Visual Computing Center (Computer Science and Engineering),
Department of Electrical and Computer Engineering,
UC San Diego

Analysis of Geometry and Deep Learning-based Methods for Visual Odometry

A Thesis Defense
by
You-Yi Jau

Professor Manmohan Krish Chandraker (Chair)
Professor Nikolay A. Atanasov (Co-Chair)
Professor Hao Su
Professor Nuno M. Vasconcelos

Outline

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Outline

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Why Visual Odometry?

Autonomous driving
- Waymo, Tesla

Virtual reality
- HoloLens, Oculus

Augmented reality
- Magic Leap

https://www.pcmag.com/news/waymo-gets-first-permit-to-test-fully-driverless-cars-in-california

https://www.uctoday.com/collaboration/the-future-is-now-everything-you-need-to-know-about-microsoft-hololens/

https://circuitstream.com/blog/magic-leap-breakdown/

Problem formulation

Camera

Images

Where am I

What does the world look like

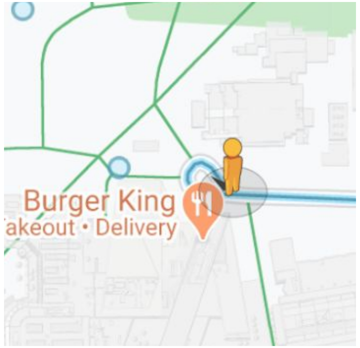


Image source: https://www.amazon.com/Appbot-Riley-Controlled-Movable-Safety/dp/B01LWXF28H

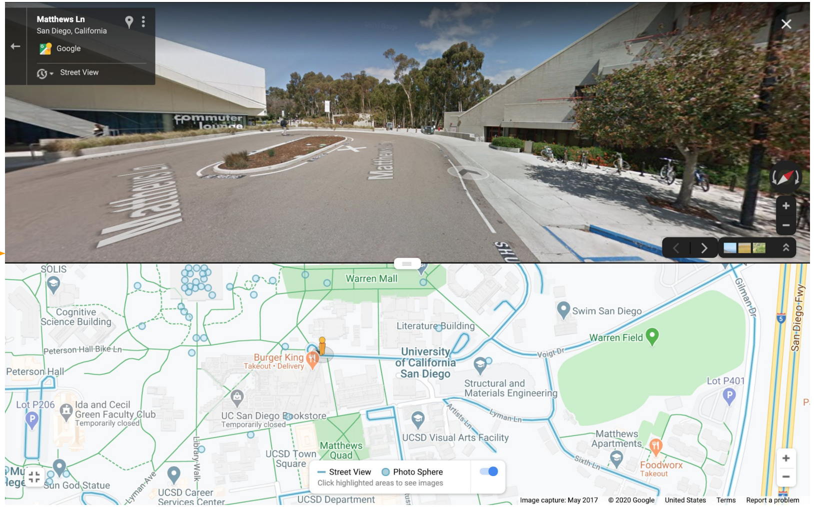
Image source: KITTI dataset

Image source: https://www.amazon.com/Smart-Tools-co-Compass/dp/B00K3HHA9Y

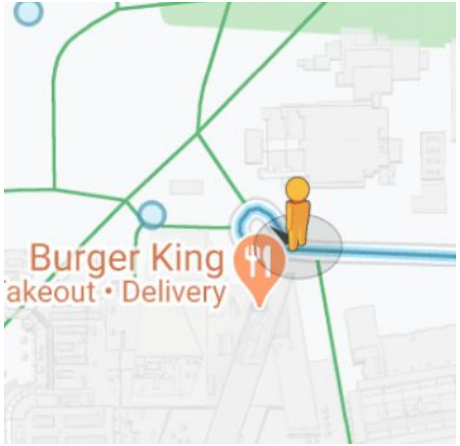
Driving to Price center

Driving to Price center

Moving from image A to image B



Camera pose in six degrees of freedom (6 DoF)



Position (3 DoF)

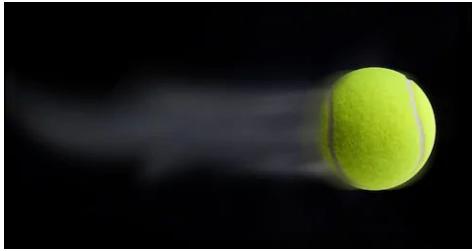
+

Orientation (3 DoF)

Camera pose in mathematical representation

Rotation	Matrix	translation

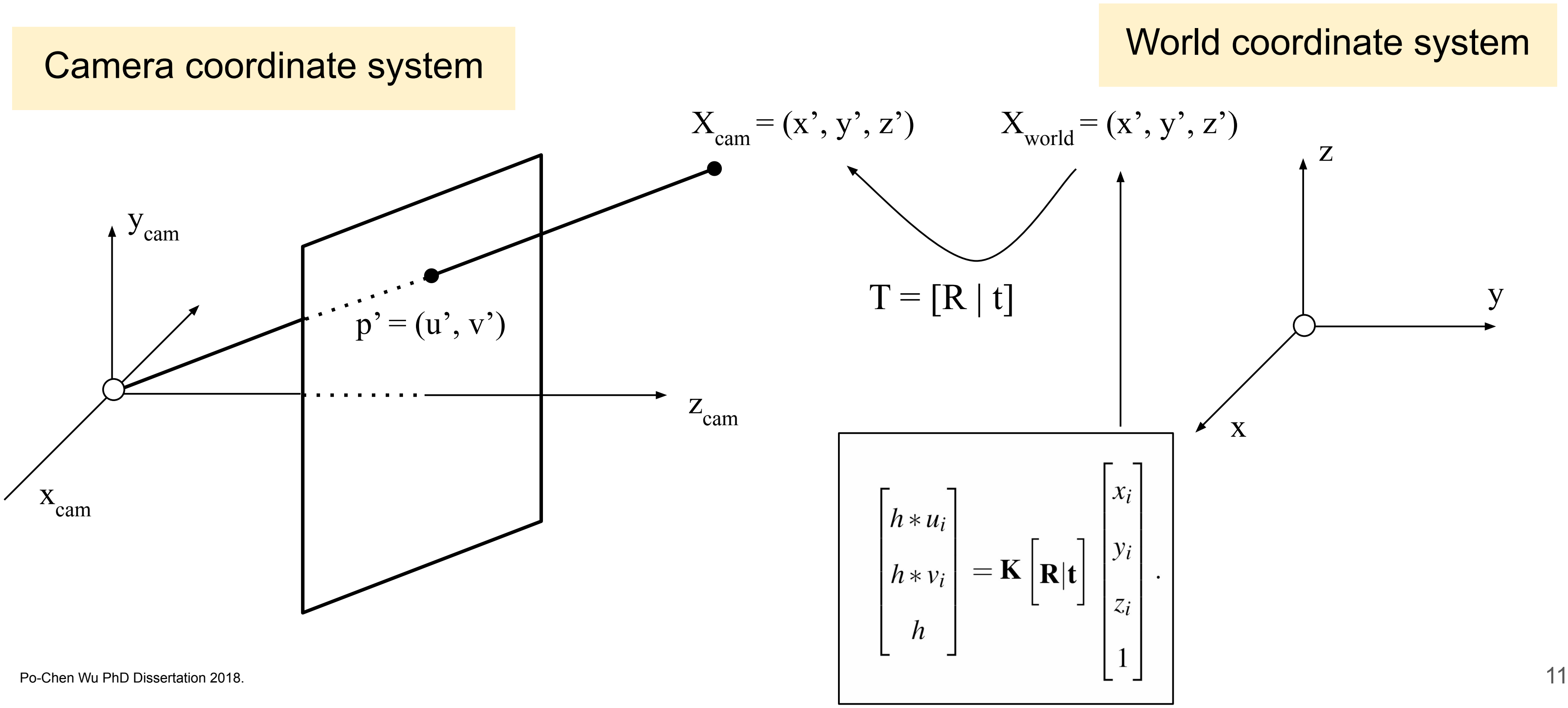
$$\tilde{\mathbf{T}} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$$



https://www.wired.com/2015/09/mind-bending-physics-tennis-balls-spin/

https://www.shutterstock.com/search/tennis+ball+moving%5D?image_type=photo

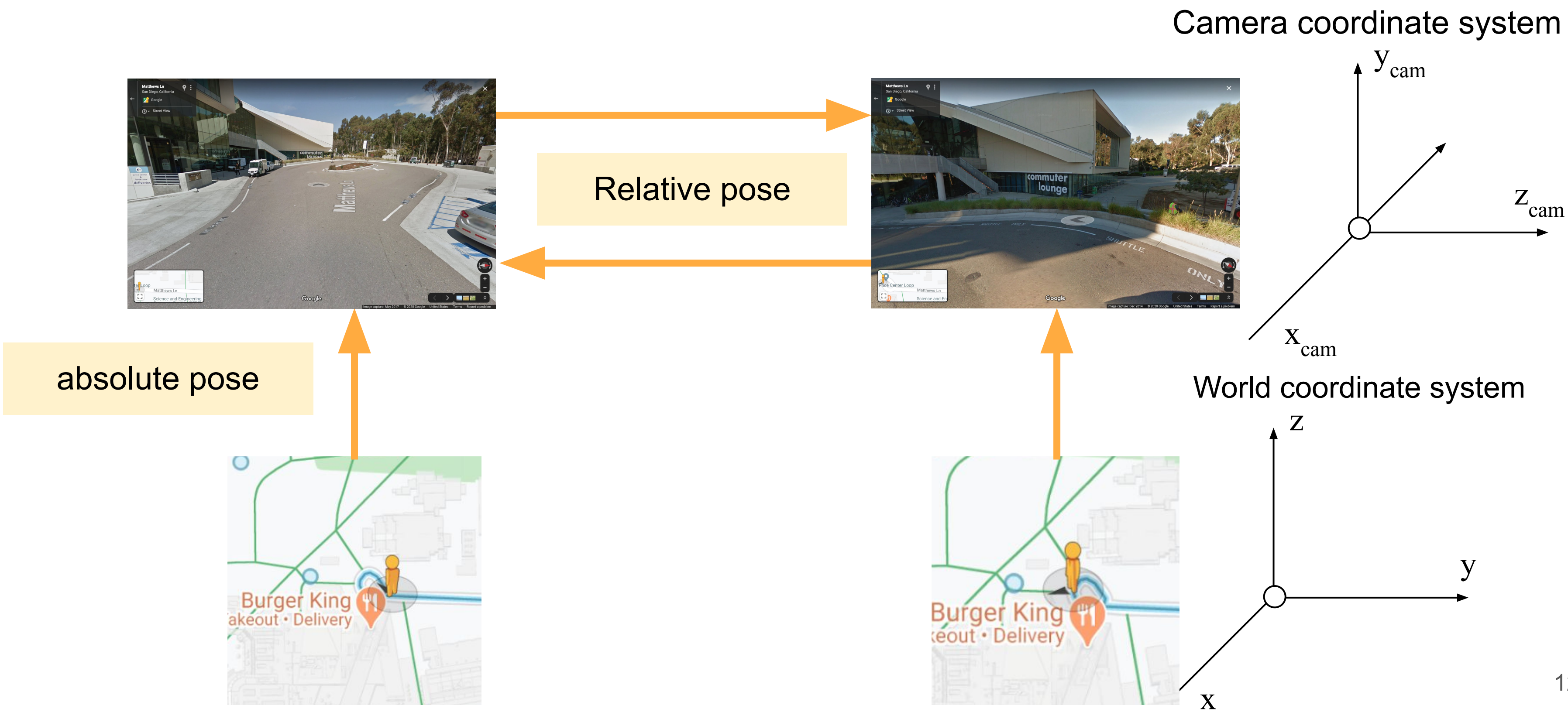
Camera projection model



$$\begin{bmatrix} h * u_i \\ h * v_i \\ h \end{bmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{R} | \mathbf{t} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}.$$

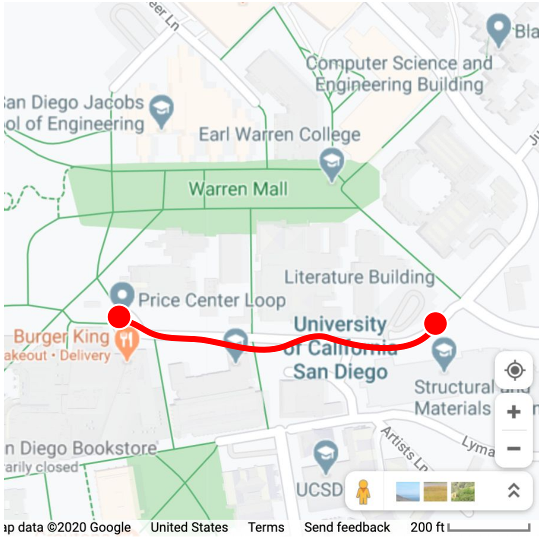
Po-Chen Wu PhD Dissertation 2018.

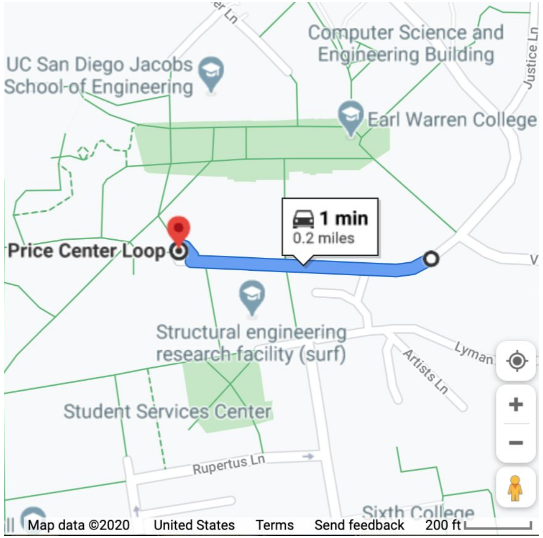
Pose representation



Trajectory evaluation

Estimated poses	Ground truth poses	Error metrics



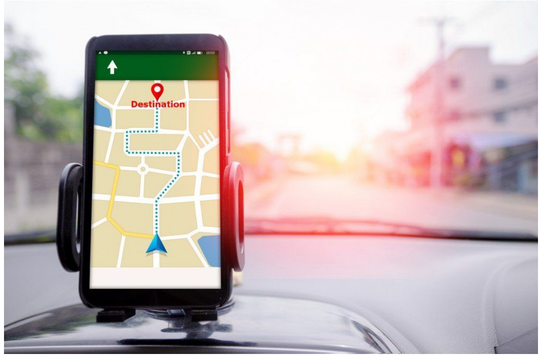
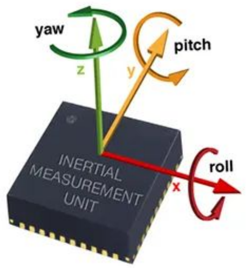


Absolute Pose Error (APE)

Relative Pose Error (RPE)

...

Why do we use cameras?

