Do we still need geometry for Visual Localization and Mapping?

Paul-Edouard Sarlin

50th Pattern Recognition and Computer Vision Colloquium - CVUT
2025-10-09

About me

Research scientist at Google Geo

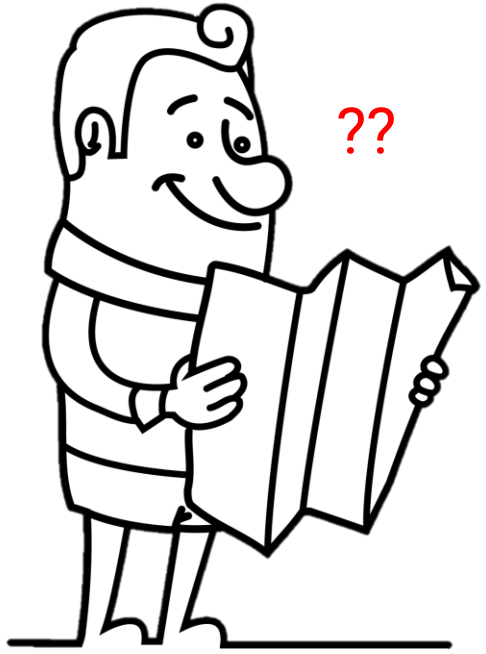
PhD at ETH Zurich 2020-2024

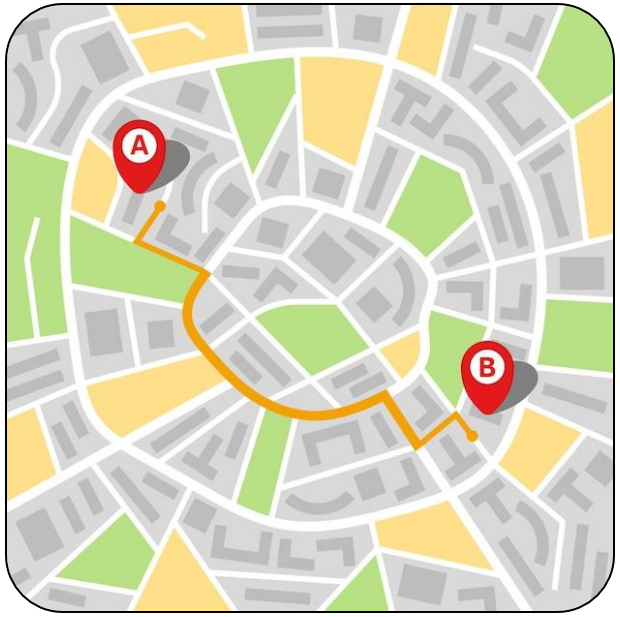
More info: psarlin.com

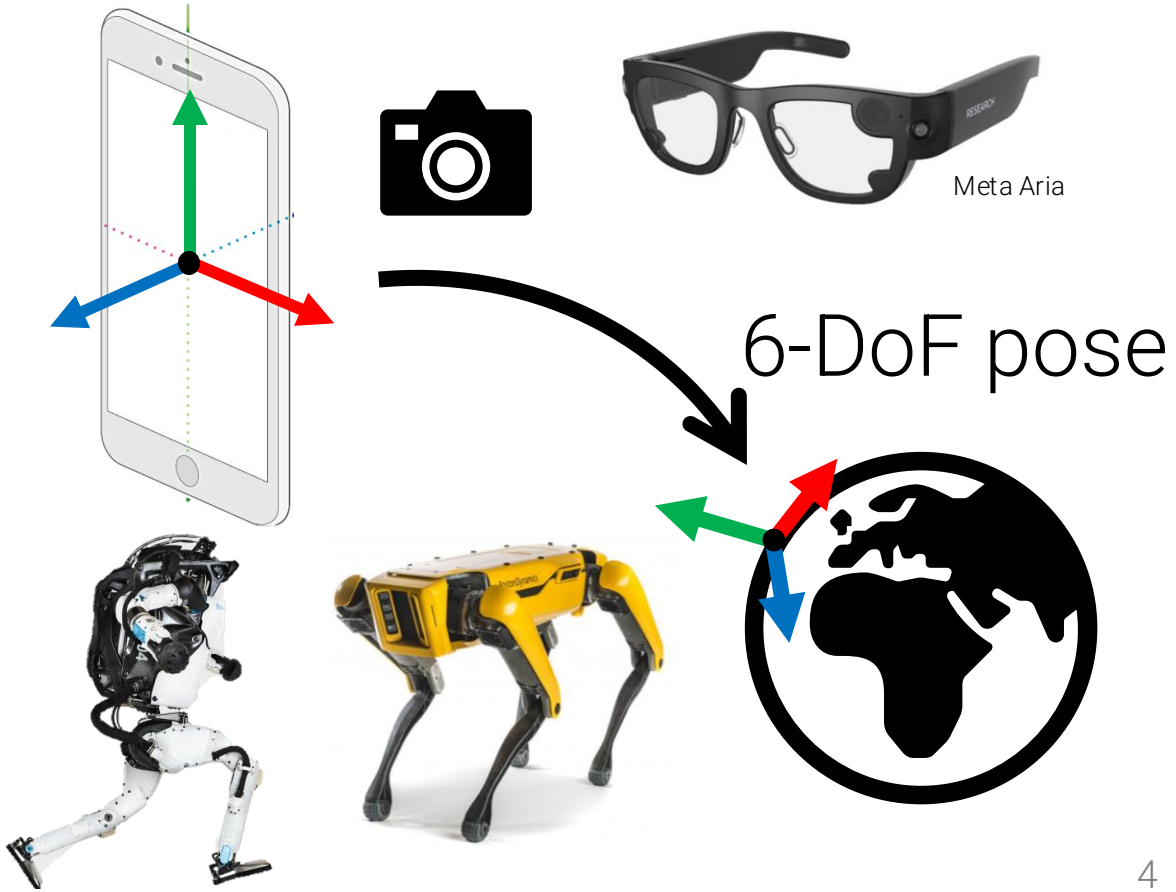
Do we still need geometry for Visual Localization and Mapping?

