Deep Visual SLAM Frontends: SuperPoint, SuperGlue, and SuperMaps

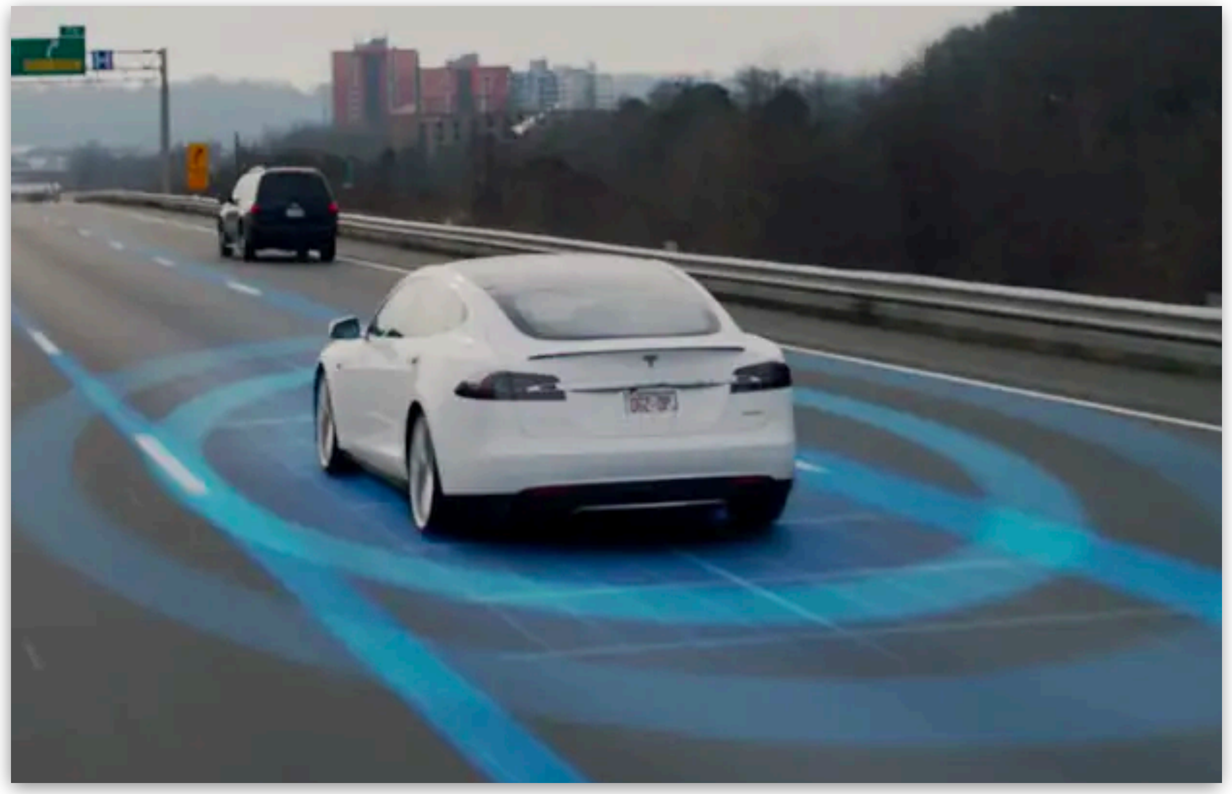
Tomasz Malisiewicz
June 14, 2020

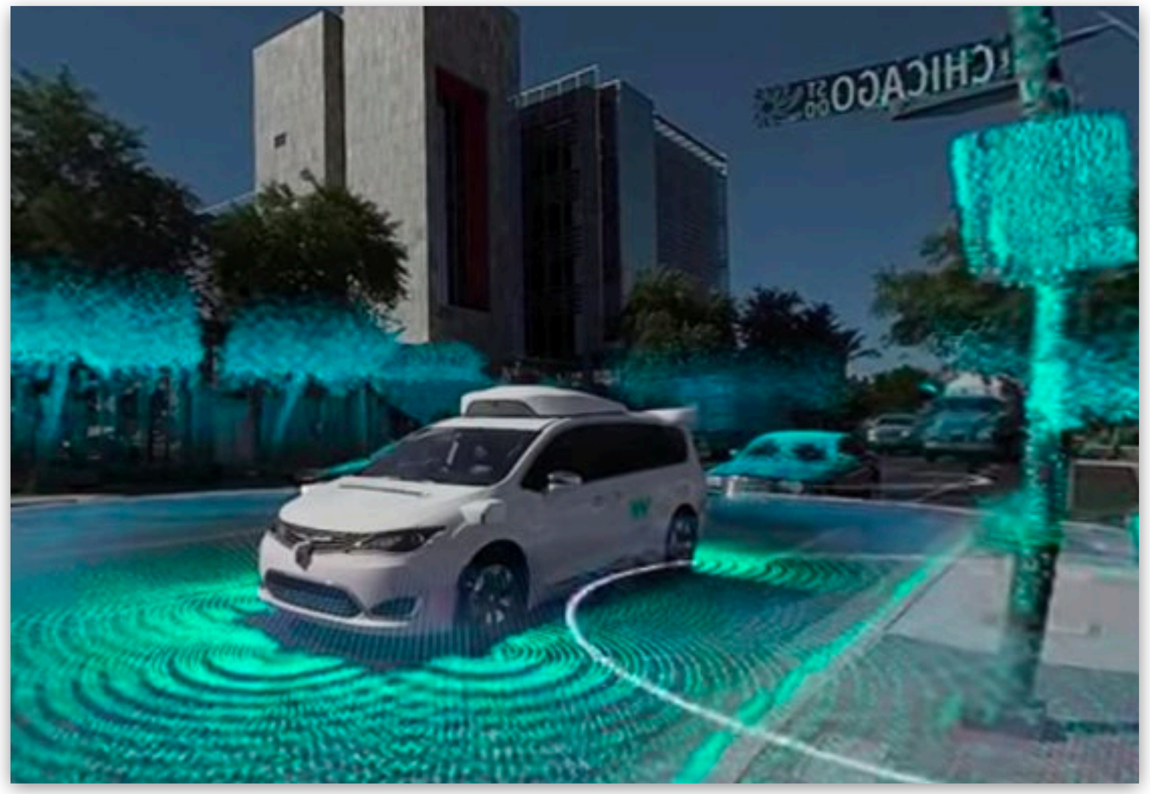
Joint Workshop on Long-Term Visual Localization, Visual Odometry and Geometric and Learning-based SLAM @ CVPR 2020

 https://tom.ai/












ClickList
Kroger
ClickList
Kroger
nuro

prime

Talk Outline

- **SuperPoint**: architectures and training paradigms you *need* to know to replace local features with Convolutional Neural Networks

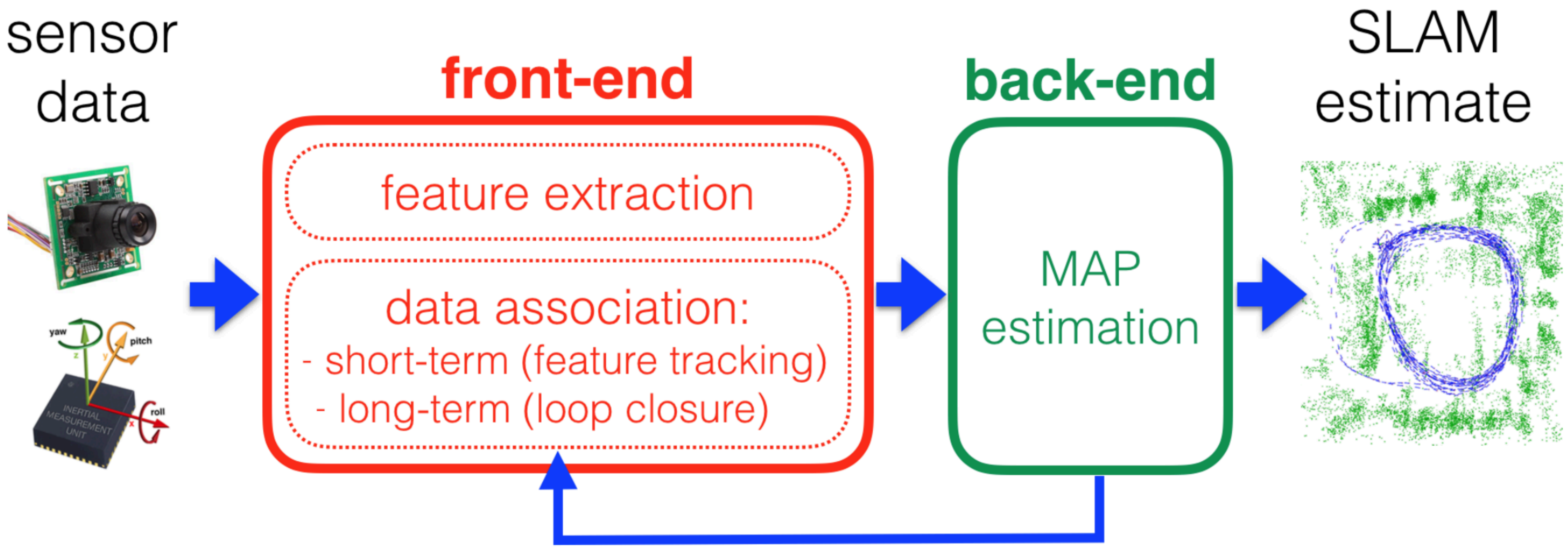
- **SuperGlue**: how to utilize Graph Neural Networks and Attention to improve feature matching

- **SuperMaps**: moving beyond pairwise matching and a roadmap towards end-to-end Deep Visual SLAM

Part I: SuperPoint

The art and craft of designing ConvNets to replace SIFT.

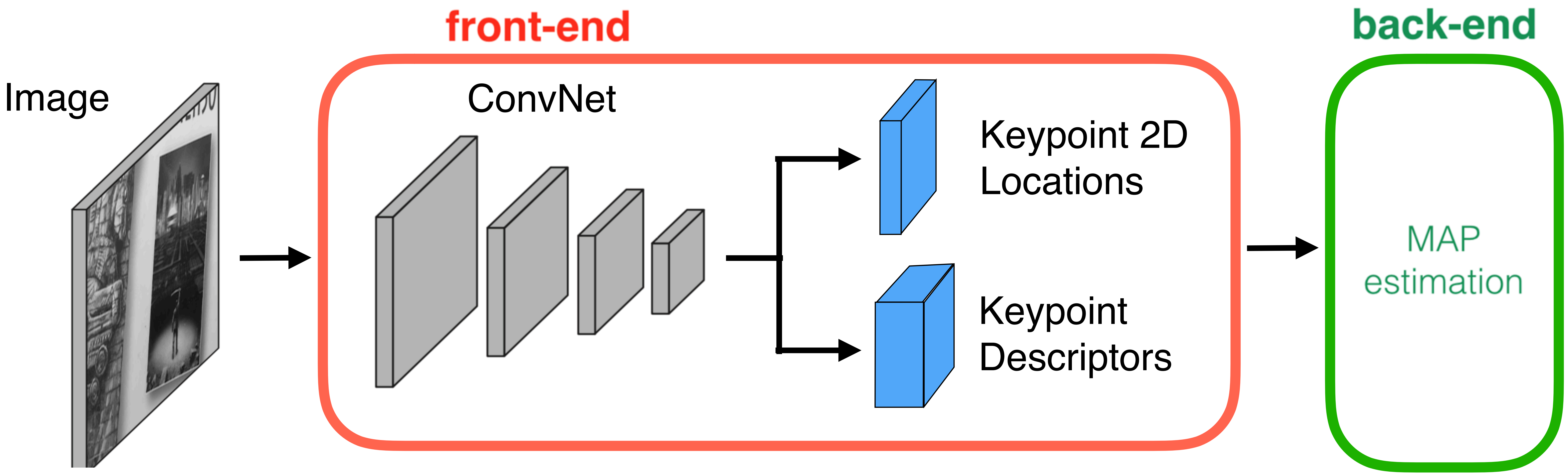
Two parts of Visual SLAM



- **Frontend**: Image inputs
  - Deep Learning success: Images + ConvNets
- **Backend**: Optimization over pose and map quantities
  - Use Bundle Adjustment

Photo Credit: Cadena et al 2016

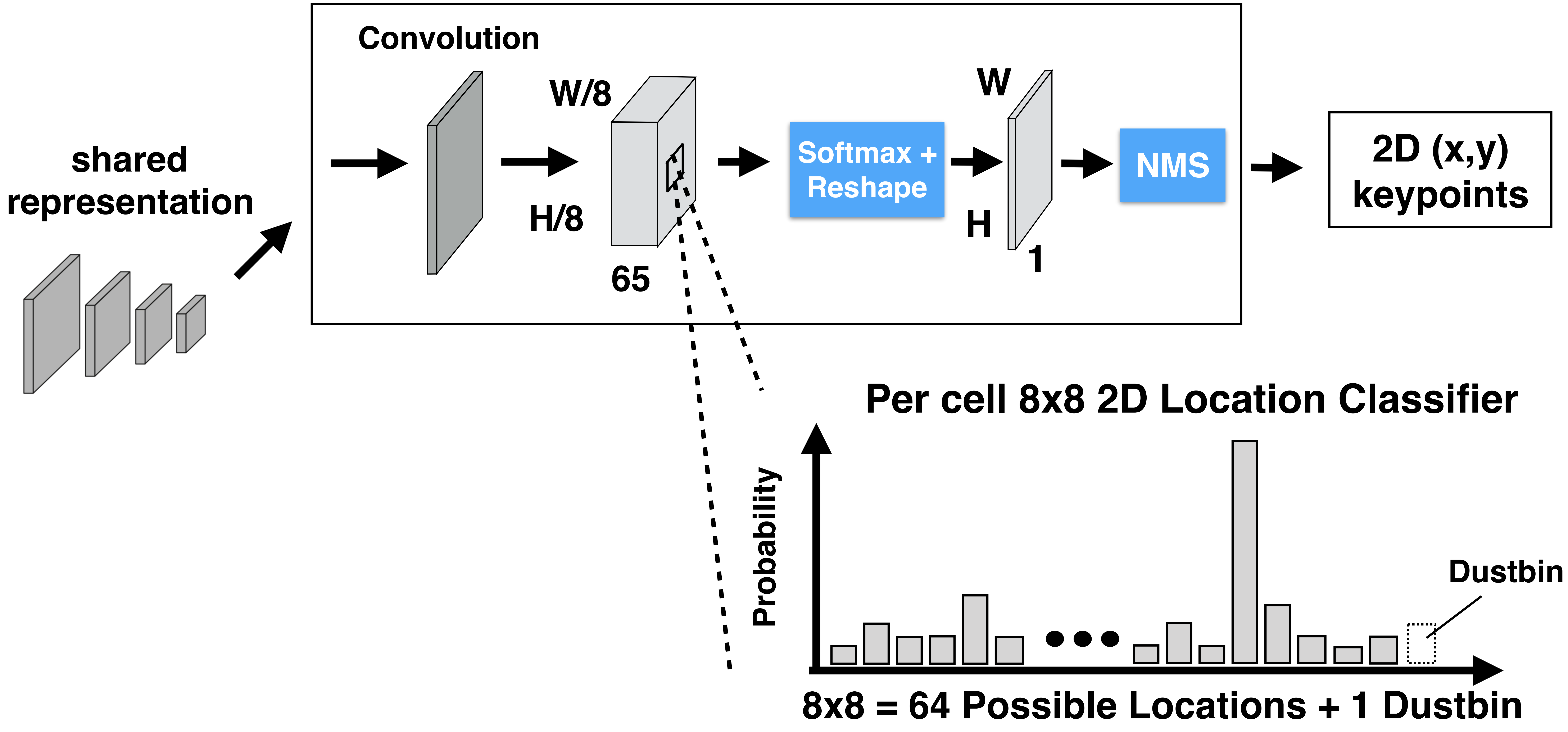
SuperPoint: A Deep SLAM Front-end



- Powerful fully convolutional design
  - Points + descriptors computed jointly, **No Patches**
  - Share VGG-like backbone
- Designed for real-time processing on a GPU
  - Medium-sized backbone. Tasks share ~90% of compute

DeTone, D., Malisiewicz, T., Rabinovich, A. [SuperPoint: Self-Supervised Interest Point Detection and Description.](#) In CVPR Deep Learning for Visual SLAM Workshop, 2018.

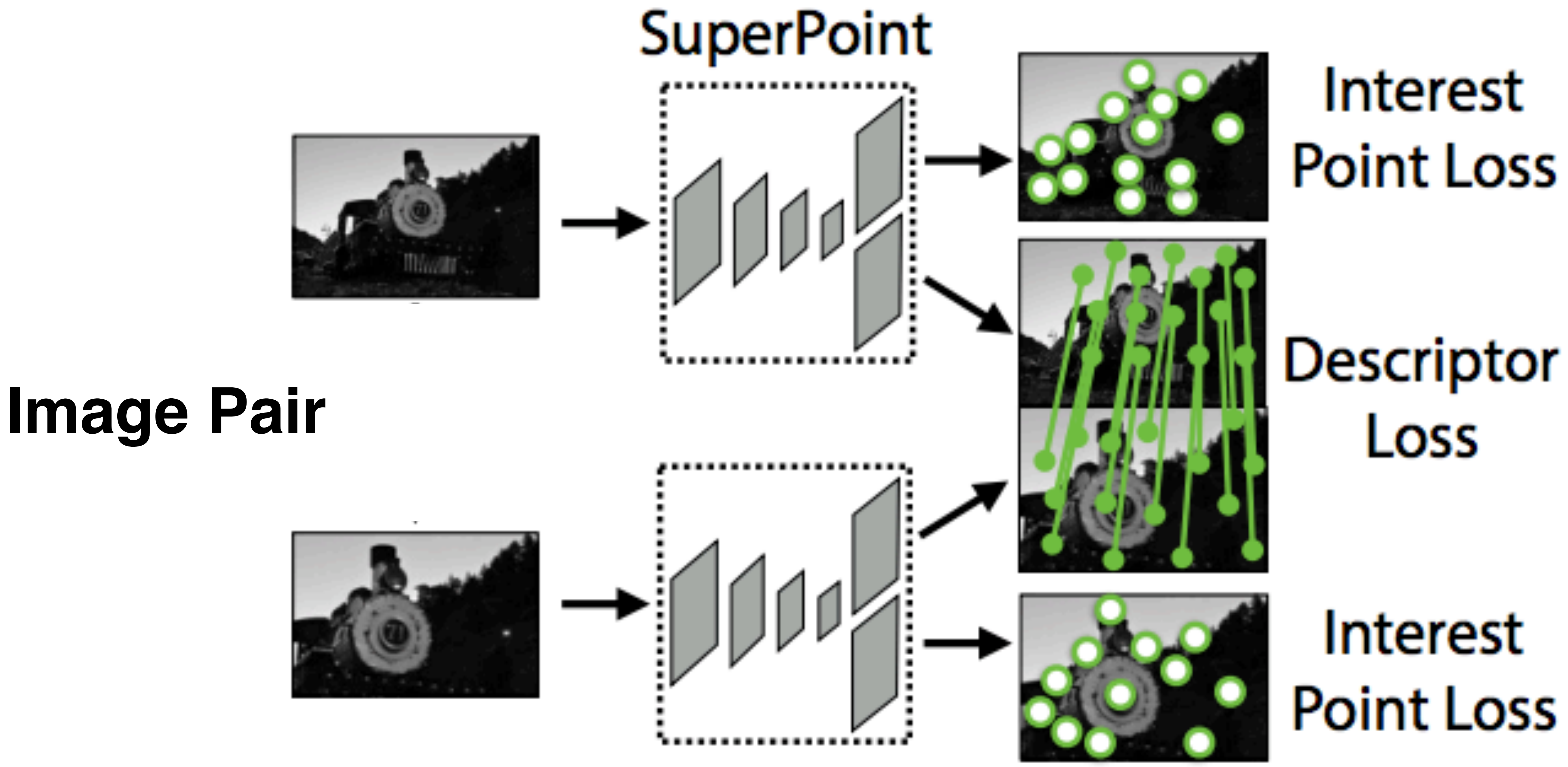
Keypoint / Interest Point Decoder



- No deconvolution layers
- Each output cell responsible for local 8x8 region

How To Train SuperPoint?

Setting up the Training



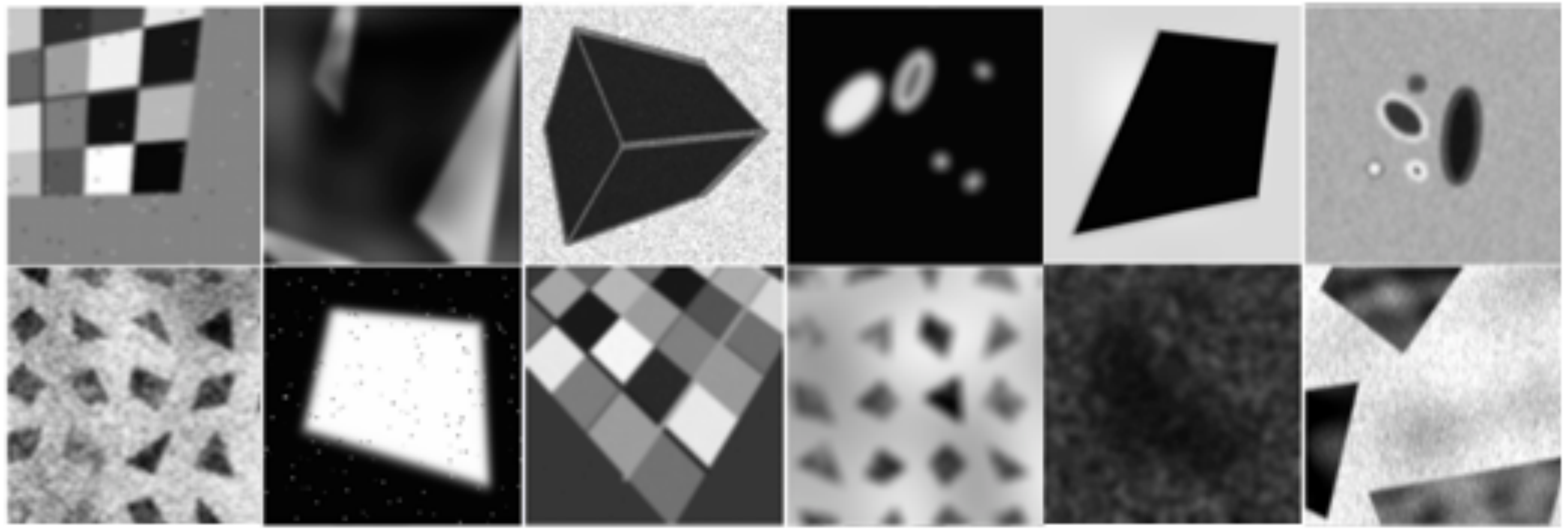
- Siamese training with pairs of images
- Descriptor trained via metric learning (contrastive loss)
  - Straightforward given correspondence
- Keypoints trained via supervised keypoint labels
  - Where do these come from?

How to get Keypoint Labels for Natural Images?