GPS	IMU	Camera
		
• Inaccurate • Low throughput	• Cheap IMUs: inaccurate, drifting • Expensive IMUs: not easily available	• Available everywhere, like human eyes

https://www.scienceabc.com/innovation/how-gps-global-positioning-system-works-satellite-smartphone.html

https://vrtracker.xyz/handling-imu-drift/

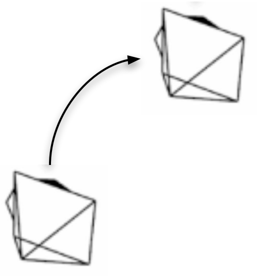
https://www.theguardian.com/technology/2019/oct/01/iphone-11-review-iphone-xr-dual-camera-a13-smartphone

Outline

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

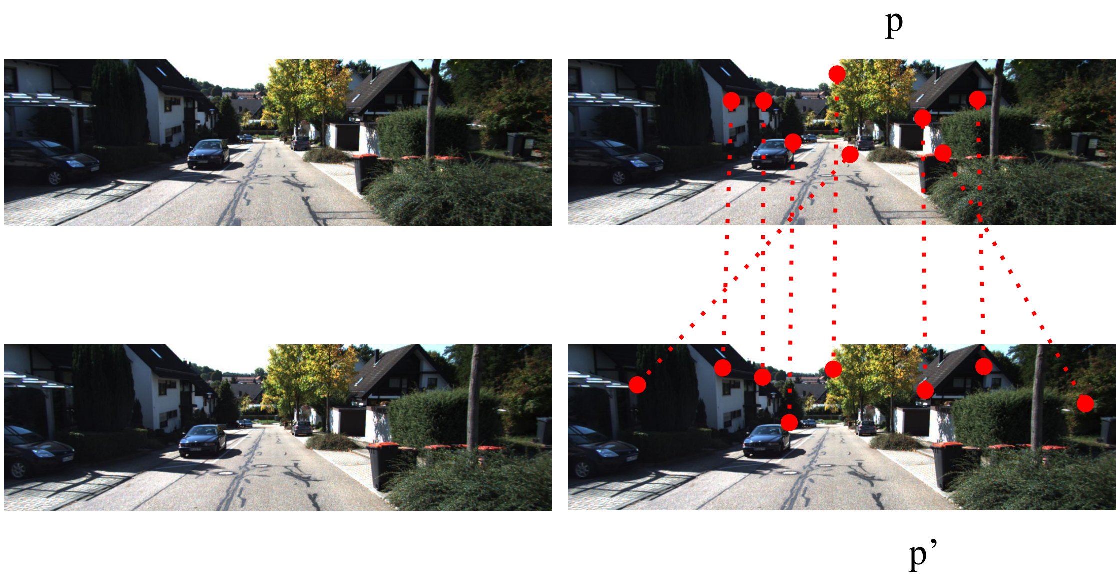
Visual odometry overview

Input → Detection → Description → Matching → Estimation → Output



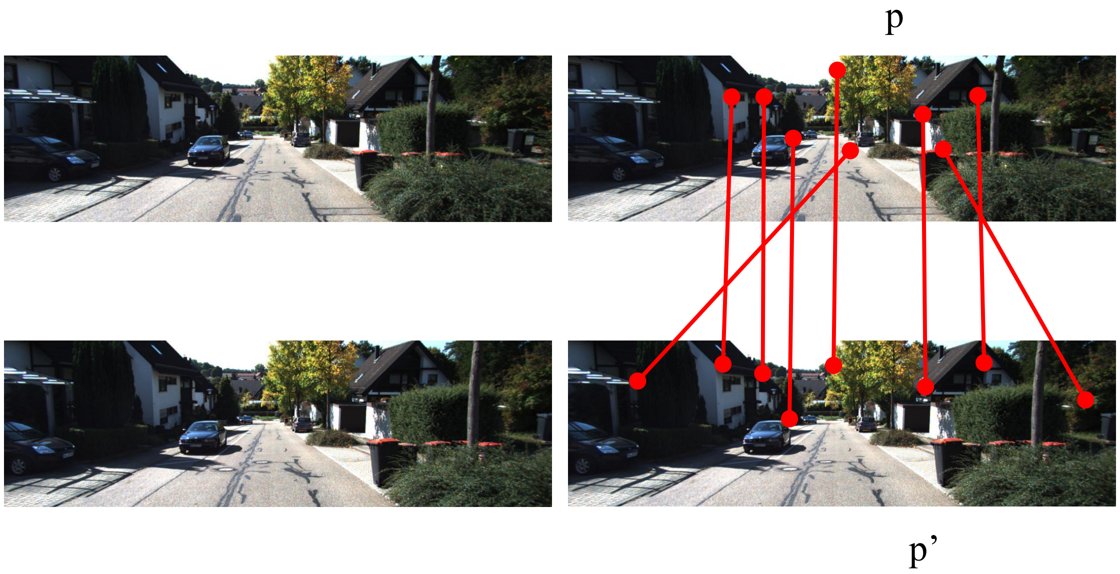
Visual odometry overview

Input → Detection → Description → Matching → Estimation → Output



Visual odometry overview

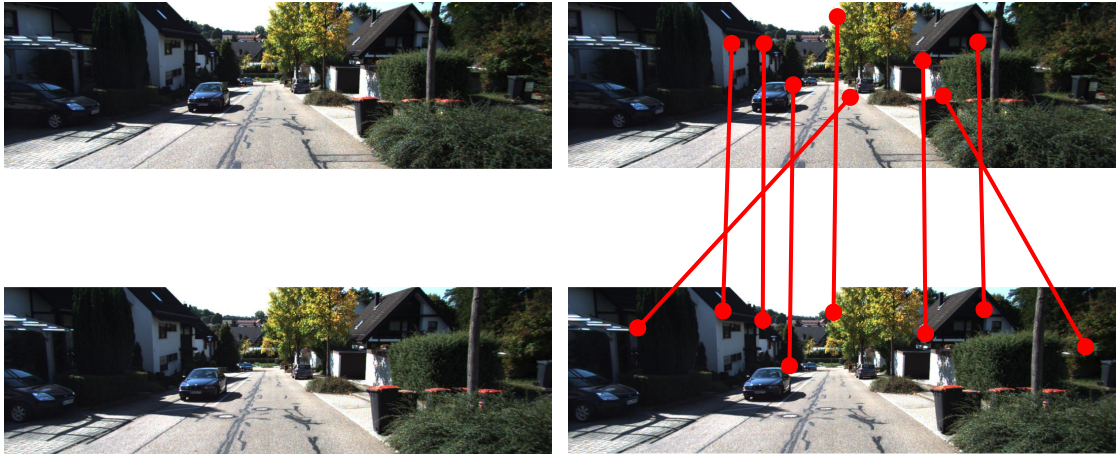
Input → Detection → Description → Matching → Estimation → Output



Visual odometry overview

Input → Detection → Description → Matching → Estimation → Output

p



p'

RANSAC

$$\mathbf{p}'^T \mathbf{F} \mathbf{p} = 0$$

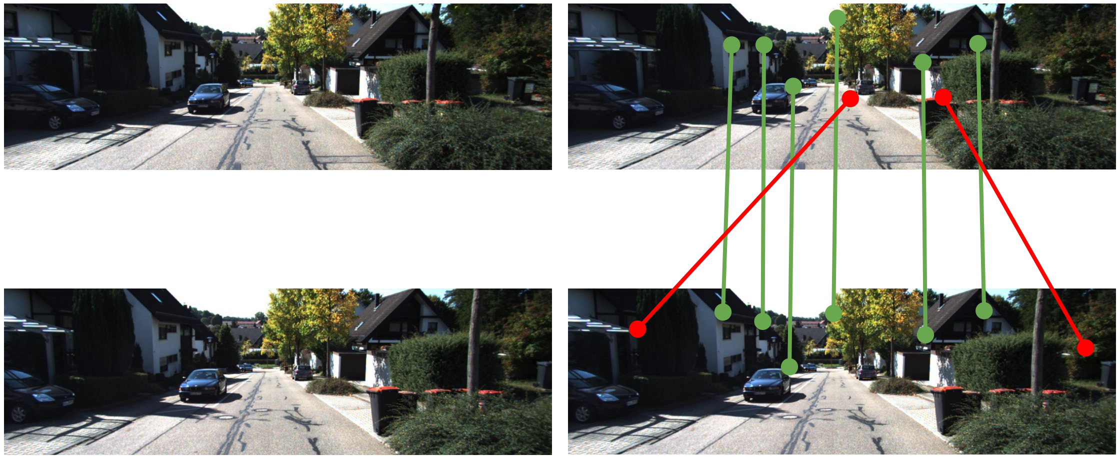
$$\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}.$$

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$$

Visual odometry overview

Input → Detection → Description → Matching → Estimation → Output

p



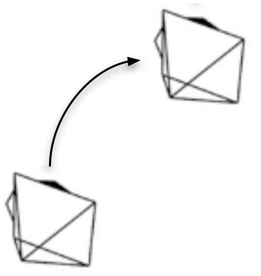
p’

RANSAC

$$\mathbf{p}'^T \mathbf{F} \mathbf{p} = 0$$

$$\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}.$$

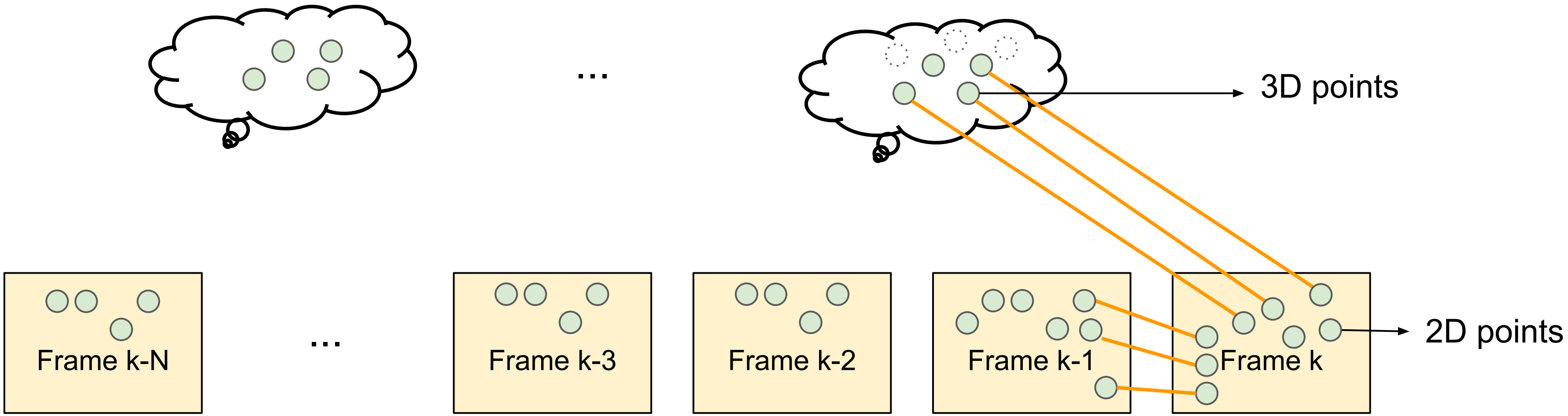
$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$$



$$\tilde{\mathbf{T}} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$$

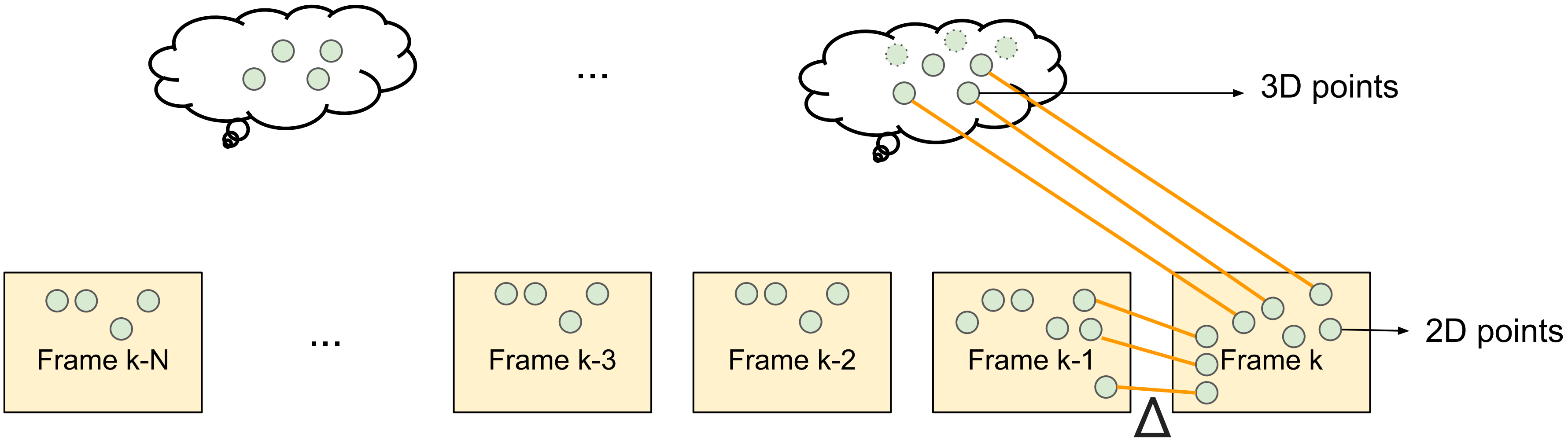
ORB-SLAM Overview

- Tracking
  - 2D-3D correspondences
  - Absolute pose estimation
- Mapping
  - 3D points



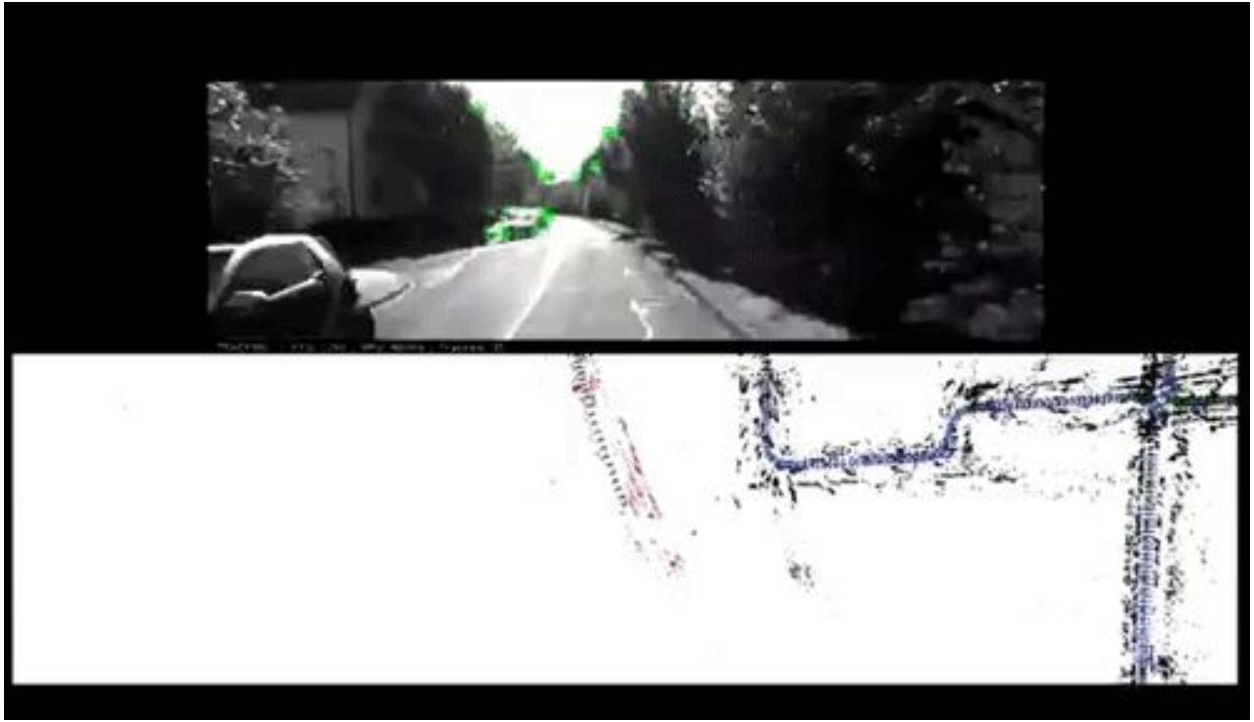
ORB-SLAM Overview

- Tracking
  - 2D-3D correspondences
  - Absolute pose estimation
- Mapping
  - 3D points