Visual Localization and Mapping

Finding where I am

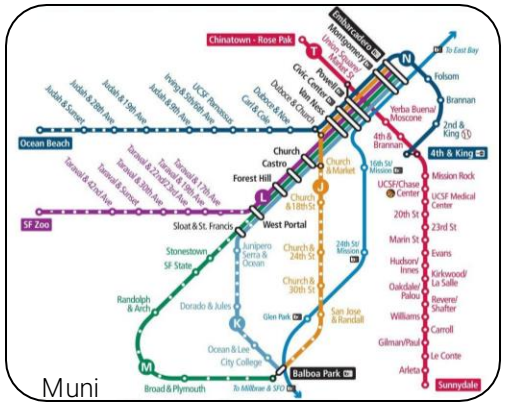


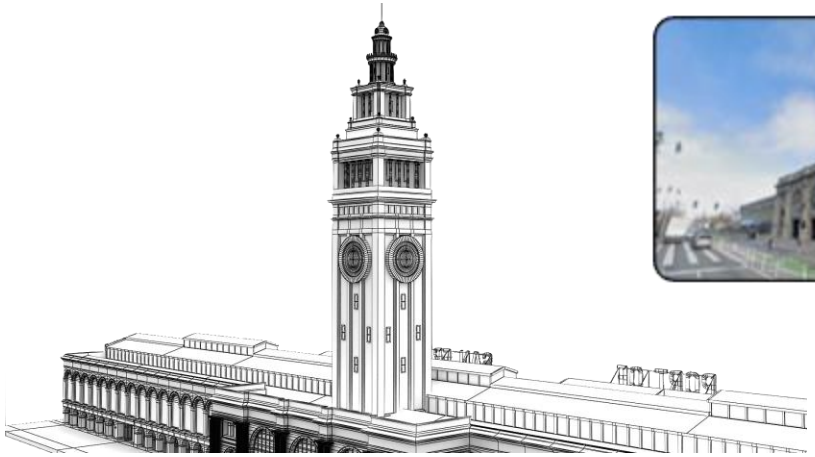


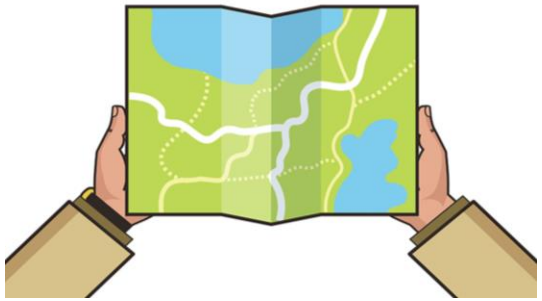


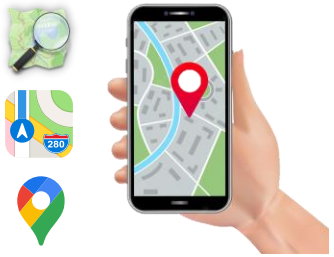
Visual Localization and Mapping

What is in the world and where









Visual Localization and Mapping

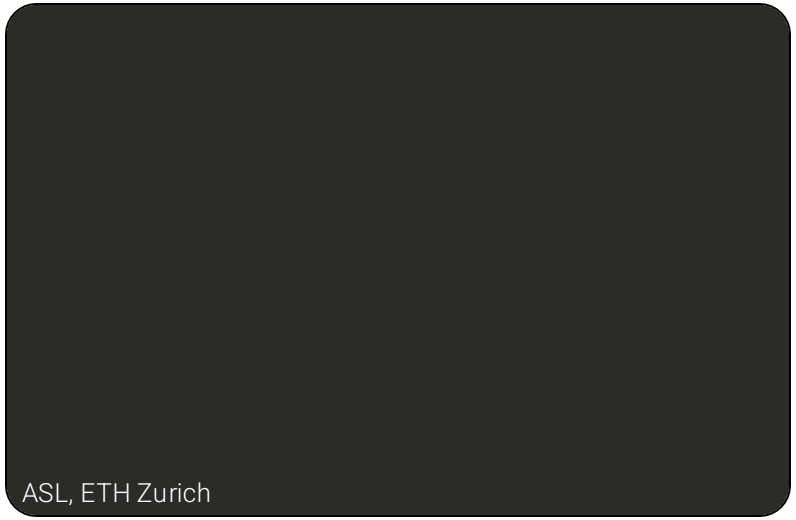
Creating maps from observations

needs the sensor pose

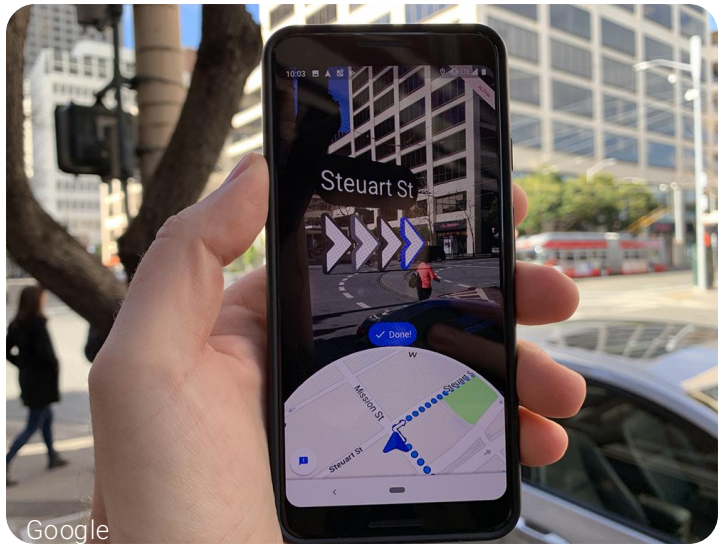
mapping

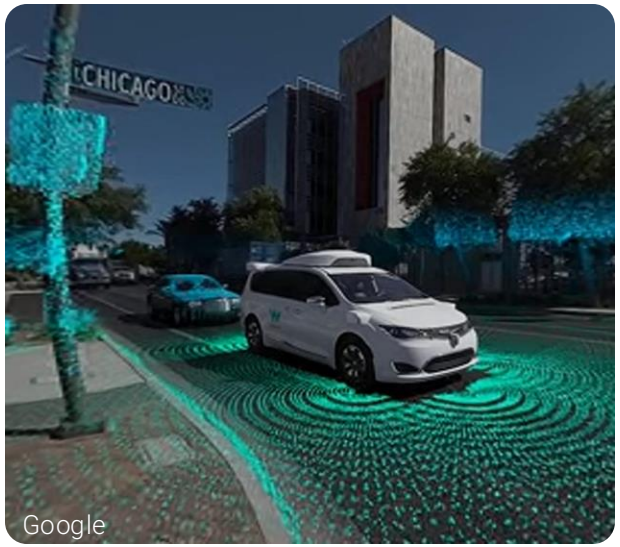
localization

needs a map

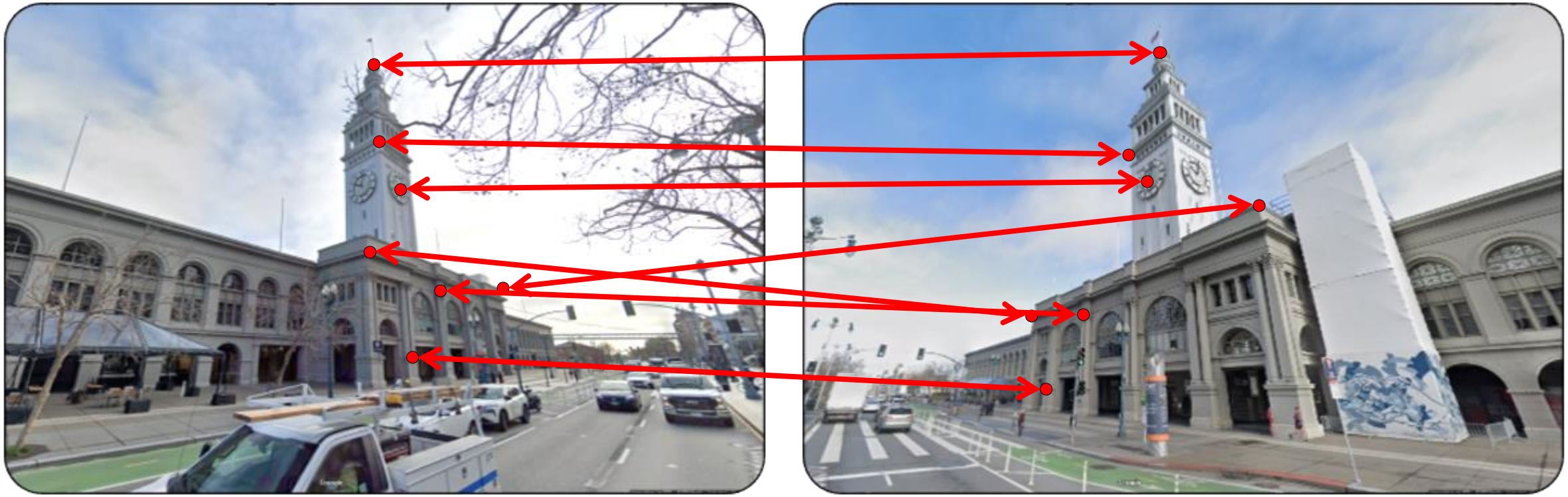


Why should we care?



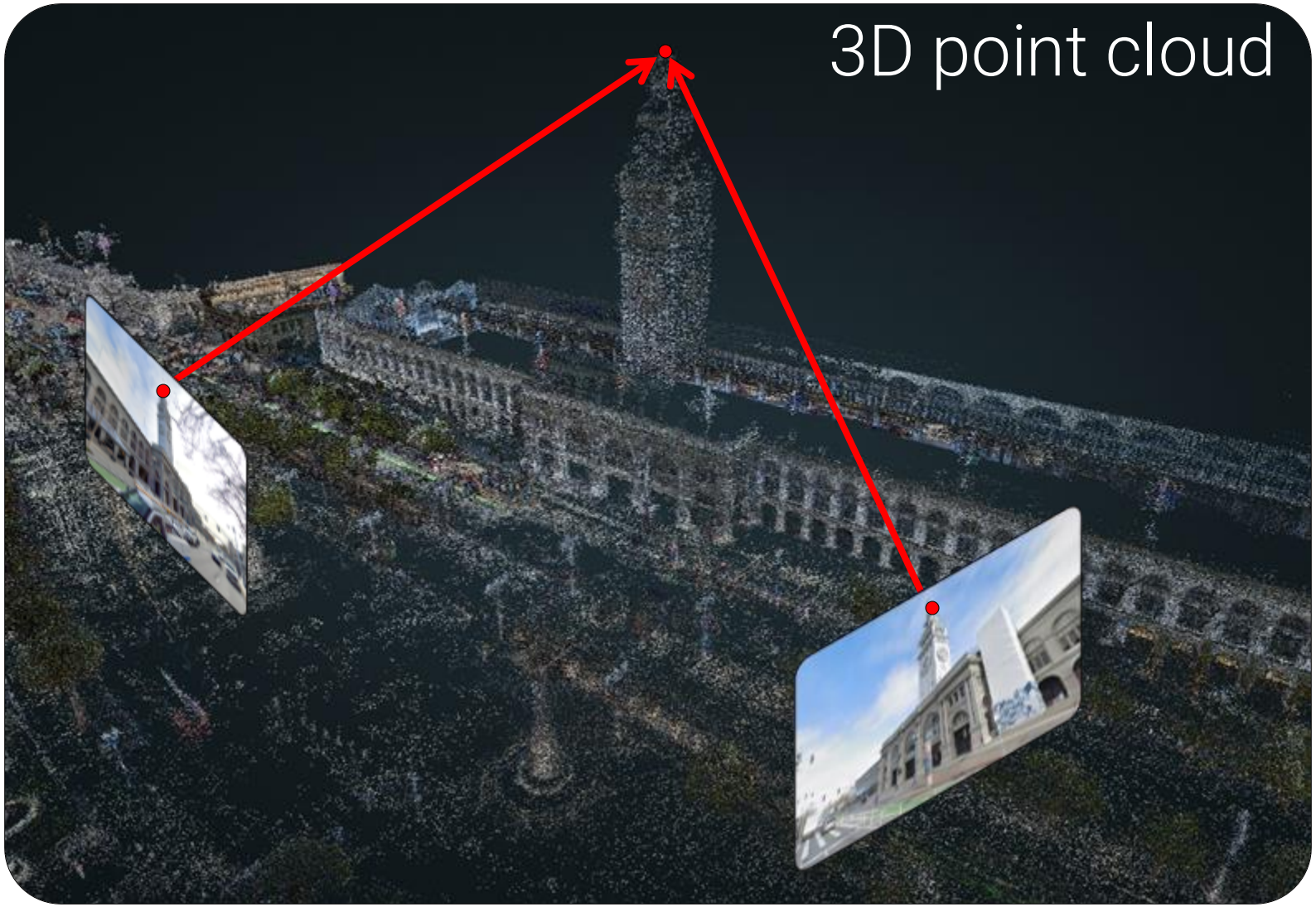


3D Geometry for Mapping



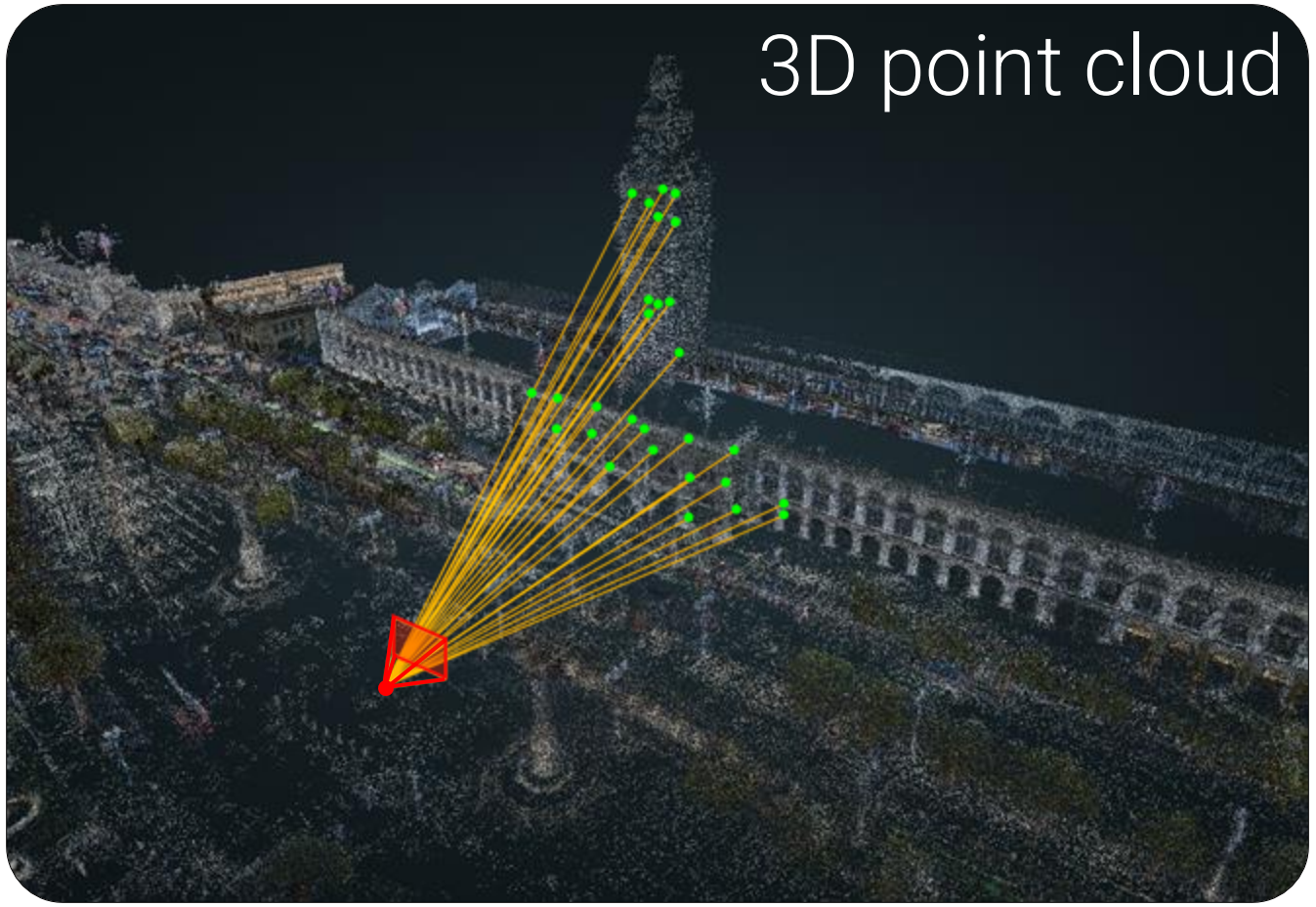
mapping images

3D Geometry for Mapping



3D Geometry for Localization

query image

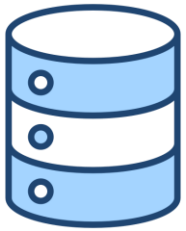


Real-world challenges

temporal changes

Google StreetView

cost & scalability



symmetries

Do we still need geometry for Visual Localization and Mapping?