- Need large-scale dataset of annotated images
- Too hard for humans to label

Self-Supervised Training

Synthetic Shapes (has interest point labels)



First train on this

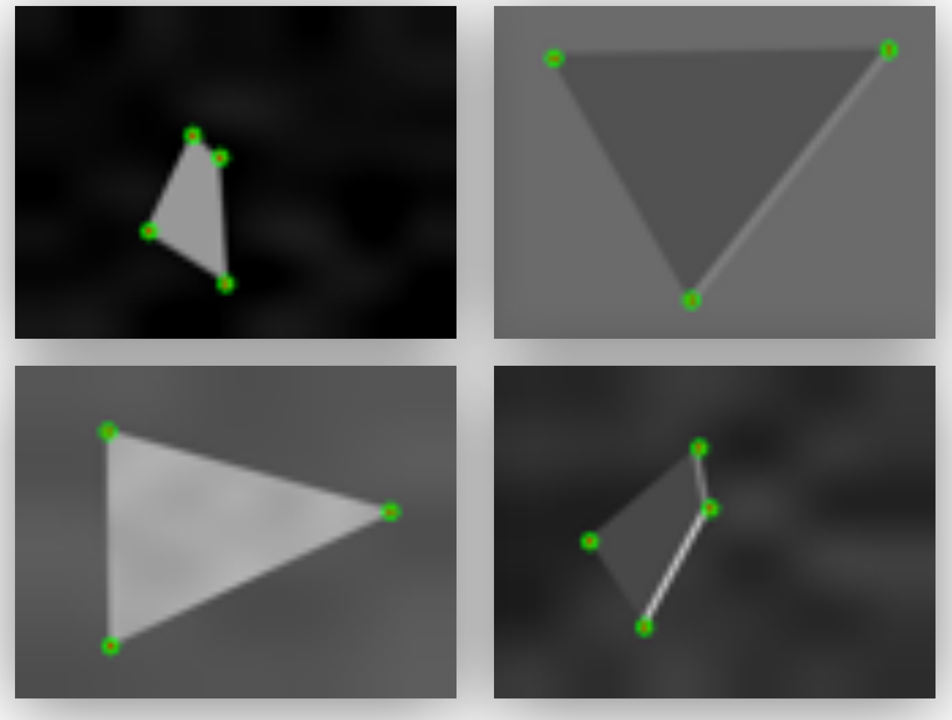
"Homographic Adaptation"

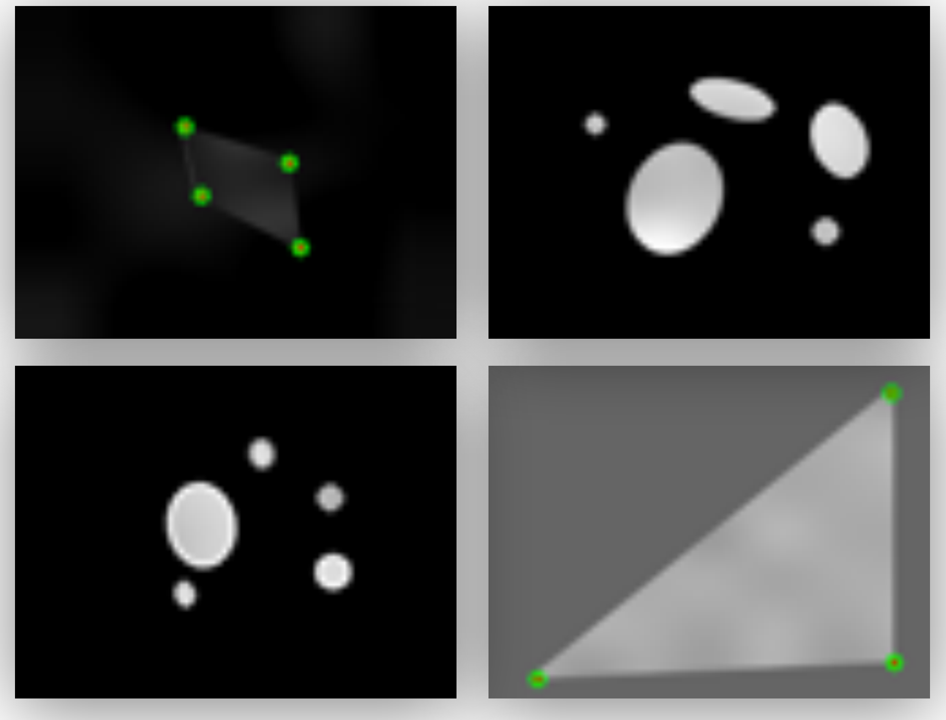
MS-COCO (no interest point labels)

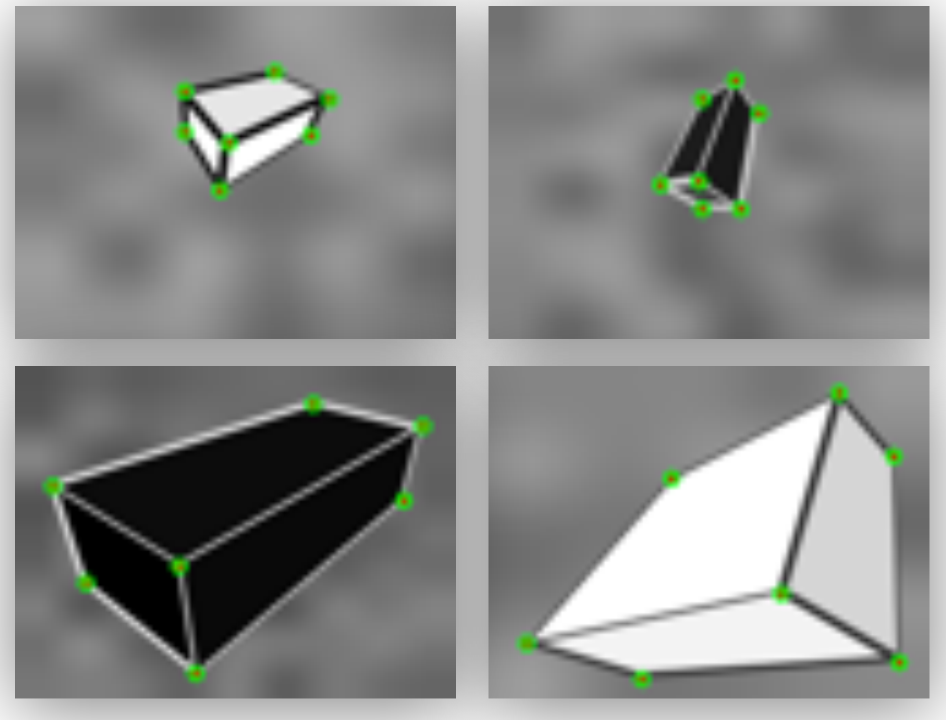
Use resulting detector to label this

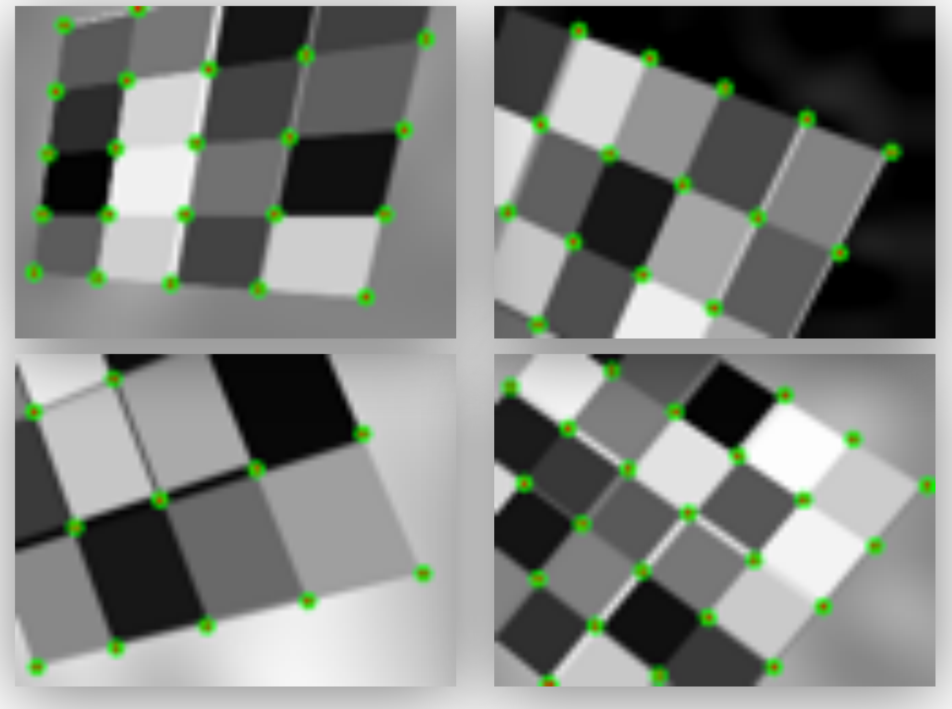
Synthetic Training

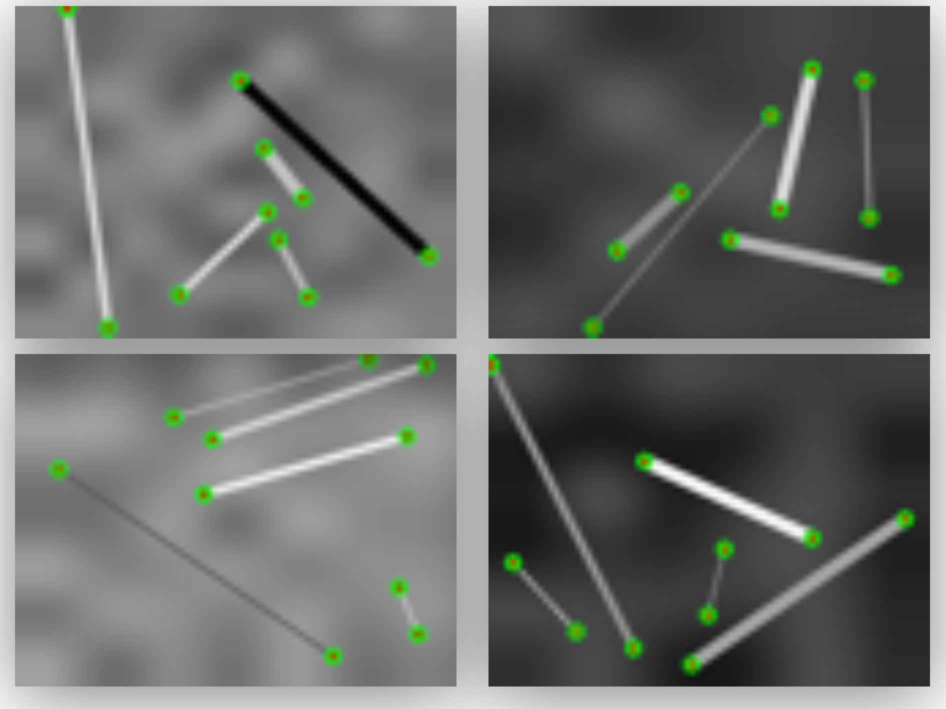
- Non-photorealistic shapes
- Heavy noise
- Effective and easy

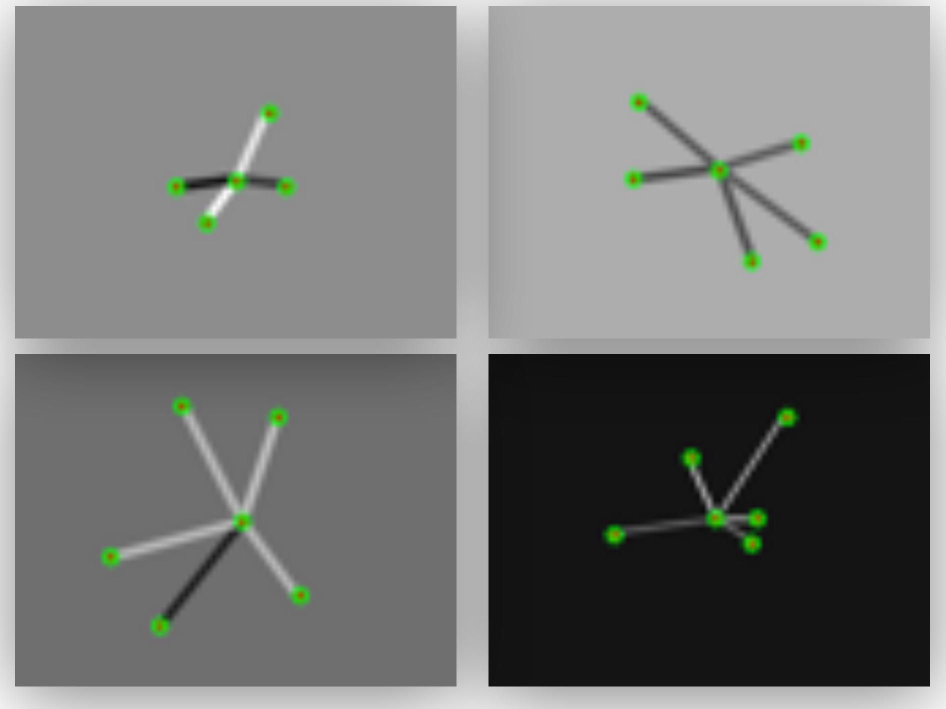

Quads/Tris


Quads/Tris/Ellipses

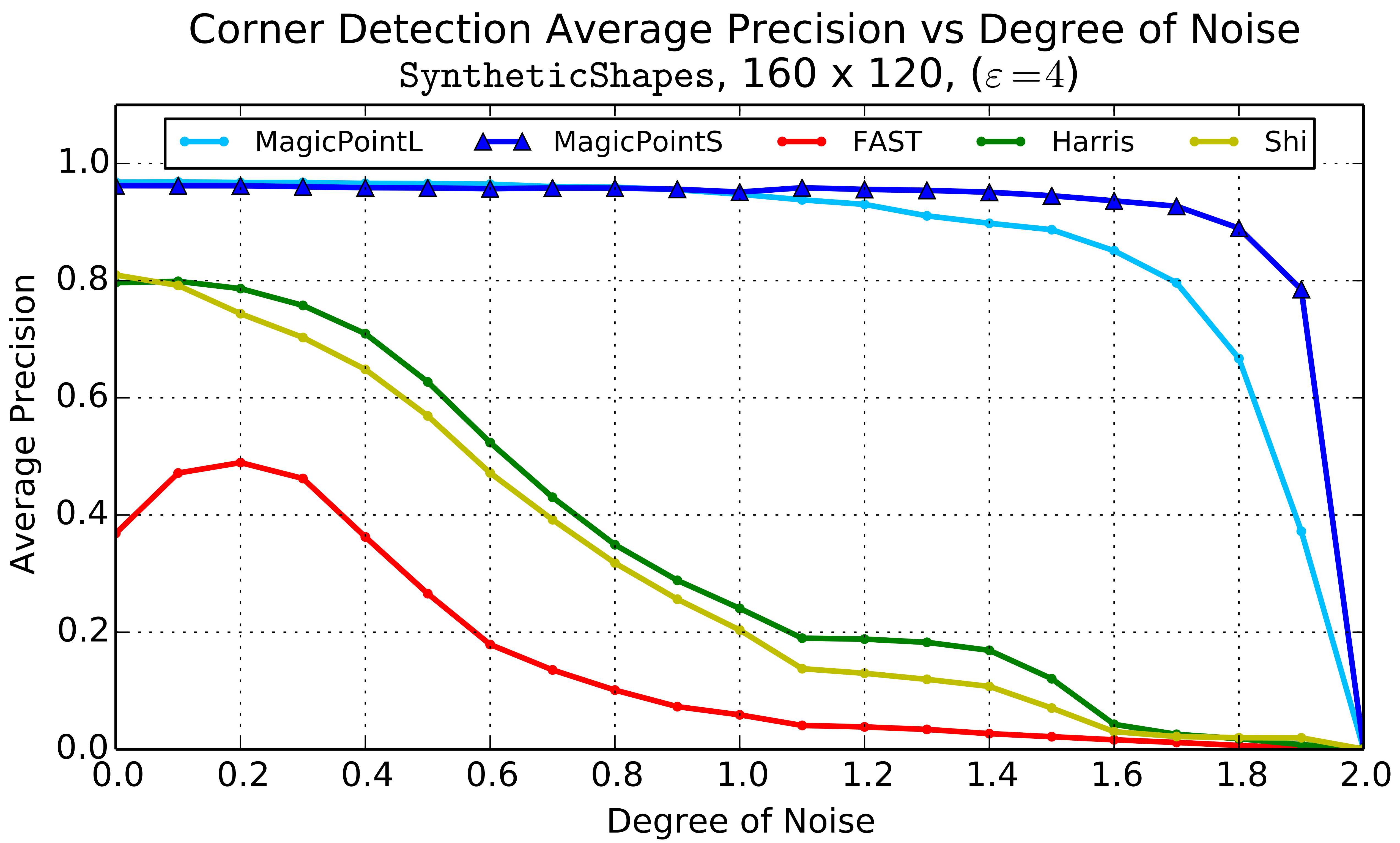

Cubes

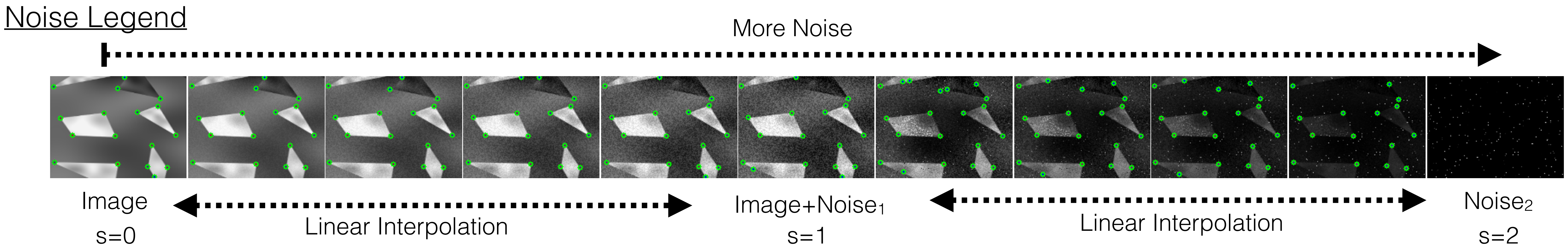

Checkerboards


Lines

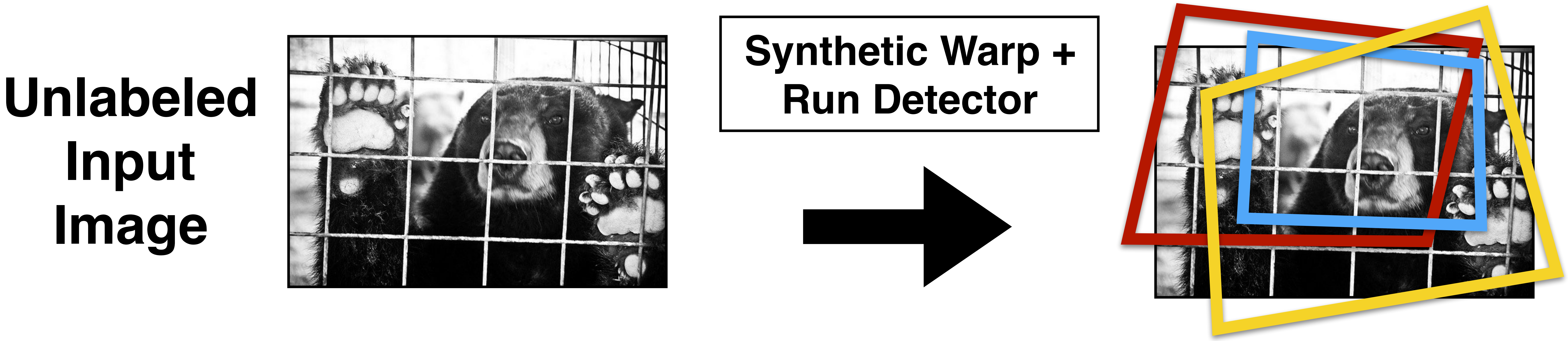

Stars

Early Version of SuperPoint (MagicPoint)



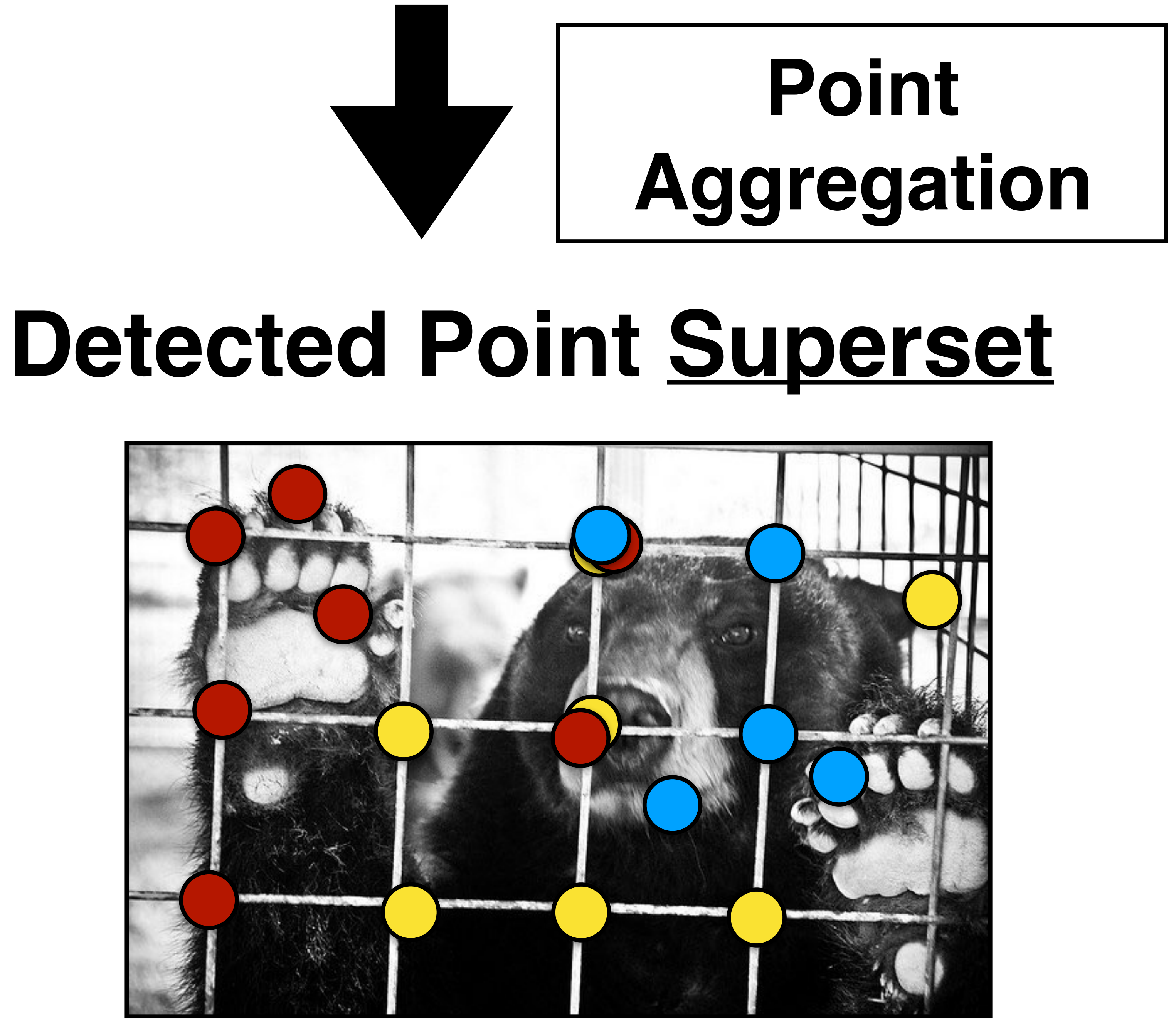


DeTone, D., Malisiewicz, T., Rabinovich, A. [Toward Geometric DeepSLAM.](#) In arXiv:1707.07410. July, 2017.