ORB-SLAM demo

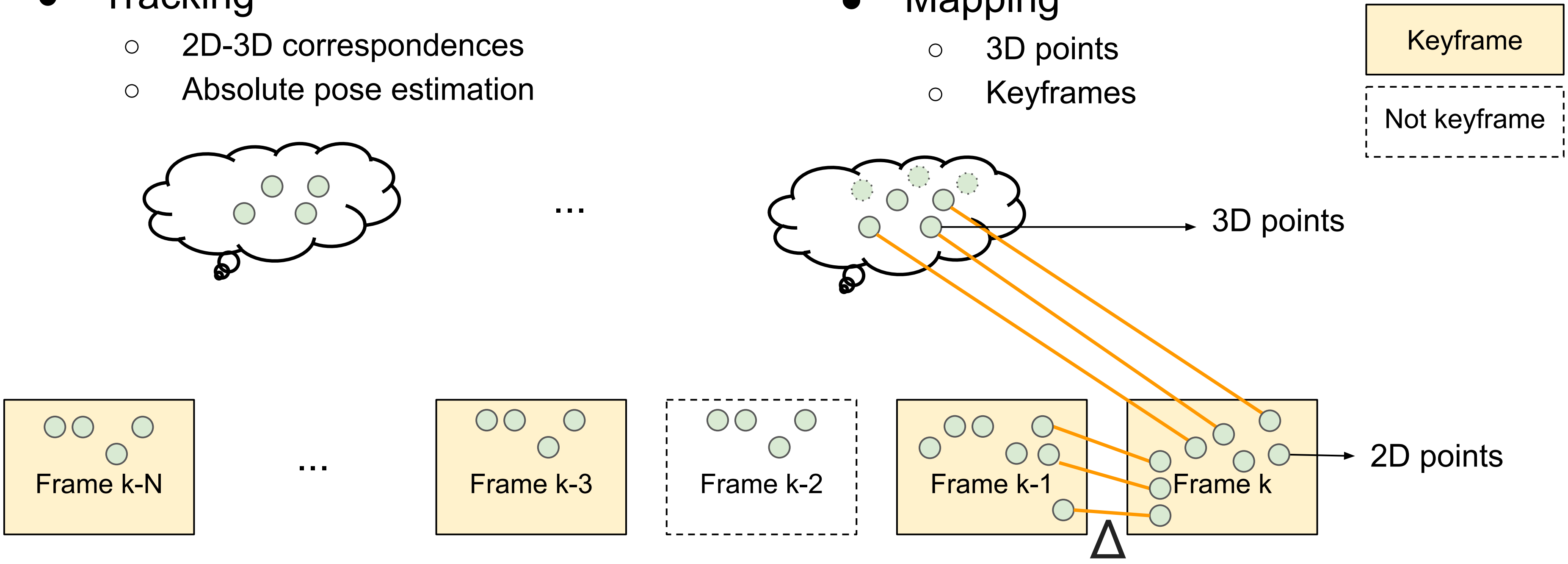
- Tracking
- Mapping



https://thumbs.gfycat.com/IgnorantDangerousDevilfish-size_restricted.gif

ORB-SLAM Overview

- Tracking
  - 2D-3D correspondences
  - Absolute pose estimation
- Mapping
  - 3D points
  - Keyframes

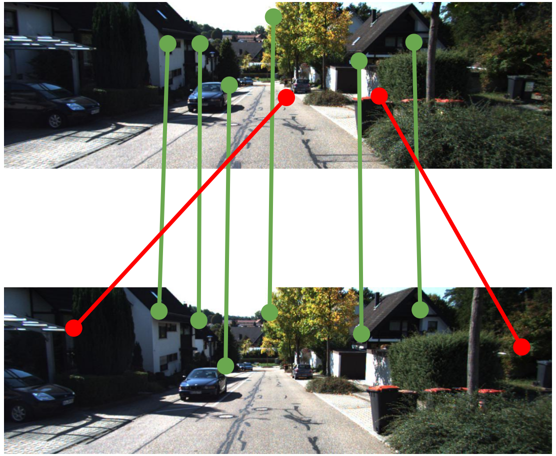


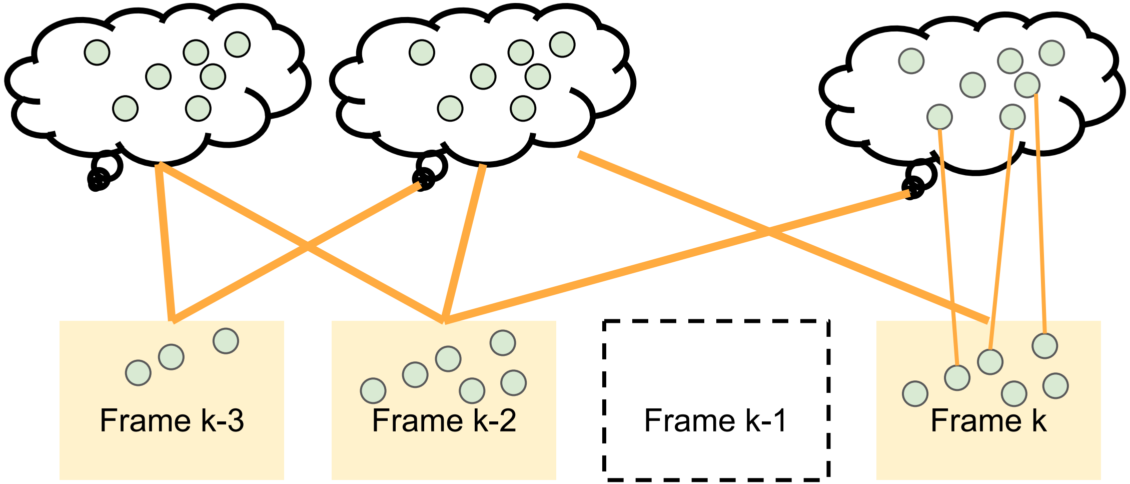
Successful factors for ORB-SLAM

Outlier rejection

Keyframe-based

Bundle adjustment



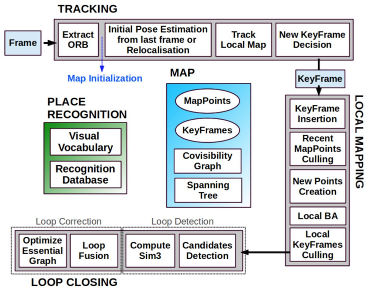


Outlines

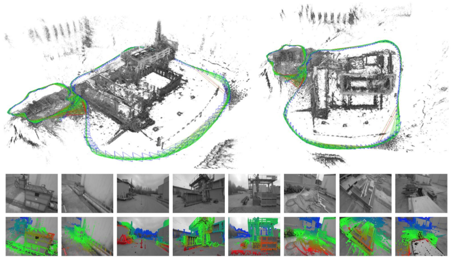
- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Related work

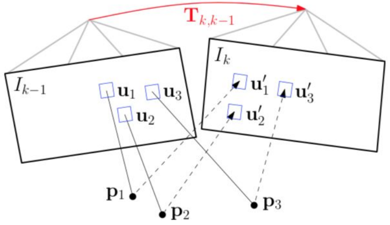
- Geometry-based visual odometry



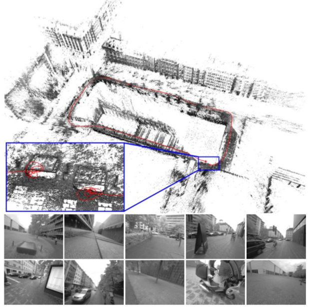
ORB-SLAM
Mur-Artal et al. 2015



LSD-SLAM
J. Engel et al. 2015

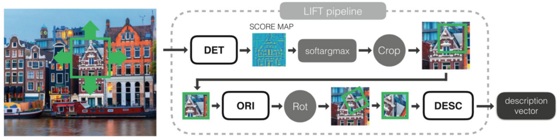


Semi-Direct VO (SVO)
C. Forster et al. 2014

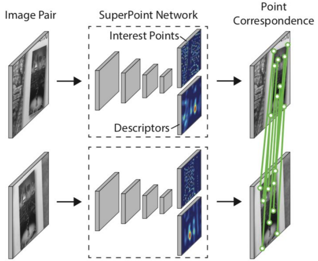


Direct SO (DSO)
J. Engel et al. 2018

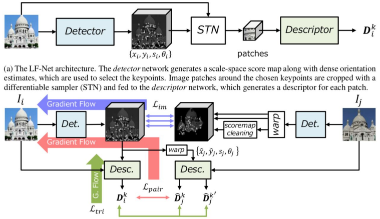
- Learning-based feature extraction and matching



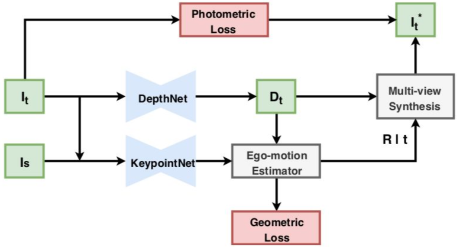
LIFT
Kwang Moo Yi et al. 2016



SuperPoint
Daniel DeTone et al. 2018



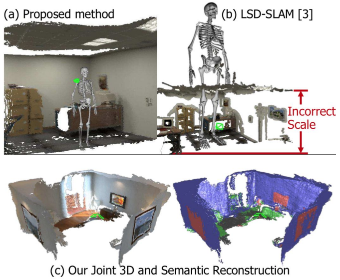
LF-Net
Yuki Ono et al. 2018



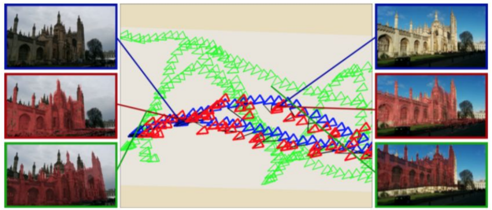
Self-supervised 3D keypoint
JiexiongTang et al. 2019

Related work

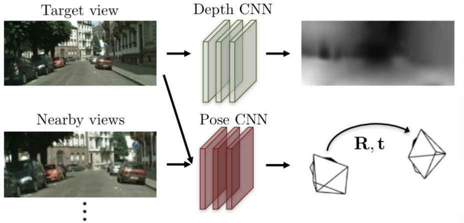
- Learning-based visual odometry & camera pose estimation



CNN-SLAM
Keisuke Tateno et al. 2017

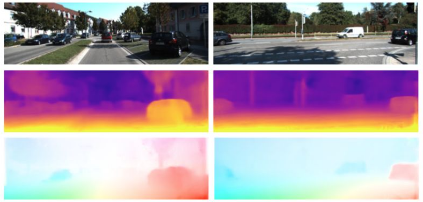


PoseNet
Alex Kendall et al. 2016

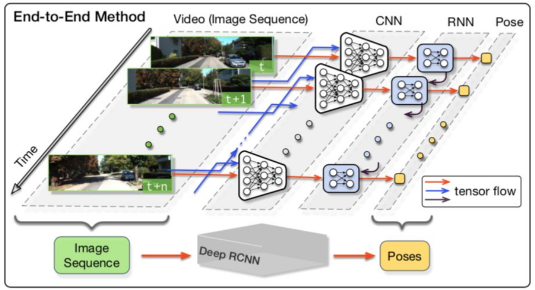


(b) Testing: single-view depth and multi-view pose estimation.

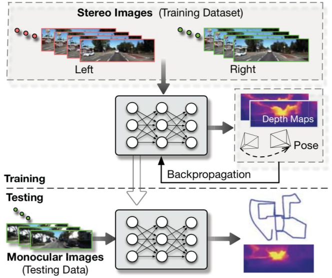
Sfm-learner
T. Zhou et al. 2017



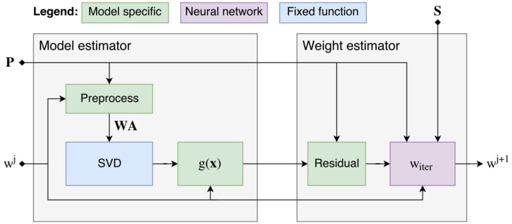
GeoNet
Zhichao Yin et al. 2018



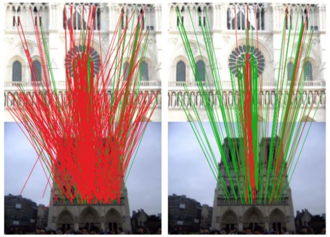
DeepVO
Sen Wang et al. 2017



UnDeepVO
Ruihao Li et al. 2018



DeepF
Ranftl et. al. 2018



(a) RANSAC (b) Our approach

Good Correspondences
Kwang Moo Yi et al. 2018

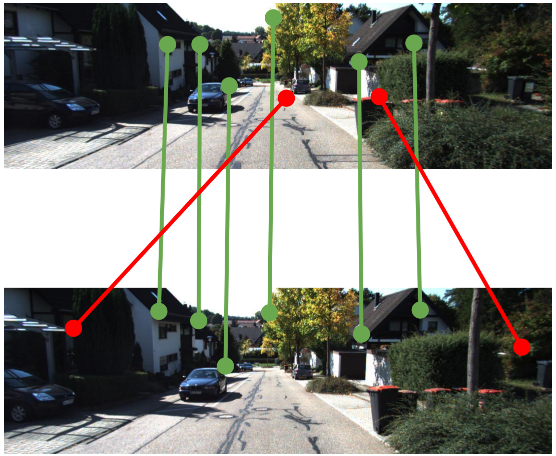
Outlines

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Motivation and problem description

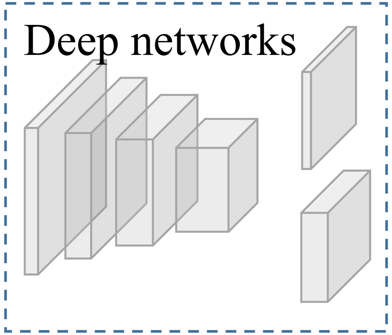
<u>Camera pose estimation</u>

- Key for visual odometry and SLAM
- SIFT + RANSAC



<u>Deep learning-based method</u>

- Learn from data
- Models to replace SIFT, RANSAC
- Modules not optimized together

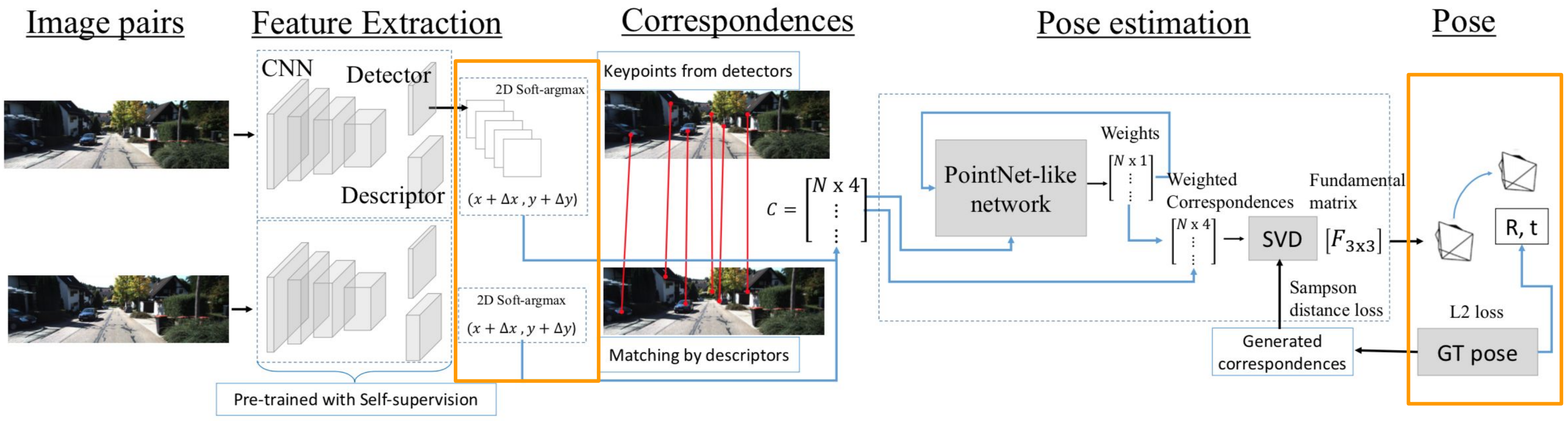


Contributions

End-to-end framework
- Feature extraction, matching
- relative pose estimation

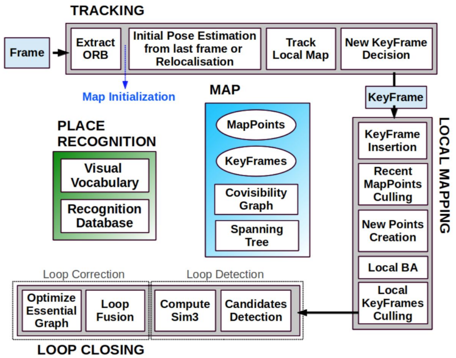
Novel modules
- *Softargmax* bridge
- Pose objective

Ablation study
- KITTI, ApolloScape
- Cross-dataset setting

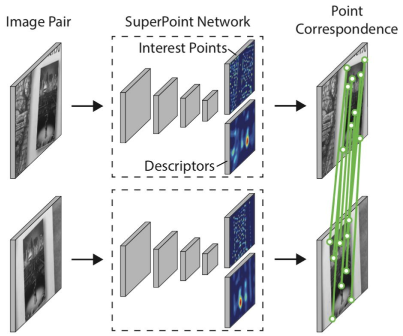


The pipeline is inspired by ...