SfM in 2019

Deep Learning

Deep Learning

Deep Learning

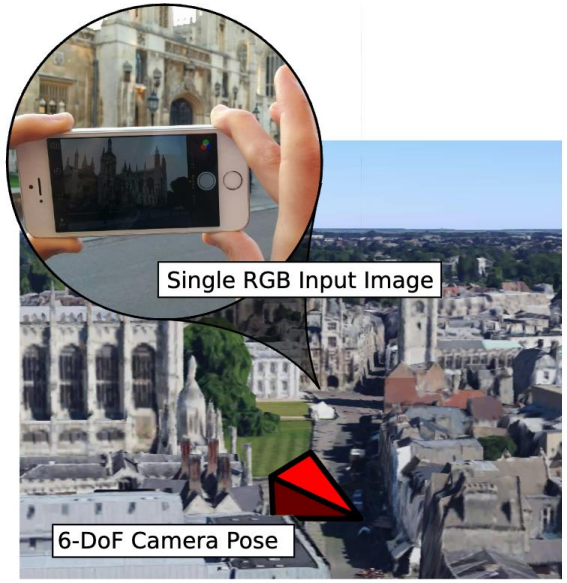
More Deep Learning

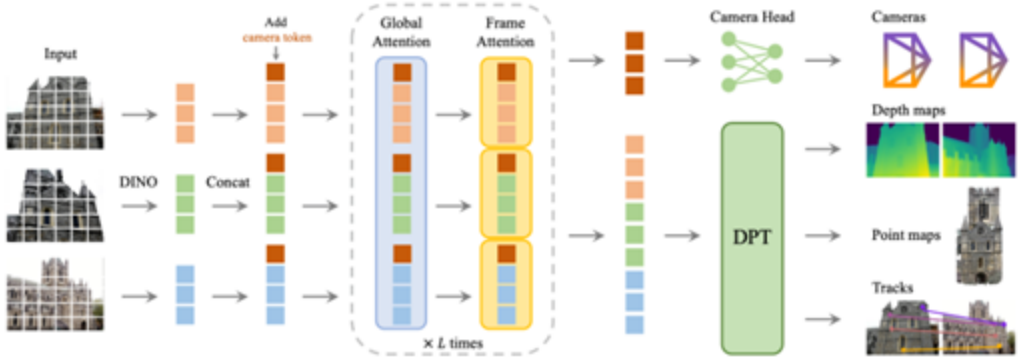
We now have end-to-end learning that works

2016 2023 2025

fast, robust, generalizes

PoseNet DUSt3R VGGT





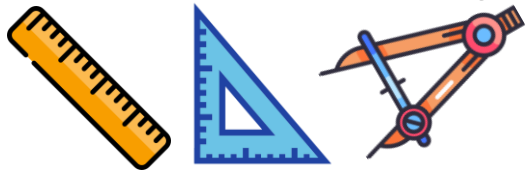
But what did we lose?

From Geometry to Deep Learning?

accurate, scalable, interpretable

mapping images

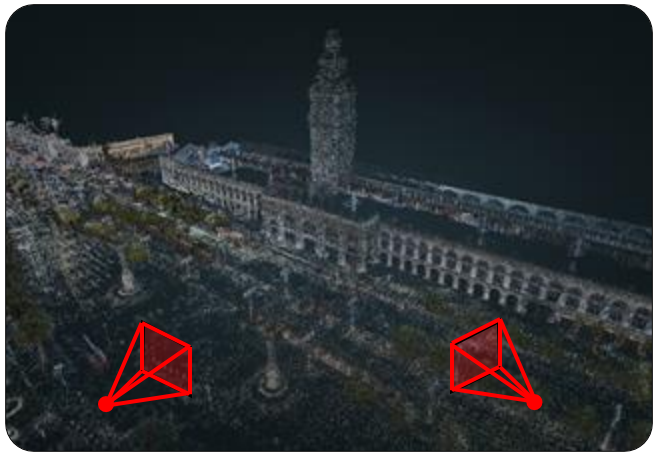
3D geometry



vs

black box deep learning

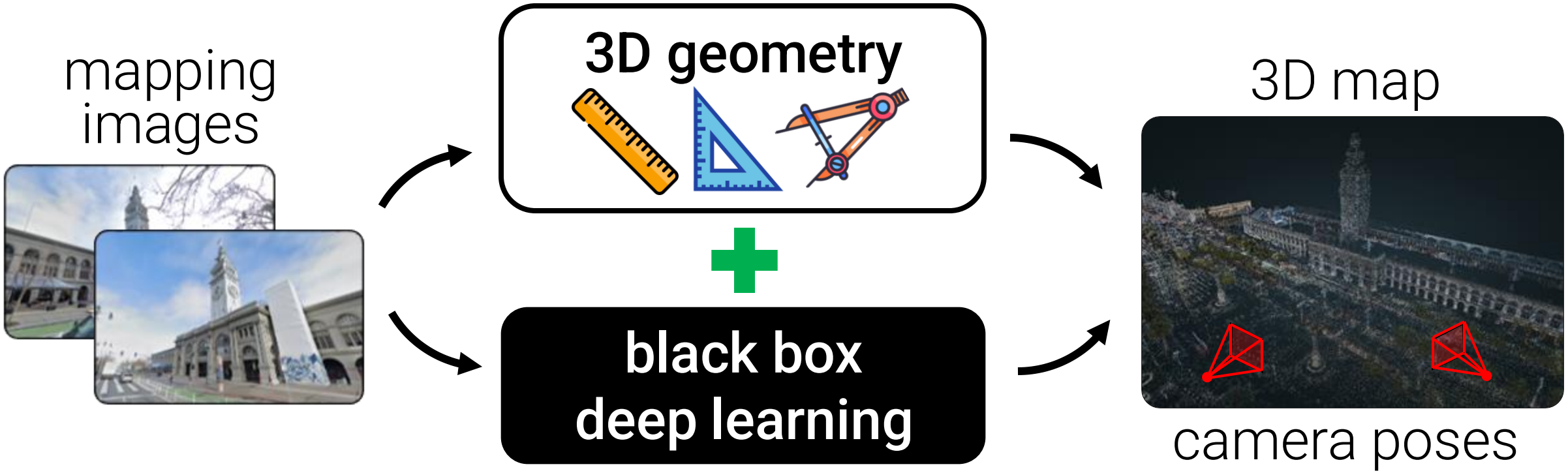
3D map



camera poses

none of the above! limited to <<1k views

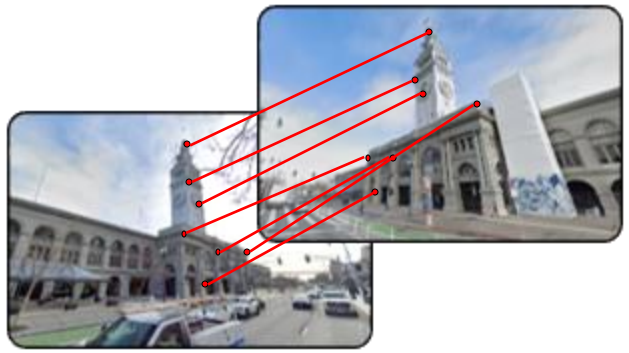
From Geometry to Deep Learning?



Do we have to choose one?

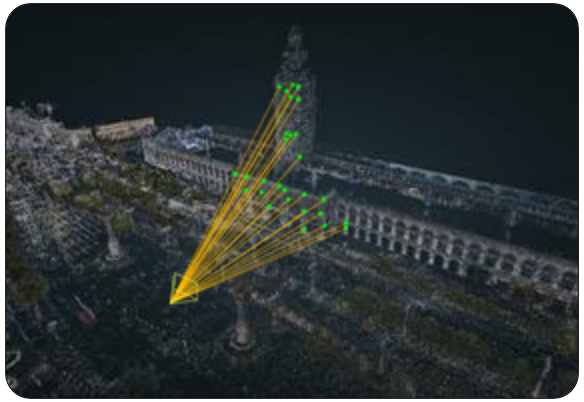
One step at a time

image matching



SuperGlue [CVPR'20]
LightGlue [ICCV'23]

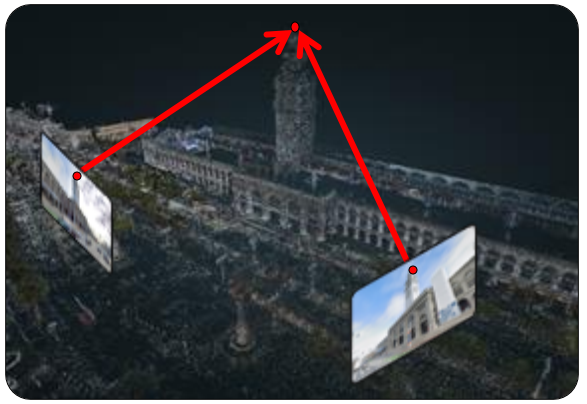
pose estimation



BACK TO THE FEATURE

PixLoc [CVPR'21]

bundle adjustment



PixSfM [ICCV'21]

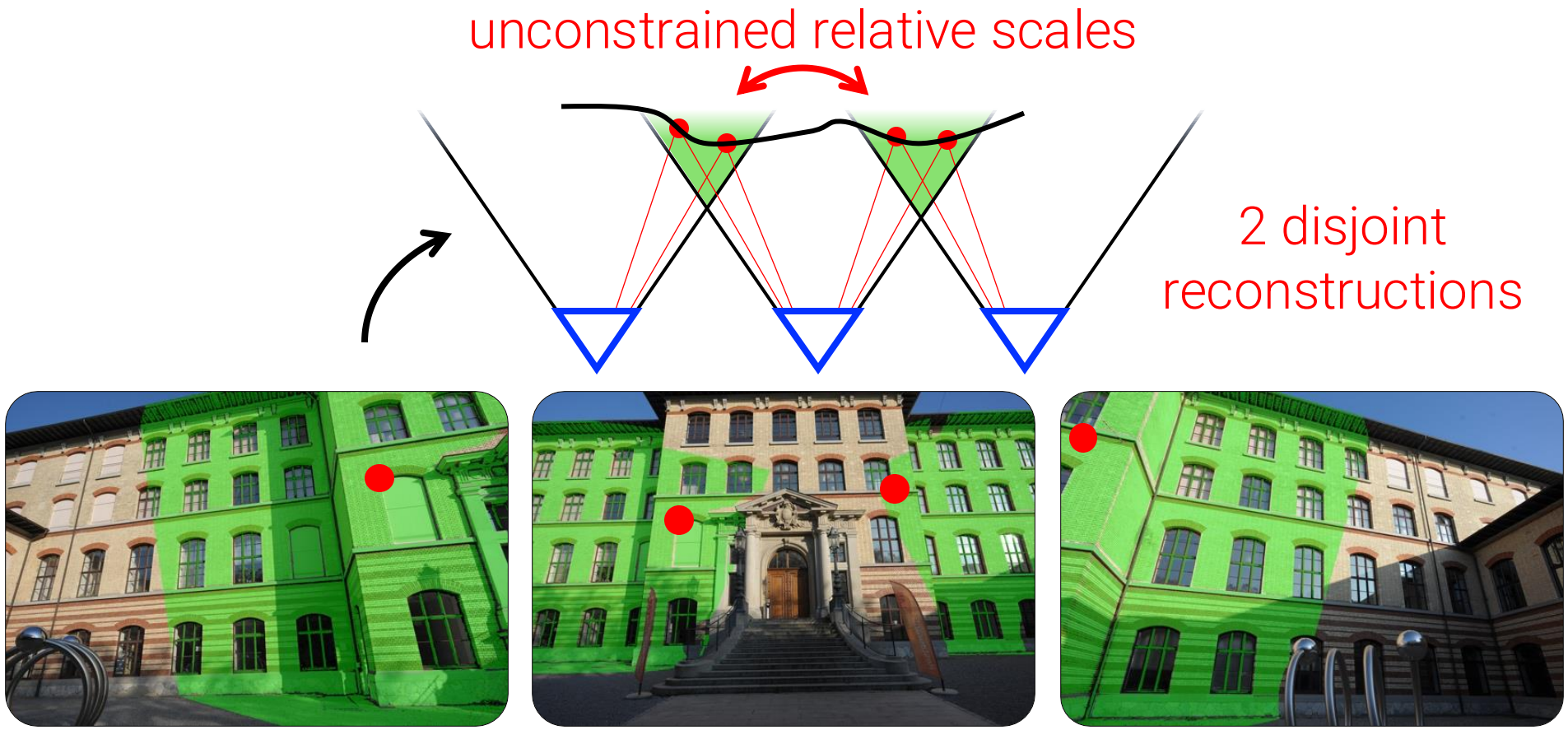
calibration

GeoCalib [ECCV'24]