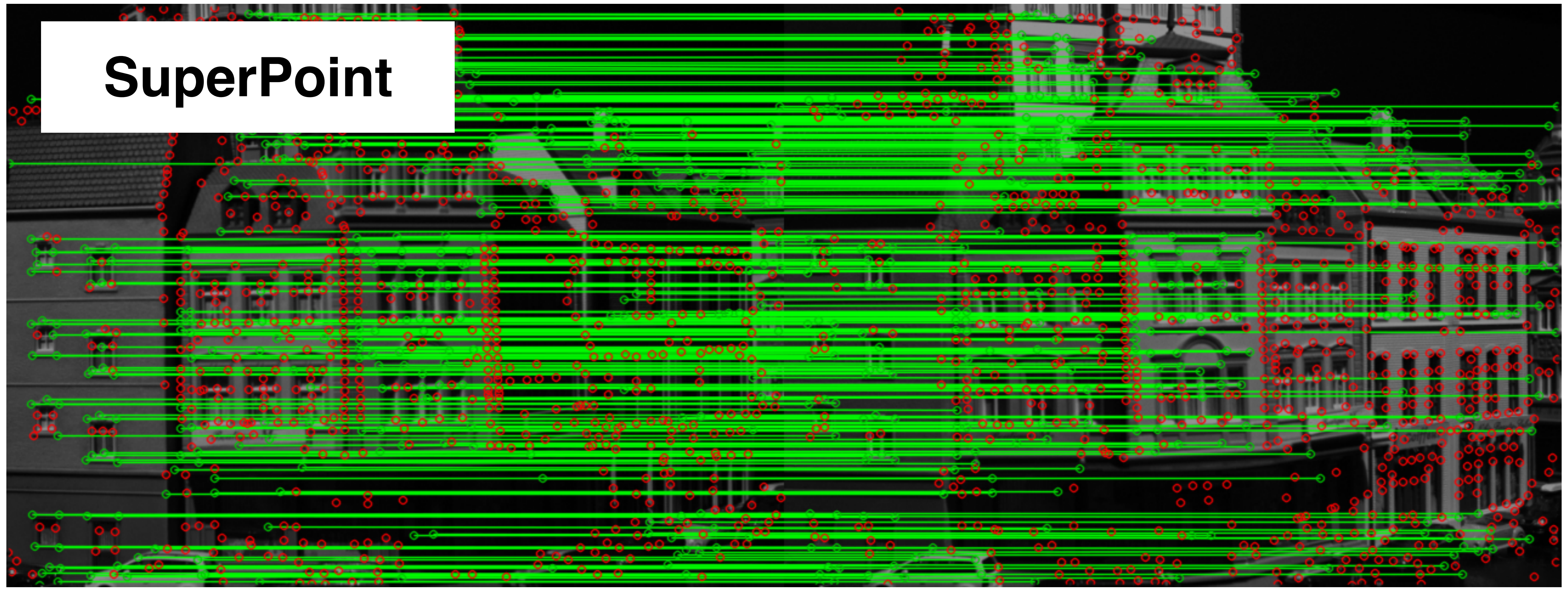


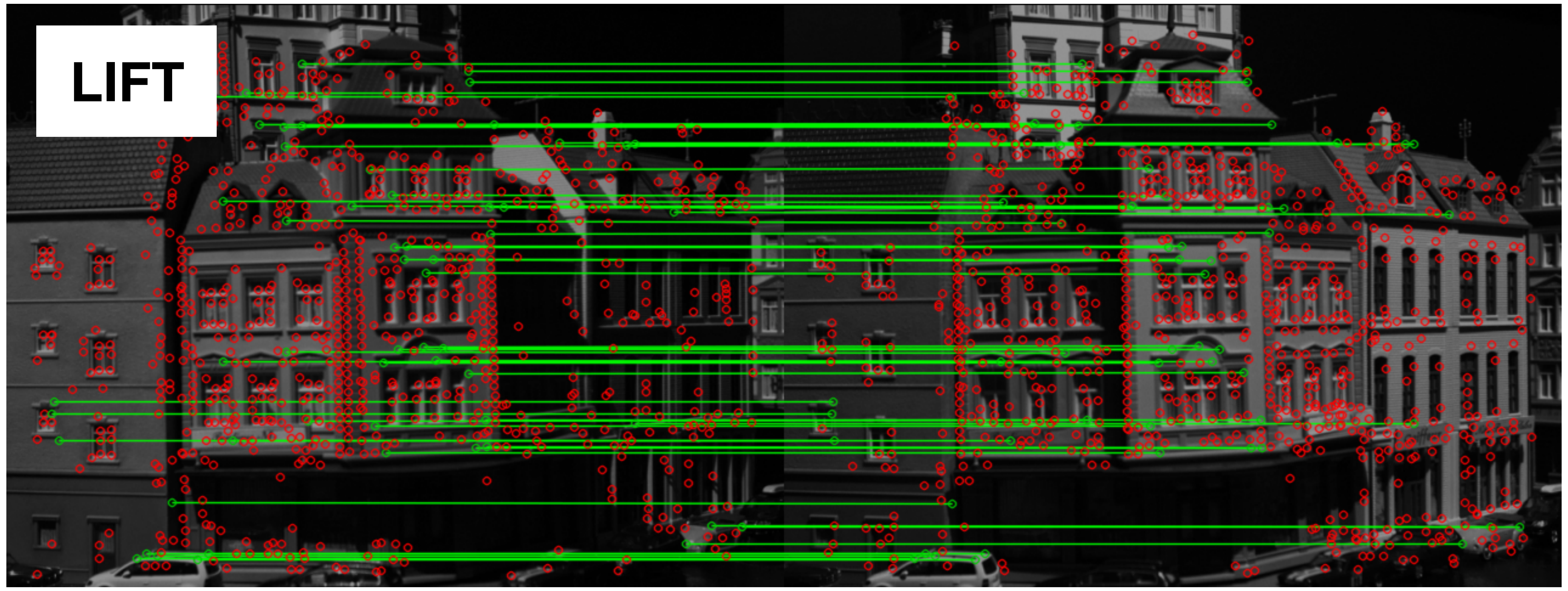
Homographic Adaptation

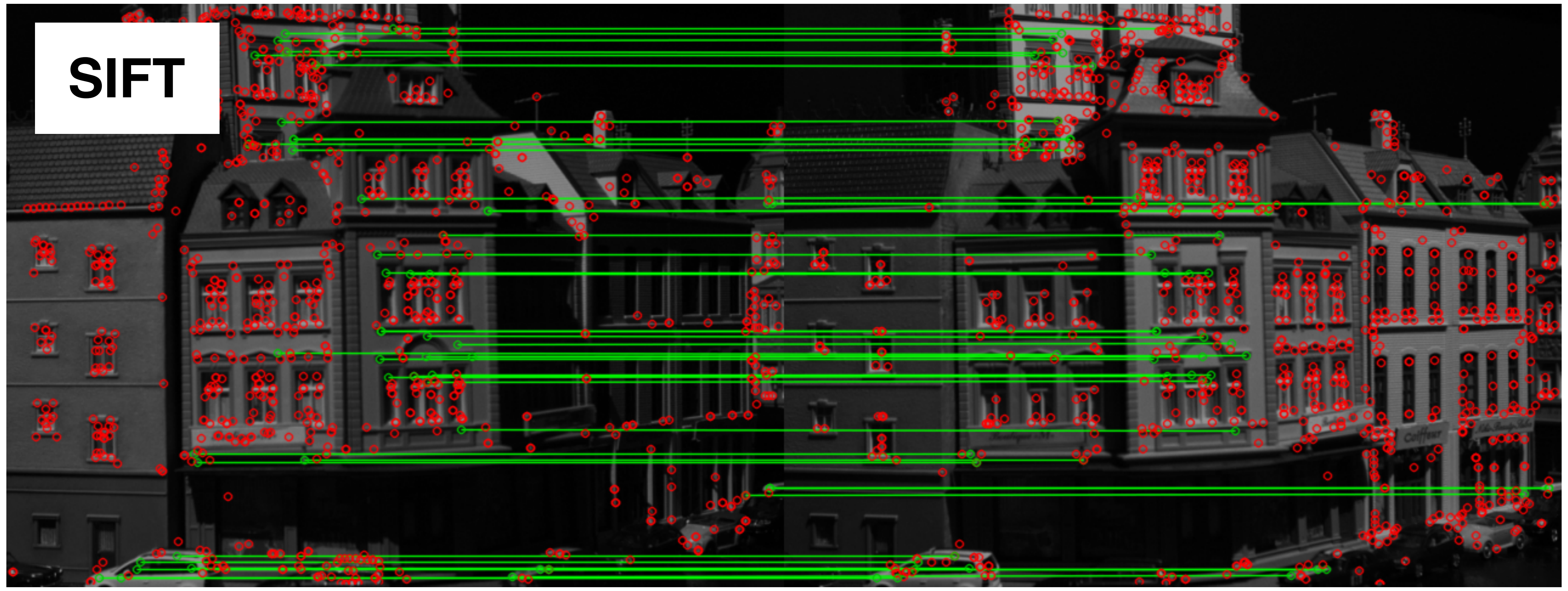
- Simulate planar camera motion with homographies
- Self-labelling technique
  - Suppress spurious detections
  - Enhance repeatable points

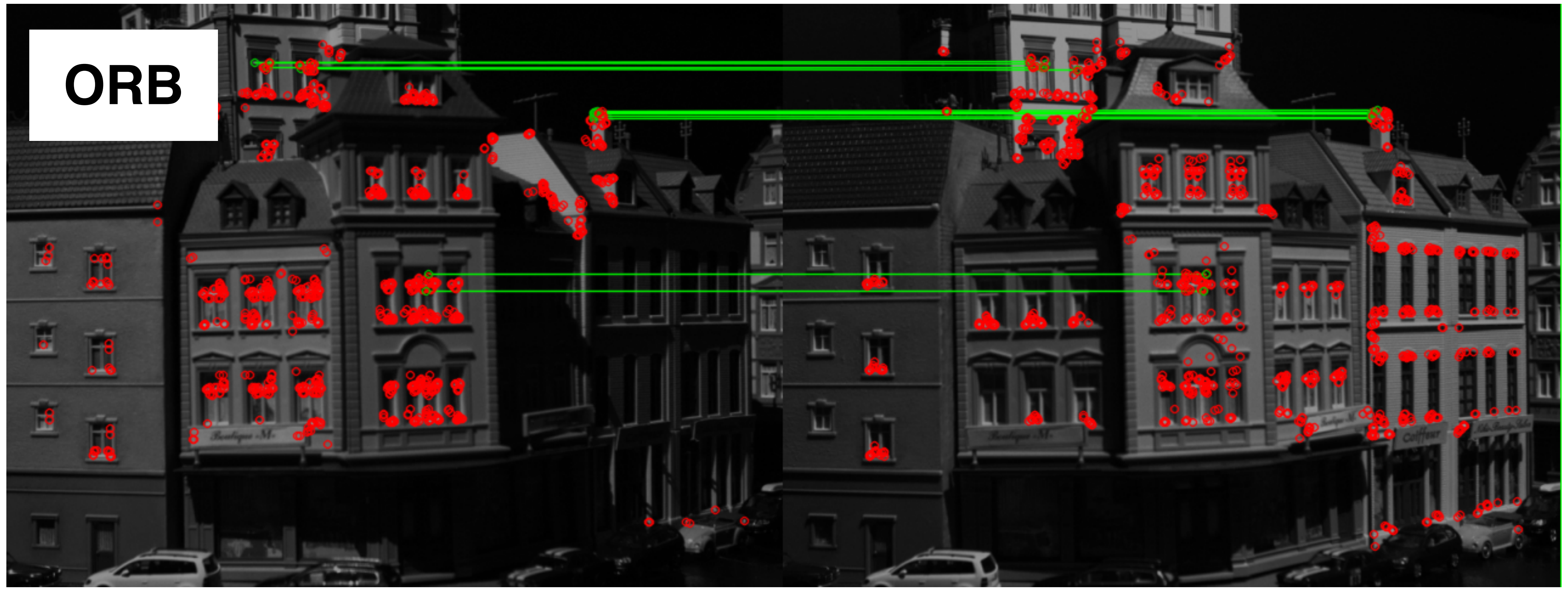


SuperPoint Example #1

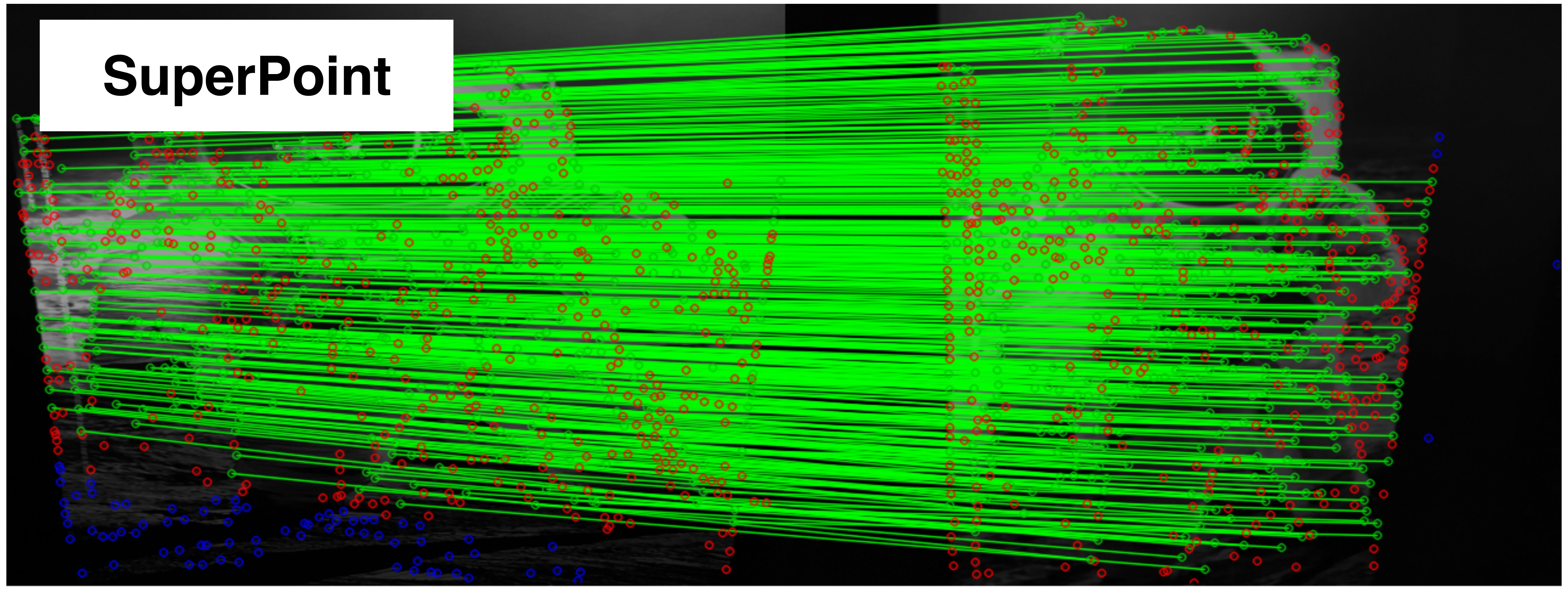


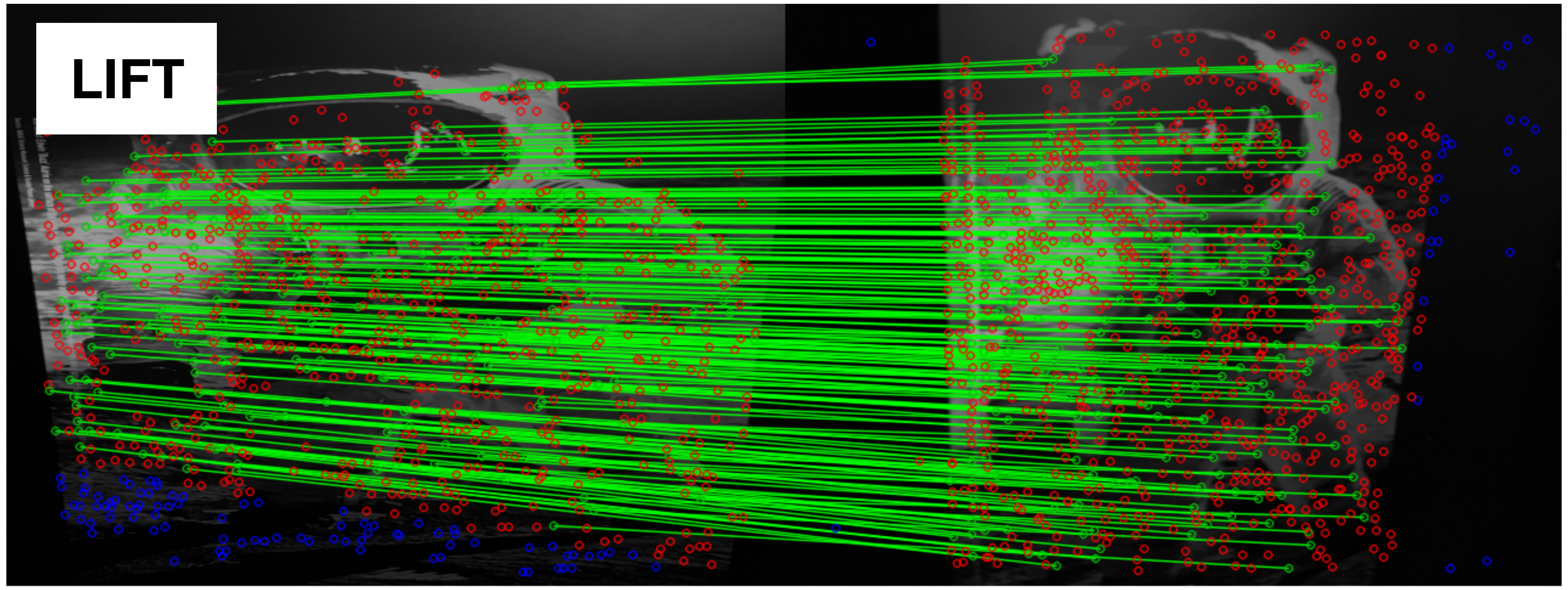


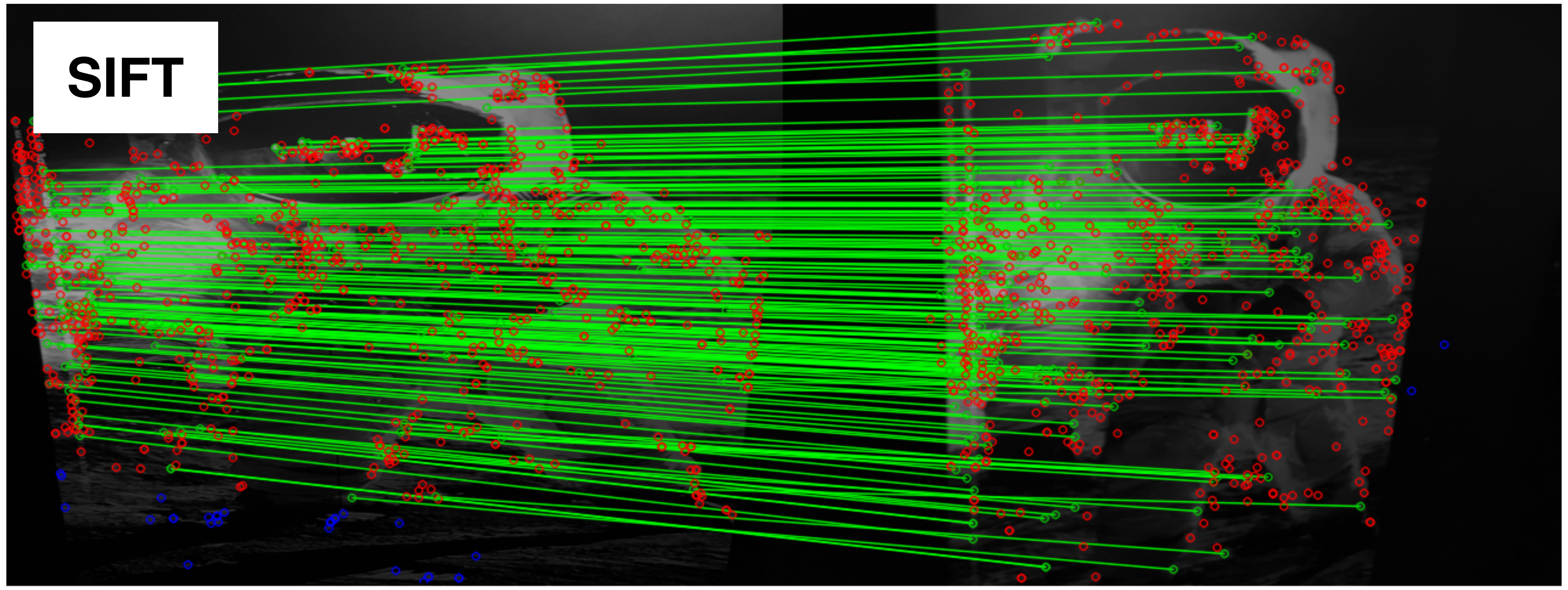


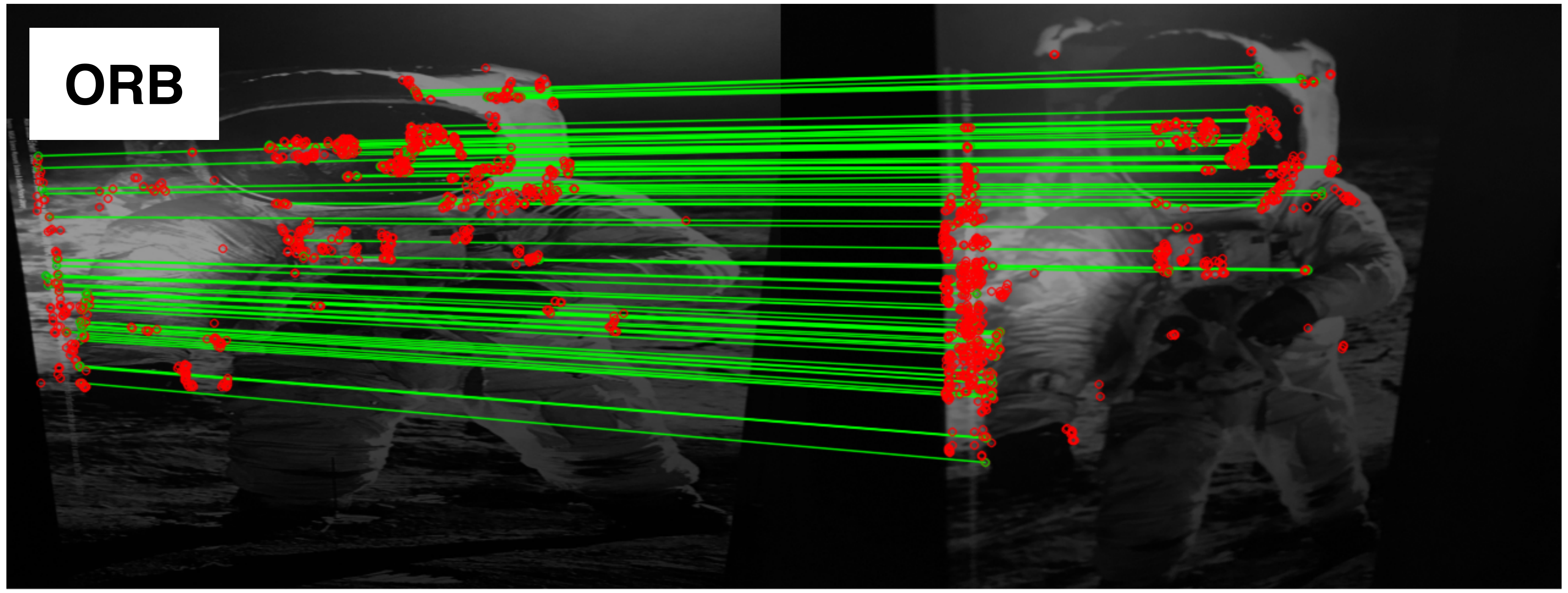


SuperPoint Example #2

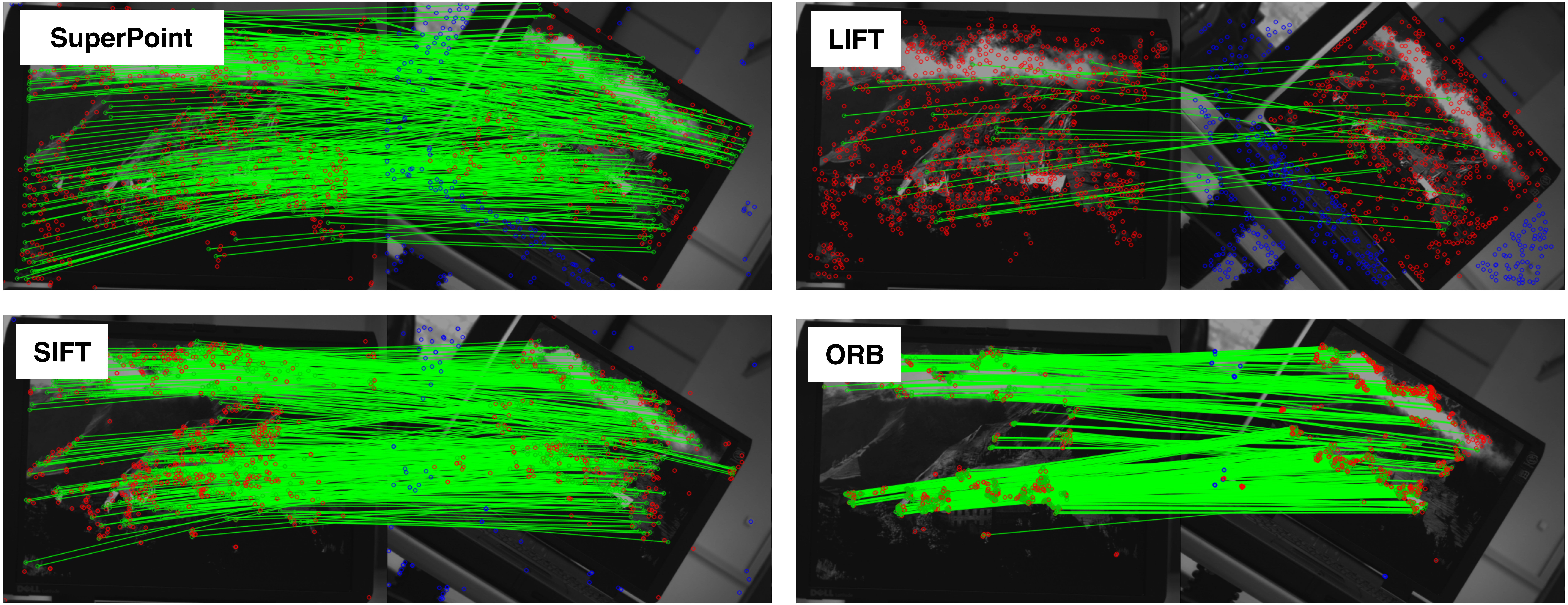








SuperPoint Example #3

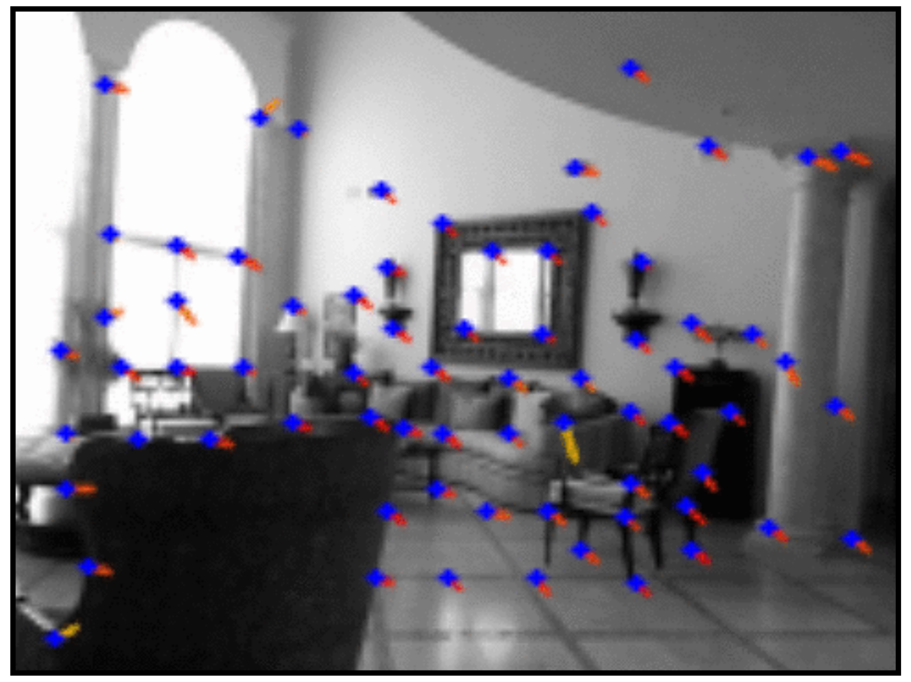


3D Generalizability of SuperPoint

- Trained+evaluated on planar, does it generalize to 3D?
- “Connect-the-dots” using nearest neighbor matches
- Works across many datasets / input modalities / resolutions!