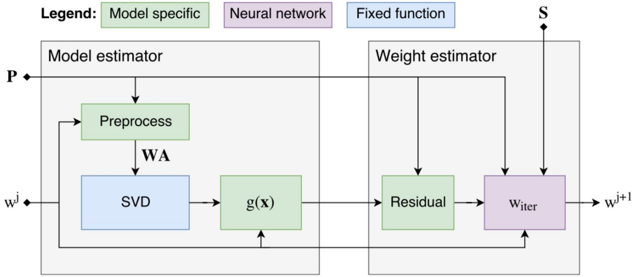
- ORB-SLAM
- SuperPoint (Magic Leap)
- Deep fundamental matrix estimation (DeepF) (Intel Lab)



ORB-SLAM
Mur-Artal et. al. 2015

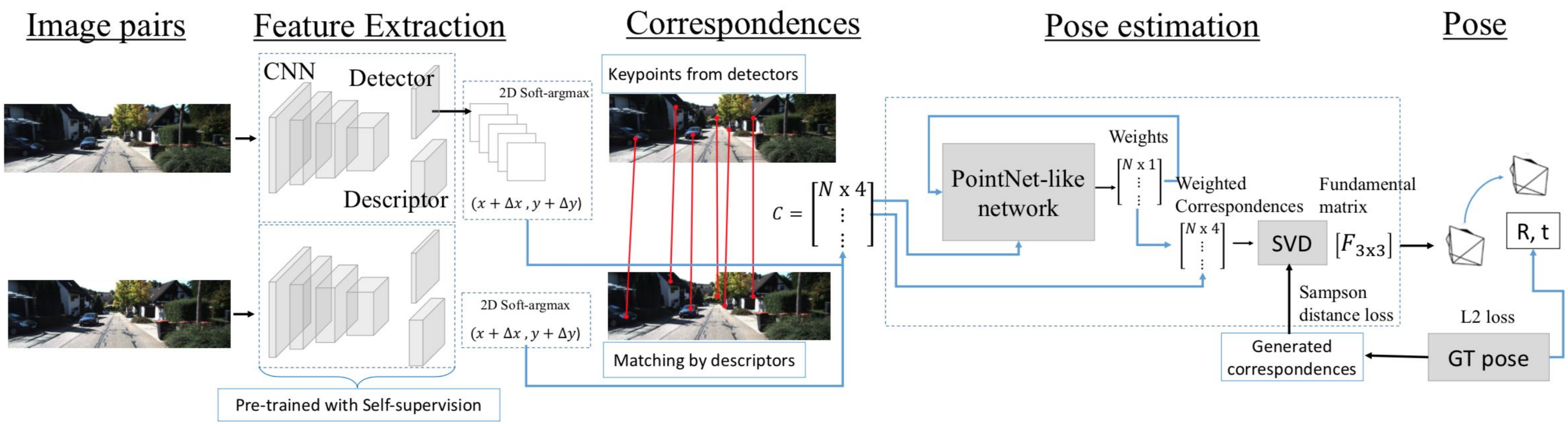


SuperPoint
DeTone et. al. 2017

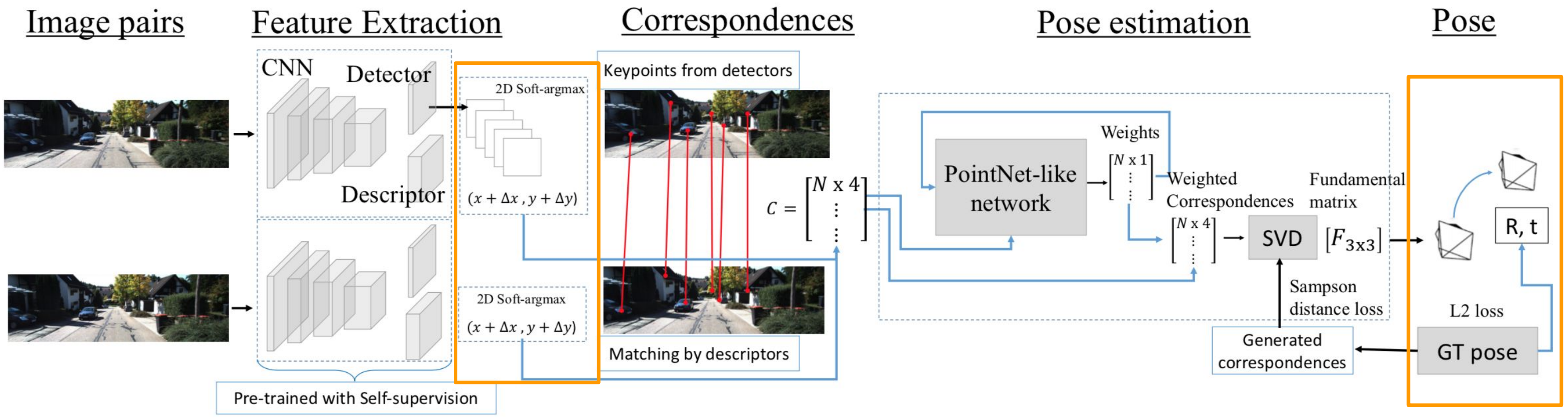


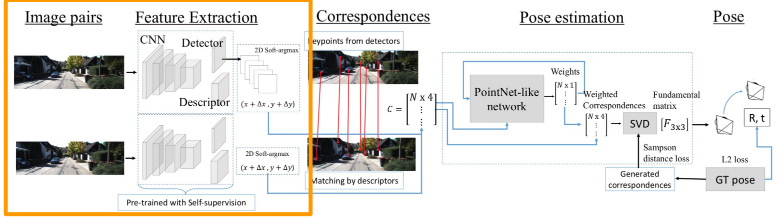
DeepF
Ranftl et. al. 2018

Pipeline Overview

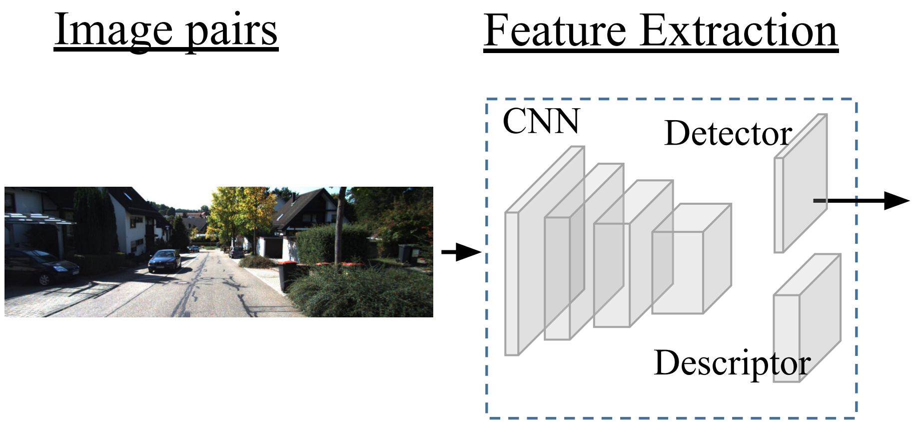


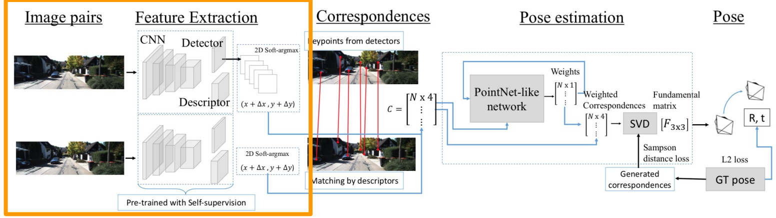
Pipeline Overview





Keypoint detection



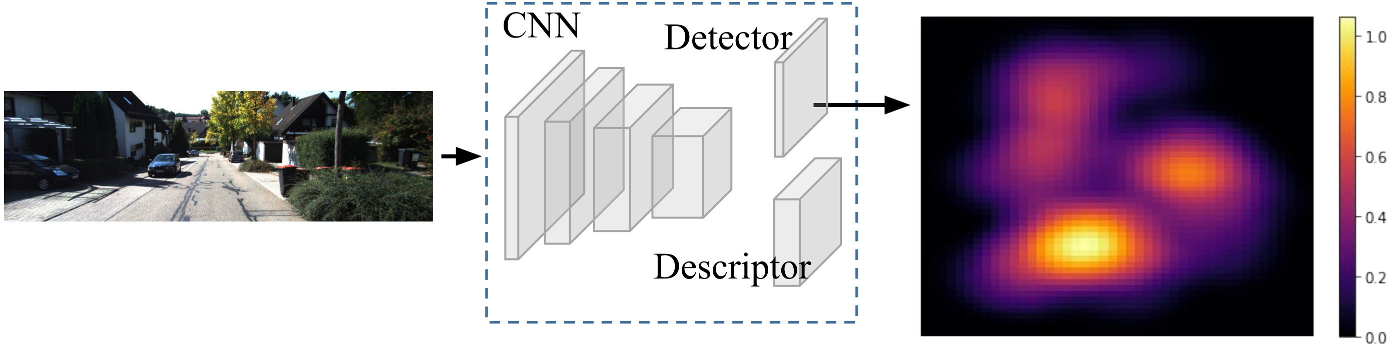


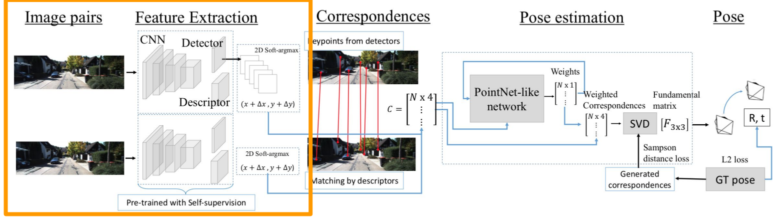
Keypoint detection

Image pairs

Feature Extraction

Detection heatmap



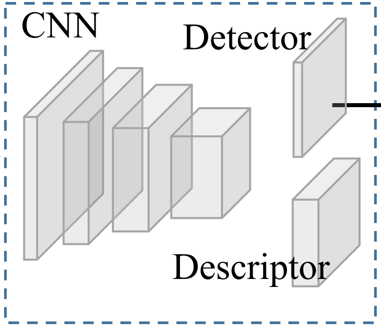


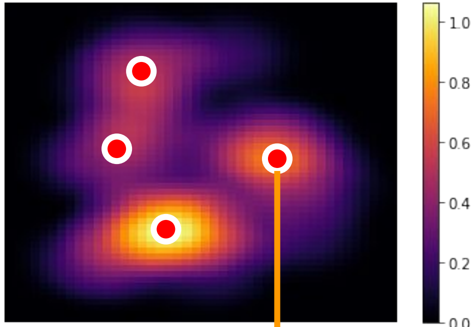
Keypoint detection

Image pairs

Feature Extraction

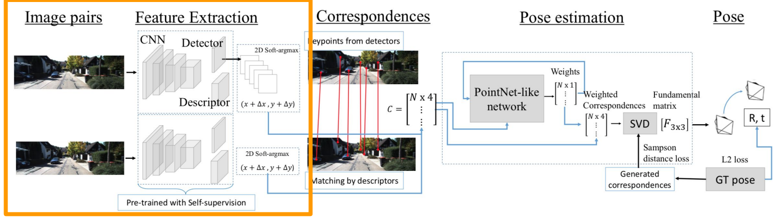
Detection heatmap





Non-Maximum Suppression (NMS)

$u_0, v_0 = (100,150)$

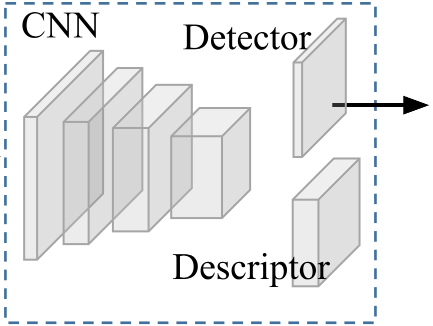


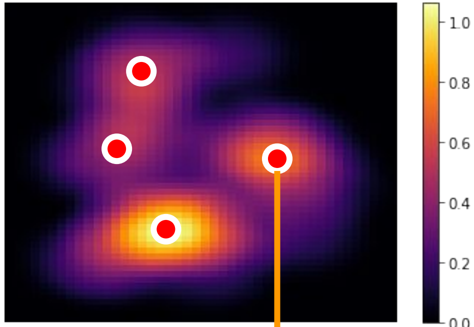
Keypoint detection

Image pairs

Feature Extraction

Detection heatmap





✘ Not differentiable

✘ Integer level

Non-Maximum Suppression (NMS)

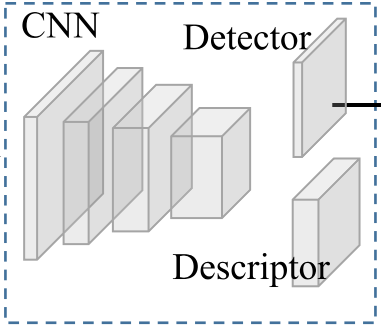
$u_0, v_0 = (100,150)$

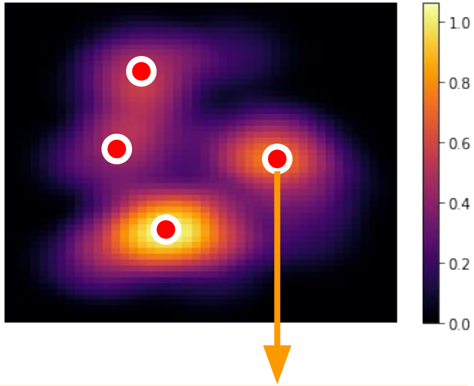
How to make the keypoints differentiable?

Image pairs

Feature Extraction

Detection heatmap





Non-Maximum Suppression (NMS)

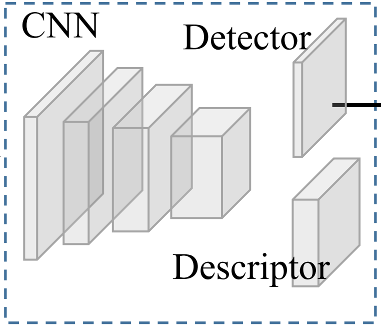
$u_0, v_0 = (100,150)$

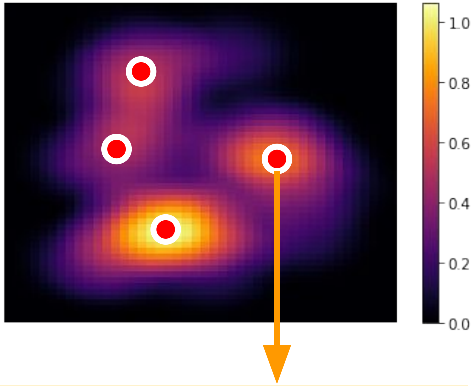
How to make the keypoints differentiable?

Image pairs

Feature Extraction

Detection heatmap





✗ Add network to predict residual

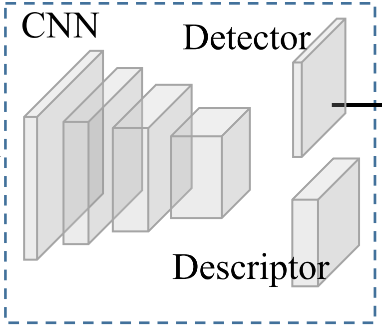
Non-Maximum Suppression (NMS)

$u_0, v_0 = (100,150)$

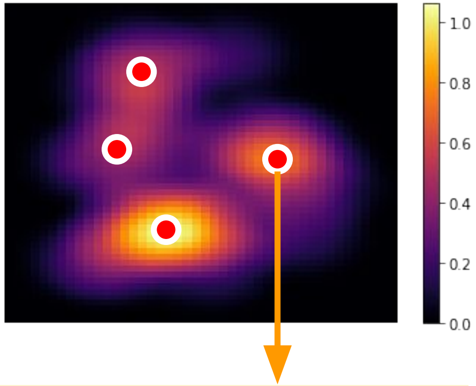
How to make the keypoints differentiable?