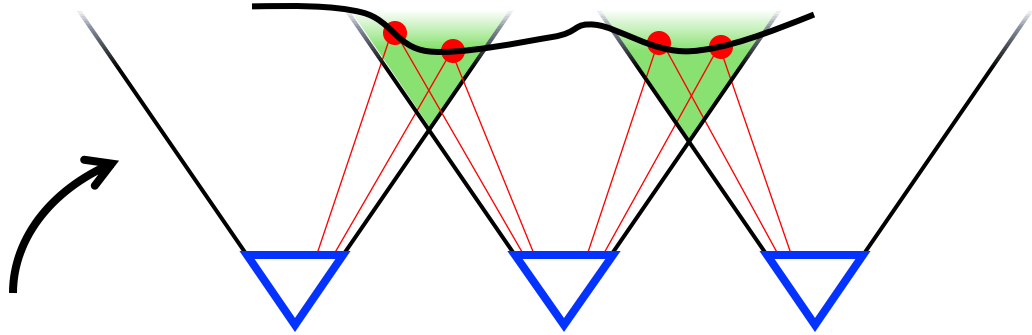
A simple recipe

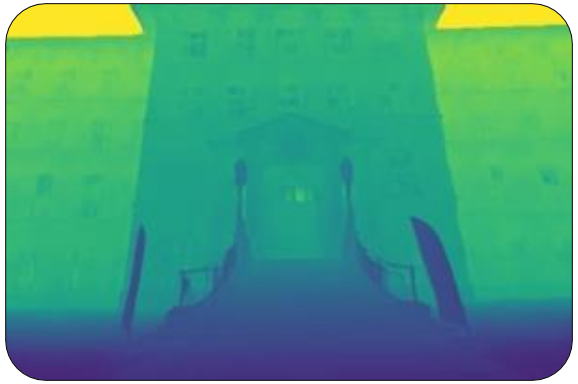
1. Identify **structural weaknesses** in geometry
   - unconstrained? unobservable? noisy inputs?
2. Think ML: what **priors** do we need? Inferred from what **data**?
3. What **geometry** shouldn't be learned? Camera models!
4. Bake it into an **optimization problem** with learned data terms
   - Learned priors as data terms
   - Flexible constraints
   - Predictive uncertainty

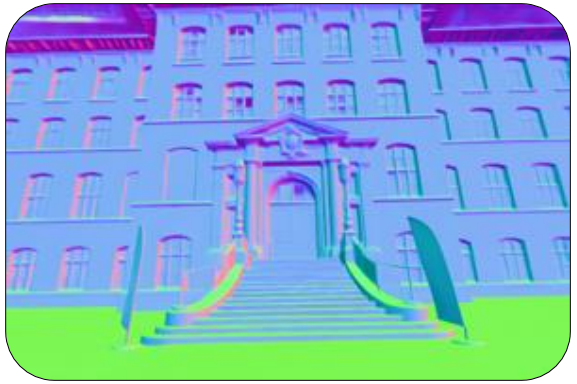
Example: three-view overlap



Example: three-view overlap

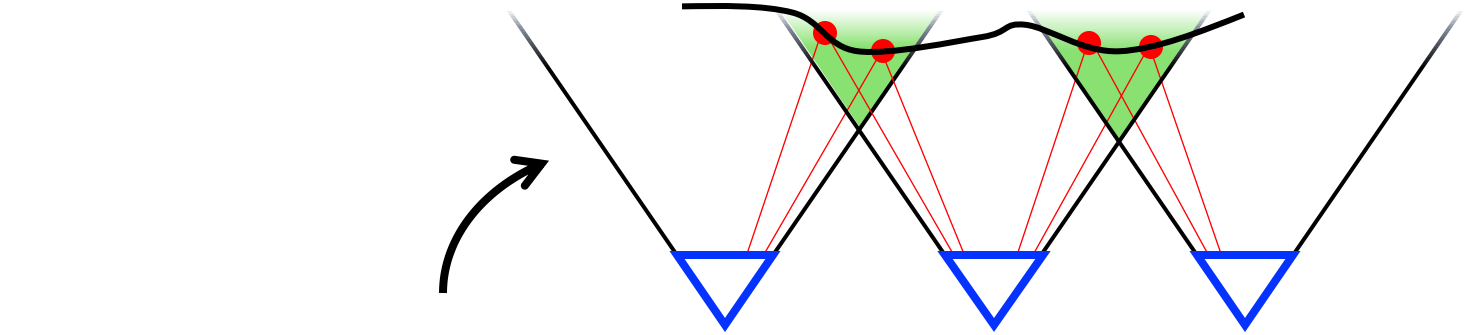


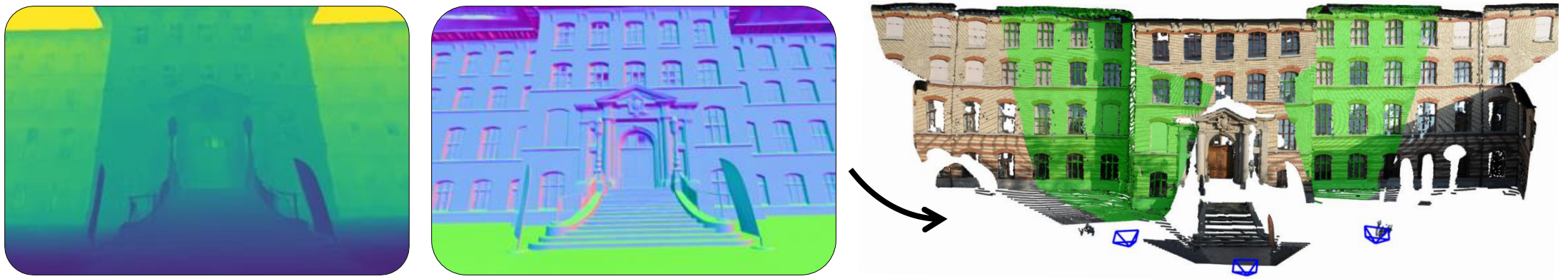




Monocular surface priors
depth & normals

Example: three-view overlap





ETH zürich [1] Google [2] Microsoft [3]

MP-SfM
Monocular Surface Priors for Robust Structure-from-Motion

CVPR 2025

Zador Pataki[1]

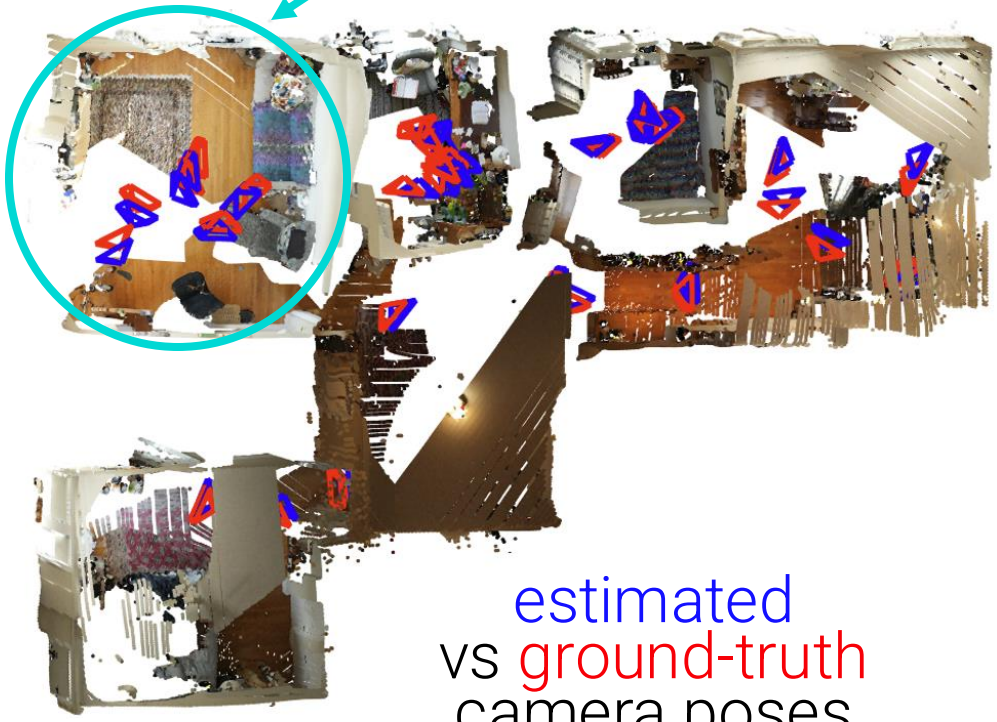
Paul-Edouard Sarlin[2]

Johannes Schönberger[1,3]

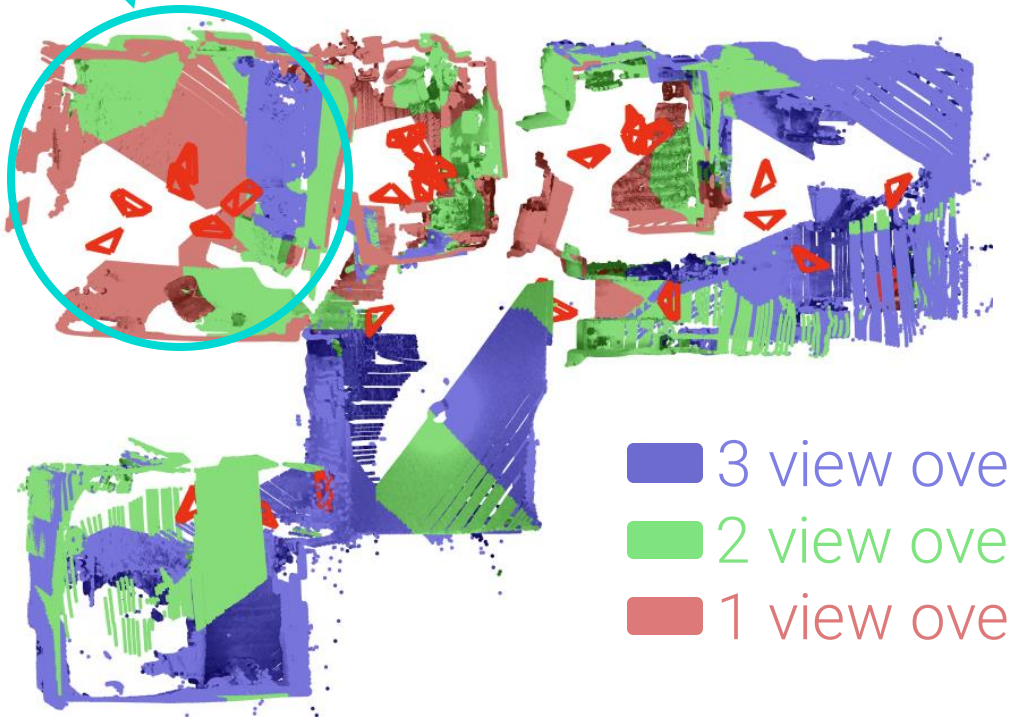
Marc Pollefeys[1,3]

github.com/cvg/mpsfm

Common scenario
for non-expert users

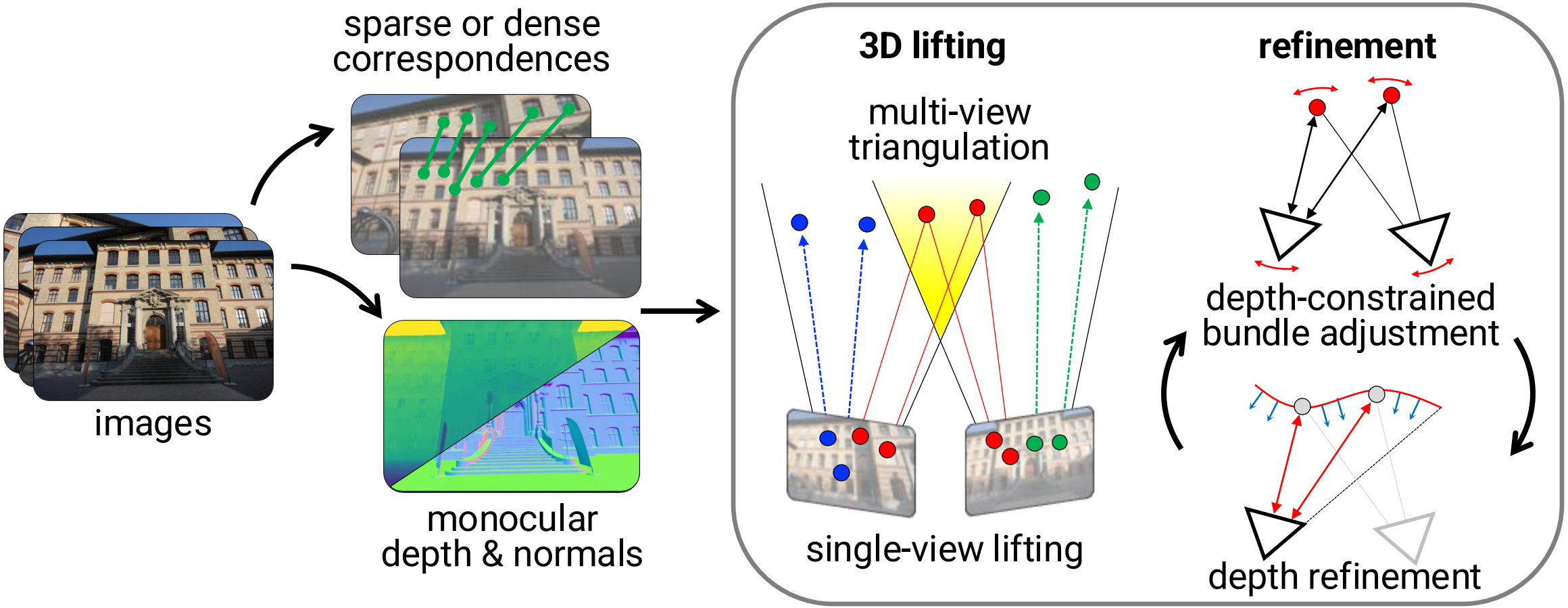


estimated
vs ground-truth
camera poses

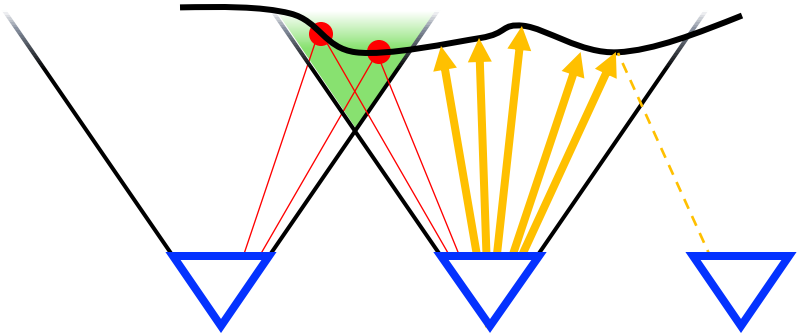


3 view overlap
2 view overlap
1 view overlap

How it works


sparse or dense
correspondences
images
monocular
depth & normals
3D lifting
multi-view
triangulation
single-view lifting
refinement
depth-constrained
bundle adjustment
depth refinement

Single-view lifting with depth

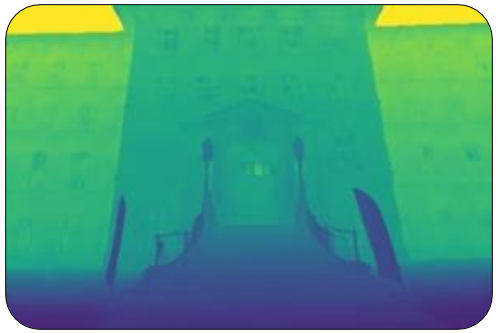


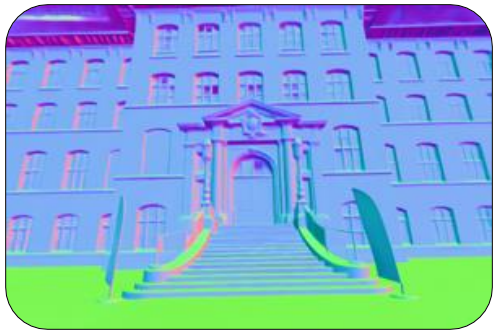
Complement regular multi-view 3D points with single-view tracks from depth

Bundle adjustment with priors

- Naïve: constrain the depth of 3D points
- But: monocular depth is rarely accurate!
- Surface normals are easier to predict

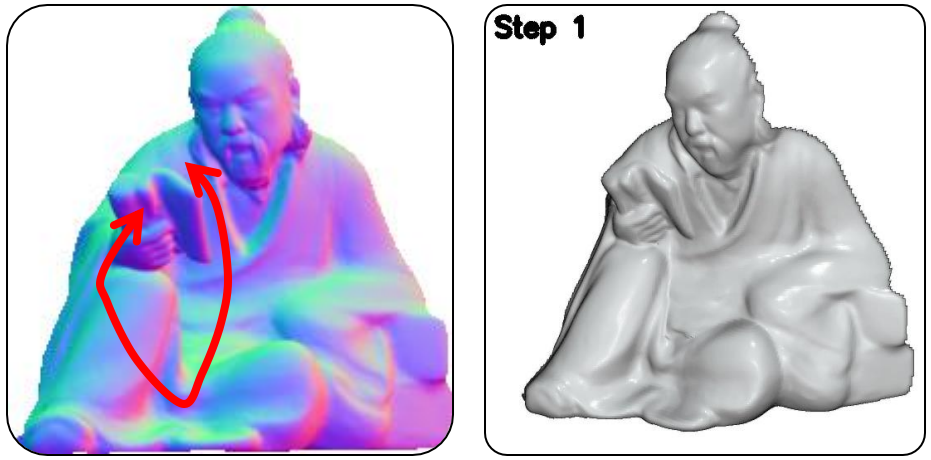
mono
vs GT depth



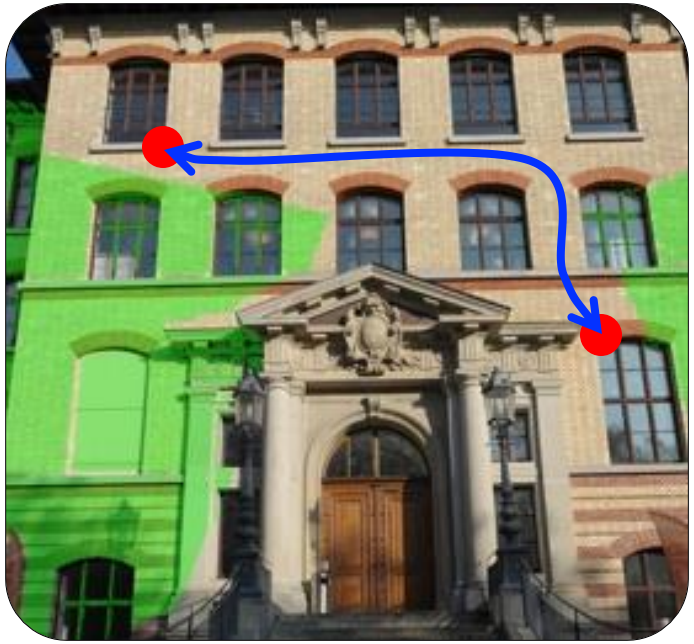


Depth refinement as normal integration

- Constrain 3D points that are on the same surface
- Optimize a dense depth map with NLS
- Estimate the depth discontinuity with IRLS

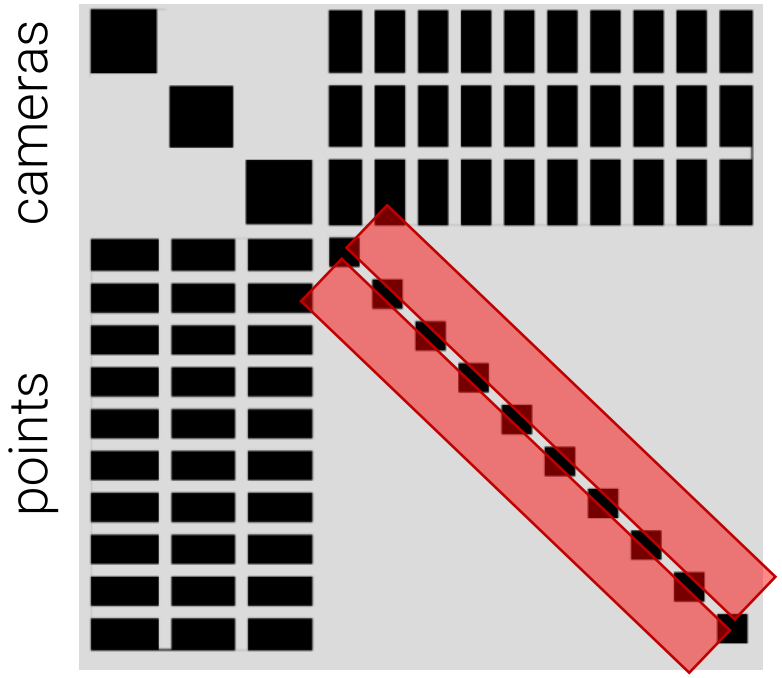


Bilateral Normal Integration, Cao et al., ECCV 2022

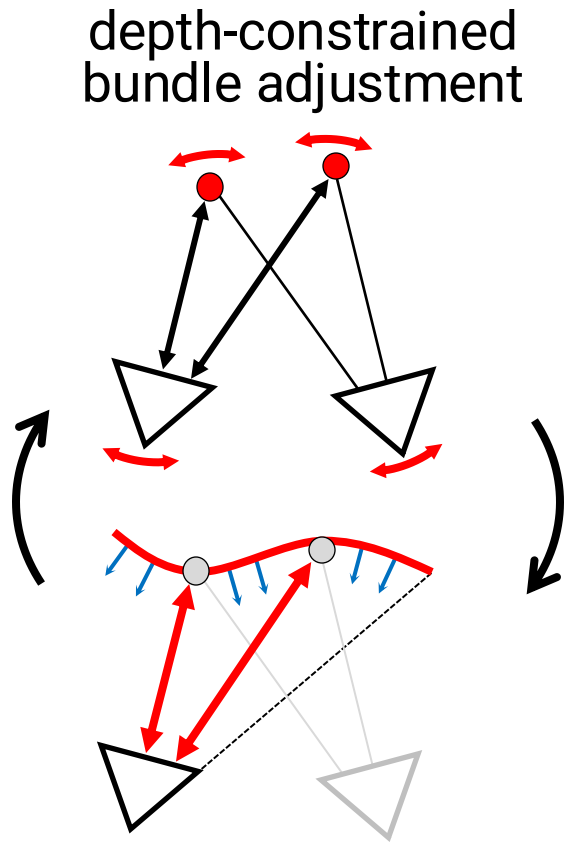


Depth refinement as normal integration

point-to-point constraints break the Schur Complement



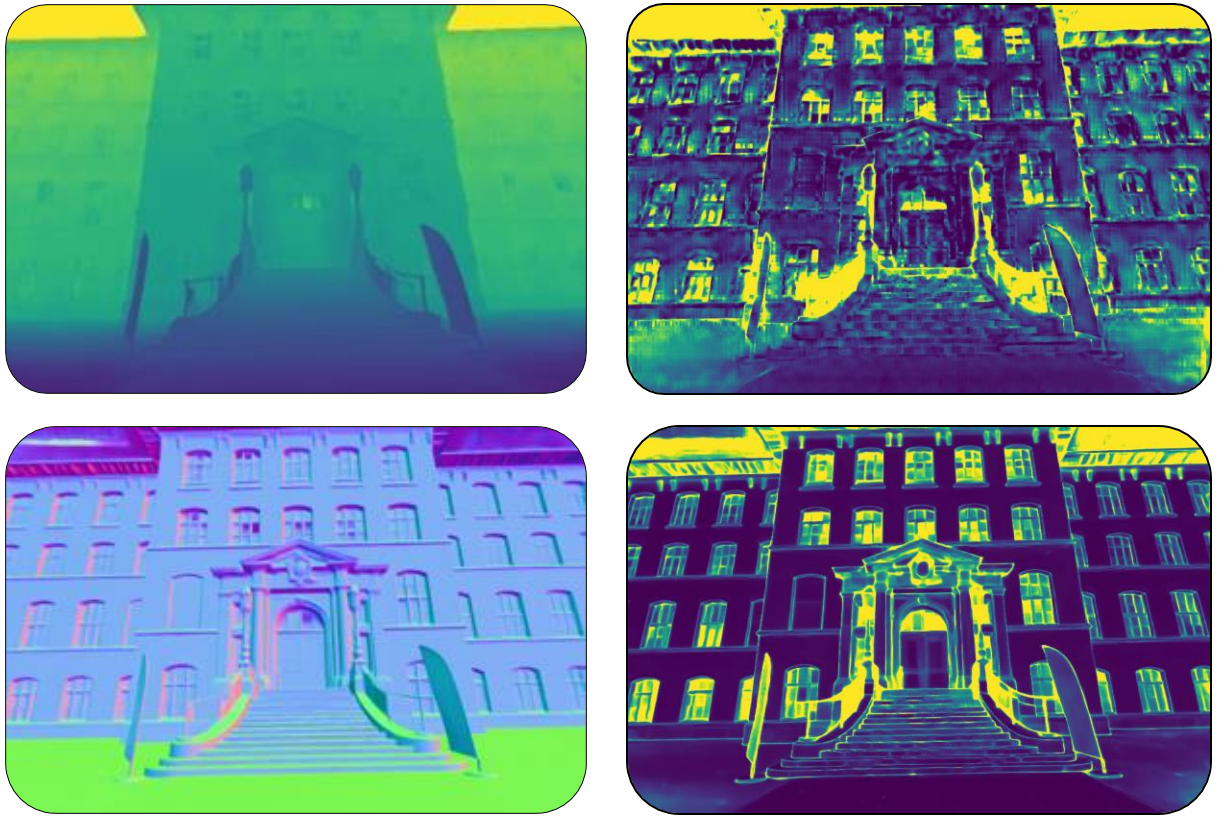
Solution: alternate optimization

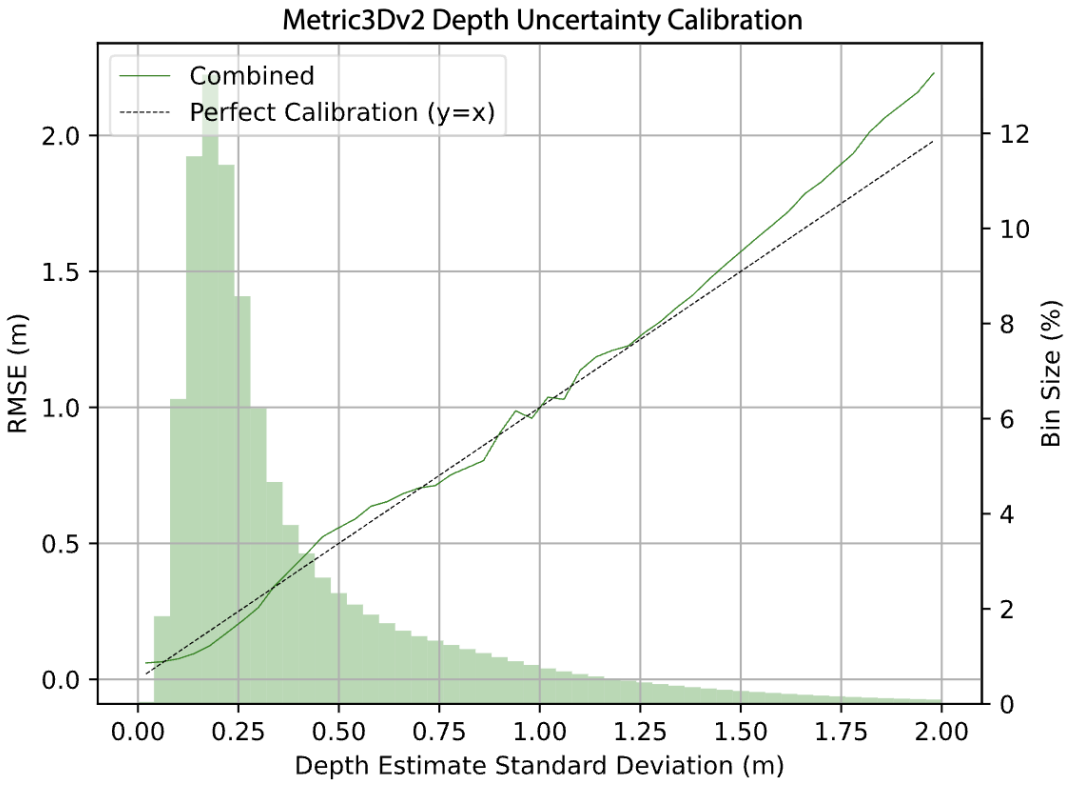


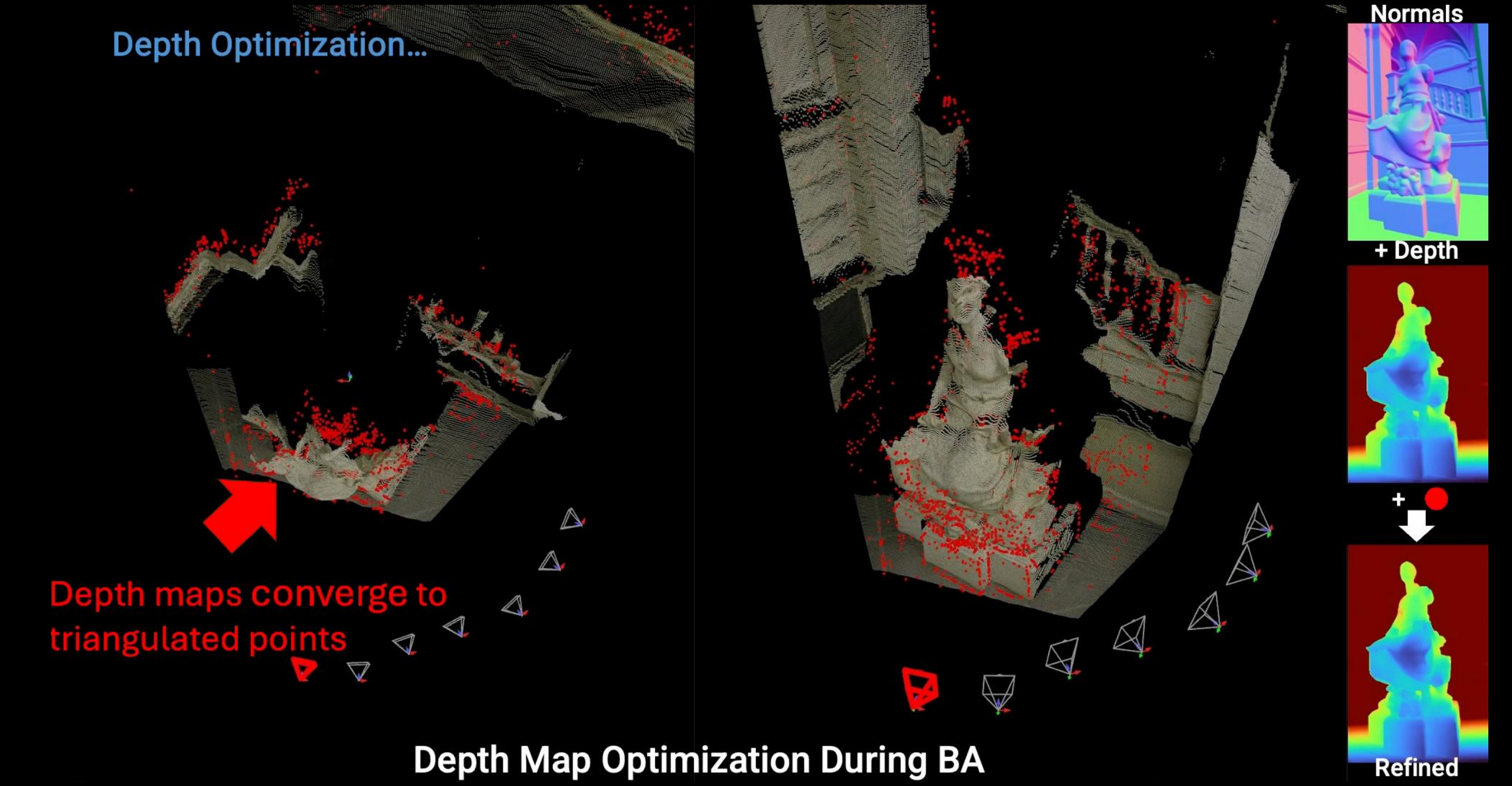
freeze
dense depth

freeze
camera poses
and 3D points

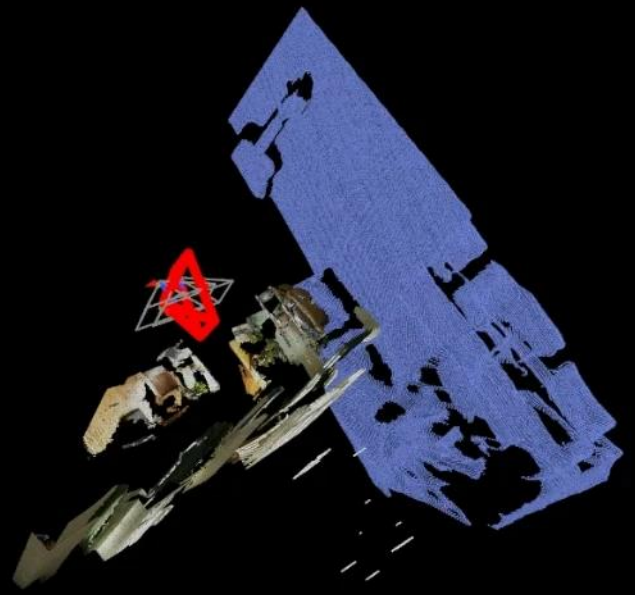
Joint optimization requires calibrated uncertainties







