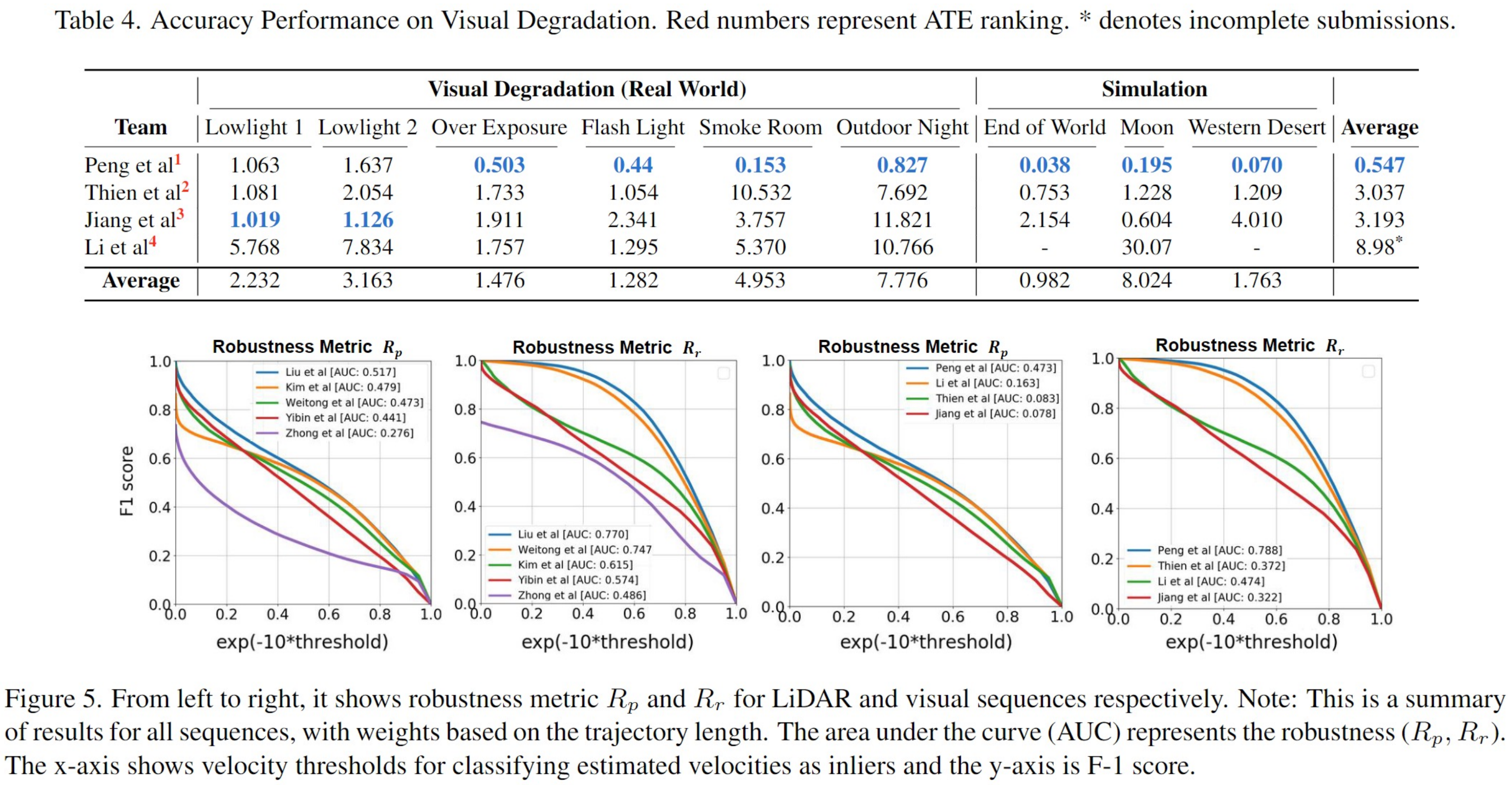


Table 4. Accuracy Performance on Visual Degradation. Red numbers represent ATE ranking. * denotes incomplete submissions.

	Visual Degradation (Real World)						Simulation			
Team	Lowlight 1	Lowlight 2	Over Exposure	Flash Light	Smoke Room	Outdoor Night	End of World	Moon	Western Desert	**Average**
Peng et al[1]	1.063	1.637	**0.503**	**0.44**	**0.153**	**0.827**	**0.038**	**0.195**	**0.070**	**0.547**
Thien et al[2]	1.081	2.054	1.733	1.054	10.532	7.692	0.753	1.228	1.209	3.037
Jiang et al[3]	**1.019**	**1.126**	1.911	2.341	3.757	11.821	2.154	0.604	4.010	3.193
Li et al[4]	5.768	7.834	1.757	1.295	5.370	10.766	-	30.07	-	8.98*
Average	2.232	3.163	1.476	1.282	4.953	7.776	0.982	8.024	1.763	

Figure 5. From left to right, it shows robustness metric R_p and R_r for LiDAR and visual sequences respectively. Note: This is a summary of results for all sequences, with weights based on the trajectory length. The area under the curve (AUC) represents the robustness (R_p, R_r). The x-axis shows velocity thresholds for classifying estimated velocities as inliers and the y-axis is F-1 score.

The area under the curve represents the robustness metric

Shibo Zhao et.al, SubTMRS: SubT-MRS Dataset: Pushing SLAM Towards All-weather Environments, CVPR 2024

Questions?