Image pairs

Feature Extraction



Detection heatmap

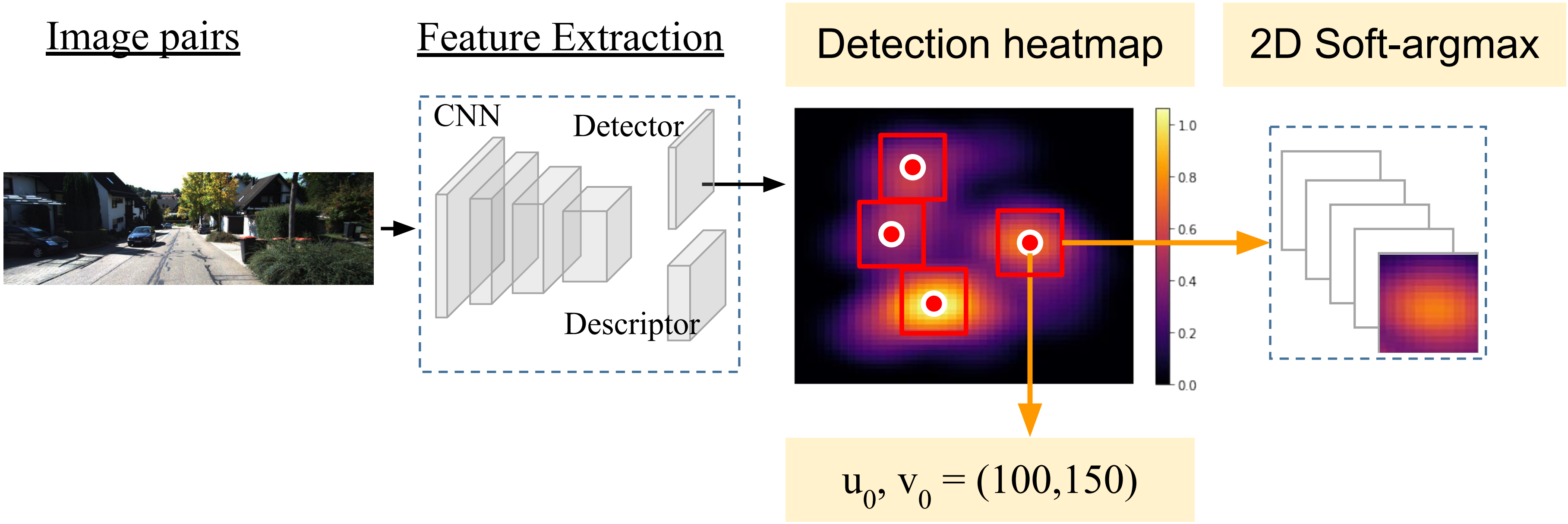


✓ Residual from the heatmap

Non-Maximum Suppression (NMS)

$u_0, v_0 = (100,150)$

Keypoint residual with 2D Soft-argmax



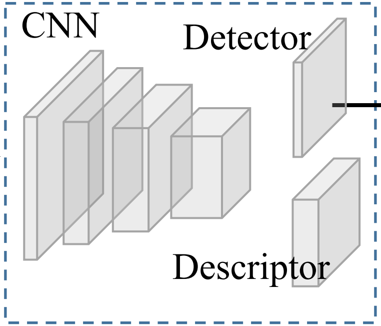
Keypoint residual with 2D Soft-argmax

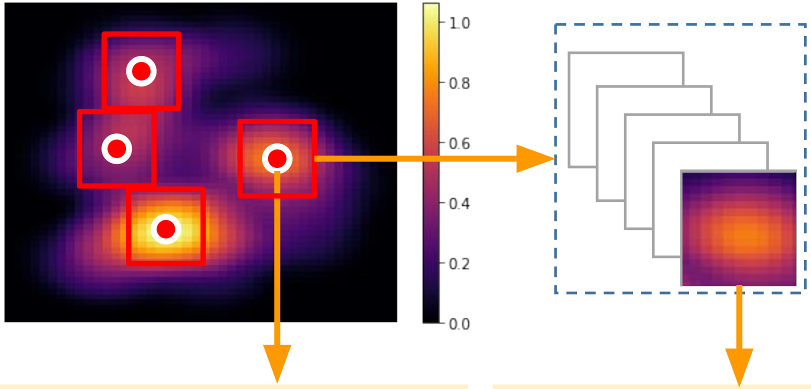
Image pairs

Feature Extraction

Detection heatmap

2D Soft-argmax





$$(u',v') = (u_0,v_0) + (\delta u, \delta v),$$

$u_0, v_0 = (100,150)$

$\delta u, \delta v = (0.3,0.5)$

$u', v' = (100.3,150.5)$

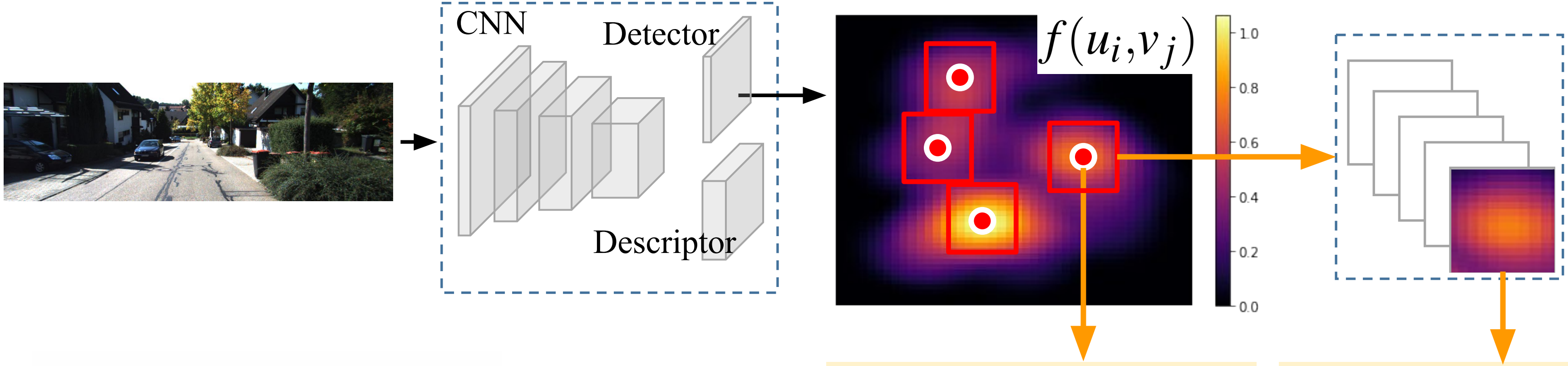
Keypoint residual with 2D Soft-argmax

Image pairs

Feature Extraction

Detection heatmap

2D Soft-argmax



$$(u', v') = (u_0, v_0) + (\delta u, \delta v),$$

$$\delta u = \frac{\sum_j \sum_i e^{f(u_i, v_j)} i}{\sum_j \sum_i e^{f(u_i, v_j)}}, \delta v = \frac{\sum_j \sum_i e^{f(u_i, v_j)} j}{\sum_j \sum_i e^{f(u_i, v_j)}}.$$

$u_0, v_0 = (100,150)$

$\delta u, \delta v = (0.3,0.5)$

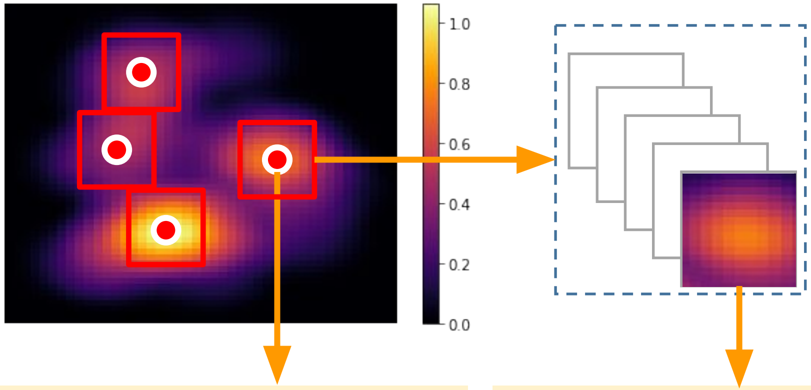
$u', v' = (100.3,150.5)$

Differentiable keypoint

- Soft-argmax detector head
  - ✔ Subpixel accuracy
  - ✔ Differentiable

Detection heatmap

2D Soft-argmax



$u_0, v_0 = (100{,}150)$

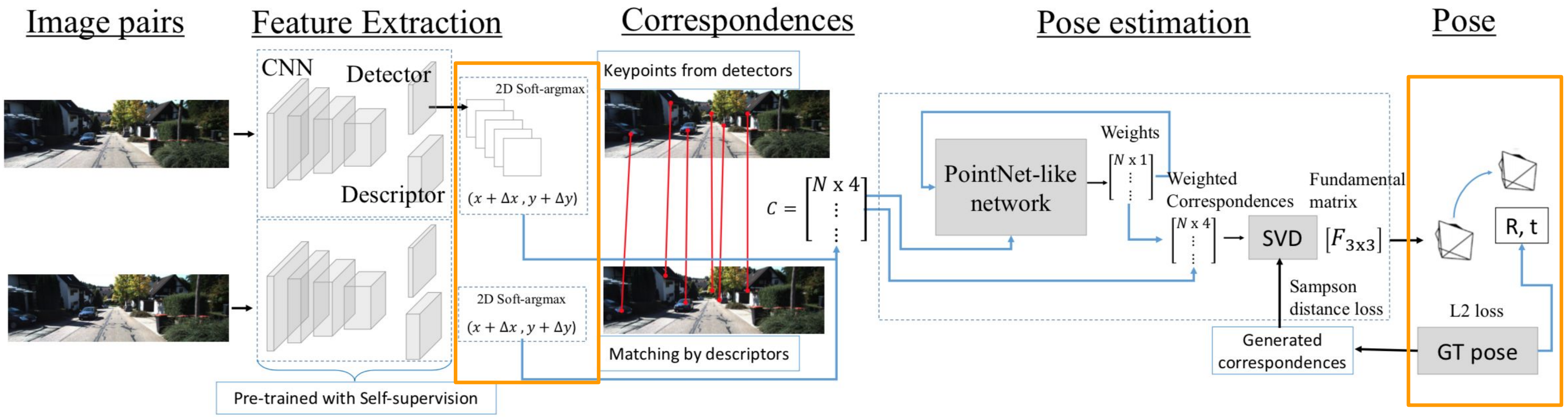
$\delta u, \delta v = (0.3{,}0.5)$

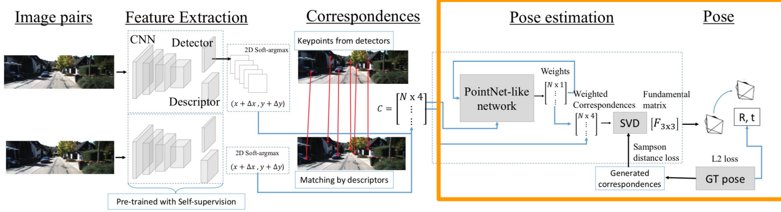
$u', v' = (100.3{,}150.5)$

$$(u', v') = (u_0, v_0) + (\delta u, \delta v),$$

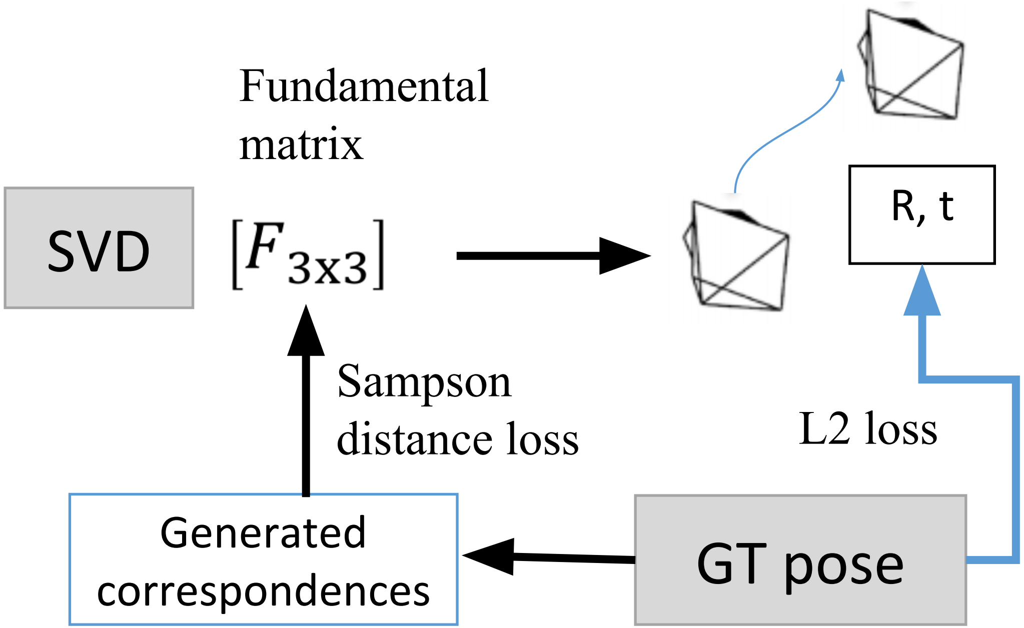
$$\delta u = \frac{\sum_j \sum_i e^{f(u_i, v_j)} i}{\sum_j \sum_i e^{f(u_i, v_j)}}, \delta v = \frac{\sum_j \sum_i e^{f(u_i, v_j)} j}{\sum_j \sum_i e^{f(u_i, v_j)}}.$$

Pipeline Overview





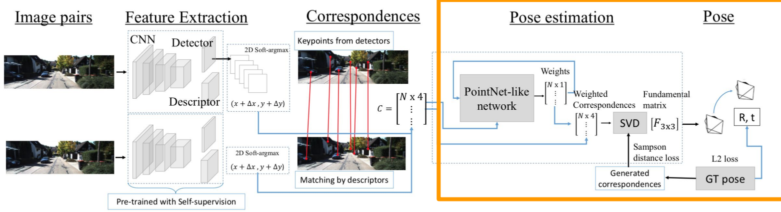
What are the losses?



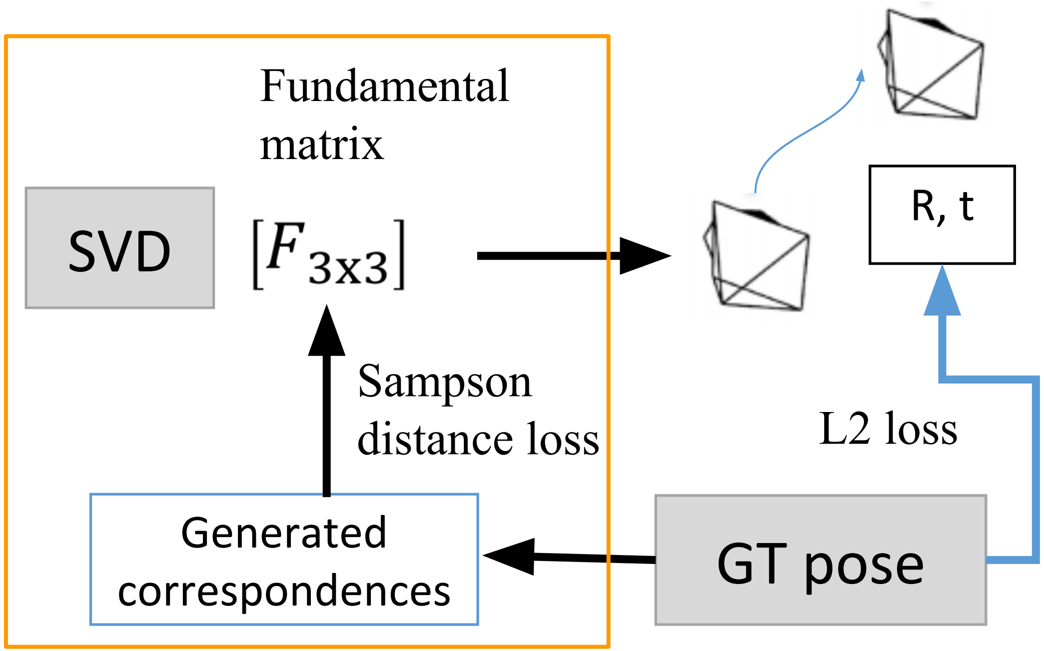
$$\mathbf{p}'^T \mathbf{F} \mathbf{p} = 0$$

$$\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}.$$

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$$



What are the losses?



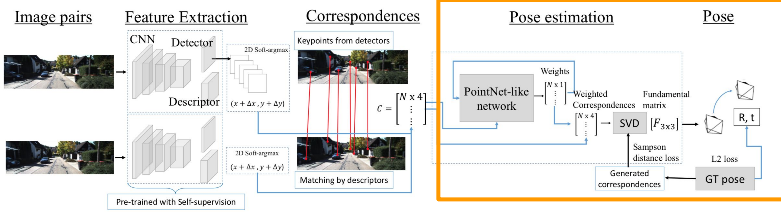
✓Put loss on $\mathbf{F}$

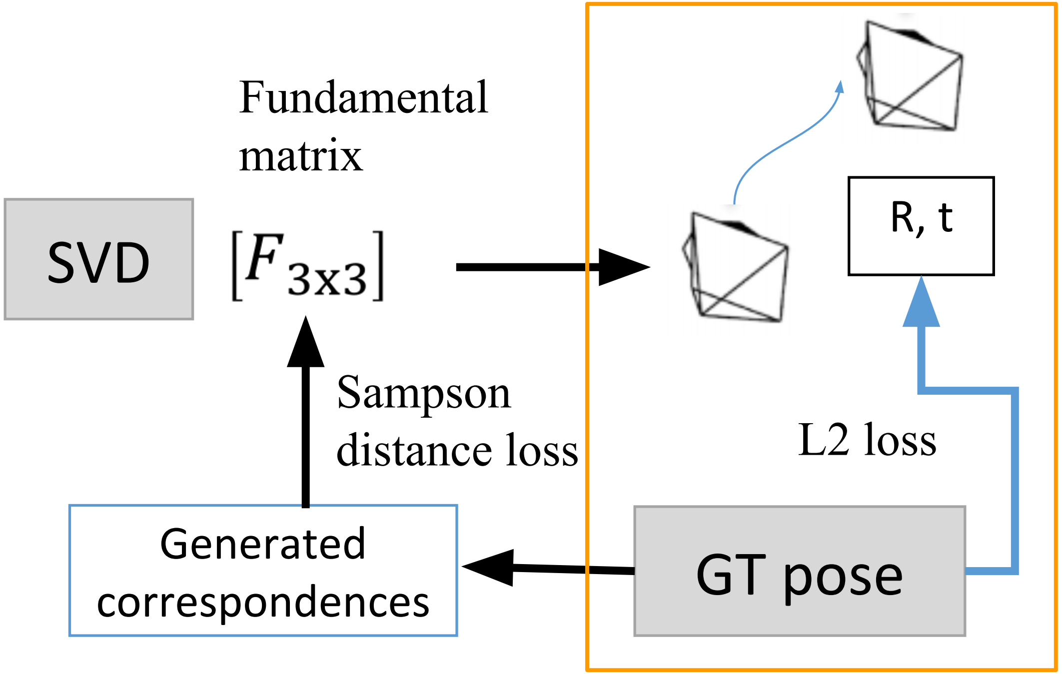
$$\mathbf{p}'^T \mathbf{F} \mathbf{p} = 0$$

$$\mathbf{E} = \mathbf{K}'^T \mathbf{F} \mathbf{K}.$$

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R}$$

What are the losses?





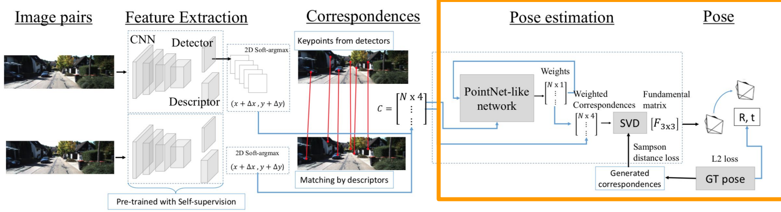
✅Put loss on **F**

✅Put loss on **R, t**

$$\mathbf{p}'^T\mathbf{F}\mathbf{p} = 0$$

$$\mathbf{E} = \mathbf{K}'^T\mathbf{F}\mathbf{K}.$$

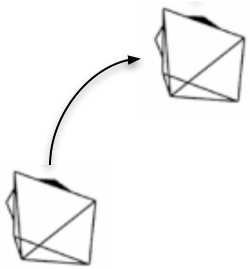
$$\mathbf{E} = [\mathbf{t}]_\times\mathbf{R}$$



Geometry-based loss

- Pose is the final output
- Handle pose decomposition

$$\mathbf{E} = [\mathbf{t}]_{\times}\mathbf{R}$$



- Loss functions

Loss = L (rot) + λ * L (trans)

L (rot) = || quaternion (GT rot) - quaternion (Est. rot) $||_2$

L (trans) = || GT trans - Est. trans$||_2$

$$\tilde{\mathbf{T}} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$$

Experiments -- baselines

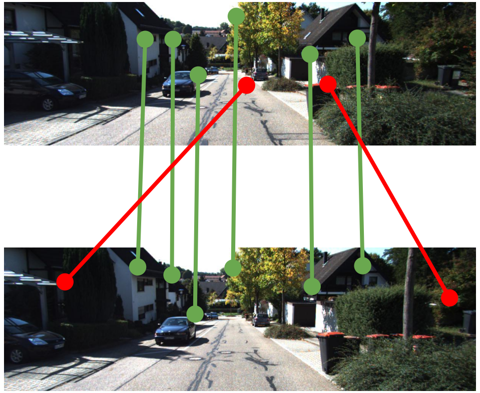
SIFT-based methods

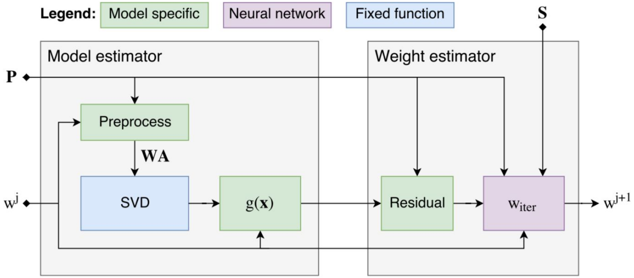
Learning-based methods

SIFT + RANSAC

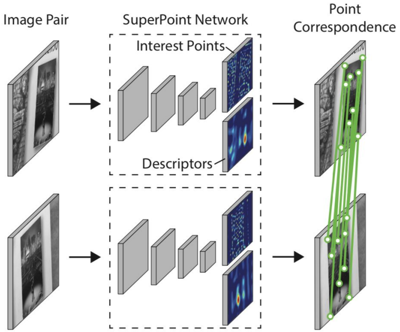
SIFT + DeepF

SuperPoint + others





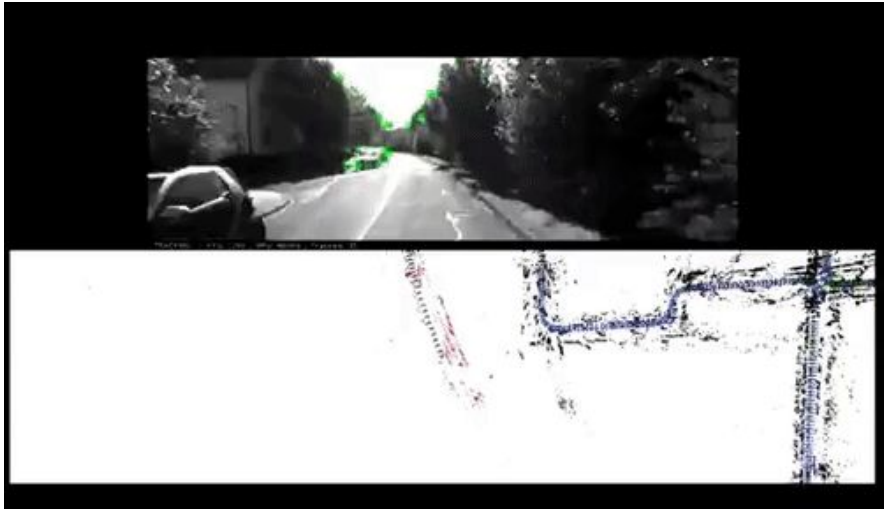
DeepF
Ranftl et. al. 2018



SuperPoint
DeTone et. al. 2017

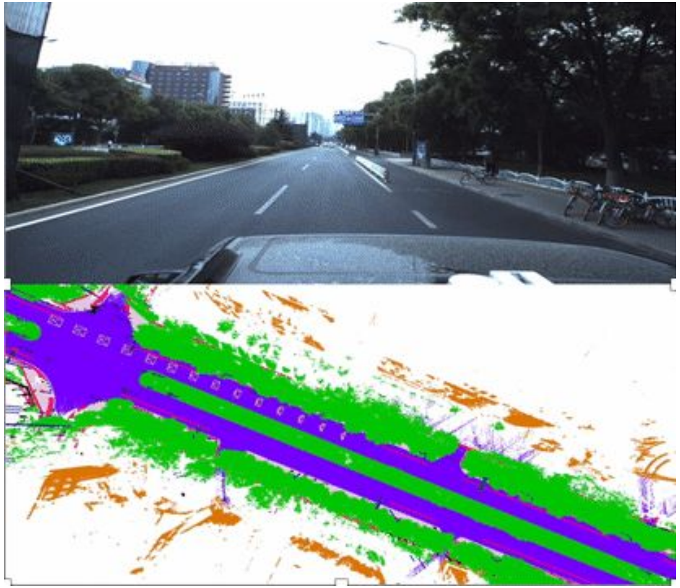
Experiments -- datasets

KITTI



https://thumbs.gfycat.com/IgnorantDangerousDevilfish-size_restricted.gif

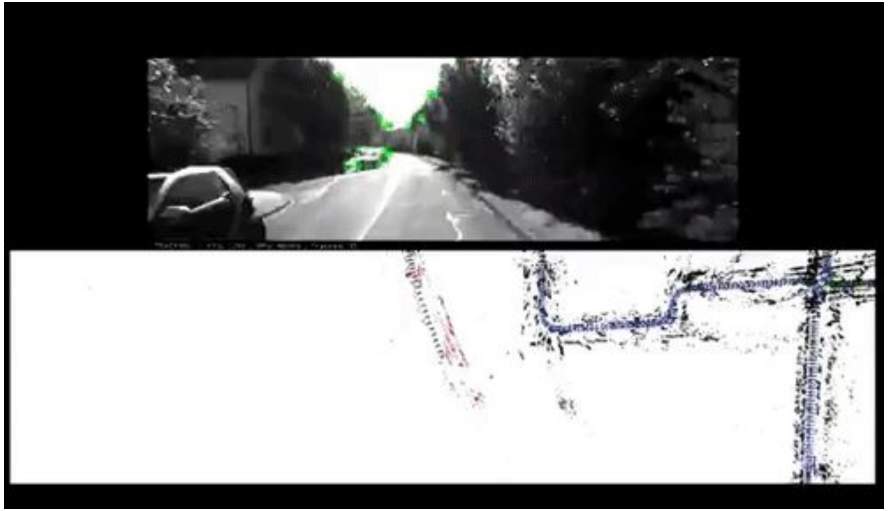
ApolloScape



http://apolloscape.auto/self_localization.html

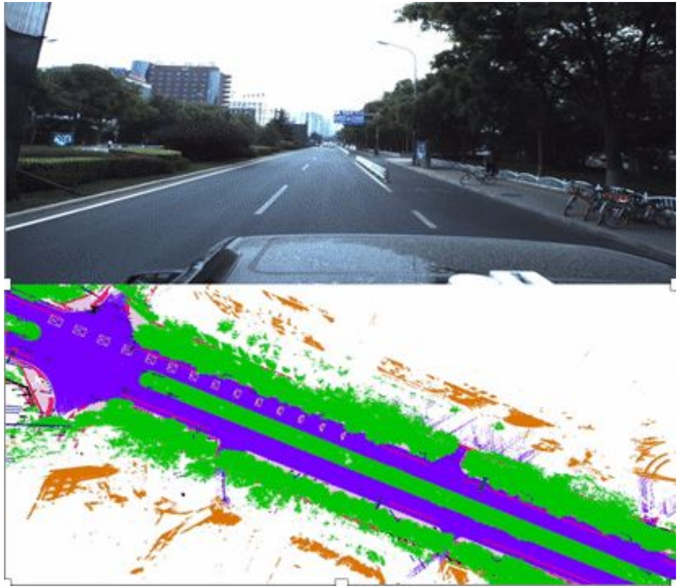
Experiments -- datasets

KITTI



https://thumbs.gfycat.com/IgnorantDangerousDevilfish-size_restricted.gif

ApolloScape



http://apolloscape.auto/self_localization.html

Qualitative results

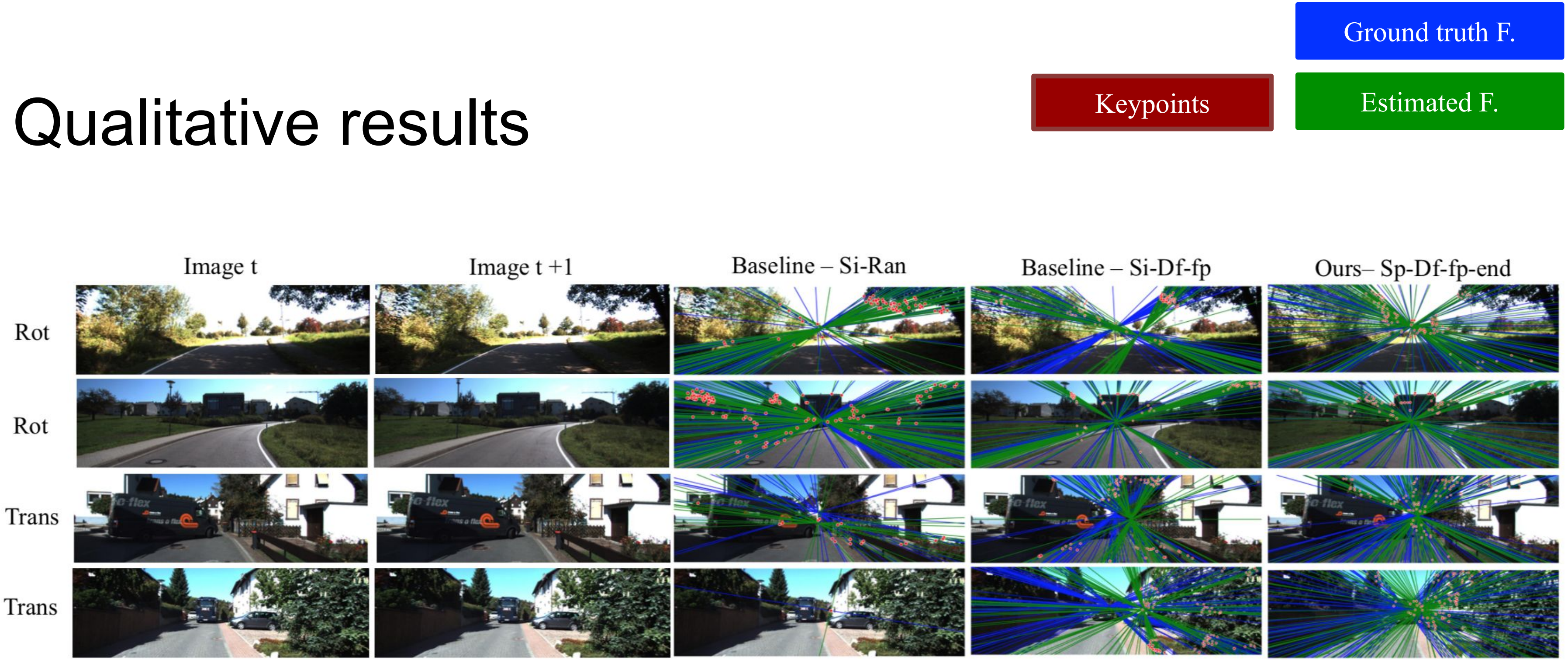
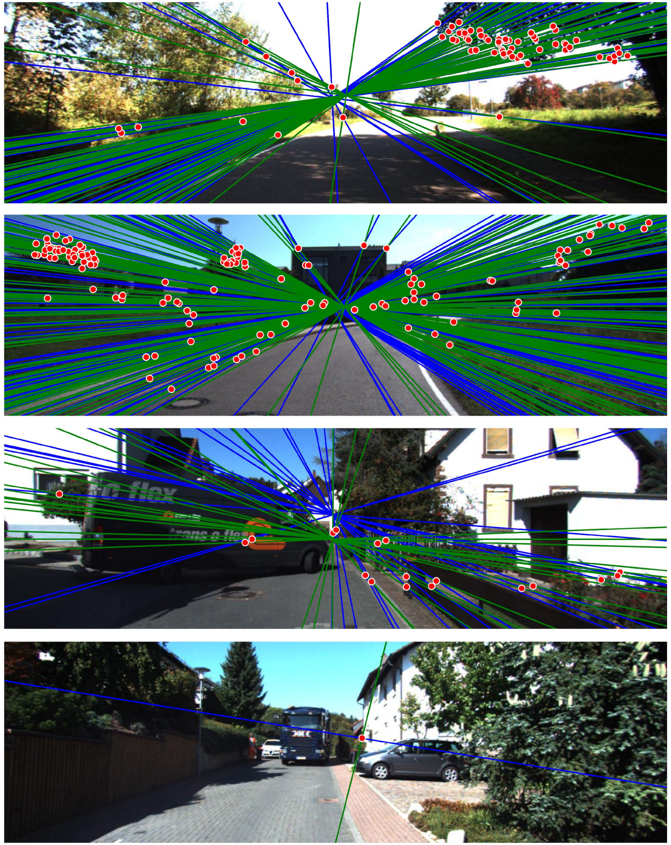


Image t

Image t +1

SIFT + RANSAC



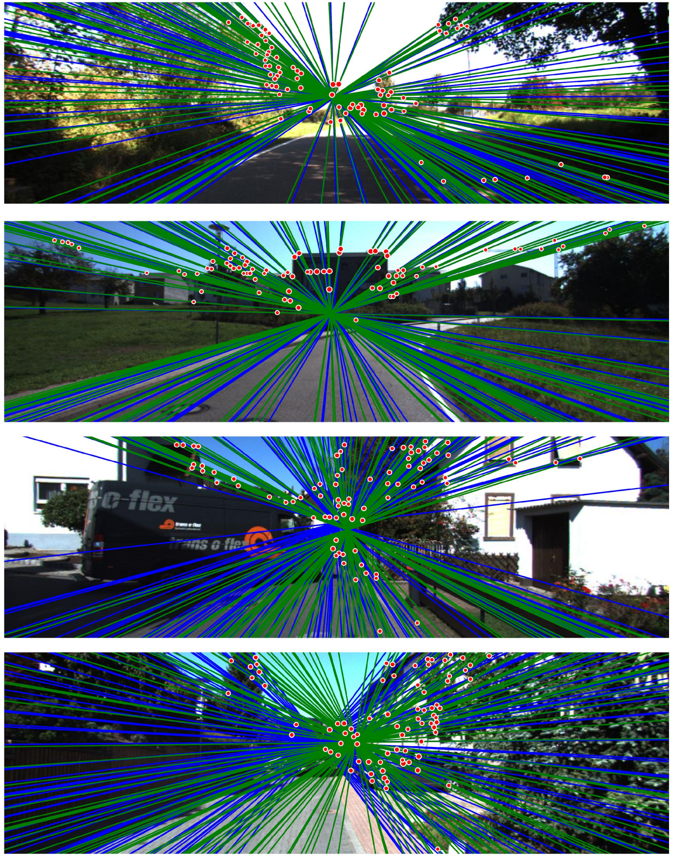
Ground truth F.

Estimated F.

Keypoints

$$\mathbf{p}'^T \mathbf{F} \mathbf{p} = 0$$

Ours – End-to-end

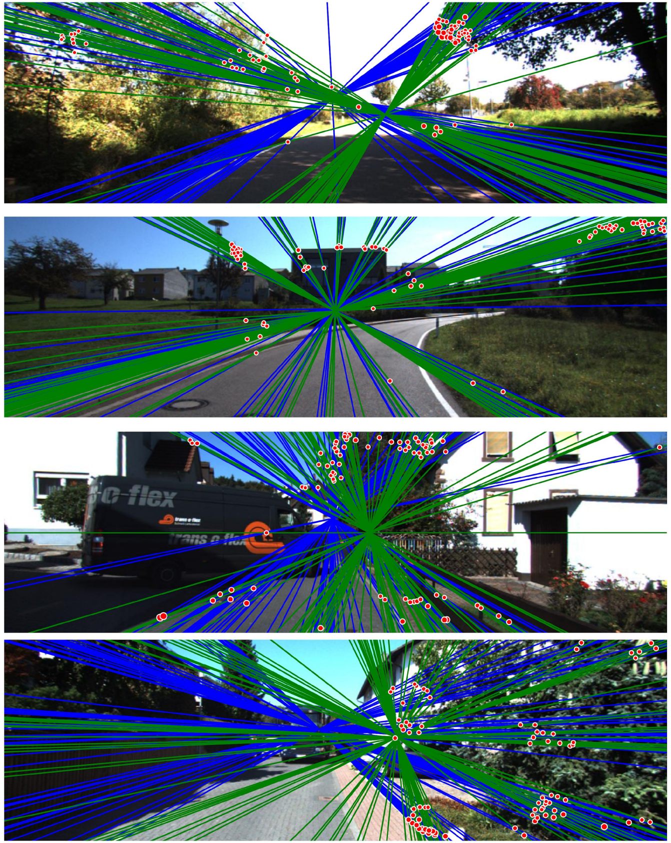


Ground truth F.

Estimated F.

Keypoints

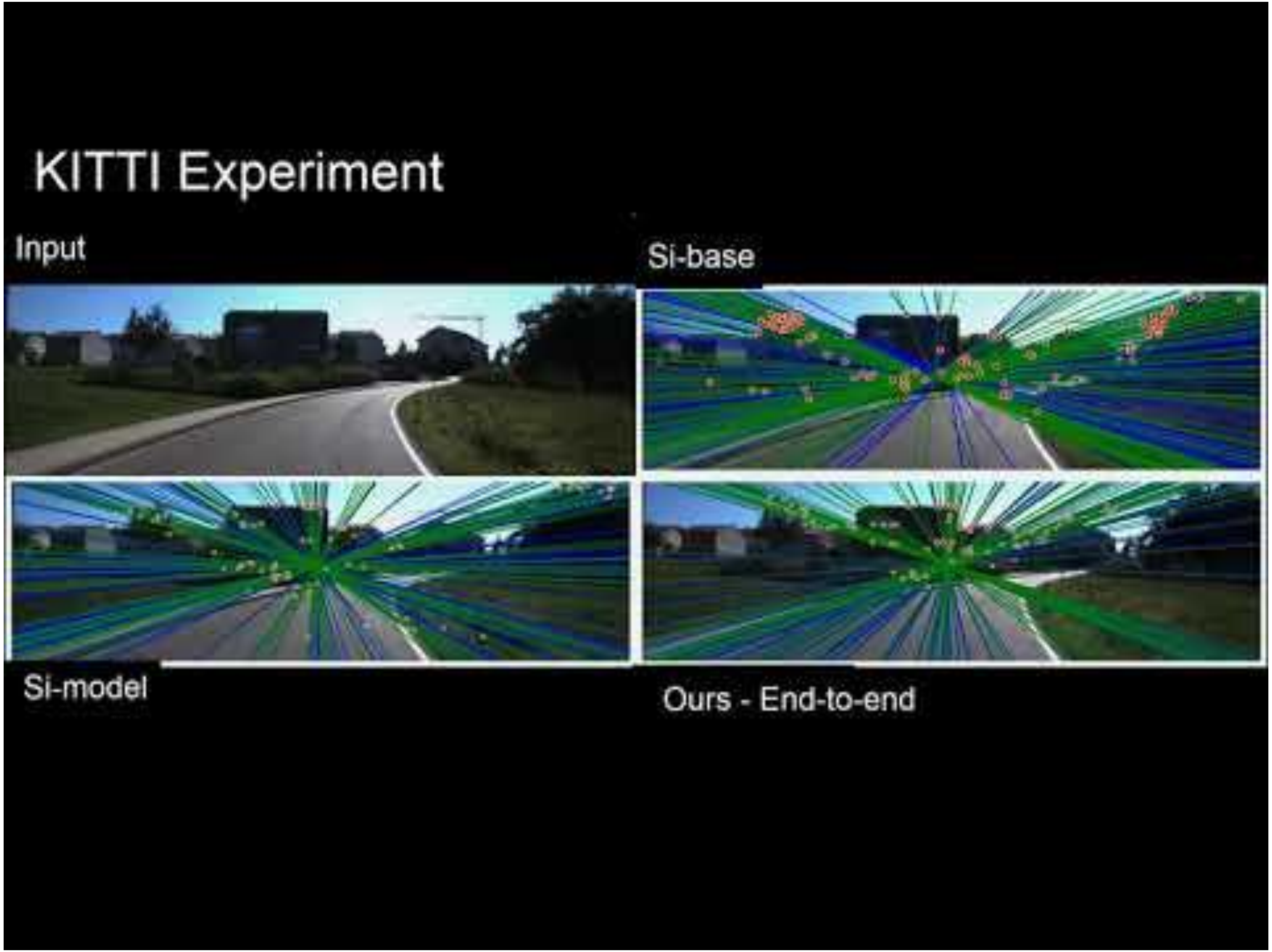
SIFT + DeepF



Ground truth F.

Estimated F.

Keypoints


KITTI Experiment
Input
Si-base
Si-model
Ours - End-to-end

Evaluation metrics

Error	Number
• Rotation error • Translation error	• Error < Threshold ? • Inlier ratio (100% is the best)

Experiment results -- KITTI dataset

- Learning-based baselines

KITTI Models	KITTI dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Sp-Ran)	0.189	0.641	0.217	0.481	5.798	2.103
Sp-Df-f	0.633	0.100	0.078	0.830	1.476	0.846
Sp-Df-p	0.875	0.130	0.047	0.887	1.719	0.539
Ours(Sp-Df-f-end)	0.915	0.053	0.042	0.905	1.662	**0.489**
Ours(Sp-Df-p-end)	**0.932**	**0.050**	**0.041**	0.905	1.600	0.503
Ours(Sp-Df-fp-end)	0.910	0.054	0.048	**0.917**	**1.062**	0.504

- SIFT-based baselines

KITTI Models	KITTI dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Si-Ran)	0.818	0.391	0.056	0.899	1.895	0.639
Si-Df-f	0.938	**0.051**	0.041	0.914	1.699	**0.484**
Si-Df-p	0.901	0.059	0.044	0.903	1.472	0.513
Si-Df-fp	**0.947**	0.111	**0.038**	0.916	1.741	0.484
Ours(Sp-Df-fp-end)	0.910	0.054	0.048	**0.917**	**1.062**	0.504

Experiment results -- KITTI dataset

- Learning-based baselines

KITTI Models	KITTI dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Sp-Ran)	0.189	0.641	0.217	0.481	5.798	2.103
Sp-Df-f	0.633	0.100	0.078	0.830	1.476	0.846
Sp-Df-p	0.875	0.130	0.047	0.887	1.719	0.539
Ours(Sp-Df-f-end)	0.915	0.053	0.042	0.905	1.662	**0.489**
Ours(Sp-Df-p-end)	**0.932**	**0.050**	**0.041**	0.905	1.600	0.503
Ours(Sp-Df-fp-end)	0.910	0.054	0.048	**0.917**	**1.062**	0.504

- SIFT-based baselines

KITTI Models	KITTI dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Si-Ran)	0.818	0.391	0.056	0.899	1.895	0.639
Si-Df-f	0.938	**0.051**	0.041	0.914	1.699	**0.484**
Si-Df-p	0.901	0.059	0.044	0.903	1.472	0.513
Si-Df-fp	**0.947**	0.111	**0.038**	0.916	1.741	0.484
Ours(Sp-Df-fp-end)	0.910	0.054	0.048	**0.917**	**1.062**	0.504

Experiment results -- ApolloScape dataset

- Learning-based baselines

KITTI Models	Apollo dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Sp-Ran)	0.407	0.205	0.118	0.583	5.645	1.670
Sp-Df-f	0.725	0.126	0.068	0.754	2.074	1.155
Sp-Df-p	0.730	0.124	0.067	0.827	1.905	0.974
Ours(Sp-Df-f-end)	0.841	0.100	0.051	0.910	**1.122**	**0.589**
Ours(Sp-Df-p-end)	0.686	0.152	0.071	0.747	2.652	1.068
Ours(Sp-Df-fp-end)	**0.864**	**0.092**	**0.051**	**0.924**	1.275	0.659

- SIFT-based baselines

KITTI Models	Apollo dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Si-Ran)	**0.922**	0.157	**0.037**	**0.979**	**0.788**	0.388
Si-Df-f	0.845	0.172	0.043	0.895	2.452	0.389
Si-Df-p	0.727	0.333	0.056	0.760	4.918	0.658
Si-Df-fp	0.840	0.148	0.044	0.911	2.103	**0.369**
Ours(Sp-Df-fp-end)	0.864	**0.092**	0.051	0.924	1.275	0.659

Experiment results -- ApolloScape dataset

- Learning-based baselines

KITTI Models	Apollo dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Sp-Ran)	0.407	0.205	0.118	0.583	5.645	1.670
Sp-Df-f	0.725	0.126	0.068	0.754	2.074	1.155
Sp-Df-p	0.730	0.124	0.067	0.827	1.905	0.974
Ours(Sp-Df-f-end)	0.841	0.100	0.051	0.910	**1.122**	**0.589**
Ours(Sp-Df-p-end)	0.686	0.152	0.071	0.747	2.652	1.068
Ours(Sp-Df-fp-end)	**0.864**	**0.092**	**0.051**	**0.924**	1.275	0.659

- SIFT-based baselines

KITTI Models	Apollo dataset - error(deg.) inlier ratio↑, mean↓, median↓					
	Rotation (deg.)			Translation (deg.)		
	0.1↑	Mean.↓	Med.↓	2.0↑	Mean.↓	Med.↓
Base(Si-Ran)	**0.922**	0.157	**0.037**	**0.979**	**0.788**	0.388
Si-Df-f	0.845	0.172	0.043	0.895	2.452	0.389
Si-Df-p	0.727	0.333	0.056	0.760	4.918	0.658
Si-Df-fp	0.840	0.148	0.044	0.911	2.103	**0.369**
Ours(Sp-Df-fp-end)	0.864	**0.092**	0.051	0.924	1.275	0.659

Summary

<u>Contributions</u>

- End-to-end framework
- Novel modules
- Cross-dataset evaluation

<u>Limitations</u>

- Camera pose estimation
  - Visual odometry

Outlines

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Motivation

- Deep learning-based method
- Various environments

Overview of SC-SfMLearner

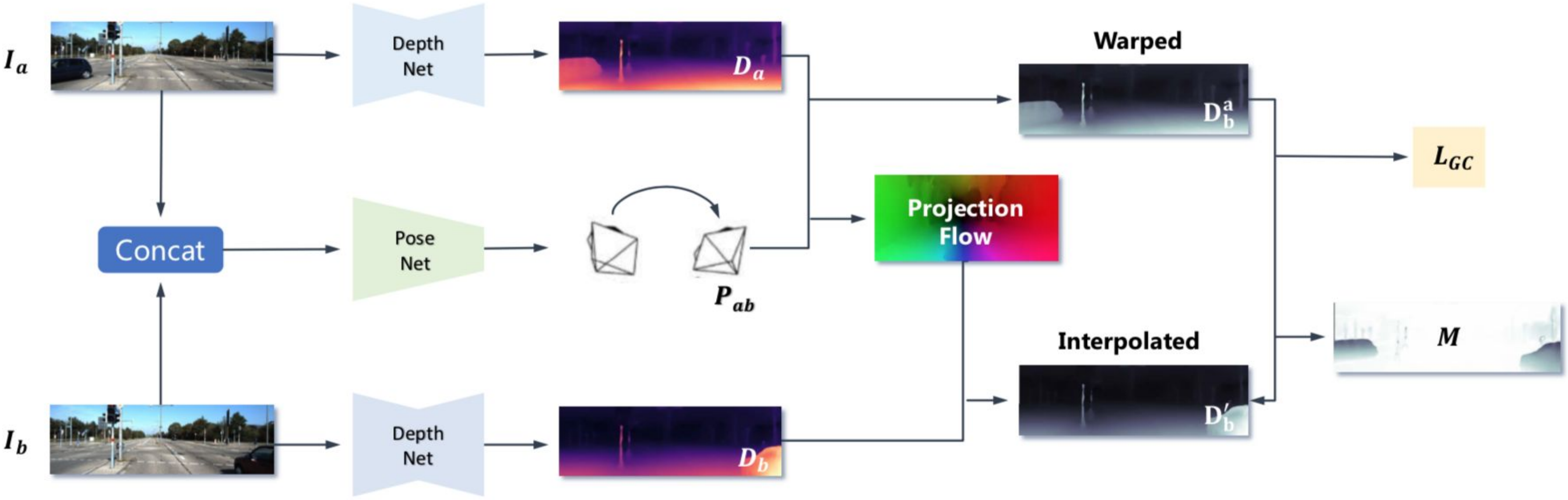


Figure 4.1. Overview of SC-SfMLearner [6].