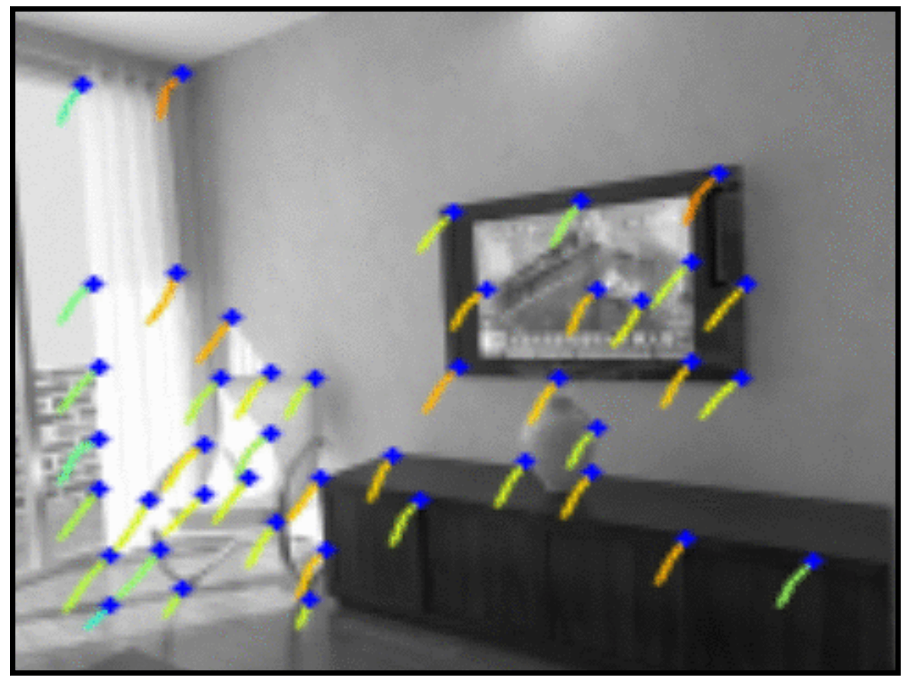
Freiburg (Kinect)

NYU (Kinect)



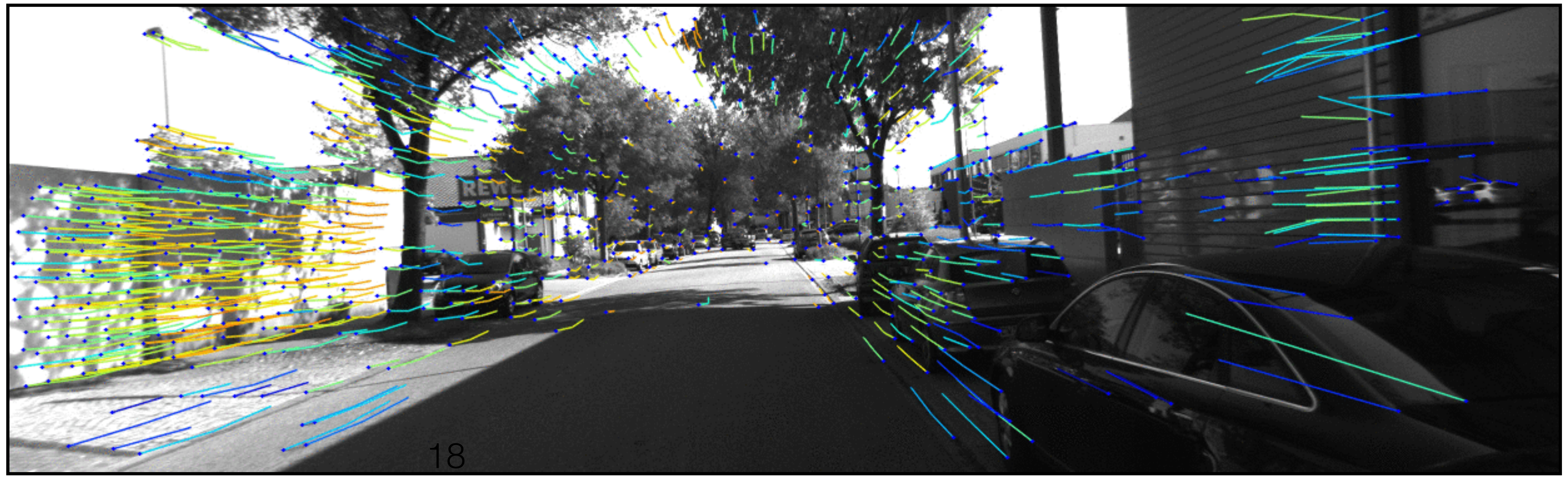
MonoVO (fisheye)

ICL-NUIM (synth)



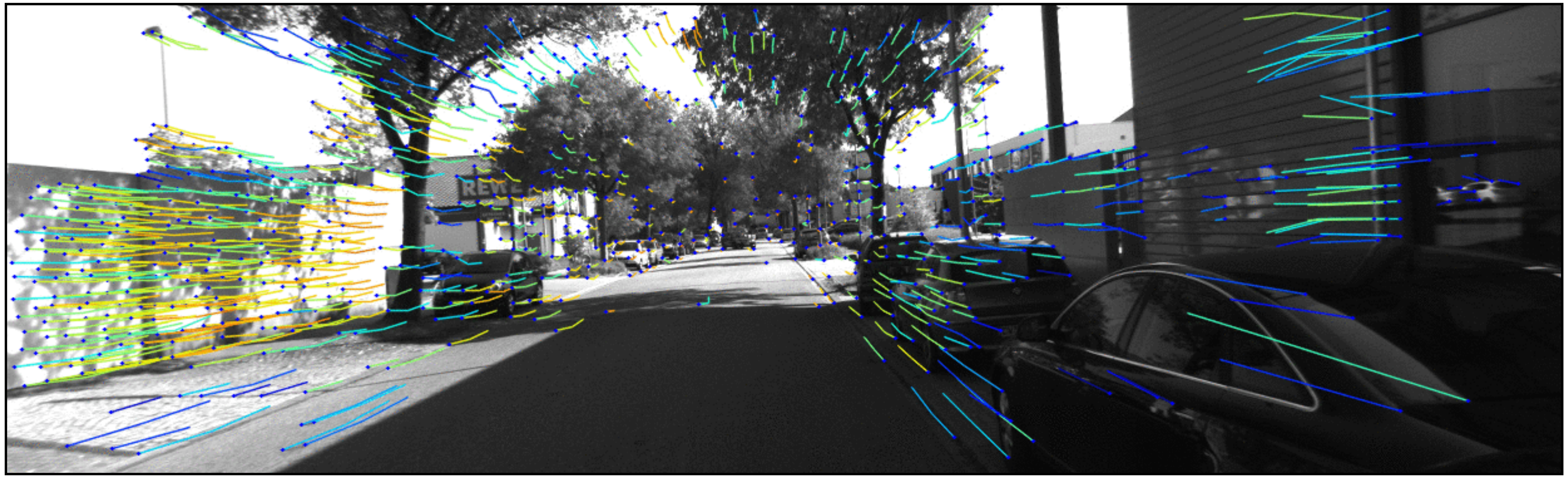
MS7 (Kinect)

KITTI (stereo)



Pre-trained SuperPoint Release

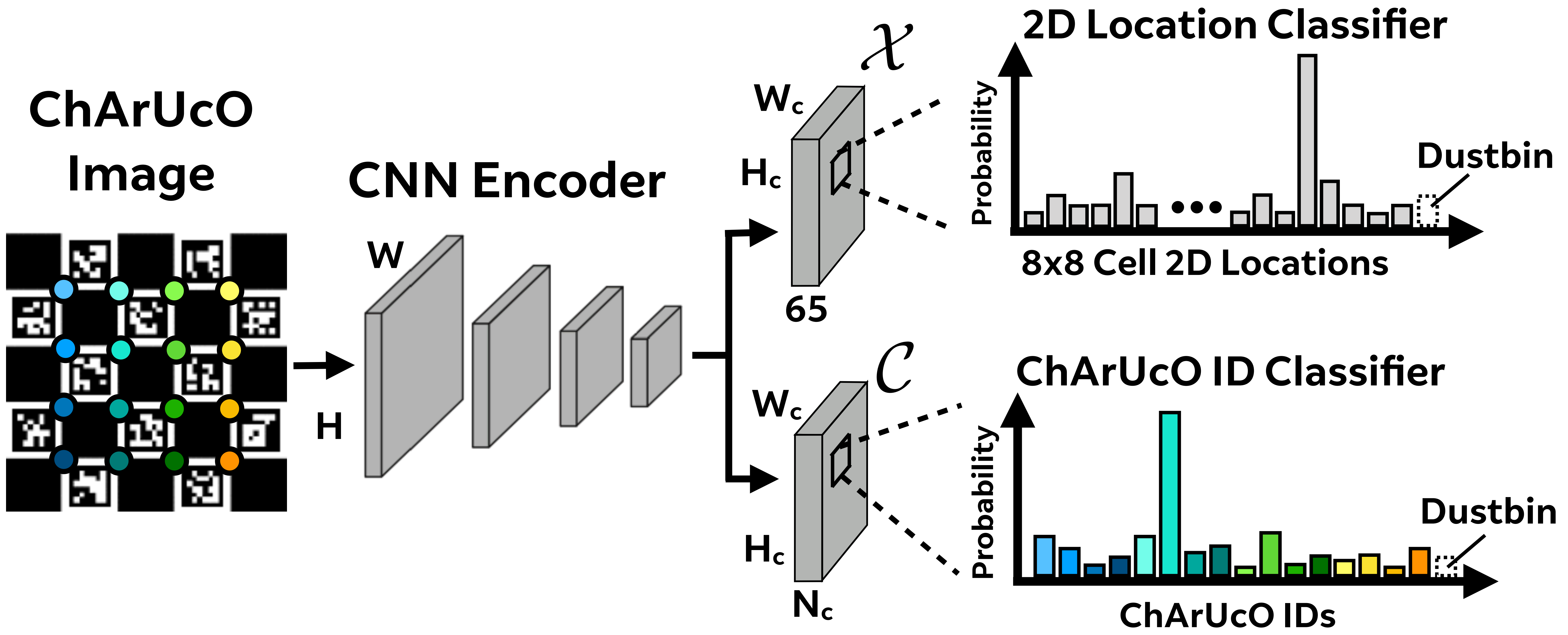
- Implemented in PyTorch
- Two files, minimal dependencies. Get up and running in 5 minutes or less!
- Released at 1st Deep Learning for Visual SLAM Workshop at CVPR 2018



github.com/magicleap/SuperPointPretrainedNetwork

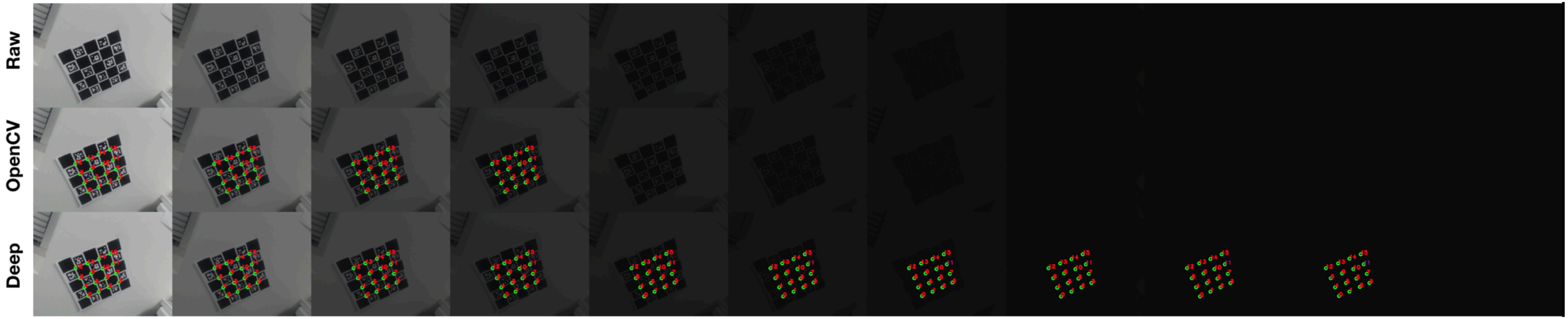
Can we apply SuperPoint to other tasks?

- What if we adapt the SuperPoint architecture to object instance detection?



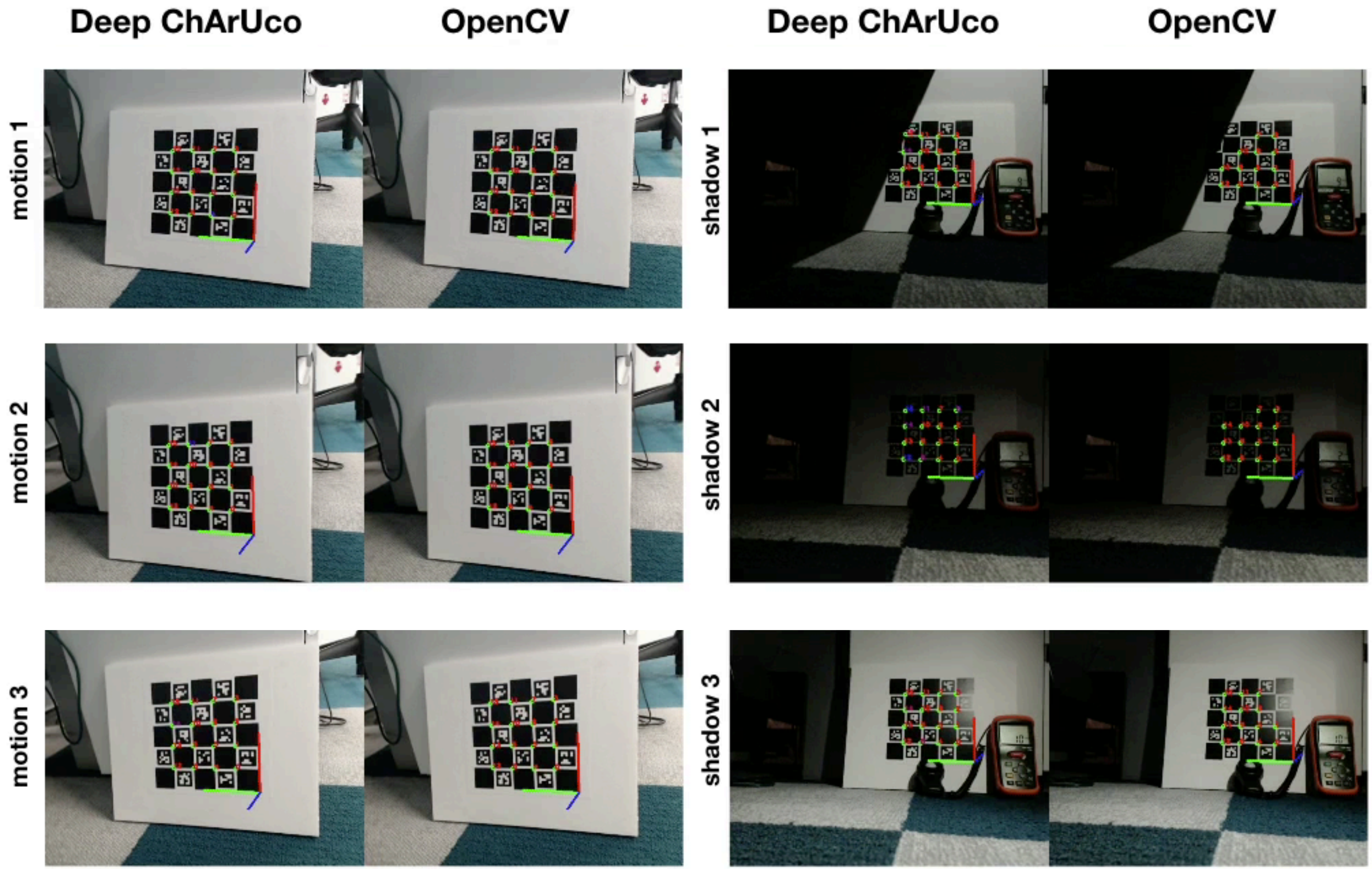
Hu D., DeTone D., Malisiewicz T. Deep ChArUco: Dark ChArUco Marker Pose Estimation. In CVPR, 2019.

CharucoNet can “see” in the dark



Increasingly Dark Images

Hu D., DeTone D., Malisiewicz T. [Deep ChArUco: Dark ChArUco Marker Pose Estimation](). In CVPR, 2019.



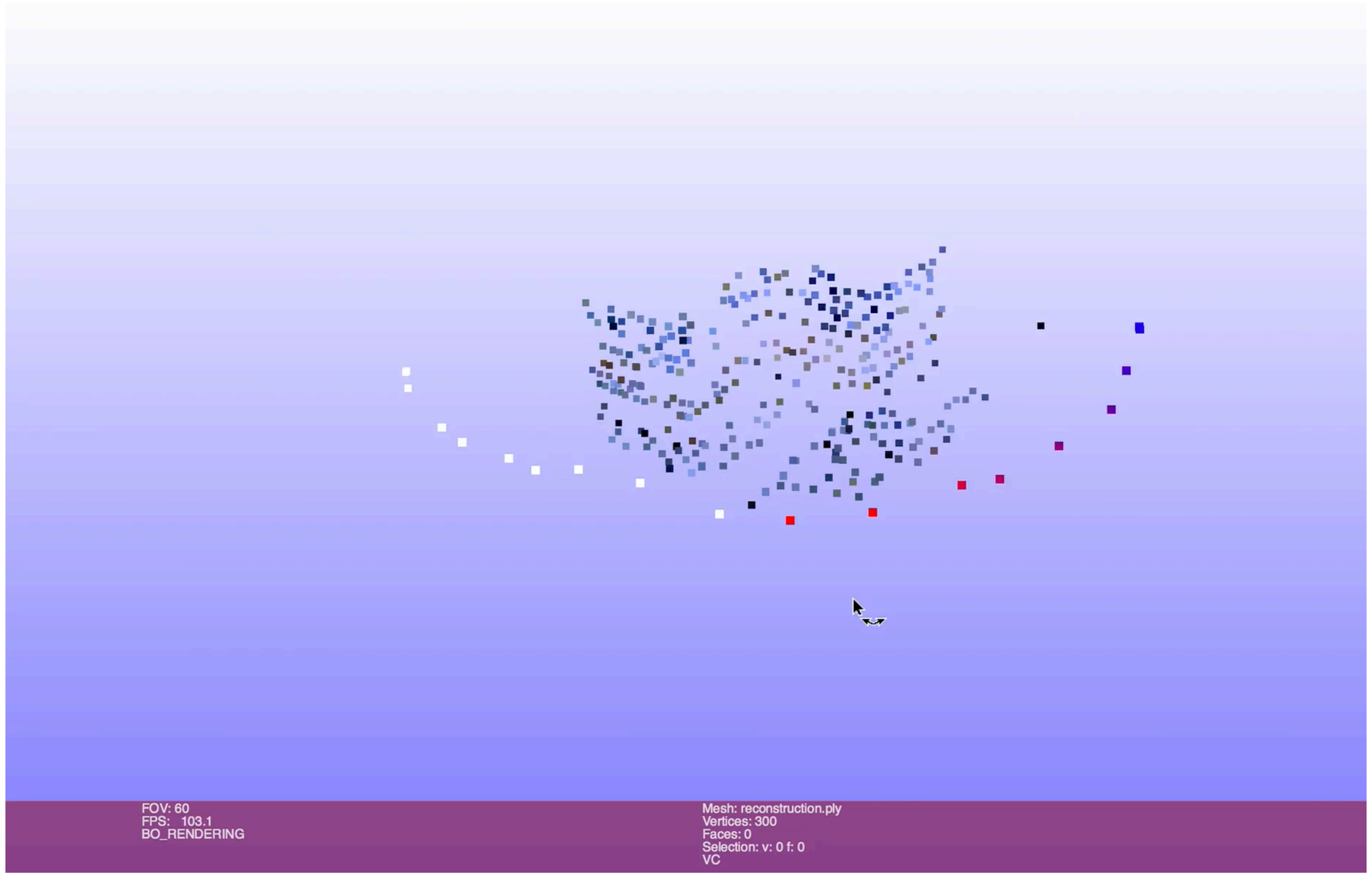
Hu D., DeTone D., Malisiewicz T. Deep ChArUco: Dark ChArUco Marker Pose Estimation. In CVPR, 2019.