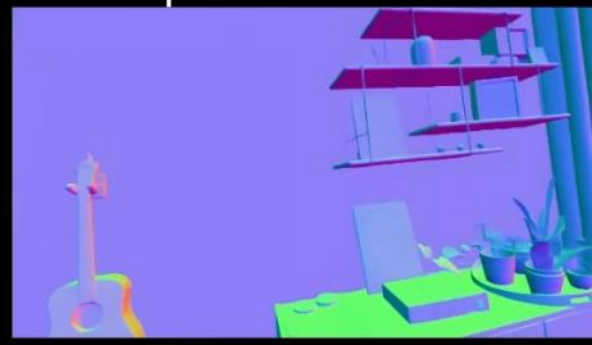
Depth Map Optimization During BA



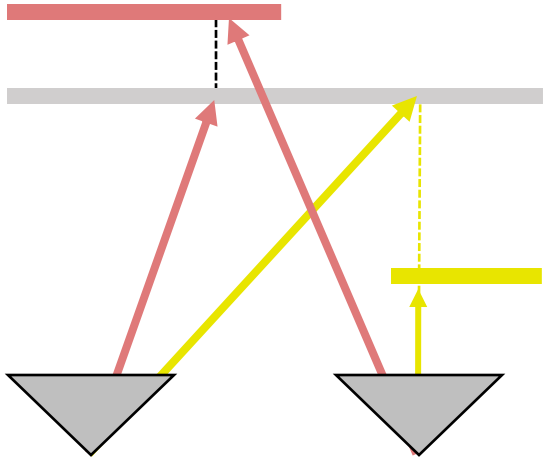
Latest Image

Input Depth

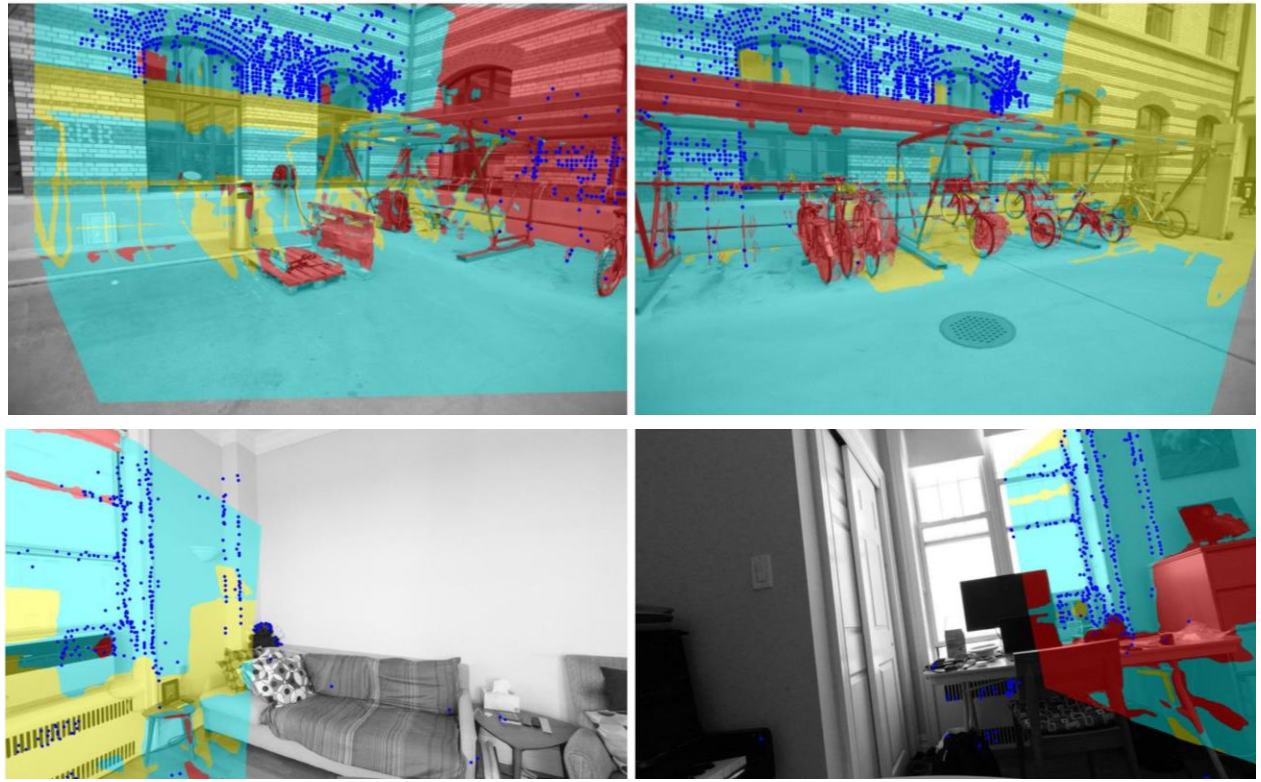
Input Normals



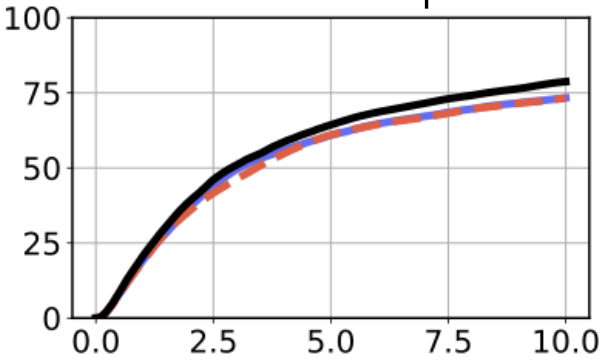
Symmetry detection with depth consistency



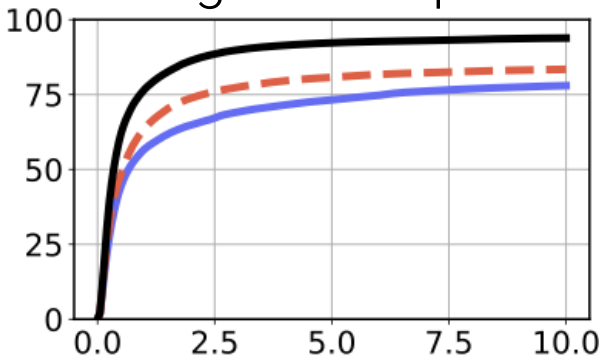
consistent occlusion conflict



low-overlap



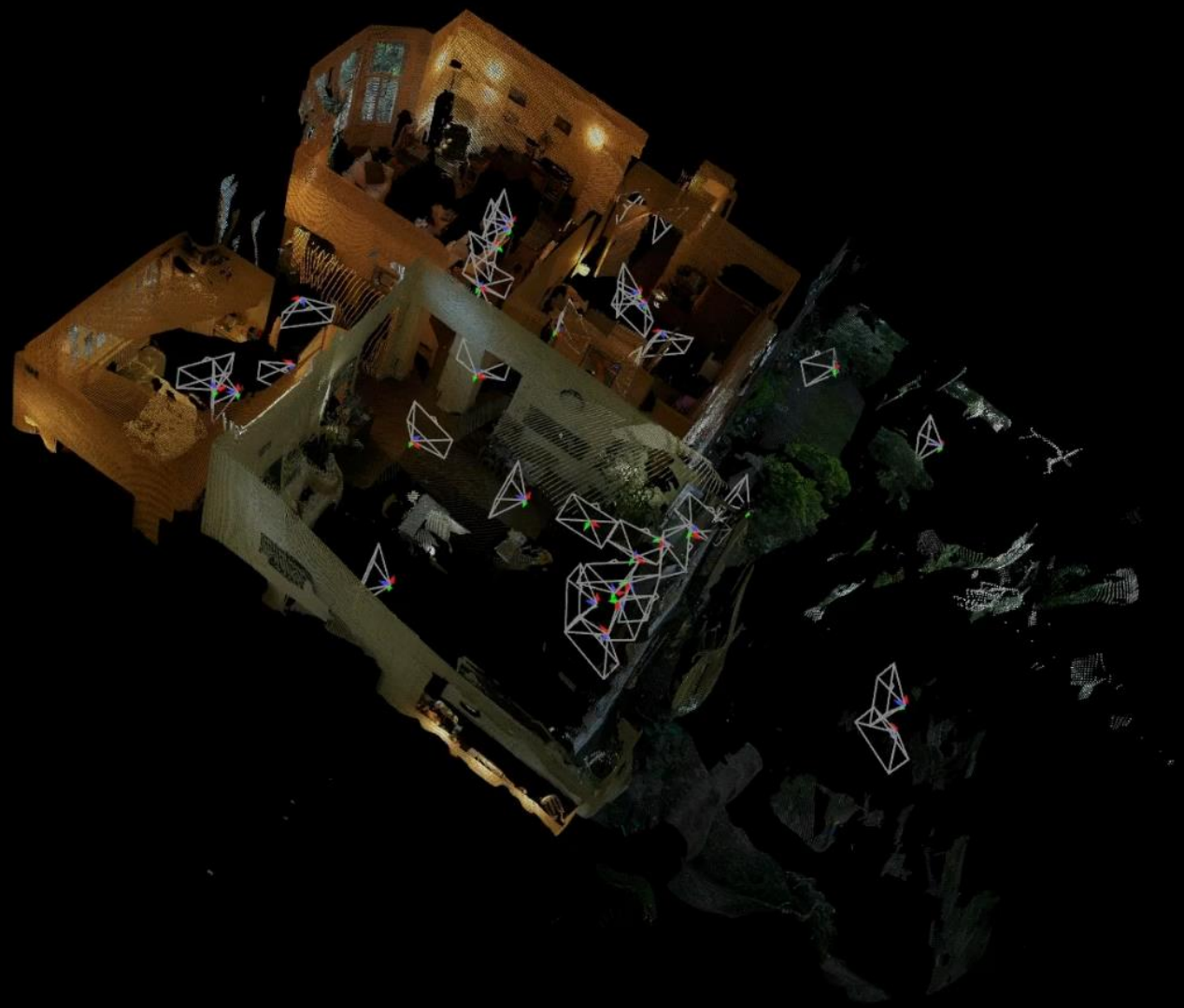
high-overlap



relative pose error [deg]

vanilla / with Doppelganger / with depth

MP-SfM (ours)



So… do we still need geometry?

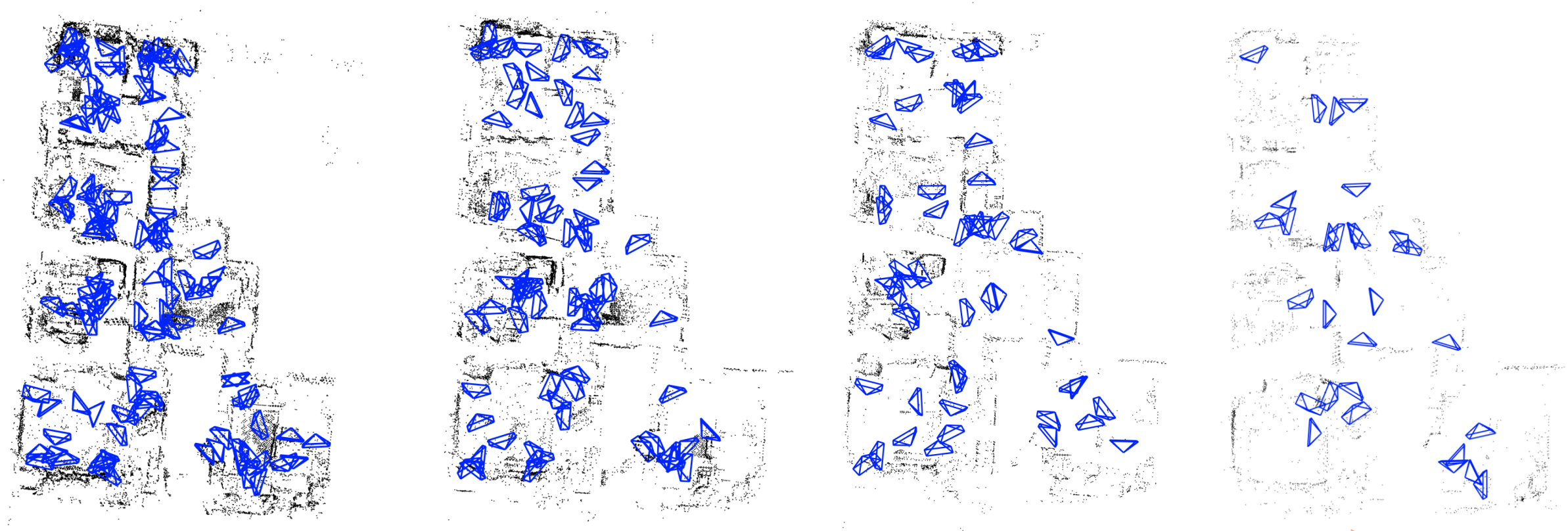
YES!

- Two/multi-view models learn strong 3D priors
- But we already have this from off-the-shelf models
- Use geometry to glue them together!
- Open question: can we learn depth/normals end-to-end in SfM?
- Grand vision: a self-tuning system, reinforced with geometry

  “Self-supervised VGGT” with hard geometric constraints

Beware of artificial evaluations!

1. Take nice, slow videos with a lot of parallax
2. Then extract images at regular interval



level of overlap

Much of real data is very different

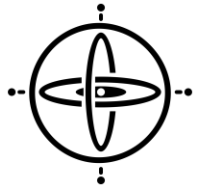
- Example: egoentric data from wearable devices
- Opportunistic capture, byproduct of the user's motion

Challenges & opportunities

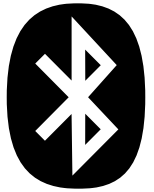
- Unconstrained motion, moving environments
- Appearance: low light / exposure changes
- Long up-time → time-varying calibration

But

- Non-image sensors: IMU, GPS, microphones, WiFi/BT
- High-frequency (>60 FPS)
- Dense coverage (crowd-sourced)



inertial units



Bluetooth

WiFi

Benchmarking Egocentric Visual-Inertial SLAM at City Scale

Anusha Krishnan[1*] Shaohui Liu[1*] Paul-Edouard Sarlin[2*] Oscar Gentilhomme[1]
David Caruso[3] Maurizio Monge[3] Richard Newcombe[3] Jakob Engel[3] Marc Pollefeys[1,4]
[1]ETH Zurich [2]Google [3]Meta Reality Labs Research [4]Microsoft Spatial AI Lab

ICCV OCT 19-23, 2025 HONOLULU HAWAII

Computer Vision and Geometry Lab

ETH zürich

Meta Microsoft Google

lamaria.ethz.ch

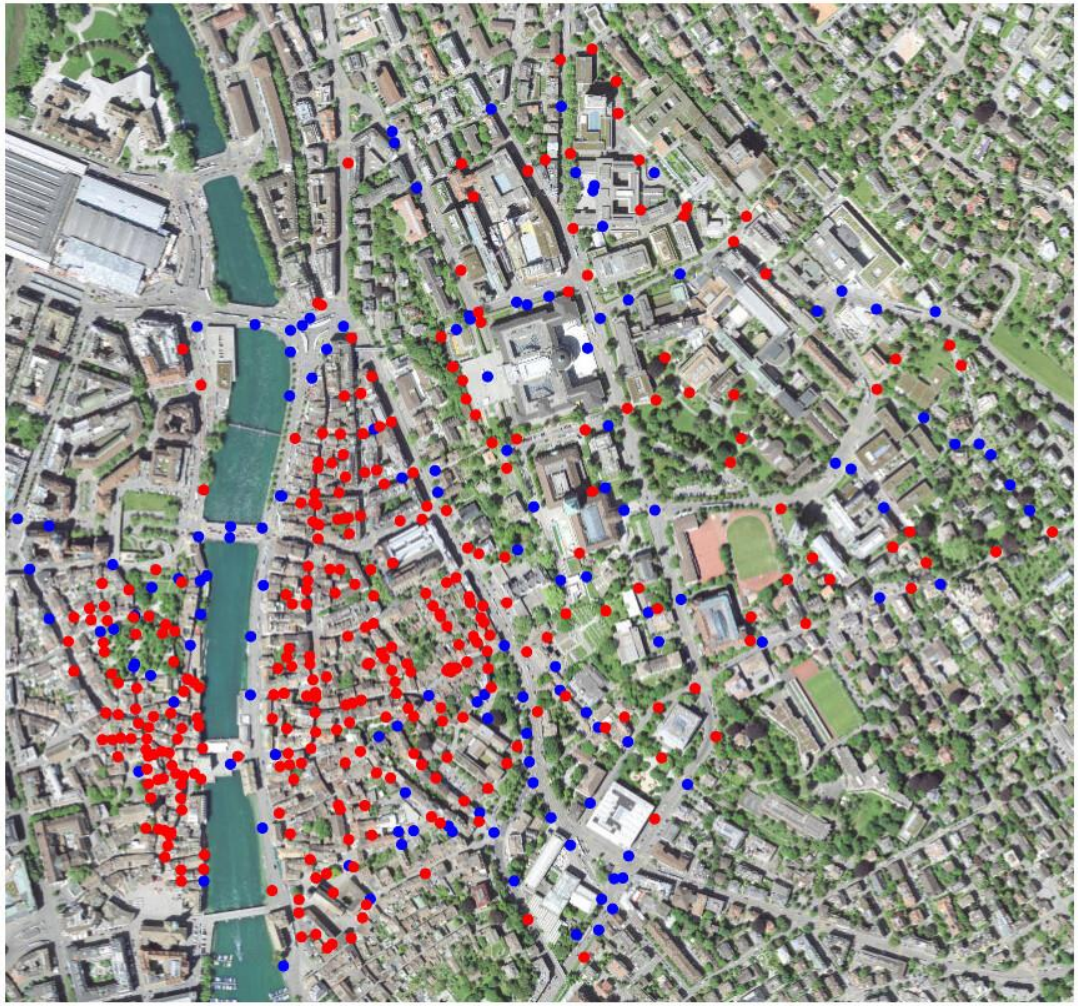
The LaMAria dataset

70km
22 hours
10-45min each

Good for multi-sensor SLAM and monocular SfM

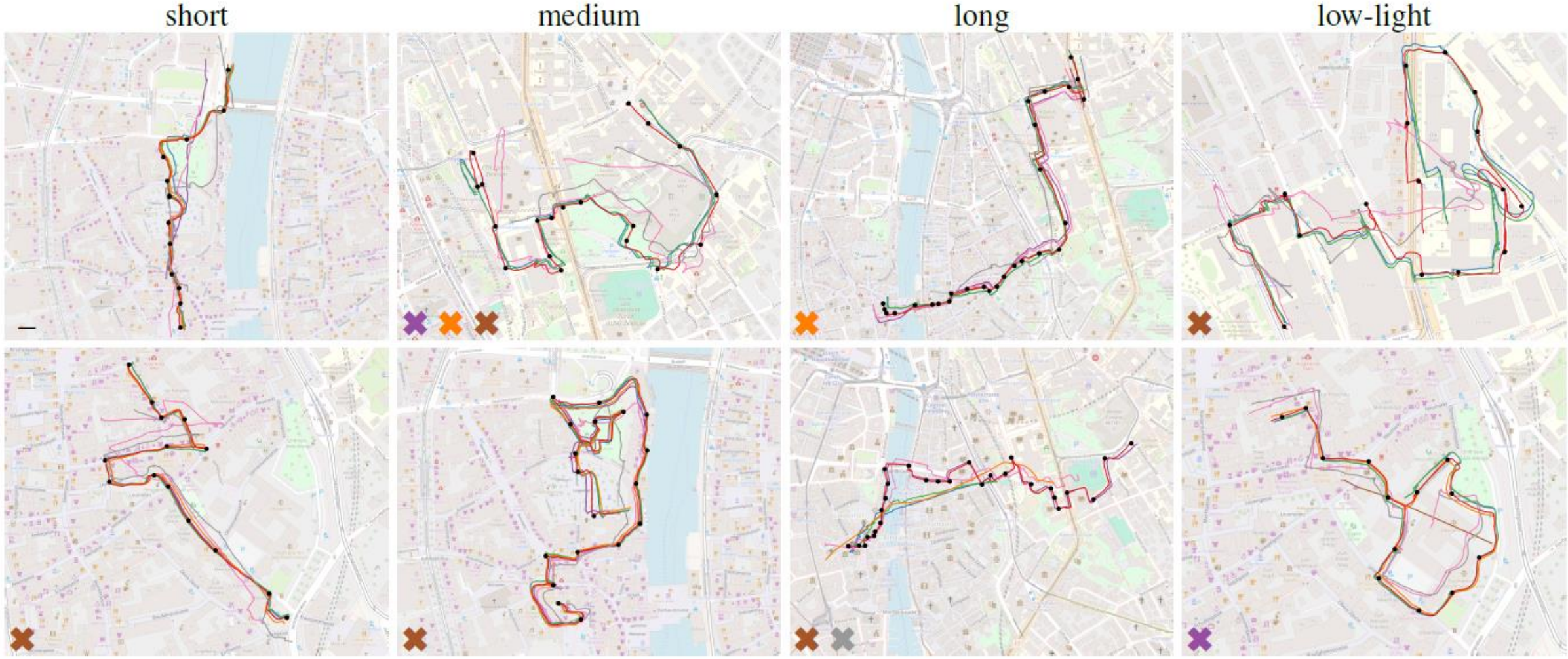
The LaMAria dataset

Control points with cm accuracy
Measure 0.1% metric scale drift



Monocular SLAM only works on easier data

Open-source systems are far behind industry



OpenVINS • OpenVINS+Maplab • ORB-SLAM3 • OKVIS2 •
Kimera VIO • DPVO • DPV-SLAM • Aria's SLAM •

Open-source systems are far behind industry

- monocular
- monocular inertial
- multi-camera inertial

method	causal	short			medium			long			challenge – low-light			challenge – moving platform		
		score ↑	CP@1m ↑	R@5m ↑	score ↑	CP@1m ↑	R@5m ↑	score ↑	CP@1m ↑	R@5m ↑	score ↑	CP@1m ↑	R@5m ↑	score ↑	CP@1m ↑	R@5m ↑
DPVO	✓	9.4	1.7	21.3	5.2	1.0	10.8	1.2	0.0	1.9	3.4	0.2	7.5	2.4	0.1	–
DPV-SLAM	✓	7.2	1.4	14.5	5.2	1.4	10.0	0.4	0.0	0.6	1.9	0.4	3.5	1.7	0.0	–
Kimera VIO	✓	6.3	2.9	12.6	6.6	1.7	15.1	6.3	1.7	14.3	4.2	2.7	6.4	7.1	1.6	–
ORB-SLAM3	x	28.3	13.4	67.1	20.3	4.4	57.0	14.2	2.3	40.6	6.2	0.6	12.5	15.7	4.1	–
OpenVINS	✓	18.1	4.4	45.7	10.9	2.3	27.9	4.7	0.5	12.3	7.9	2.4	17.6	2.4	0.6	–
OpenVINS + Maplab	x	22.9	8.1	50.8	13.1	4.1	29.0	5.8	1.3	13.3	9.6	2.9	19.3	3.7	1.2	–
OpenVINS	✓	22.2	6.2	57.9	17.8	5.7	46.1	10.6	1.7	25.8	16.9	6.2	38.2	11.5	2.4	–
OpenVINS + Maplab	x	26.0	9.5	61.1	21.3	7.3	50.6	12.6	1.9	30.3	16.5	4.6	37.9	13.0	3.0	–
OKVIS2	✓	24.3	12.0	54.8	13.6	6.8	28.2	2.3	2.5	1.7	15.4	5.3	38.6	4.1	2.8	–
Aria's SLAM	x	90.7	99.2	–	78.5	87.4	–	70.8	75.9	–	84.2	91.6	–	53.6	51.2	–

Aria's MPS SLAM (close-sourced) is far ahead of all current academic solutions.

Moving the goalpost for pose Transformers

- *How to maintain scale consistency over million of frames?*
- *Efficient joint inference over multiple sequences?*
- *How to use sensors like IMU in e2e models?*
- Ubiquitous symmetry indoors
- GT is orders of magnitude more accurate
- Plenty of extra sequences for SSL

Conclusion

- Yes, e2e models are **impressive**!
- But e2e is **not a requirement to leverage 3D priors**
- And e2e is sometimes a **liability** – e.g. sensor fusion is harder
- Open question

How to retain the benefits of geometry
while learning even higher-level priors e2e?
accurate, scalable, interpretable, flexible w.r.t. sensors

Thank you!

psarlin.com