Experiments

- Datasets
  - Outdoors: KITTI
  - Indoors: EuRoC
- Prediction
  - Depth
  - Pose

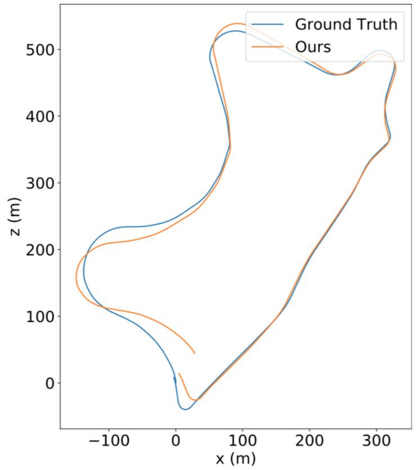
Datasets

KITTI

EuRoC

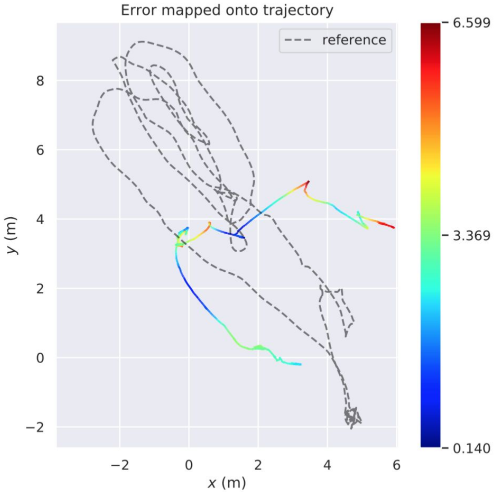
Trajectory -- Model trained on KITTI

KITTI -- seq 09



SC-SfMLearner

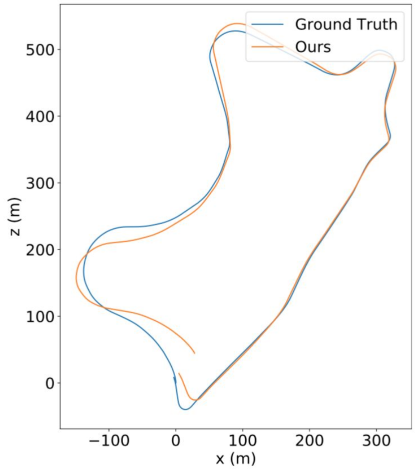
EuRoC -- MH_01_easy



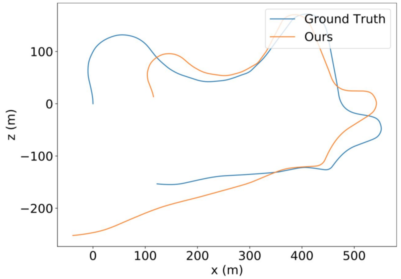
SC-SfMLearner

Comparison -- SC-SfMLearner vs. ORB-SLAM

KITTI -- seq 09

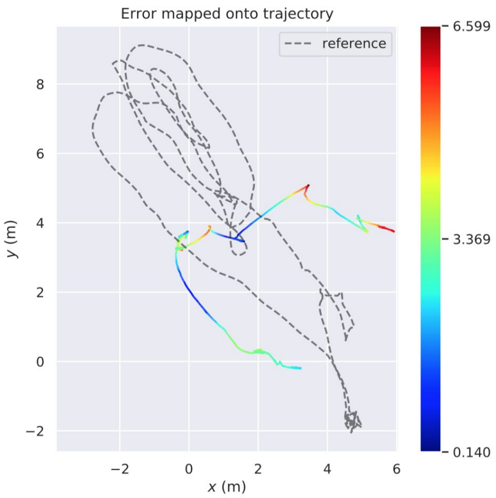


SC-SfMLearner

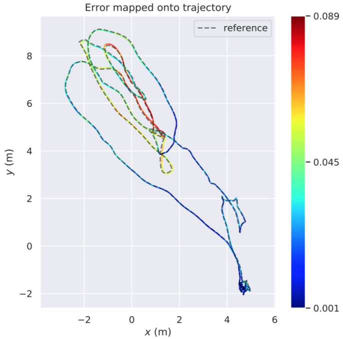


ORB-SLAM

EuRoC -- MH_01_easy



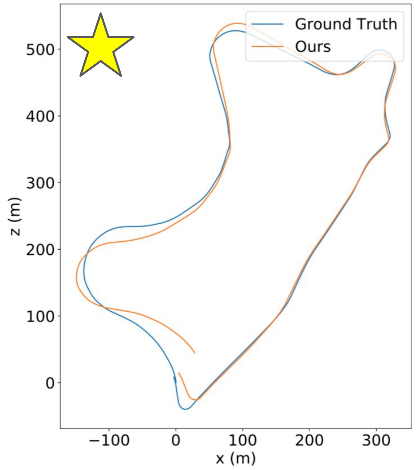
SC-SfMLearner



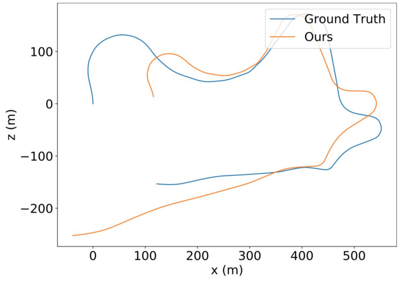
ORB-SLAM

Comparison -- SC-SfMLearner vs. ORB-SLAM

KITTI -- seq 09

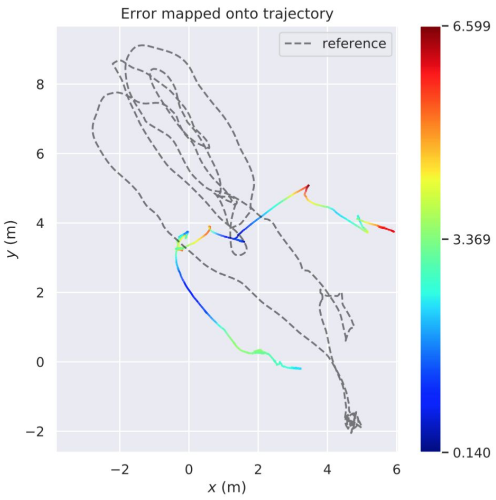


SC-SfMLearner

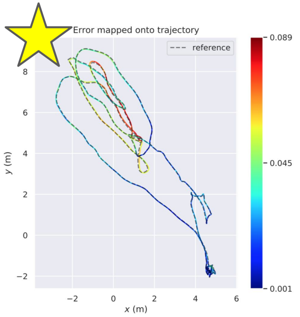


ORB-SLAM

EuRoC -- MH_01_easy



SC-SfMLearner



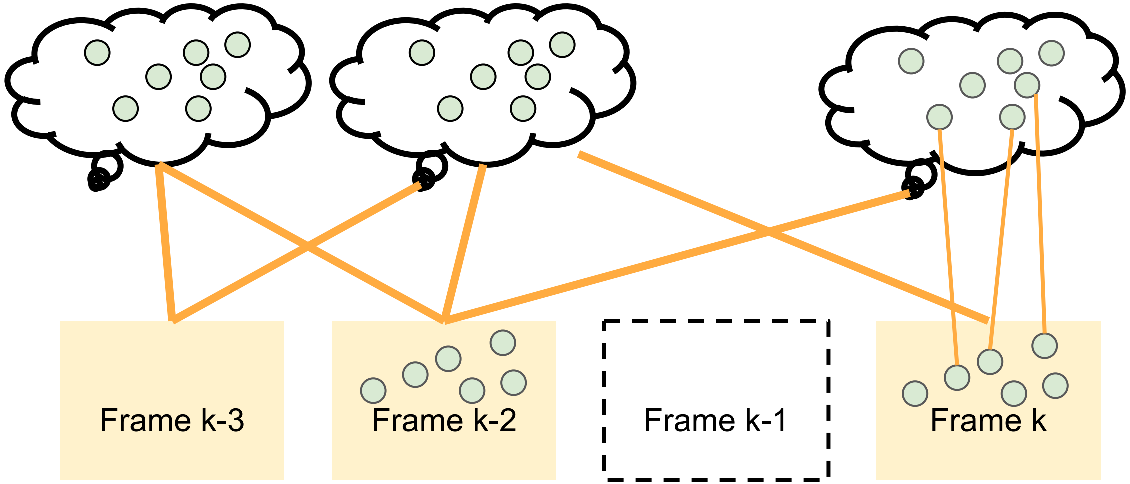
ORB-SLAM

Problems

- Domain gap
- Overfitting

Future work for deep visual odometry

- Optimization
  - Bundle adjustment
- Keyframe
  - Representative
  - Large baseline

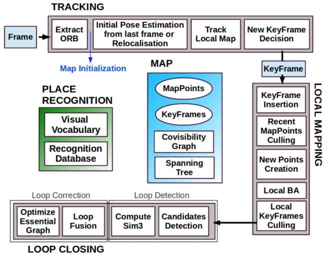


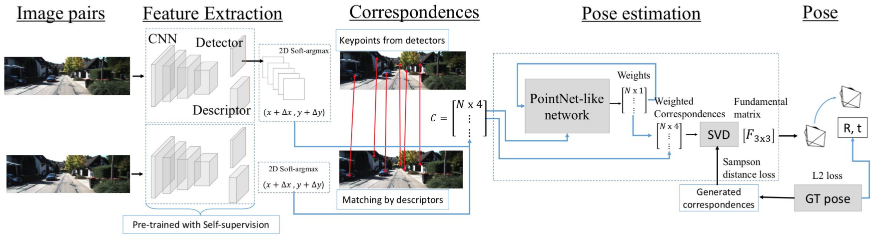
Outlines

- Introduction
- Visual odometry and SLAM
- Related work
- Deep keypoint-based camera pose estimation
- Deep learning-based visual odometry on various datasets
- Summary and future work

Summary

- Overview for visual odometry
- Analysis for geometry-based system -- ORB-SLAM
- A deep keypoint-based pipeline for camera pose estimation
- Analysis for deep learning-based system -- SC-SfMLearner





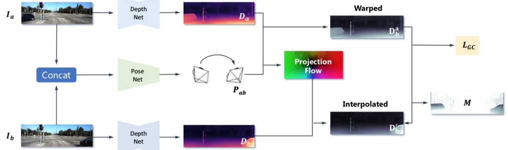


Figure 4.1. Overview of SC-SfMLearner [6].

Future work

- Key from geometry for successful visual odometry
- Deep keypoint-based pose estimation to visual odometry
- Combination of geometry-based and deep learning-based methods

Acknowledgements

Advisors

- Committee:
  - Professor Manmohan Chandraker
  - Professor Nikolay Atanasov
  - Professor Hao Su
  - Professor Nuno M. Vasconcelos
- Professor Mohan Trivedi
- Professor Shao-Yi Chien, Dr. Po-Chen Wu (NTU)
- Dr. Wei-Chao Chen, Dr. Trista Chen (Inventec)
- Stephanie Mathew (ECE)

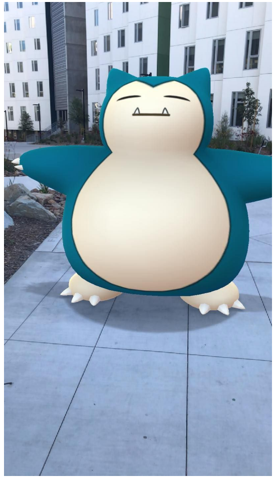
Friends, Co-workers

- Rui Zhu
- Bowen Zhang
- Giayuan Gu, Shuang Liu
- Ishit Mehta
- Fred Lin
- Joseph Li-Yuan Chiang, Vanessa Chang

Acknowledgements



Thank you



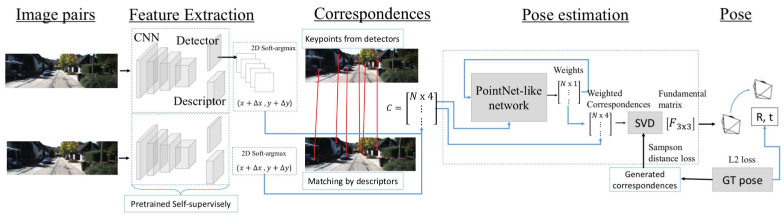
Github: https://github.com/eric-yyjau

Backup slides

Motivation and problem description

- Camera pose estimation has been the key to Simultaneous Localization and Mapping (SLAM) systems
- SIFT + RANSAC method has dominated the design of camera pose estimation pipeline for decades.
- Basic challenges for learning-based systems.
  - Not trained and optimized end-to-end for the ultimate purpose of camera poses
  - The over-fitting nature of training-based methods
  - Existing learning-based keypoint detector is weaker than SIFT

Method details and analysis



- Geometry-based loss
  - Correspondences → Fundamental matrix
  - Fundamental matrix → solve R, t
  - Optimize over the best R, t (min. error)

Loss = L(rot) + λ * L (trans)

L(rot) = || GT rot - Est. rot $||_2$

L (trans) = || GT trans - Est. trans$||_2$

$$\mathcal{L}_{pose} = min(\mathcal{L}_{rot}(\mathbf{R}_{est}, \mathbf{R}_{gt}), c_r) + \lambda_{rt} * min(\mathcal{L}_{trans}(\mathbf{t}_{est}, \mathbf{t}_{gt}), c_t),$$

$$\mathcal{L}_{rot} = min(\|q(\mathbf{R}_{est\_i}) - q(\mathbf{R}_{gt})\|_2), i = [1, 2],$$

$$\mathcal{L}_{trans} = min(\|\mathbf{t}_{est\_i} - \mathbf{t}_{gt}\|_2), i = [1, 2],$$

Contribution

- A new end-to-end trainable framework for feature extraction, matching, outlier rejection, and relative pose estimation
- The pipeline is tightly connected with the novel *Softargmax* bridge, and optimized with geometry-based objective obtained from correspondences
- The thorough study on cross-dataset setting is done to evaluate generalization ability, which is critical but not much discussed in the existing works

Experiment settings

- Baselines
  - SIFT + RANSAC (Si-base)
  - SuperPoint + RANSAC (Sp-base)
  - SIFT + DeepF[34] (Si-models)
  - Our method – no end-to-end training (Sp-models)
  - Our method - with end-to-end training (DeepFEPE)
- Datasets
  - KITTI
  - ApolloScape

[34] Rene´ Ranftl and Vladlen Koltun. Deep Fundamental Ma- trix Estimation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, Computer Vision – ECCV 2018, volume 11205, pages 292–309. Springer Inter- national Publishing, Cham, 2018.