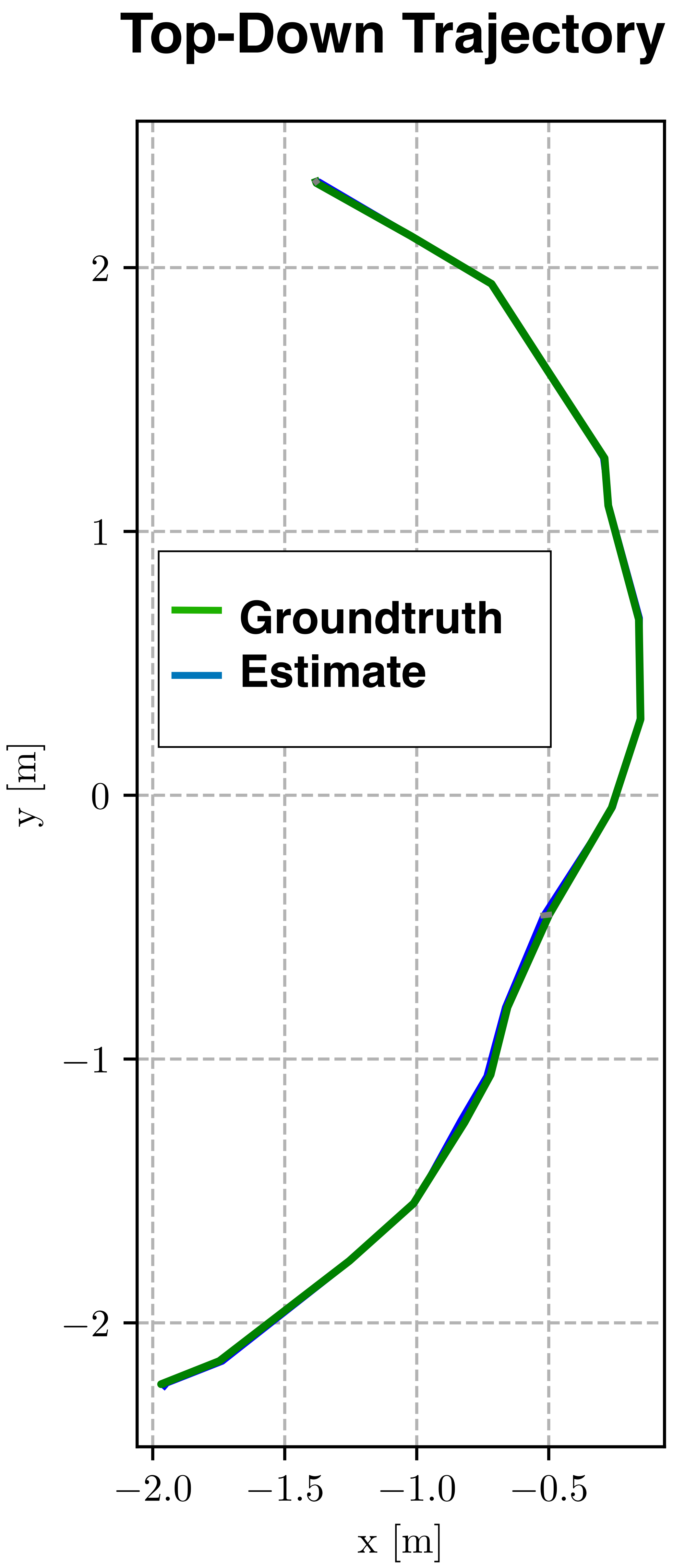
SuperPointVO

Can we improve SuperPoint with real data and a Visual Odometry backend?

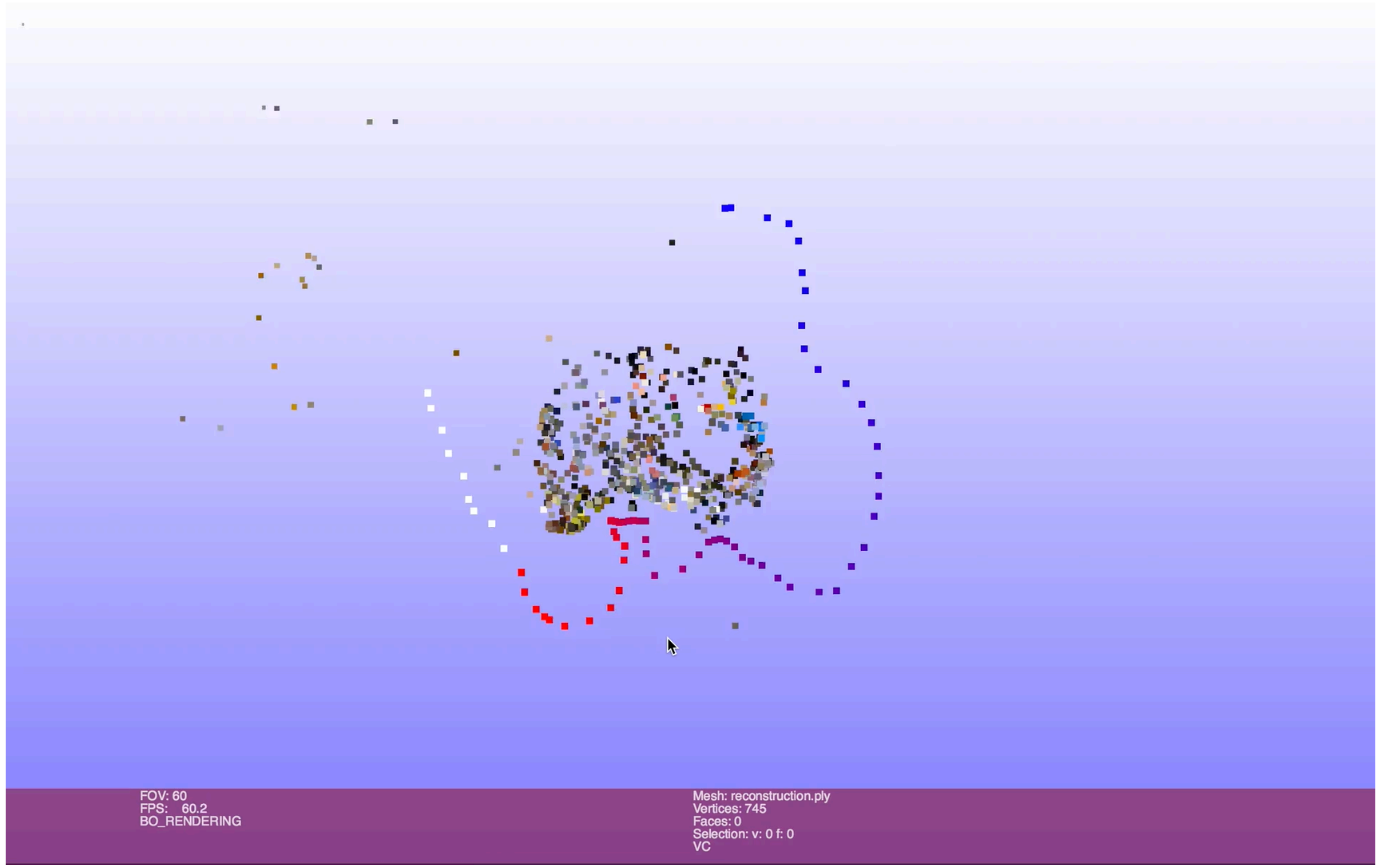
DeTone, D., Malisiewicz, T., Rabinovich, A. Self-Improving Visual Odometry In arXiv: 1812.03245. December, 2018.

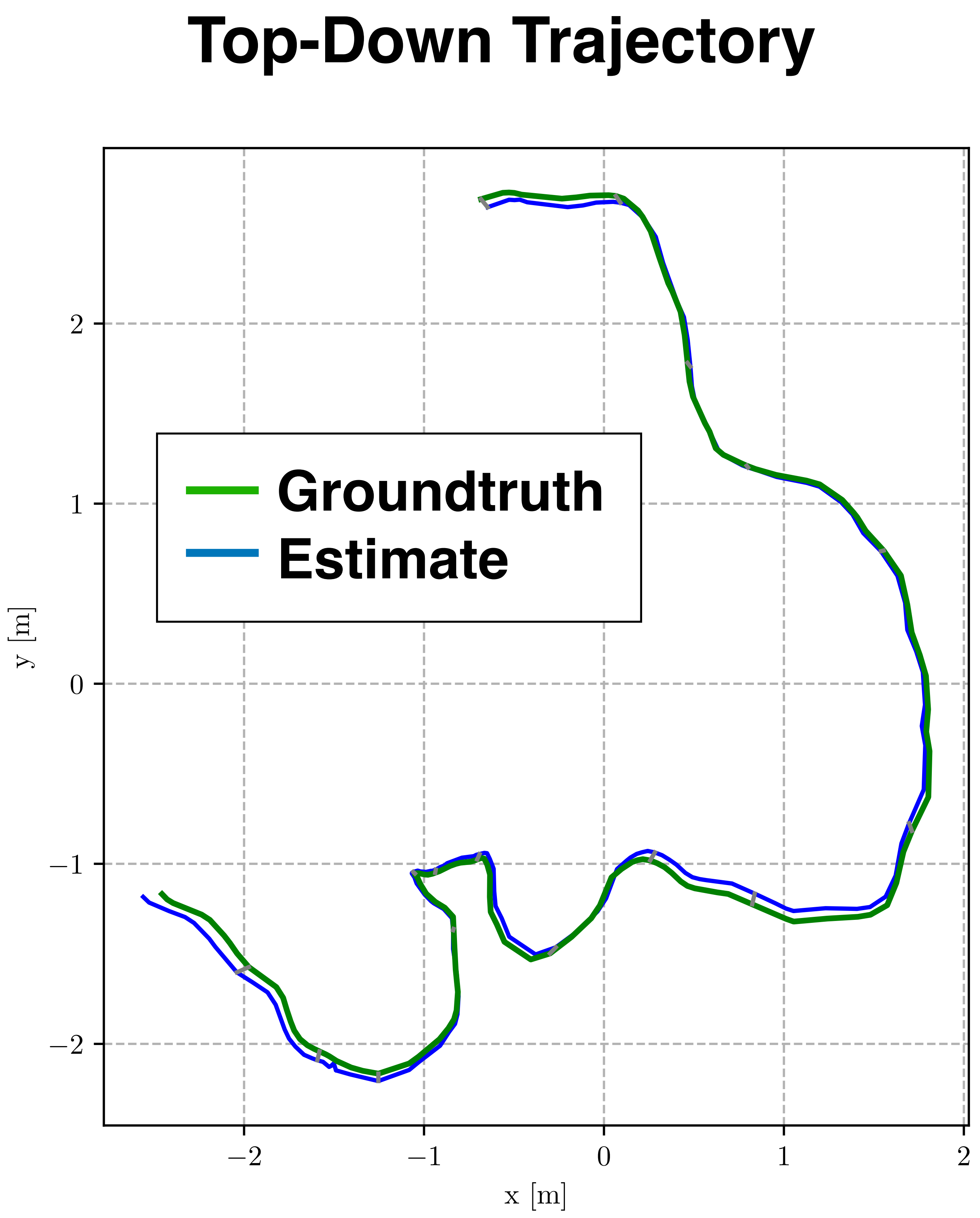
VO Reconstruction on Freiburg-TUM RGBD
‘structure_texture_far’





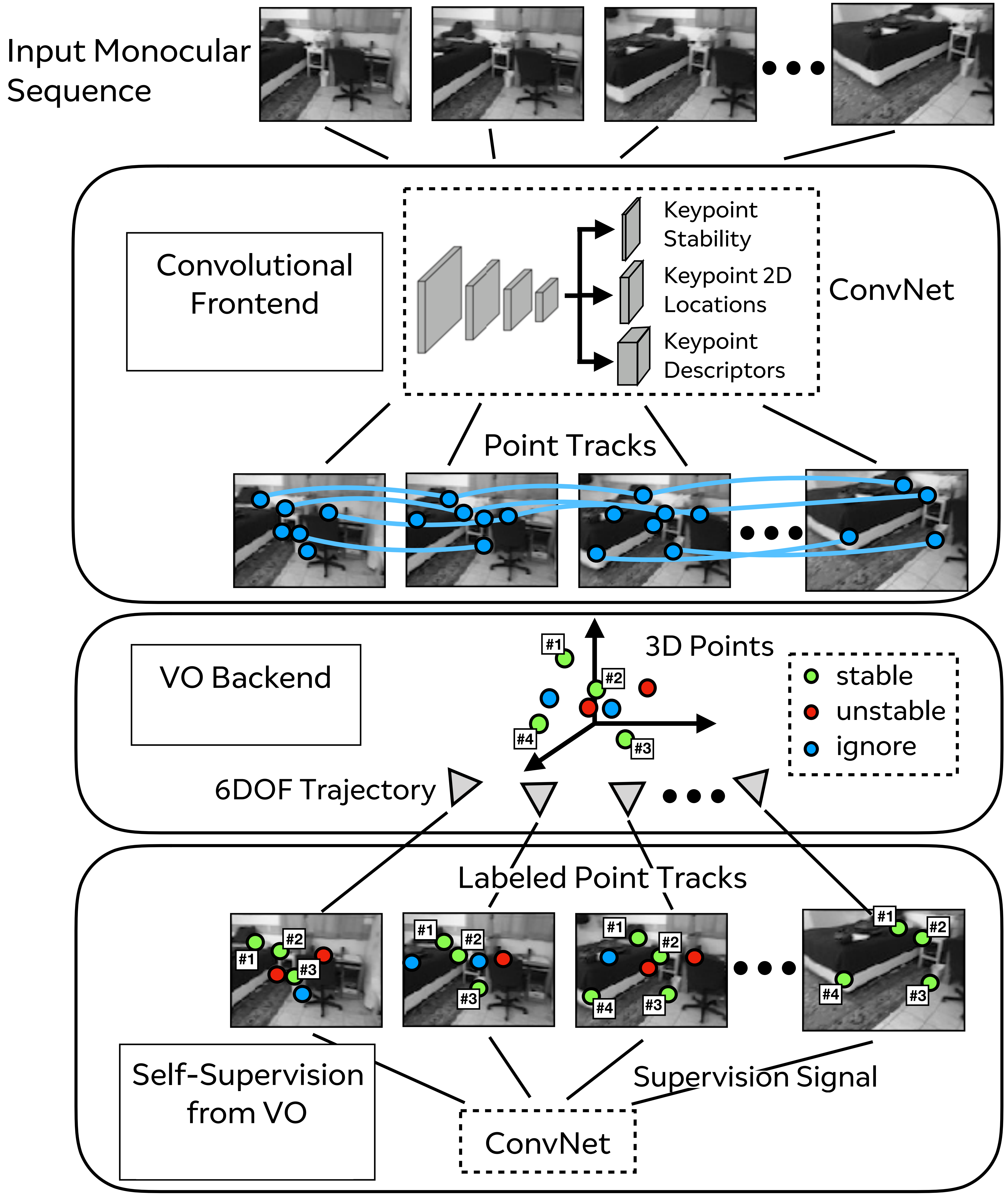
VO Reconstruction on Freiburg-TUM RGBD 'long_office_household'





Benefits of VO-based SuperPoints

- Establish correspondence across time
- Learn which points are stable


Input Monocular Sequence
Convolutional Frontend
Keypoint Stability
Keypoint 2D Locations
Keypoint Descriptors
ConvNet
Point Tracks
VO Backend
3D Points
#1
#2
#3
#4
stable
unstable
ignore
6DOF Trajectory
Labeled Point Tracks
Self-Supervision from VO
Supervision Signal
ConvNet

How to define Stability?

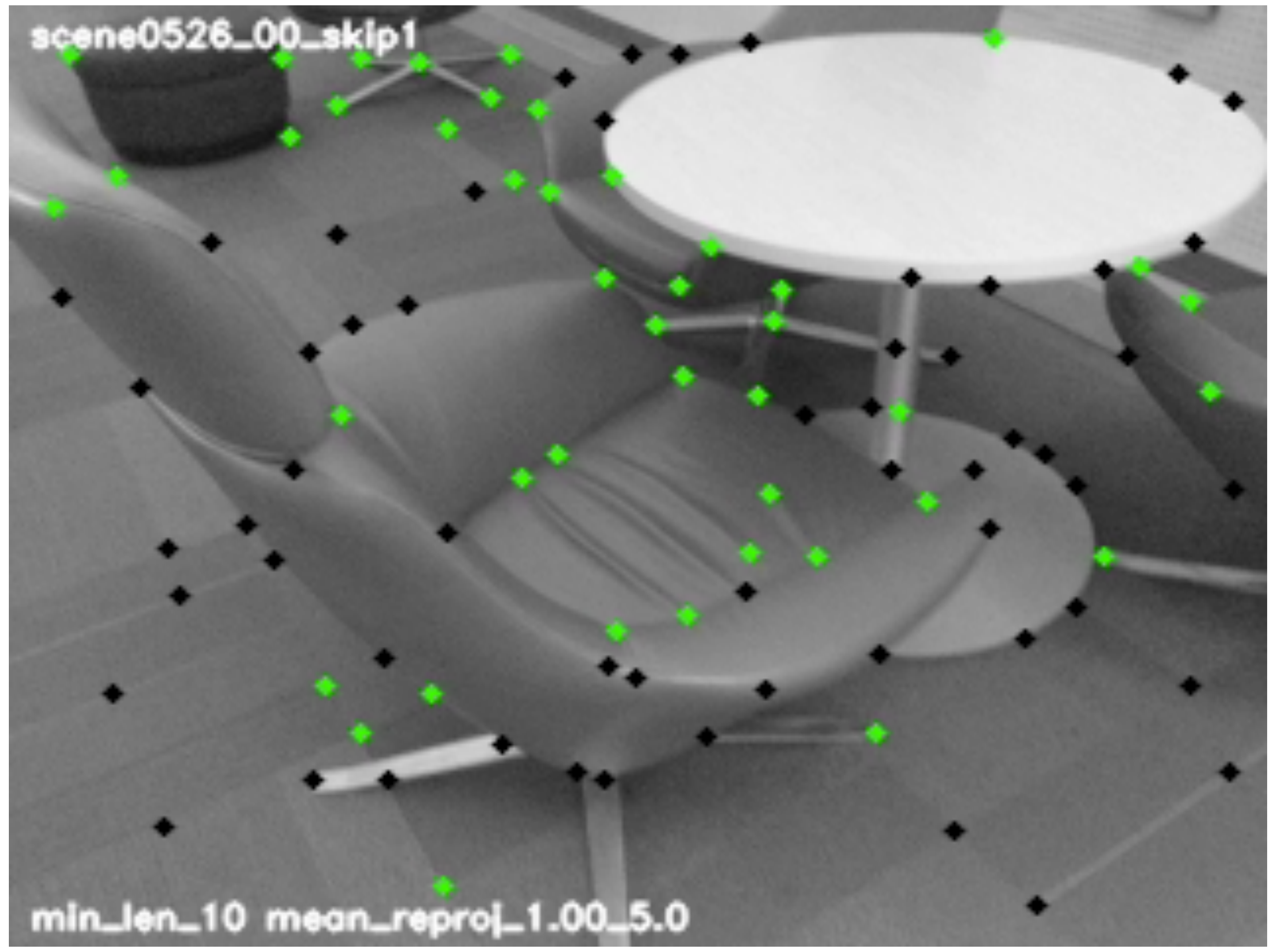
- For sufficiently long tracks, look at the reprojection error

$$X_{stable} = \begin{cases} \text{Stable} & \text{, if reprojection error is} < 1 \text{ pixel} \\ \text{Not Stable} & \text{, if reprojection error is} > 5 \text{ pixels} \\ \text{Ignore} & \text{, else} \end{cases}$$

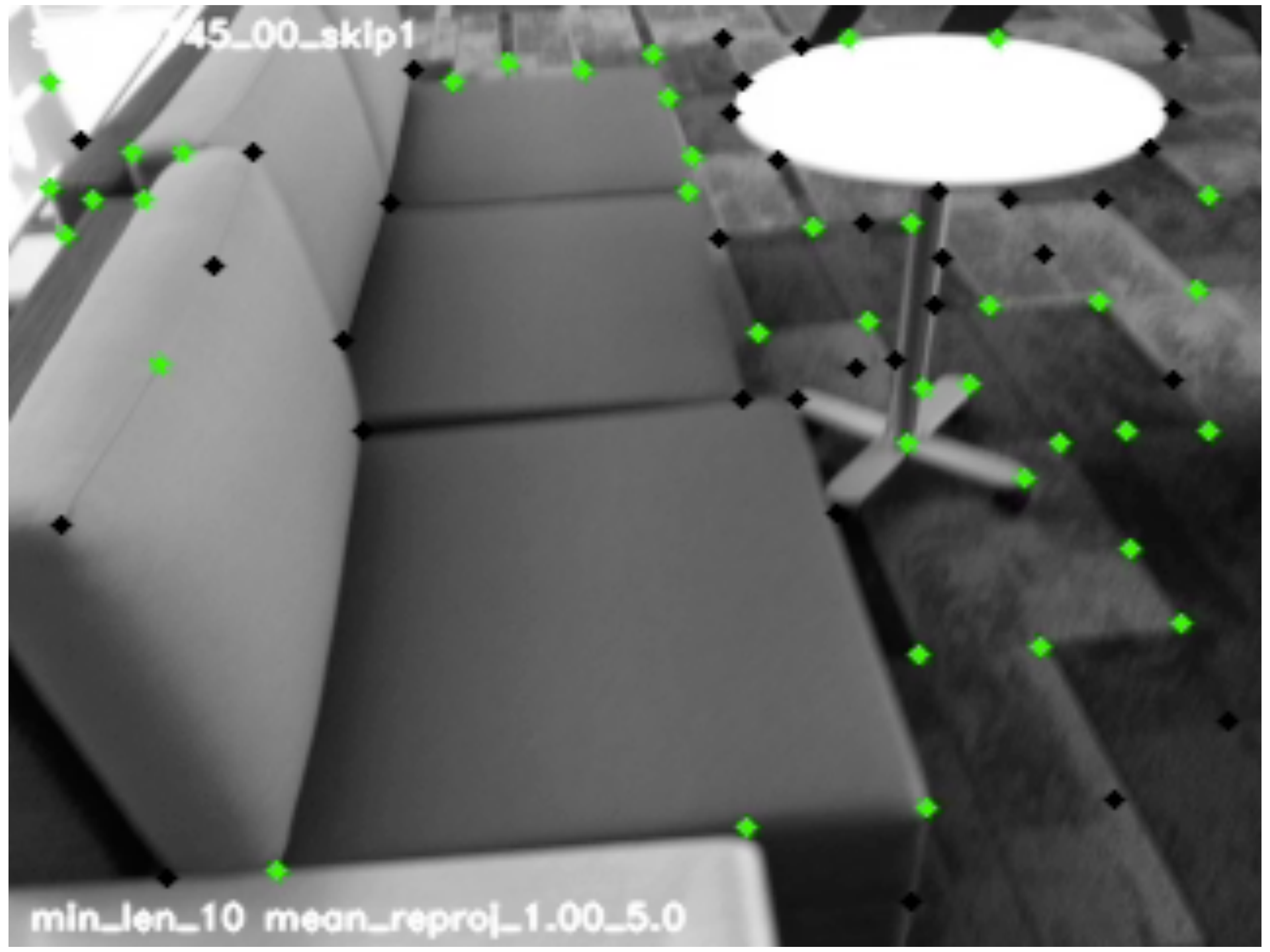
- **Stable Points: Positives**
- **Not Stable Points: Negatives**
- **Other Points: Ignore**

VO Stability Labeling

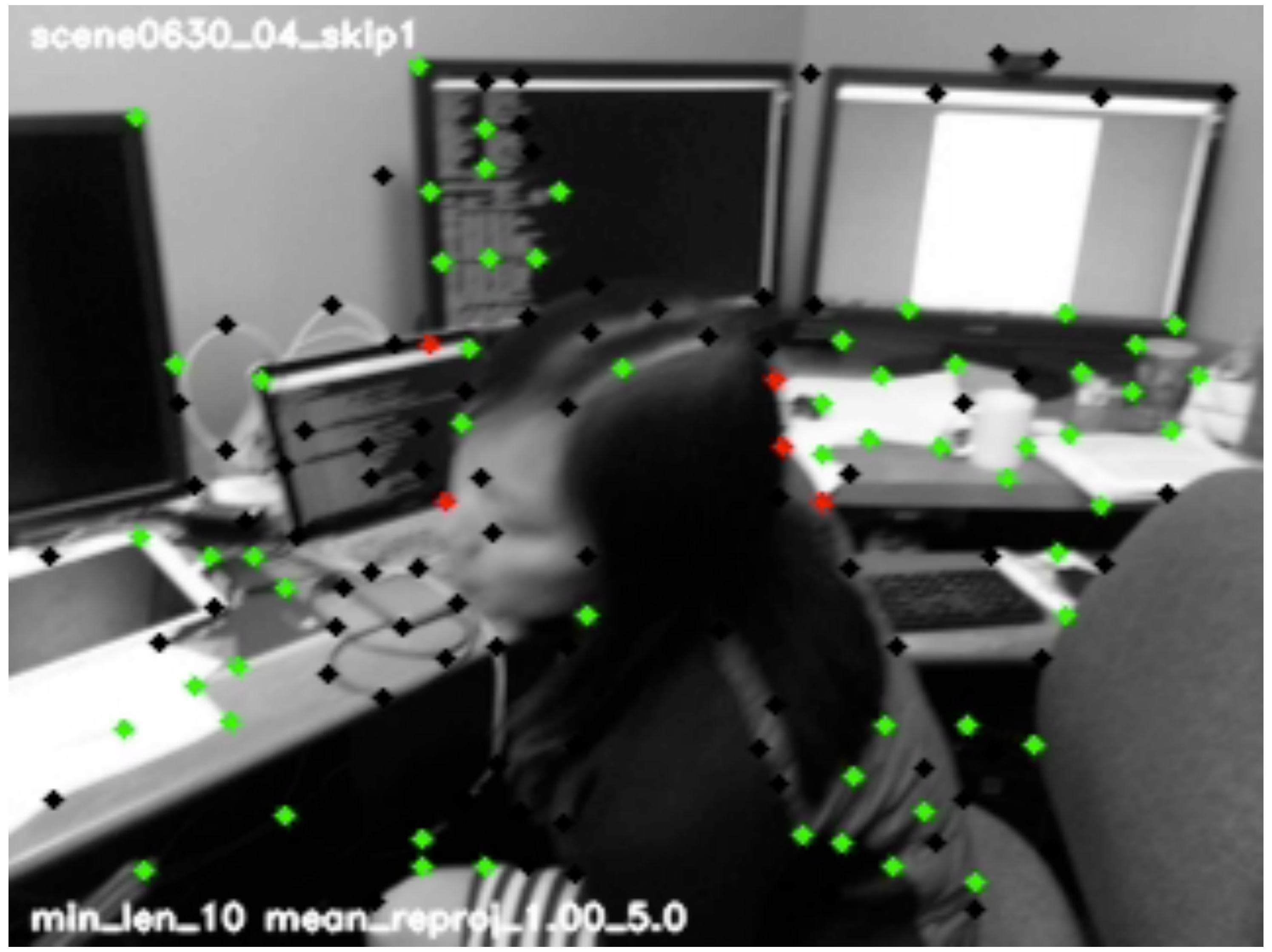
t-junctions across depth aka “sliders”



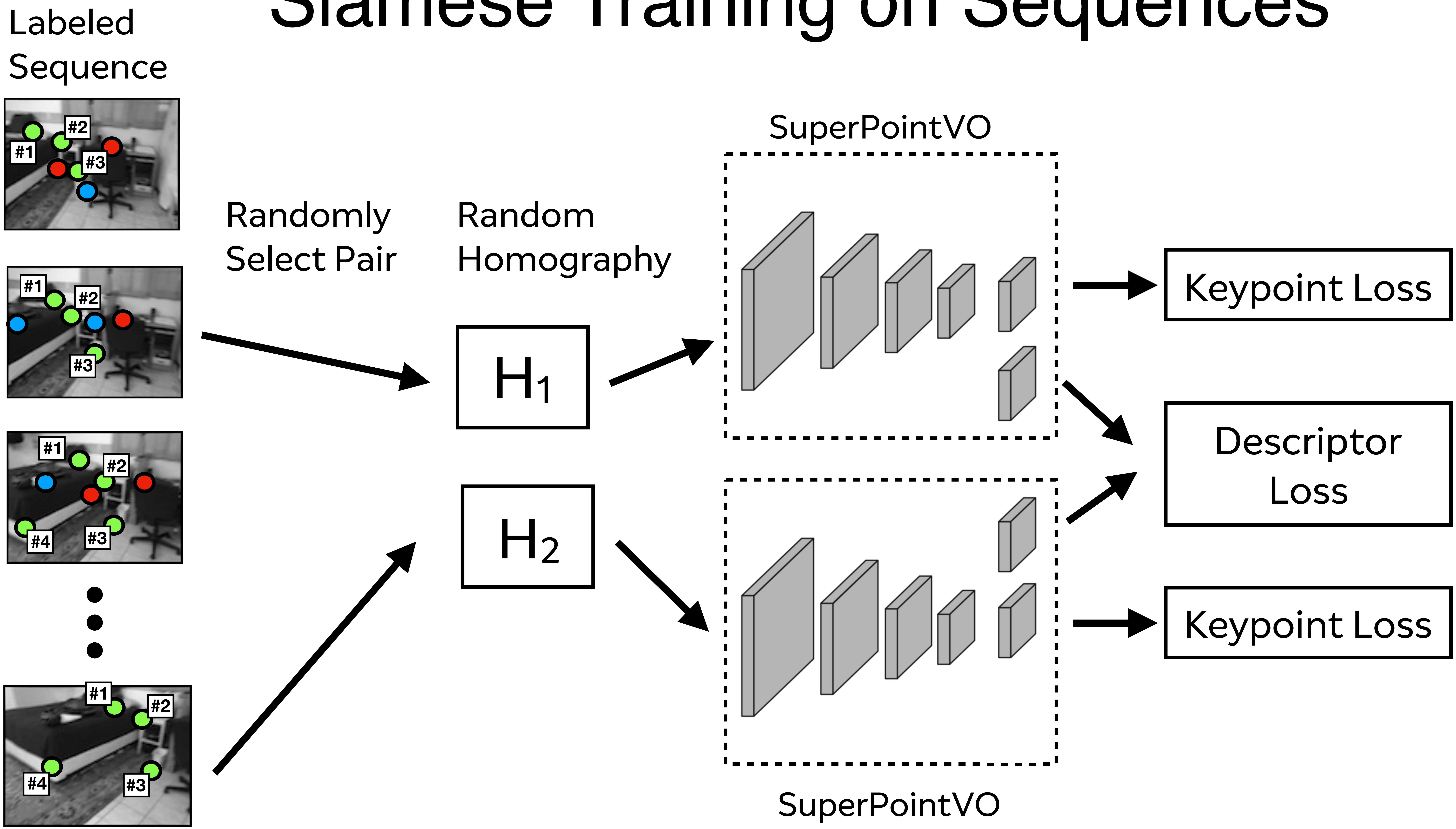
lighting highlights



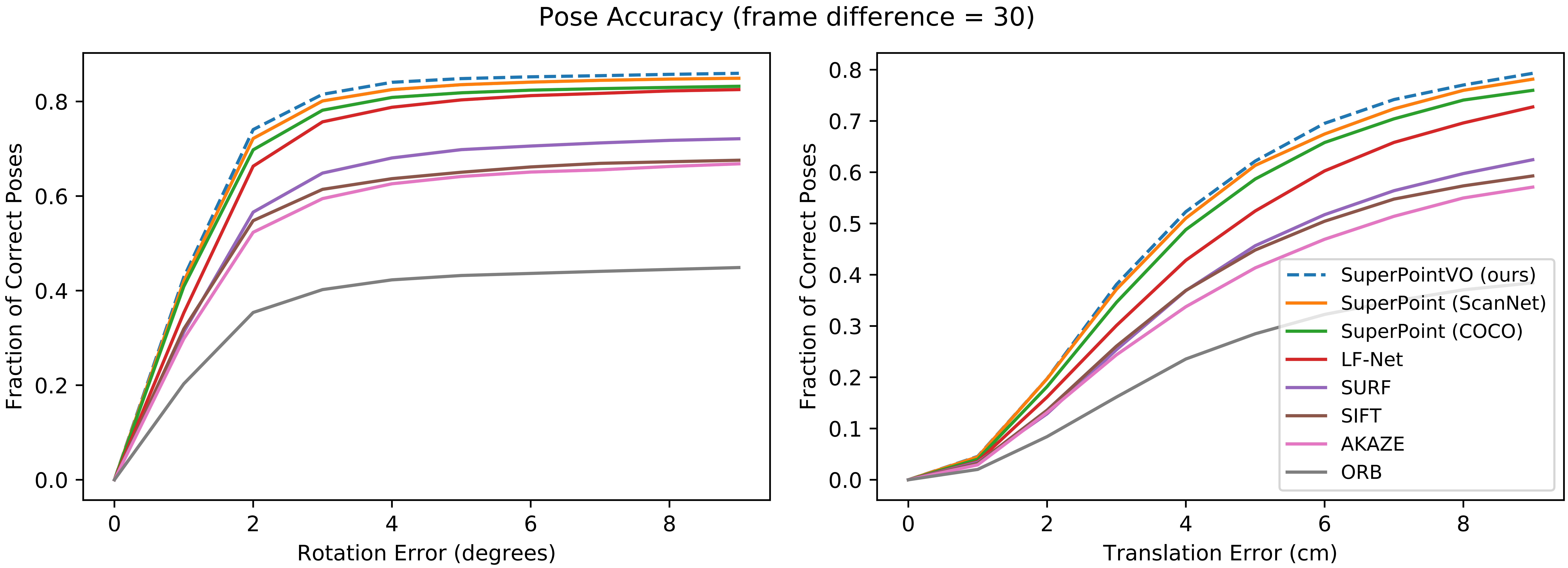
dynamic motion



Siamese Training on Sequences

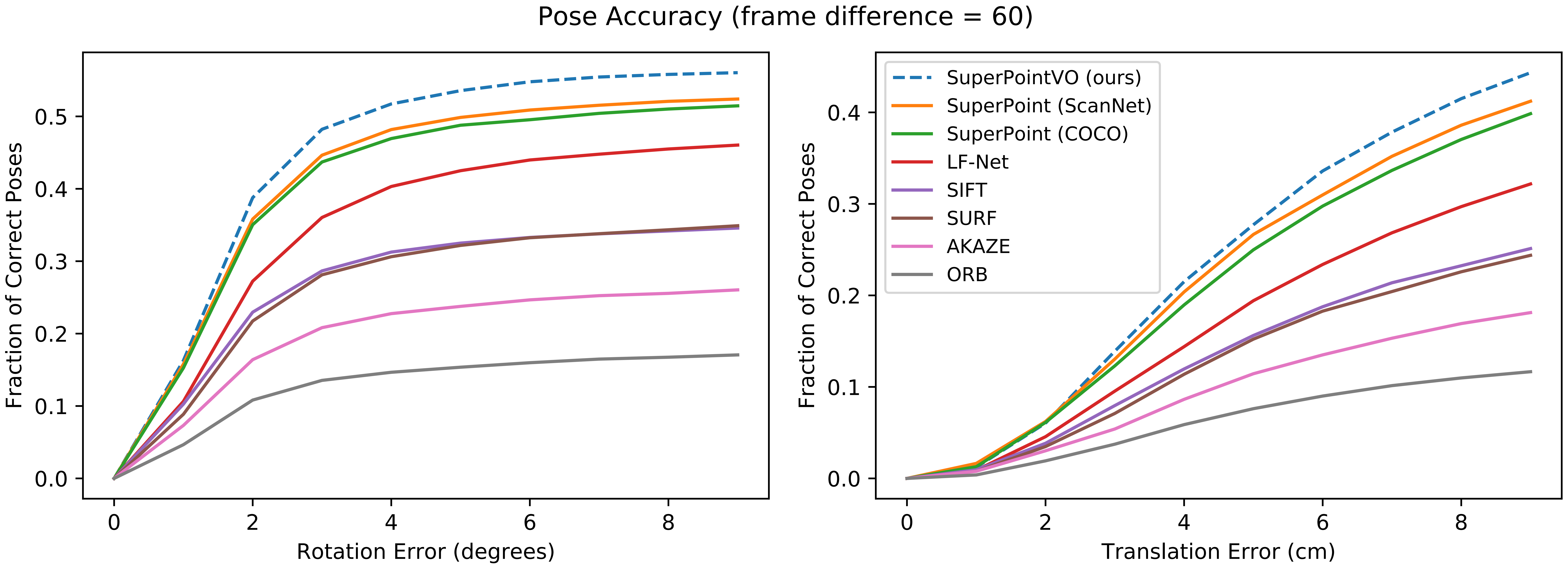


SuperPointVO: Pose Estimation on ScanNet



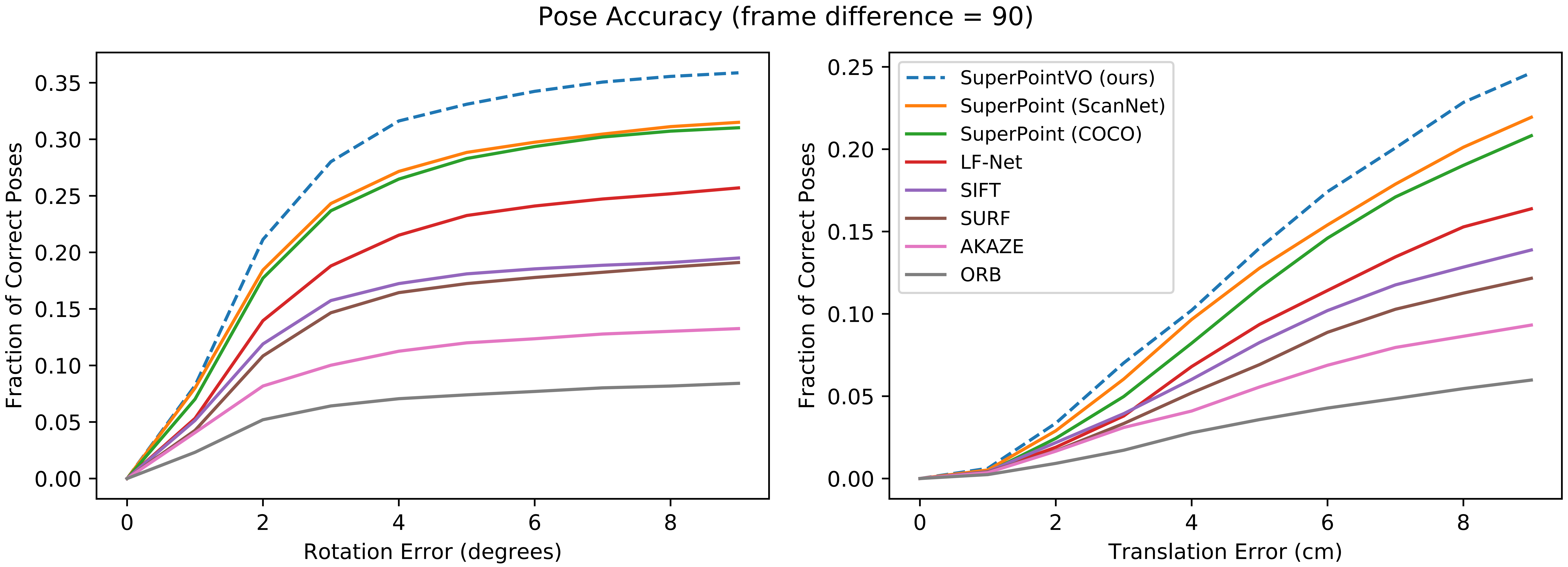
- Small baseline of ~1 second: VO helps a tiny bit

SuperPointVO: Pose Estimation on ScanNet



- Medium baseline of ~2 seconds: VO starts helping

SuperPointVO: Pose Estimation on ScanNet



- Widest baseline of ~3 seconds, biggest performance gap

Part II: SuperGlue

Deep Matching with SuperPoint: Can we learn to solve the correspondence problem?

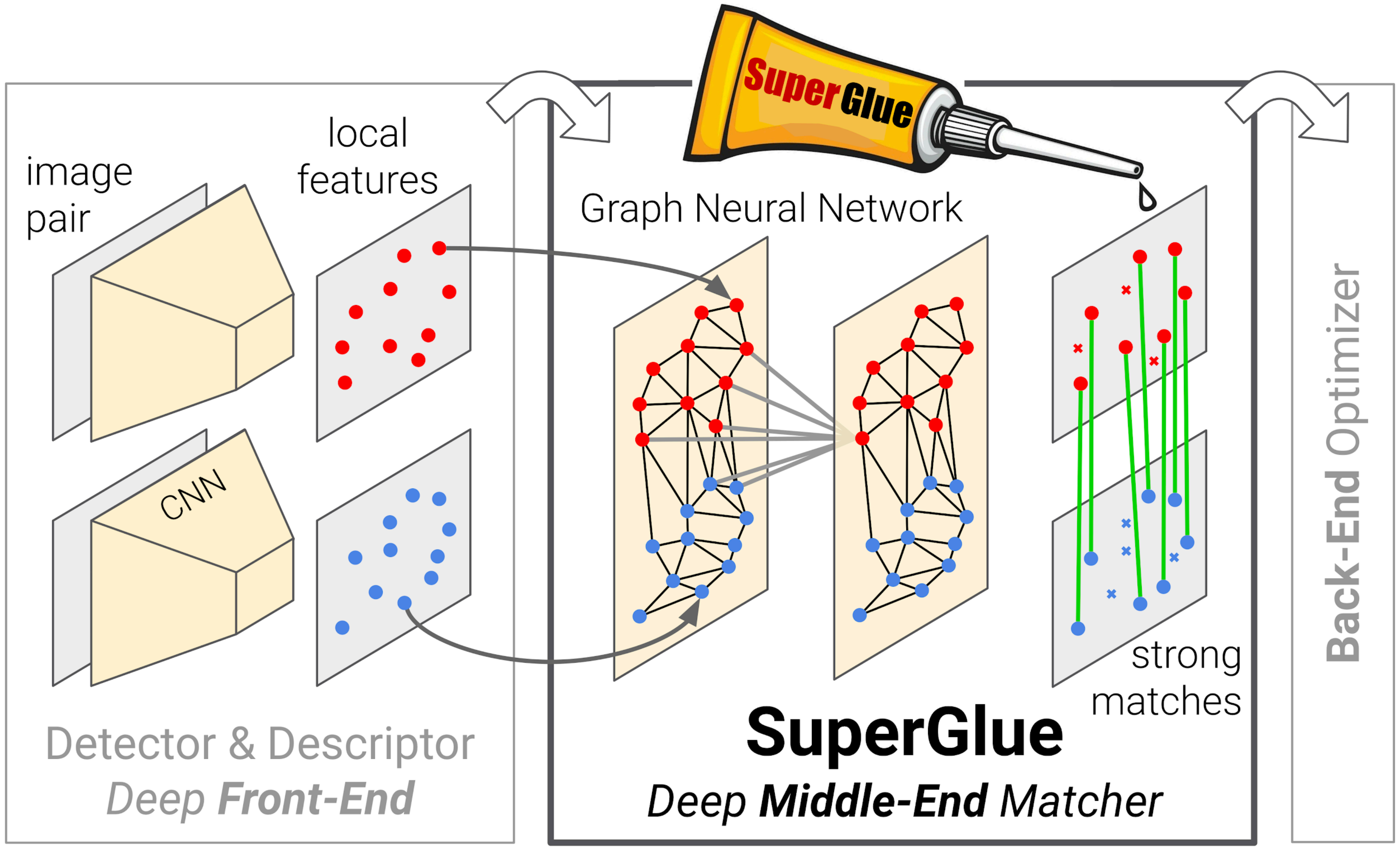
SuperGlue:
Learning Feature Matching with Graph Neural Networks

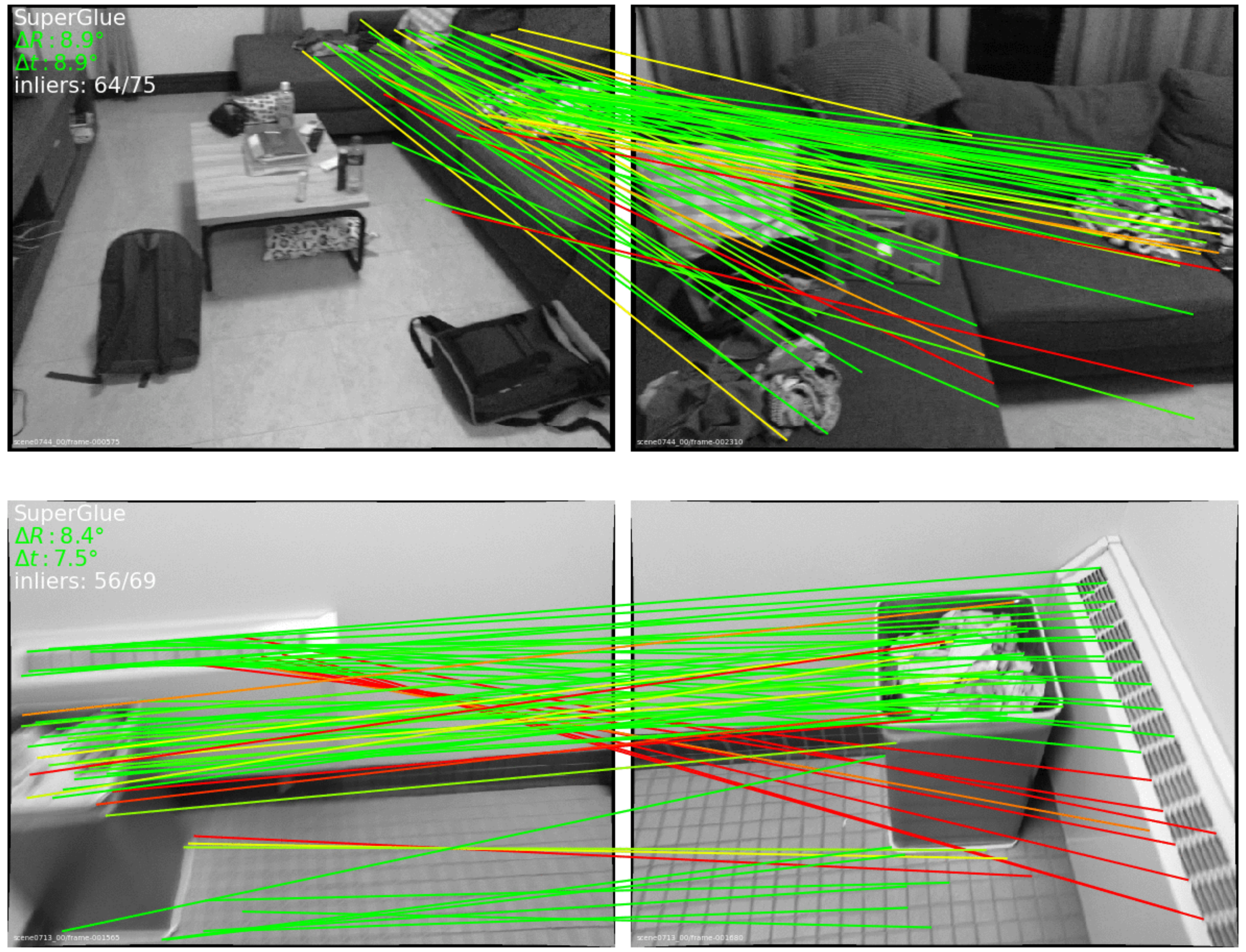
Paul-Edouard Sarlin[1] Daniel DeTone[2]
Tomasz Malisiewicz[2] Andrew Rabinovich[2]

ETH zürich
magic leap

Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A. SuperGlue: Learning Feature Matching with Graph Neural Networks. In CVPR, 2020.

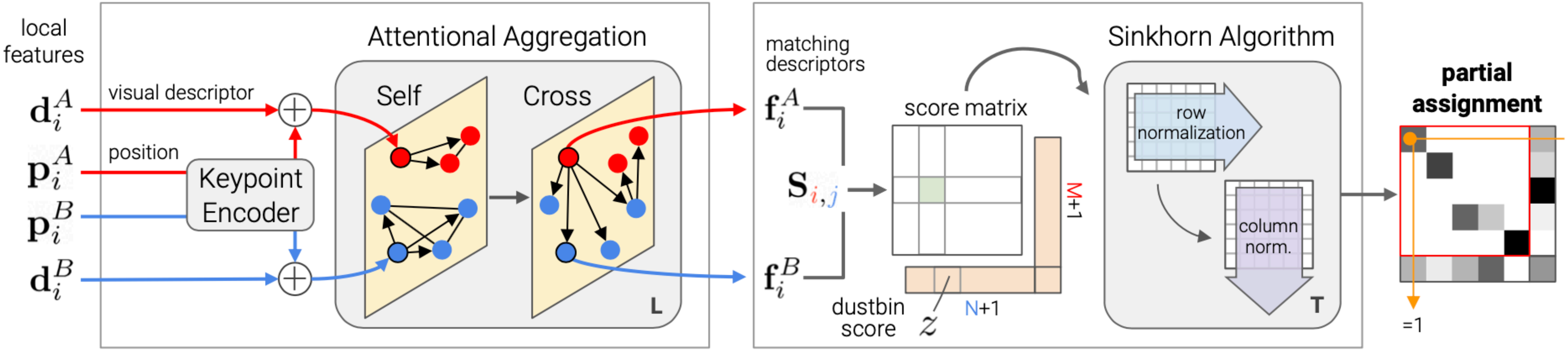
SuperGlue = Graph Neural Nets + Optimal Transport





- Extreme **wide-baseline** image pairs in **real-time on GPU**
- State-of-the-art **indoor+outdoor** matching with **SIFT** & **SuperPoint**

SuperGlue's goal is to be better than motion-guided matching without any motion model!



A Graph Neural Network with attention

Solving a partial assignment problem

Encodes **contextual cues** & priors

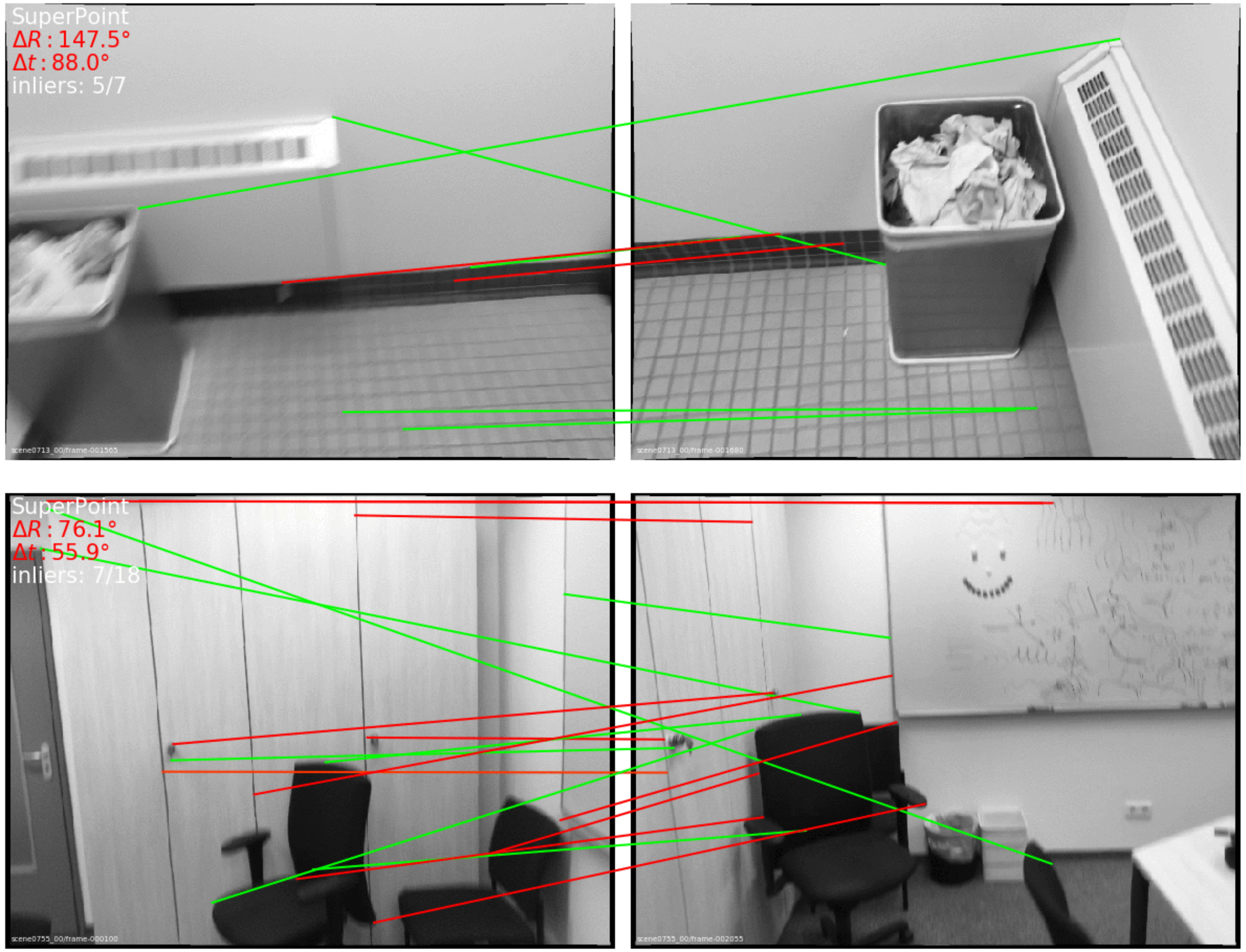
Reasons about the 3D scene

Differentiable **solver**

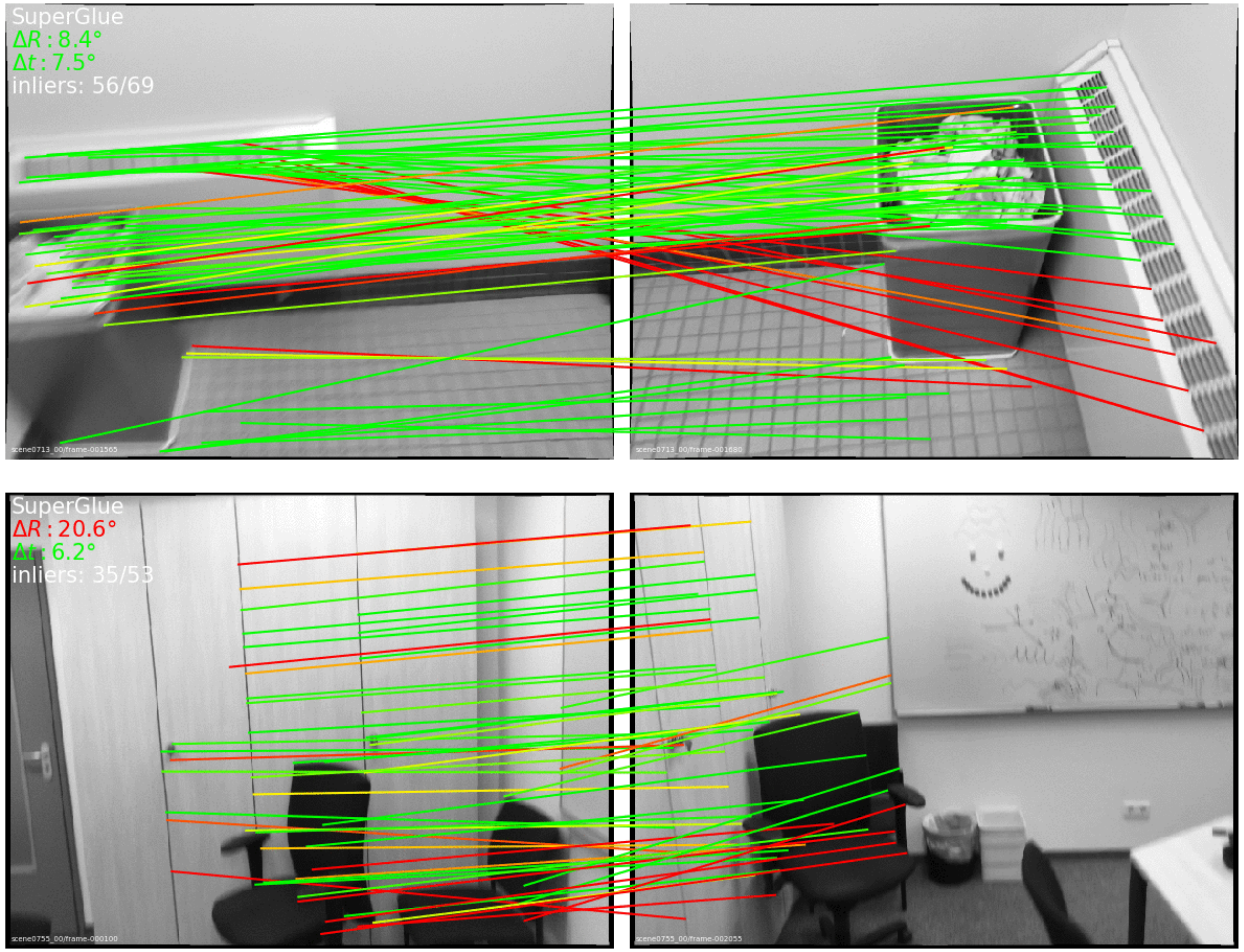
Enforces the assignment constraints = **domain knowledge**

SuperGlue requires both sets of local features: a paradigm shift in matching!

SuperPoint + NN + heuristics



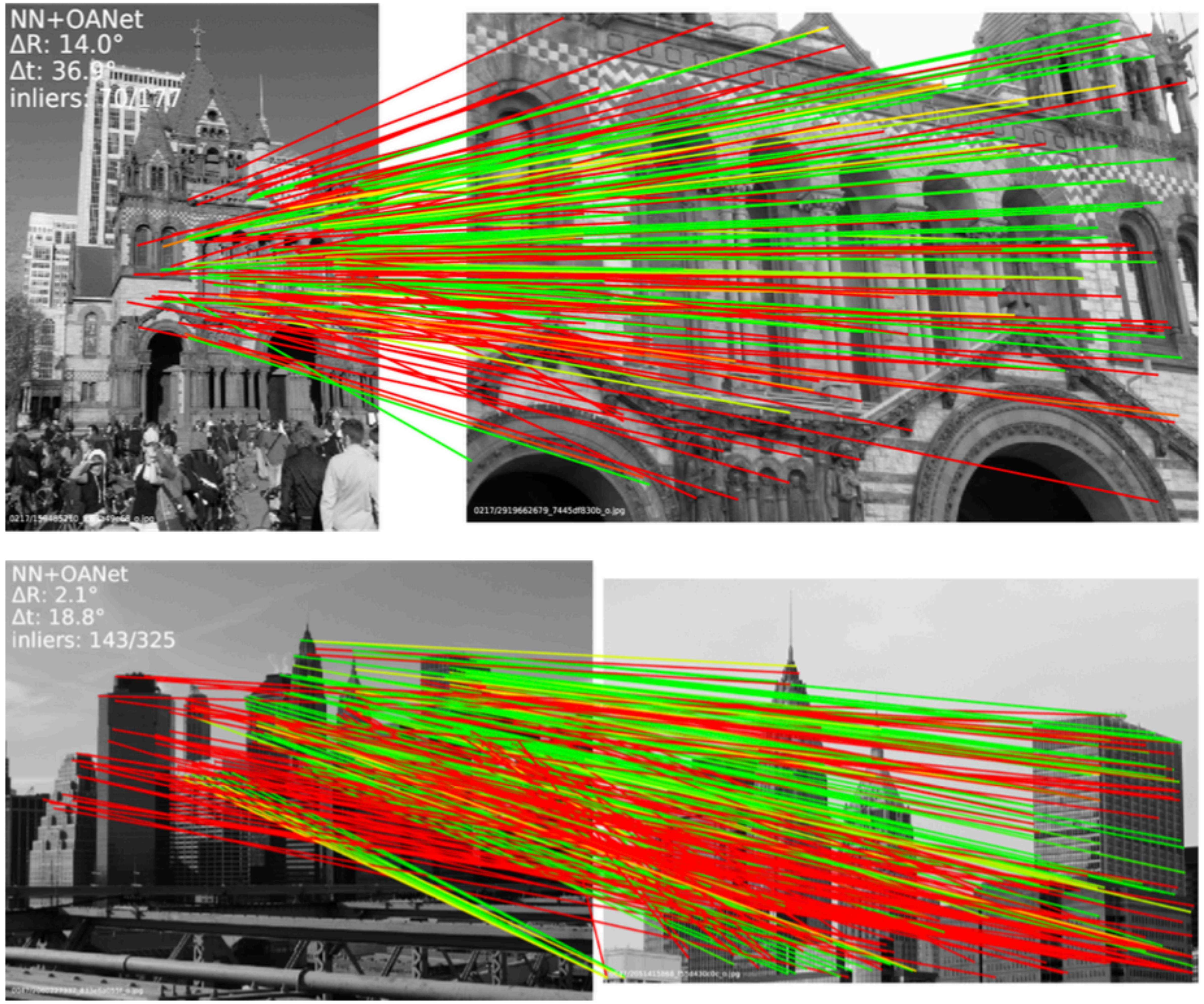
SuperPoint + **SuperGlue**



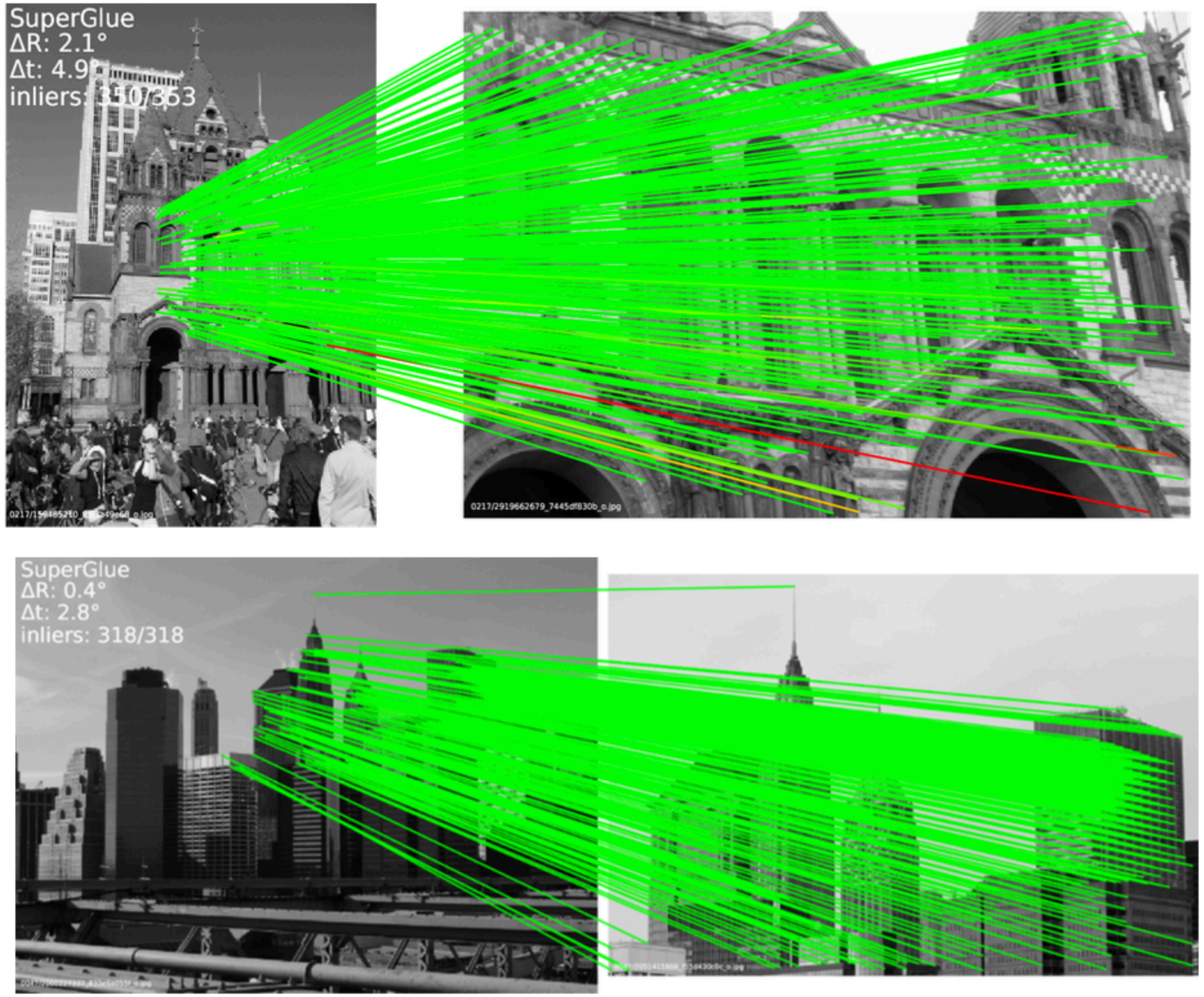
SuperGlue: more **correct matches** and fewer **mismatches**

Results: outdoor - SfM

SuperPoint + NN + OA-Net (inlier classifier)

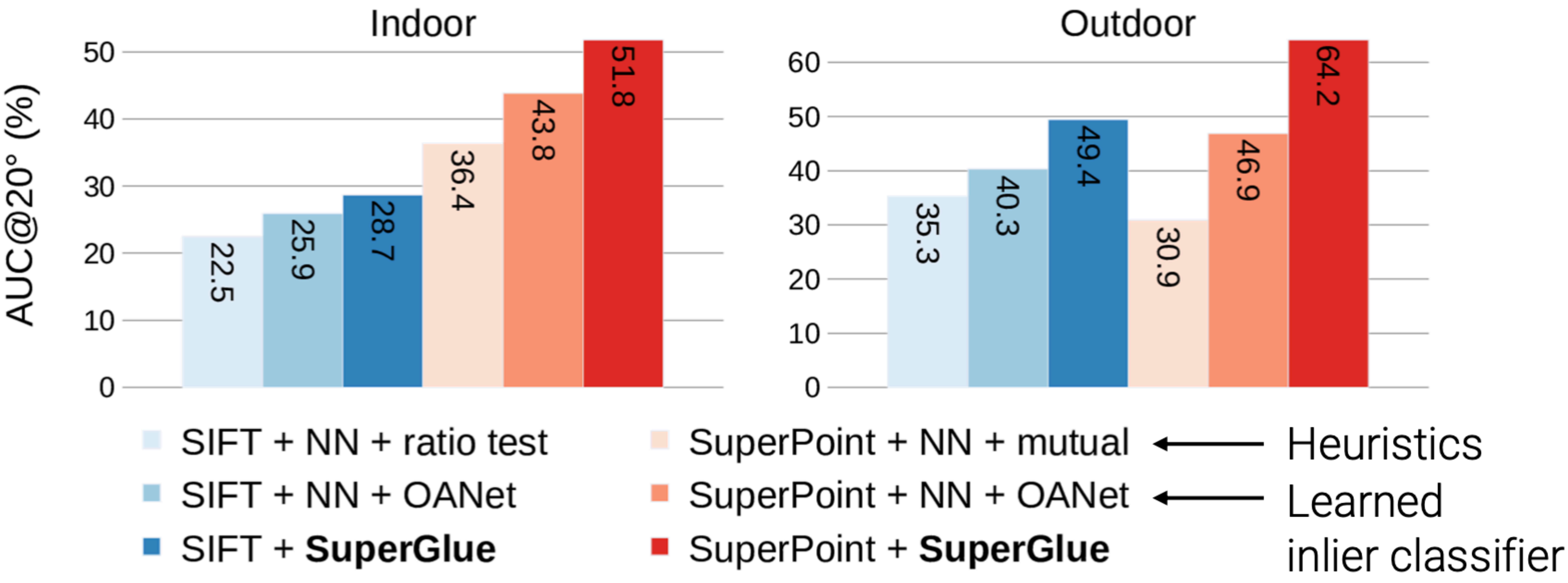


SuperPoint + **SuperGlue**



SuperGlue: more **correct matches** and fewer **mismatches**

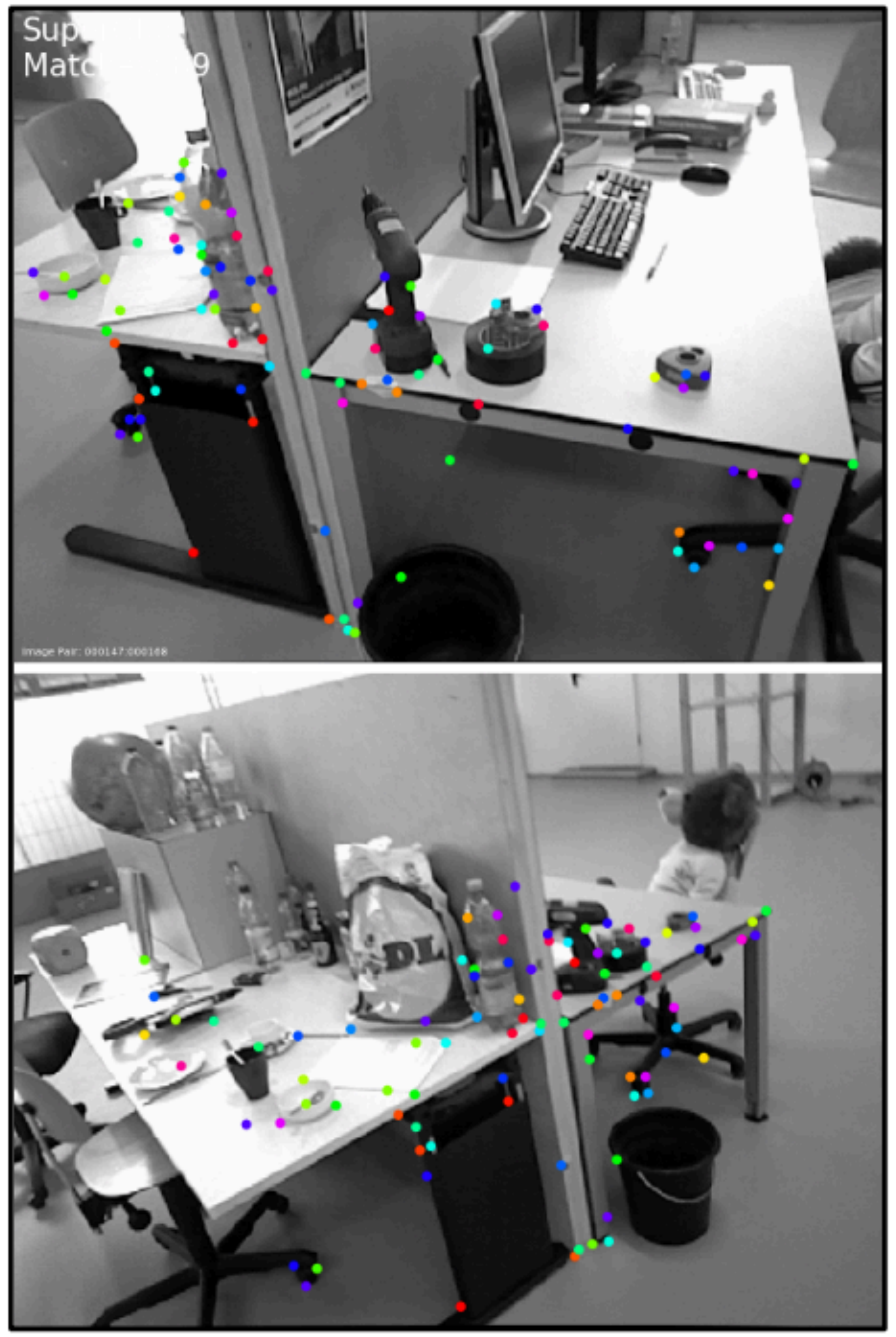
Evaluation

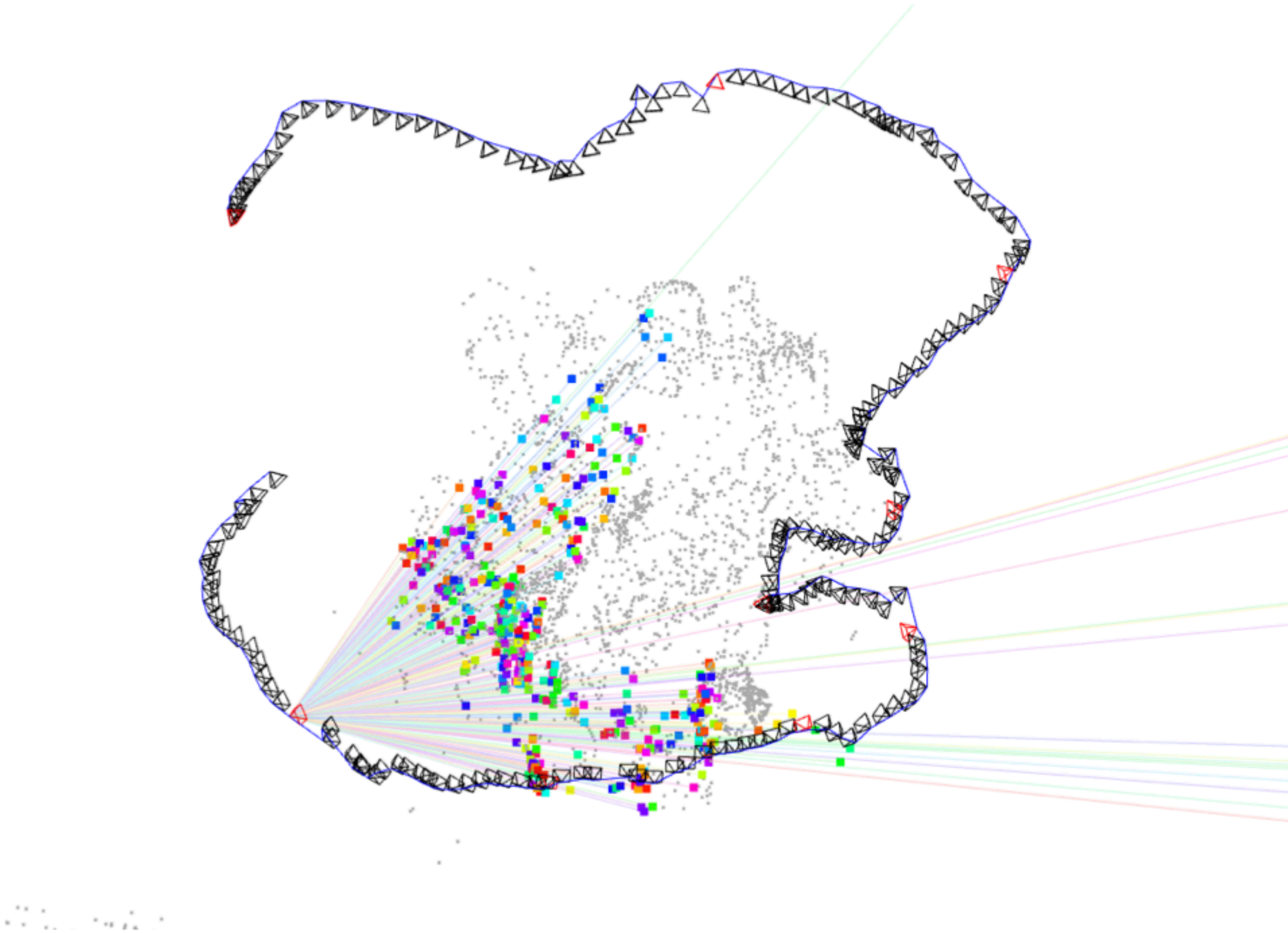


SuperGlue yields **large improvements** in all cases

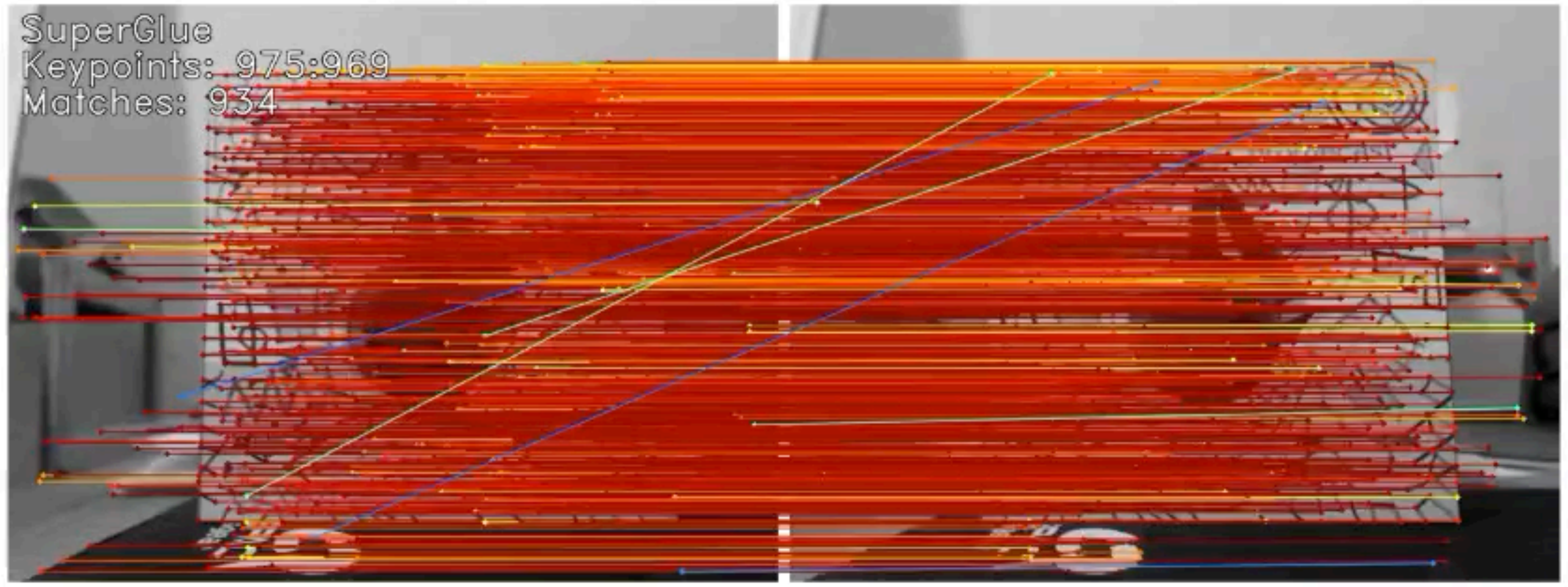
Demo: **15 FPS** for **512 keypoints** on GPU

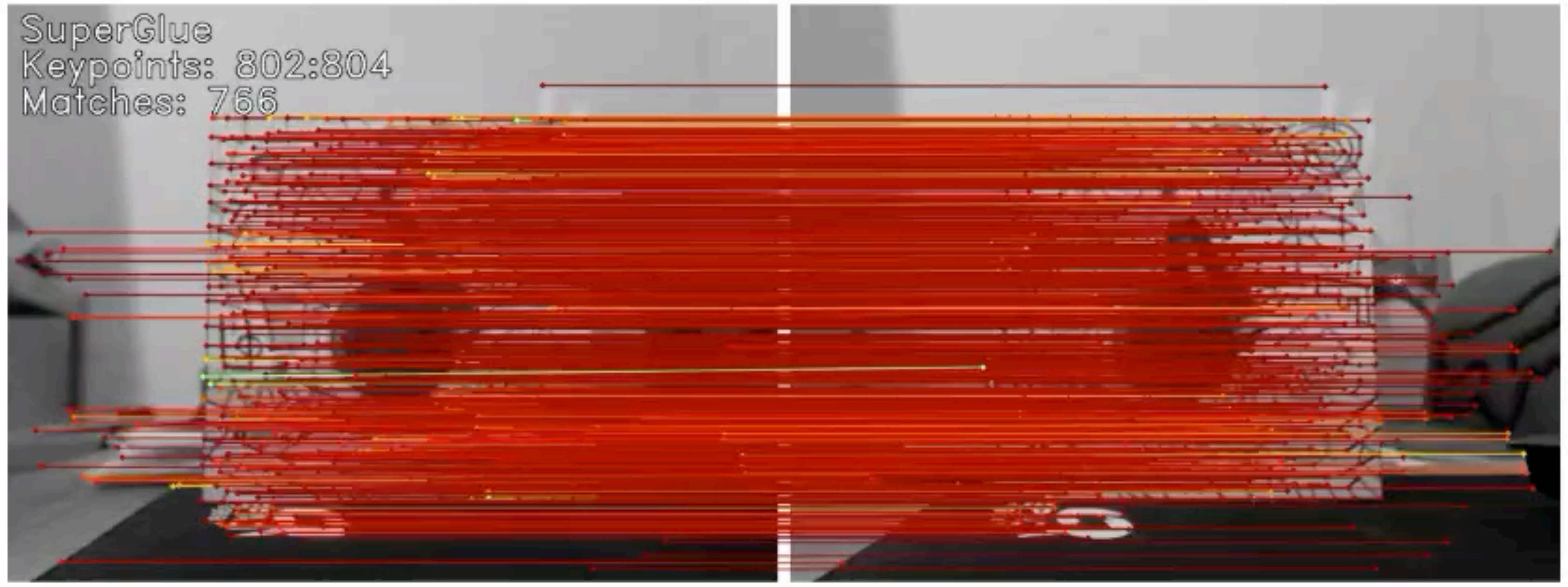
psarlin.com/superglue

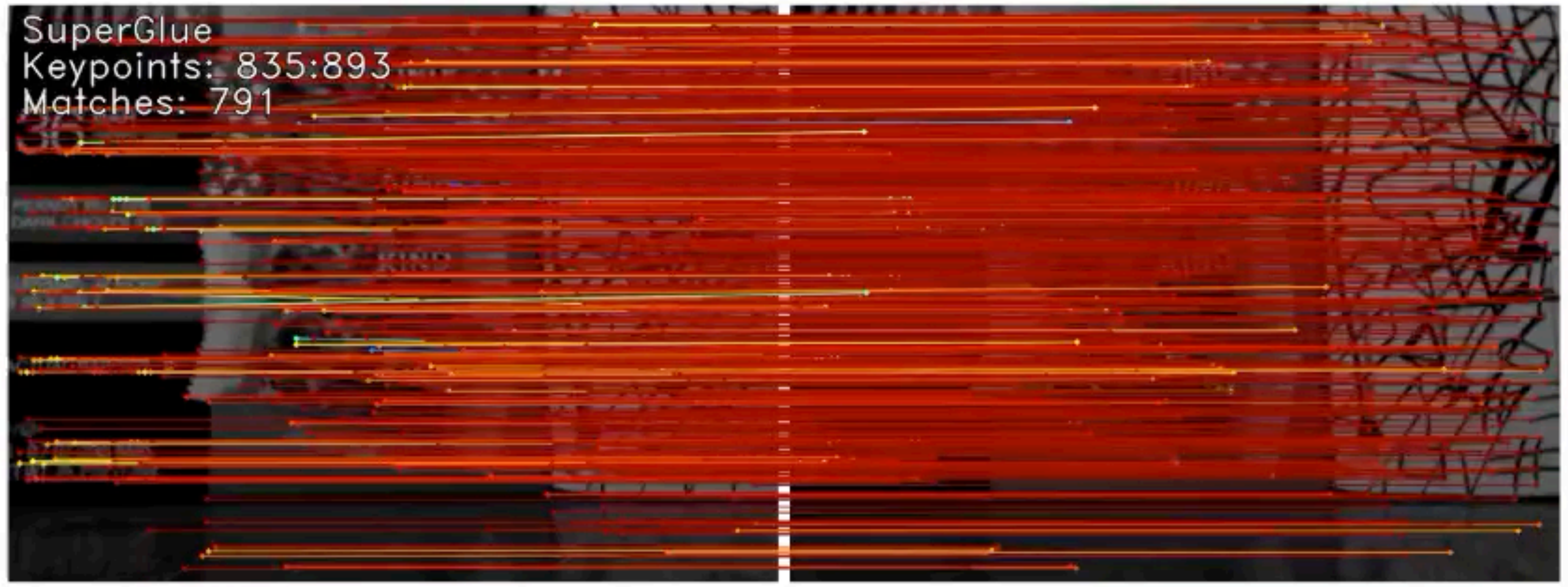


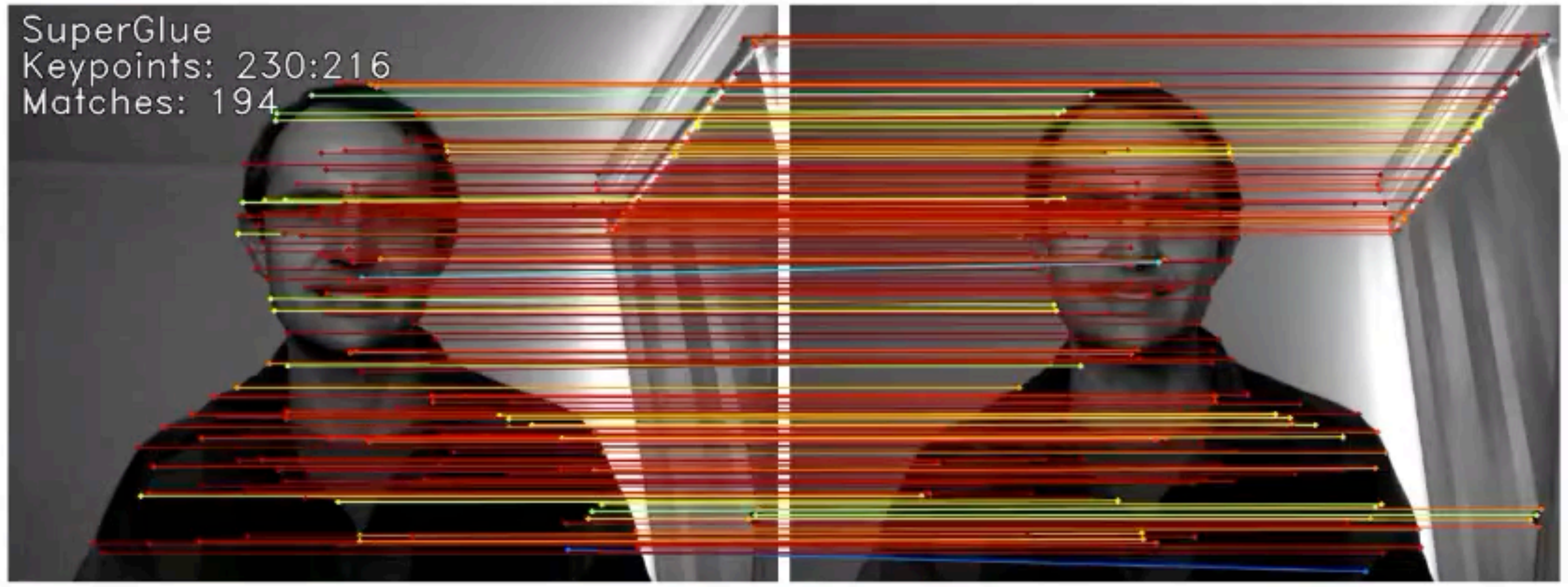


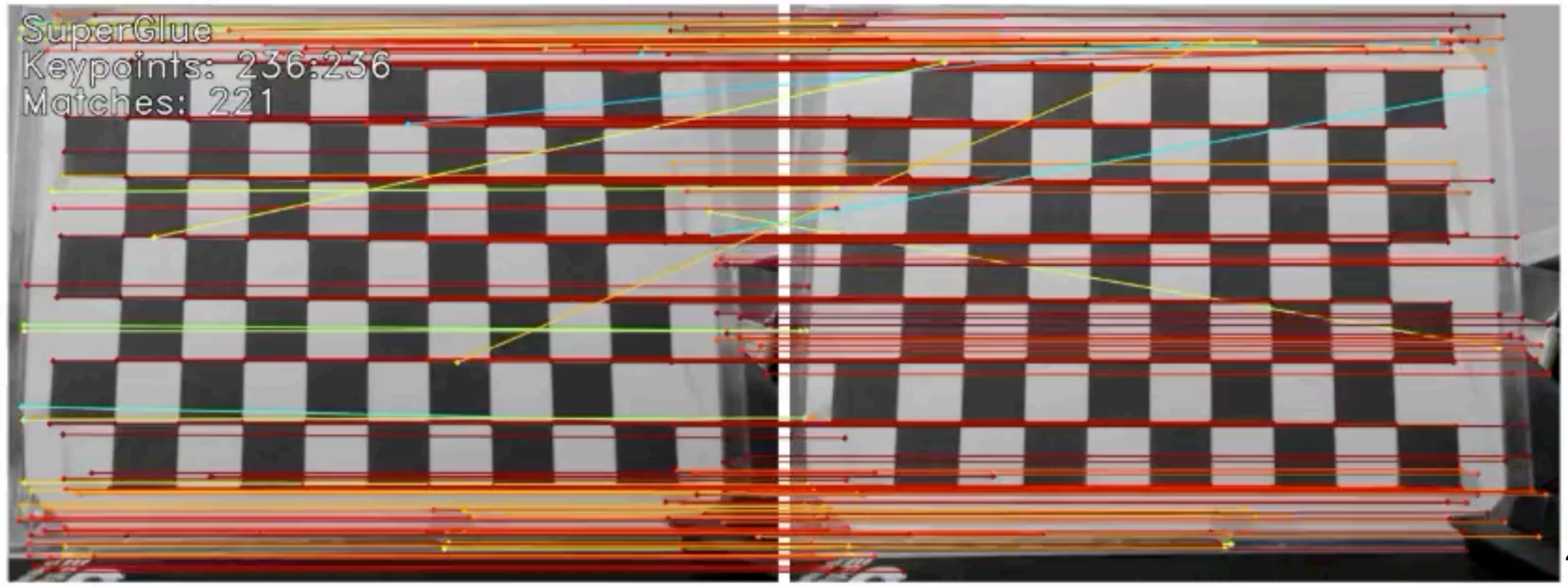
github.com/magicleap/SuperGluePretrainedNetwork

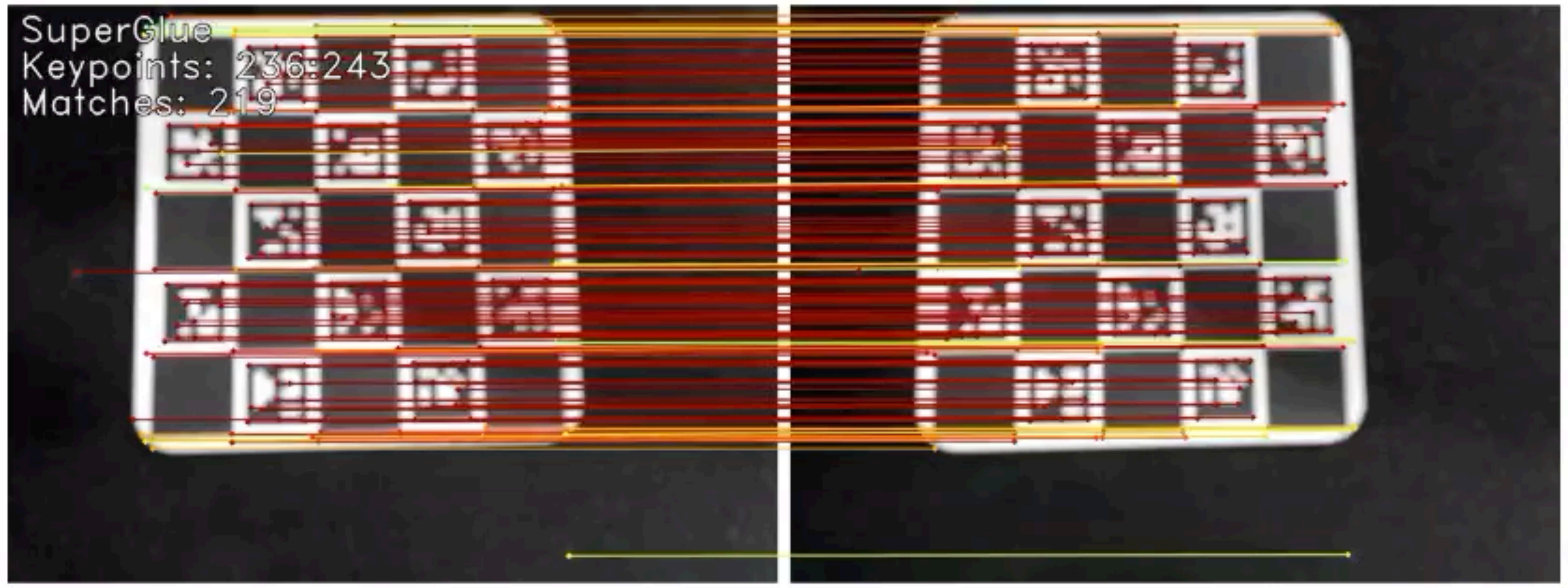

SuperGlue
Keypoints: 975:969
Matches: 934


SuperGlue
Keypoints: 802:804
Matches: 766


SuperGlue
Keypoints: 835:893
Matches: 791


SuperGlue
Keypoints: 230:216
Matches: 194


SuperGlue
Keypoints: 236:236
Matches: 221


SuperGlue
Keypoints: 236:243
Matches: 219

Part III: SuperMaps

What comes after SuperPoint + SuperGlue?

SuperPoint+SuperGlue	SuperMaps
Works with a **pair** of images	Works with a **set** of images
Uses **classical** pose estimation system	Estimates pose **inside** the network
No loop closure mechanism	**Keyframe embeddings** to close loops
Modules trained **independently**	Joint **end-to-end training**
Has **multiple** notions of receptive field	A **unified** notion of receptive field

Quō vādis Visual SLAM?

(some open problems at the intersection of DL and SLAM that will drive innovation)

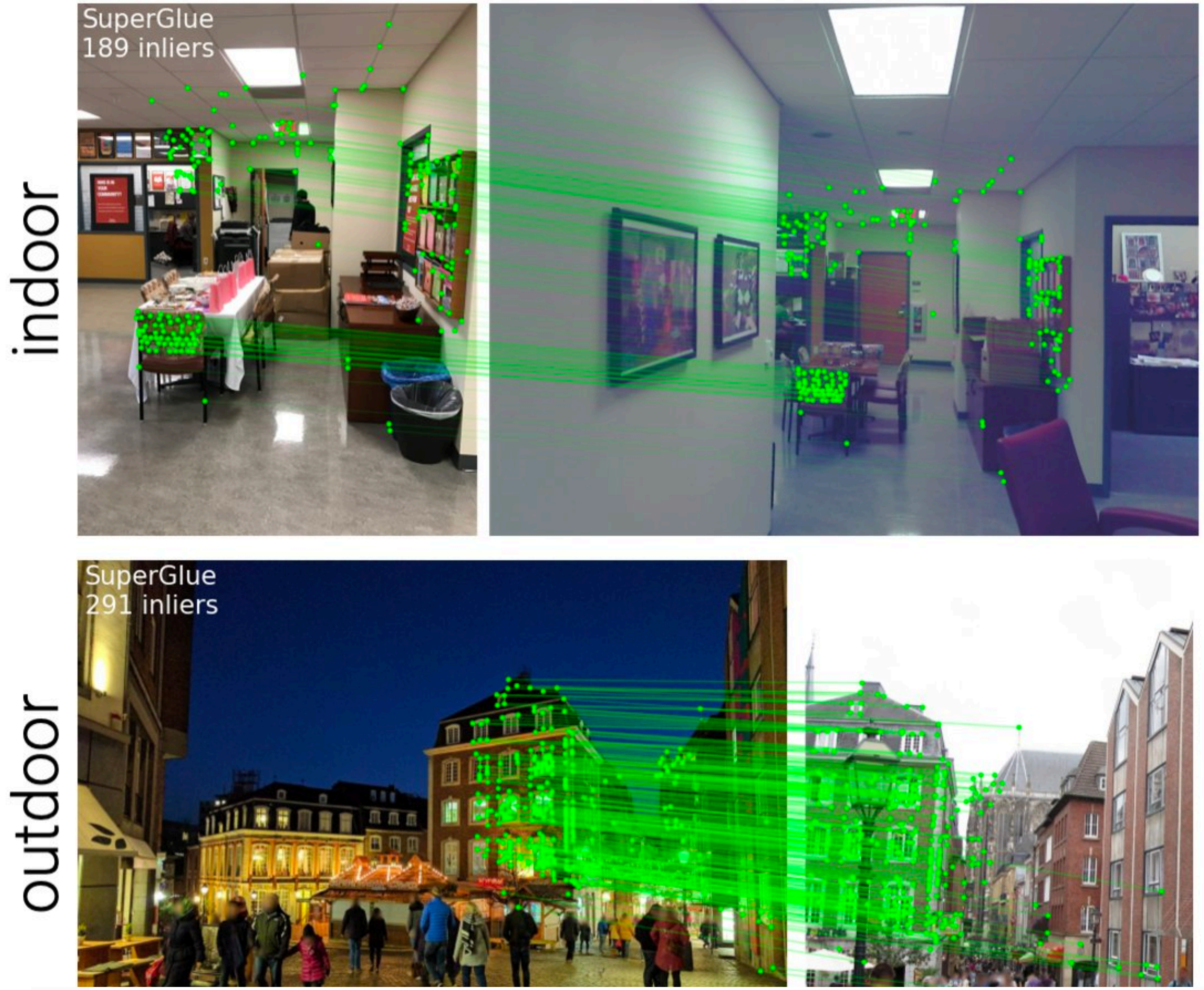
1. Multi-user SLAM: Creating representations/maps that work across a large number of agents

2. Integrating object recognition capabilities into SLAM frontends

3. Enabling life-long learning: letting the system automatically improve over time

Summary

- **SuperPoint:** A Convolutional Neural Network Architecture for Visual SLAM frontends
  - *Self-Supervised Learning via*:
    - Homographies
    - Visual Odometry Backend
  - CharucoNet: Pattern-specific SuperPoints: can “see” in the dark
- **SuperGlue:** Amazing success in applying Graph Neural Networks and Attention to wide baseline image matching problems
- **SuperMaps:** Ideas for going beyond pairwise matching and end-to-end SLAM



SuperGlue

Learning Feature Matching with Graph Neural Networks

CVPR 2020 Oral

1st place
in 2 visual localization challenges

Joint Workshop on Long-Term Visual Localization, Visual Odometry and Geometric and Learning-based SLAM

Winning entry:
restricted keypoints (2k) / standard descriptors (512 bytes)

SuperGlue Presentations @ CVPR 2020

This Workshop

Local Feature Challenge
Monday, June 15th: 9:10am PT

Handheld Devices Challenge
Monday, June 15th: 9:35am PT

3D Scene Understanding for Vision, Graphics, and Robotics Workshop
Monday, June 15th: 10:25 am PT

CVPR 2020 Oral Presentation
Wednesday, June 17th: 10:40 am PT & 10:40 pm PT

Image Matching: Local Features & Beyond Workshop
Friday, June 19th: 11:45 am PT

Paul-Edouard Sarlin
ETHZ Ph.D. Student

Thank you

Tomasz Malisiewicz
https://tom.ai/

@quantombone

Daniel DeTone
https://danieldetone.com/

@ddetone

Paul-Edouard Sarlin
https://psarlin.com/

@pesarlin

Follow us on Twitter:

Research Questions: