Radar SLAM for Autonomous Driving

TUM Data Innovation Lab

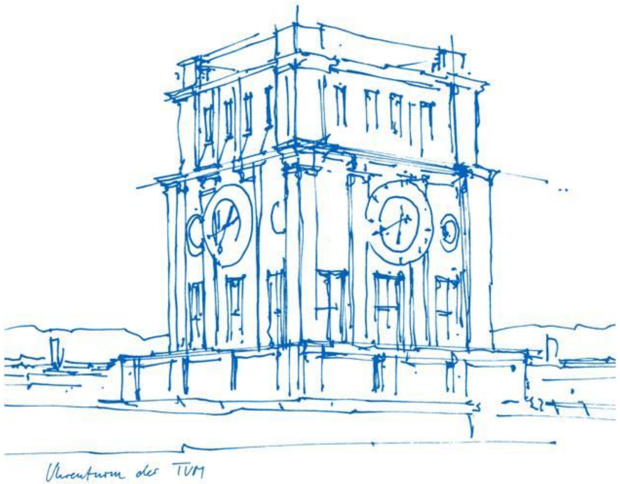
Team:
Felix Bergmann, Frithjof Winkelmann, Hans Schmiedel,
Michael Seegerer

Mentor: Dr. Georg Kuschk Astyx GmbH / Cruise

Co-Mentor: M.Sc. Fabian Wagner

Project Lead: Dr. Ricardo Acevedo Cabra

Supervisor: Prof. Dr. Massimo Fornasier



Motivation

How do cars drive on their own?

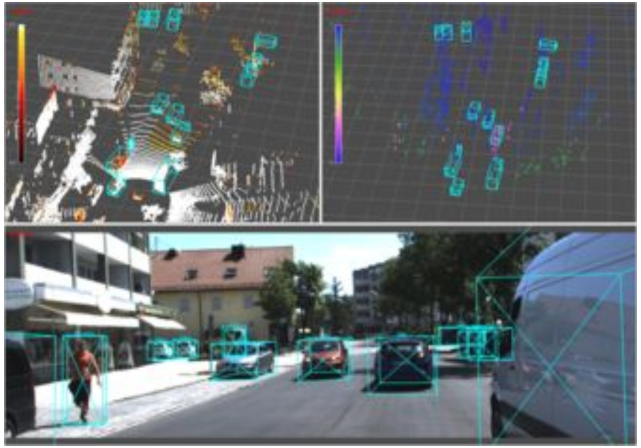
- Perceive environment with sensors
- Recognize roads, obstacles, other road users, …
- Follow path according to internal map

➔ Need map and cars position (SLAM)

Why Radar SLAM?

- almost weather-independent

Project’s goal: Evaluate different SLAM approaches to model the environment out of sensor data



@Astyx GmbH / Cruise

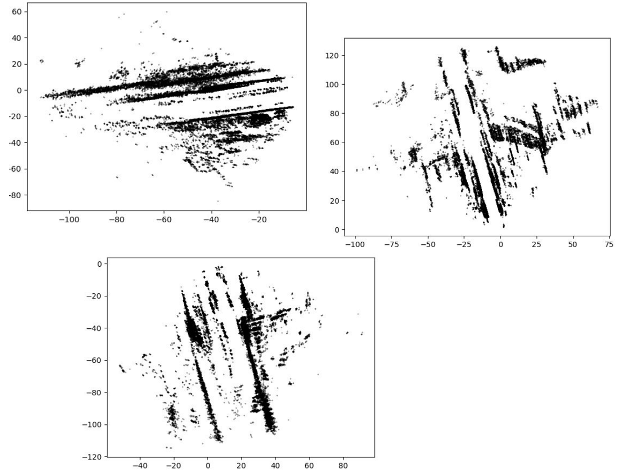
Contents

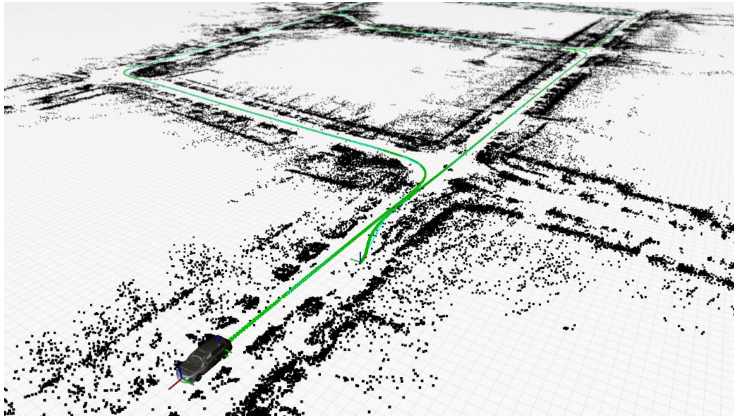
What is SLAM

Simultaneous **L**ocalization **a**nd **M**apping:

- Map a robot’s environment
- Locate itself inside this map

Sensors

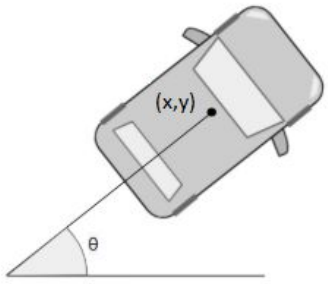


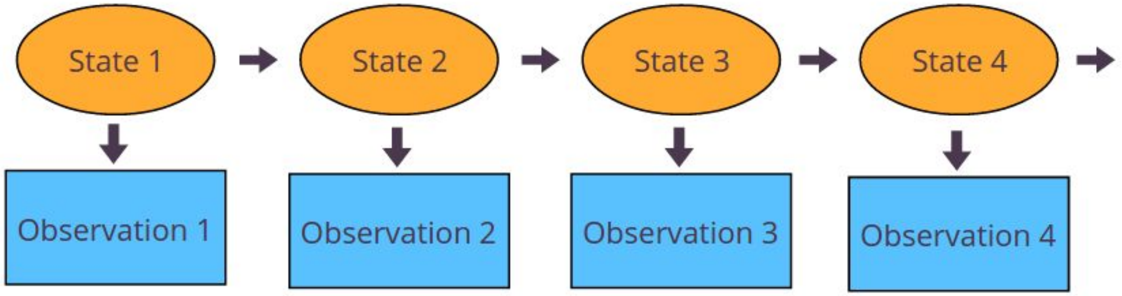


M. F. Holder, S. Hellwig, and H. Winner. “Real-Time Pose Graph SLAM based on Radar”. In: 2019 IEEE Intelligent Vehicles Symposium (IV). ttps://tuprints.ulb.tu-darmstadt.de/8756/.

What is SLAM

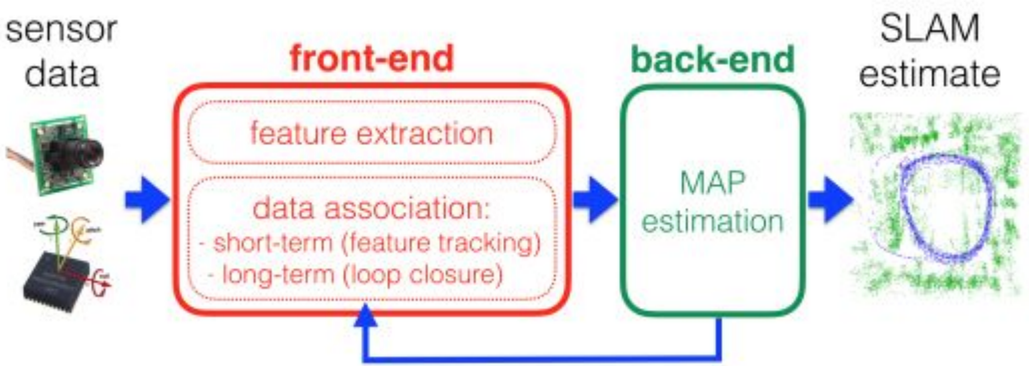
- Sequence of robot's states (position and rotation)
- Estimate next state with previous states and measurements





What is SLAM

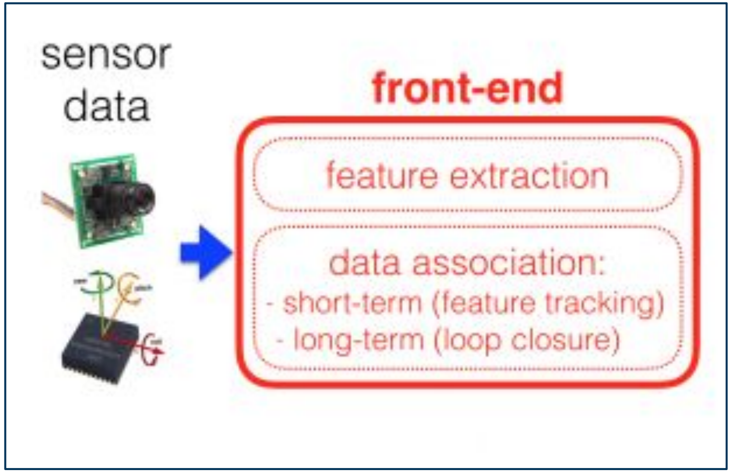
Typically divided into Front- and Back-end:



C. Cadena et al. “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age”. In: IEEE Transactions on Robotics 32.6 (2016)

Front-end

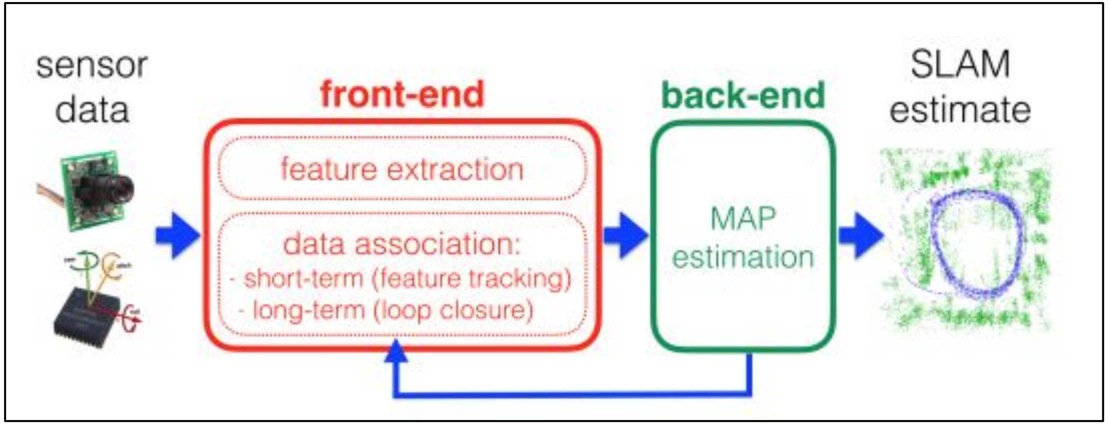
- Preprocess sensor data
- Find important features in sensor scans (feature extraction)
- Compare scans (data association / scan matching)
- Estimate new states of the robot



C. Cadena et al. “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age”. In: IEEE Transactions on Robotics 32.6 (2016)

Back-end

- Construct the map
- Correct errors and optimize state transitions
- Fix detected loops



C. Cadena et al. “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age”. In: IEEE Transactions on Robotics 32.6 (2016)

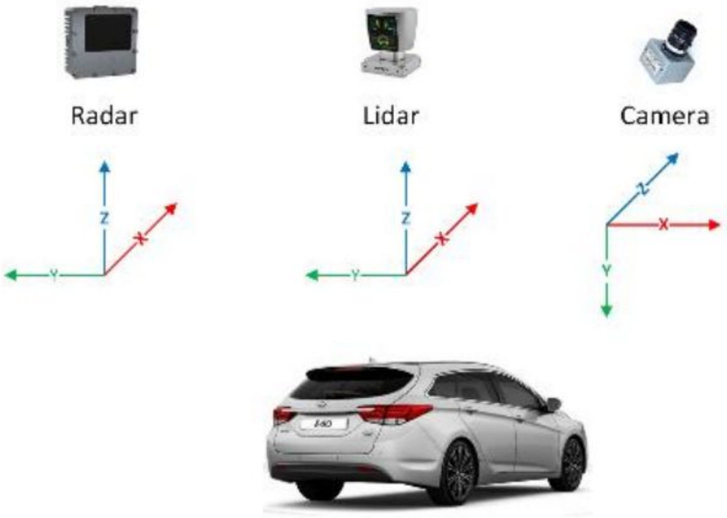
Content

1. SLAM
2. Datasets
3. Classical approaches
4. Deep learning approaches
5. Conclusion

Datasets

Master coordinate system = Radar

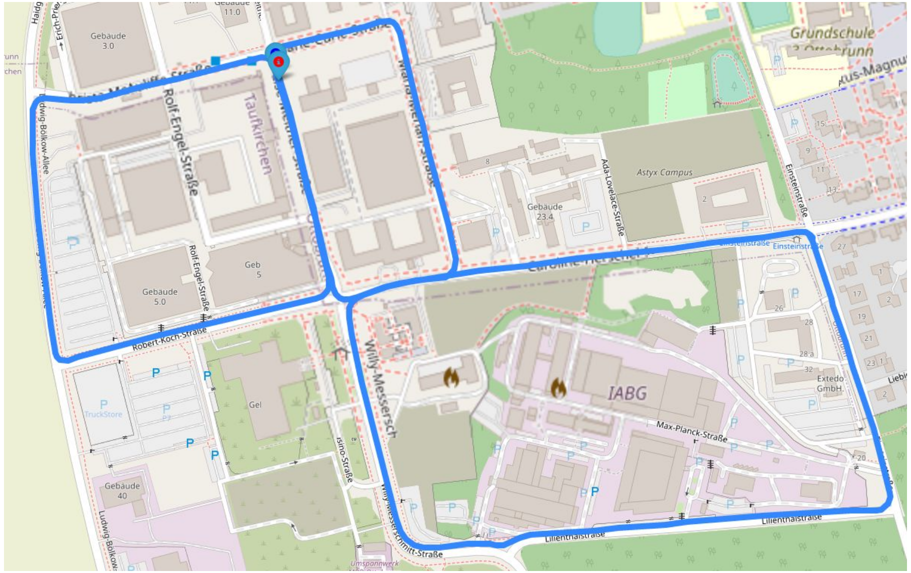
Local 3D sensor coordinate systems (COS)



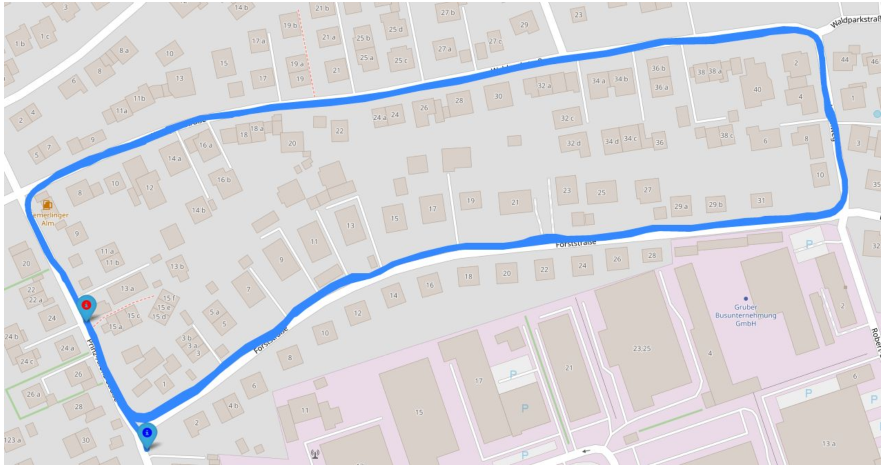
@Astyx GmbH / Cruise

Datasets

Sequence A

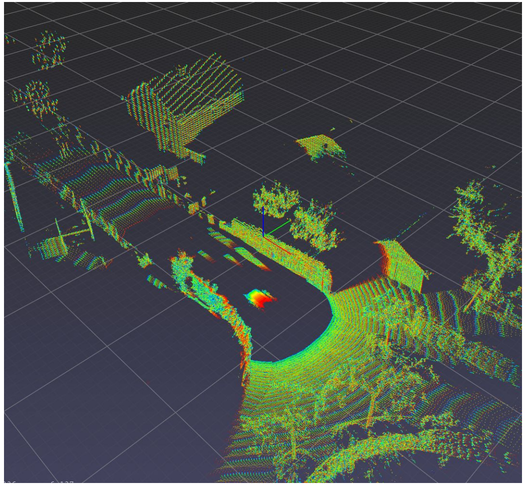


Sequence B



Astyx GmbH/Cruise Sensors

Lidar



@Astyx GmbH / Cruise

Radar Sensor

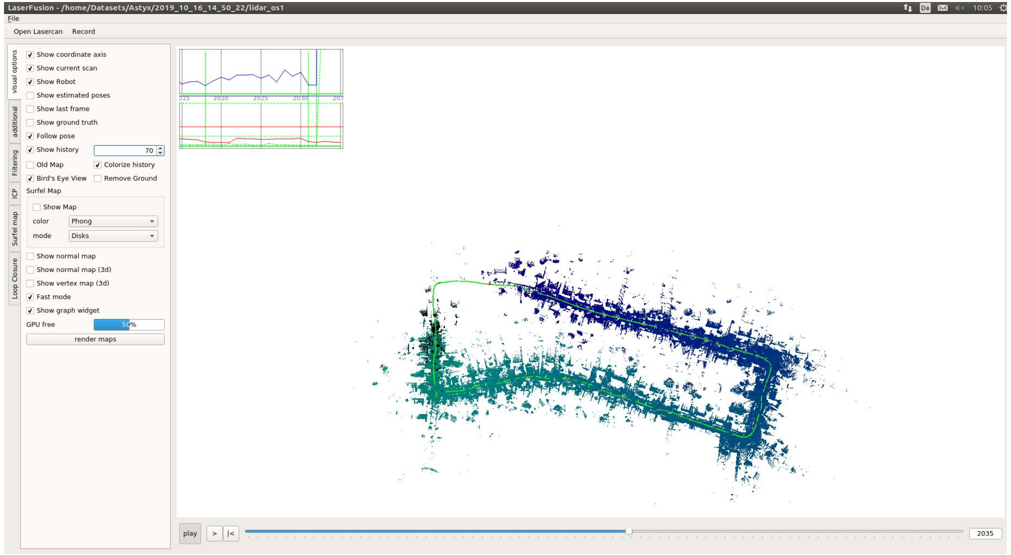
Captured at 10Hz

GPS & IMU

- Location (~0.5m accuracy)
- Orientation (~3° accuracy)
- Acceleration

Ground truth for Astyx GmbH/Cruise Datasets

~0.5m accuracy (from GPS) is not accurate enough
=> **IDEA:** Getting ground truth using Lidar SLAM approach



Existing Lidar SLAM implementations:

- Surfel-based Mapping (SuMa)
- Lidar Odometry and Mapping

For dataset B we were able to obtain Ground Truth information for approximately ~2000 scans (half of the scans from the dataset).

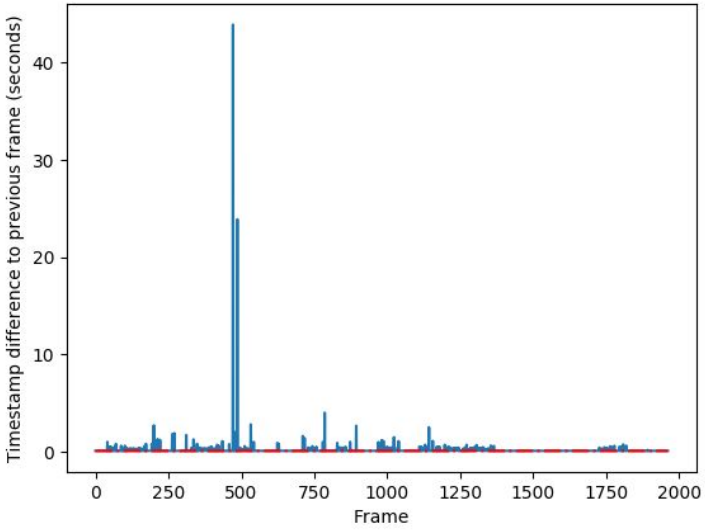
Ground truth for Astyx GmbH/Cruise Datasets

- Lidar based Ground Truth creation fails with the previous mentioned approaches.
- **Problem:** Measurement scan matching is here mostly based on Iterative Closest Point (ICP) --> unstable for large time difference between the single scans.
- Visualization of one mismatched in the ICP process on the left.

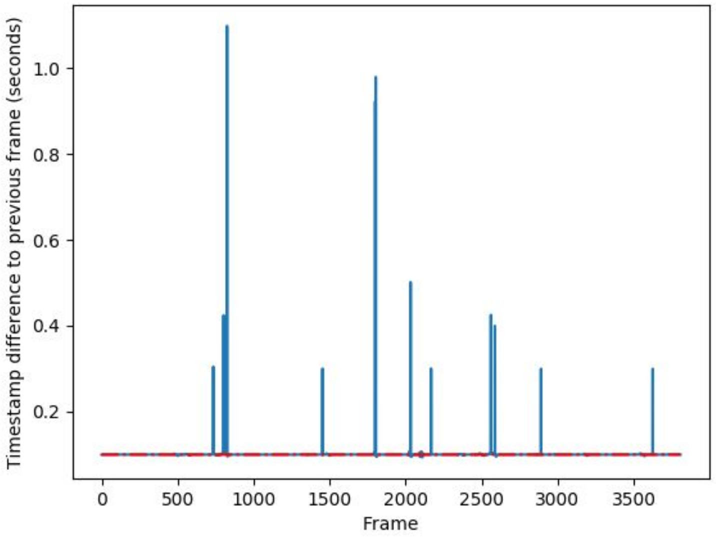
Ground truth for Astyx GmbH/Cruise Dataset

Time difference between measurements

Sequence A

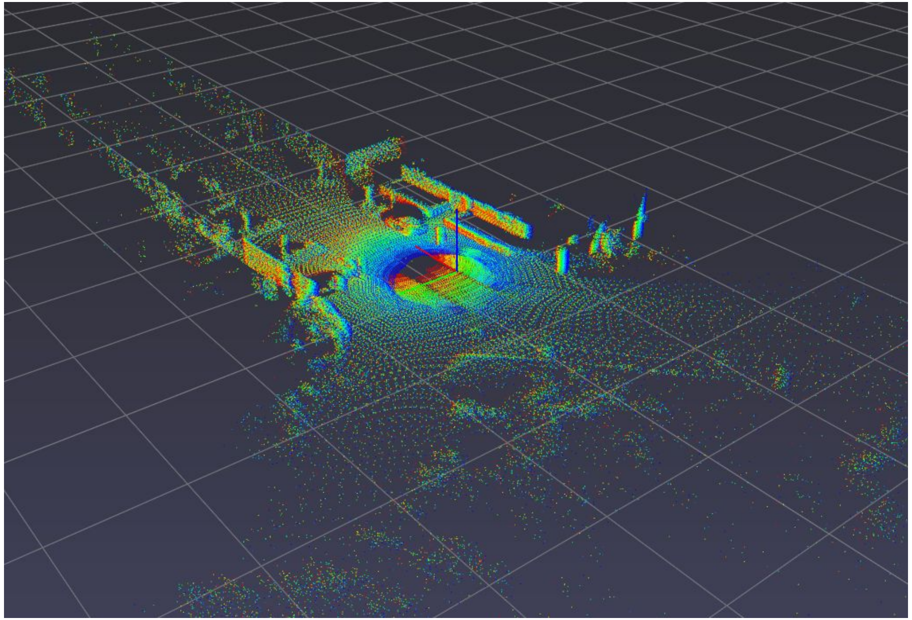


Sequence B



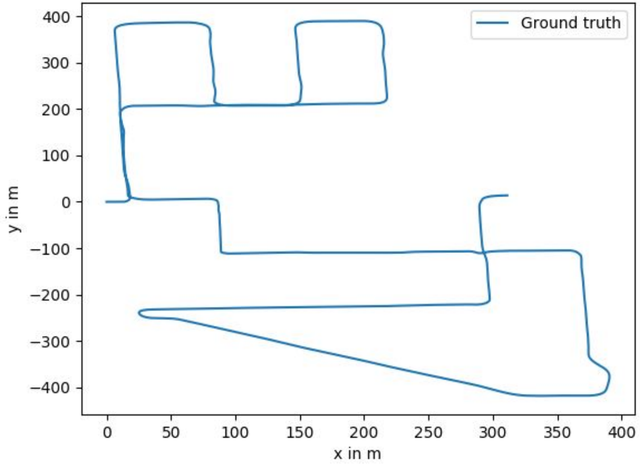
KITTI Odometry dataset

Lidar



GPS

- Location (~0.01m accuracy)
- Orientation (~0.03° accuracy)

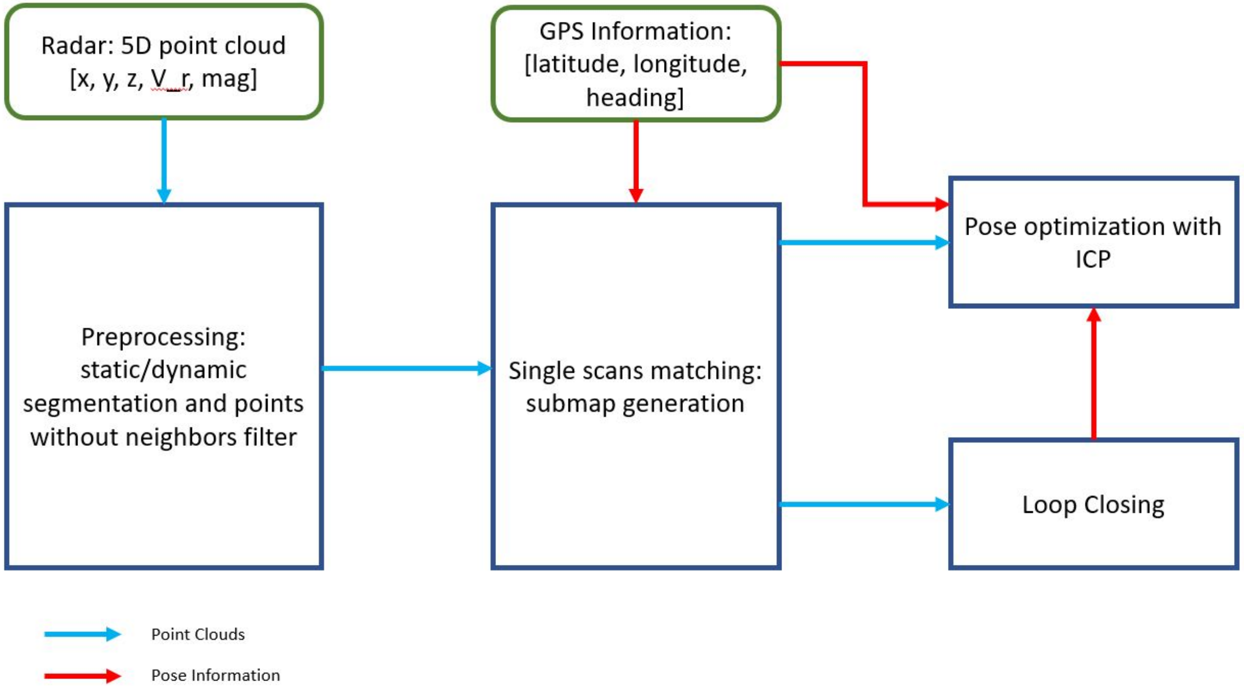


Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: Conference on Computer Vision and Pattern Recognition (CVPR). 2012.

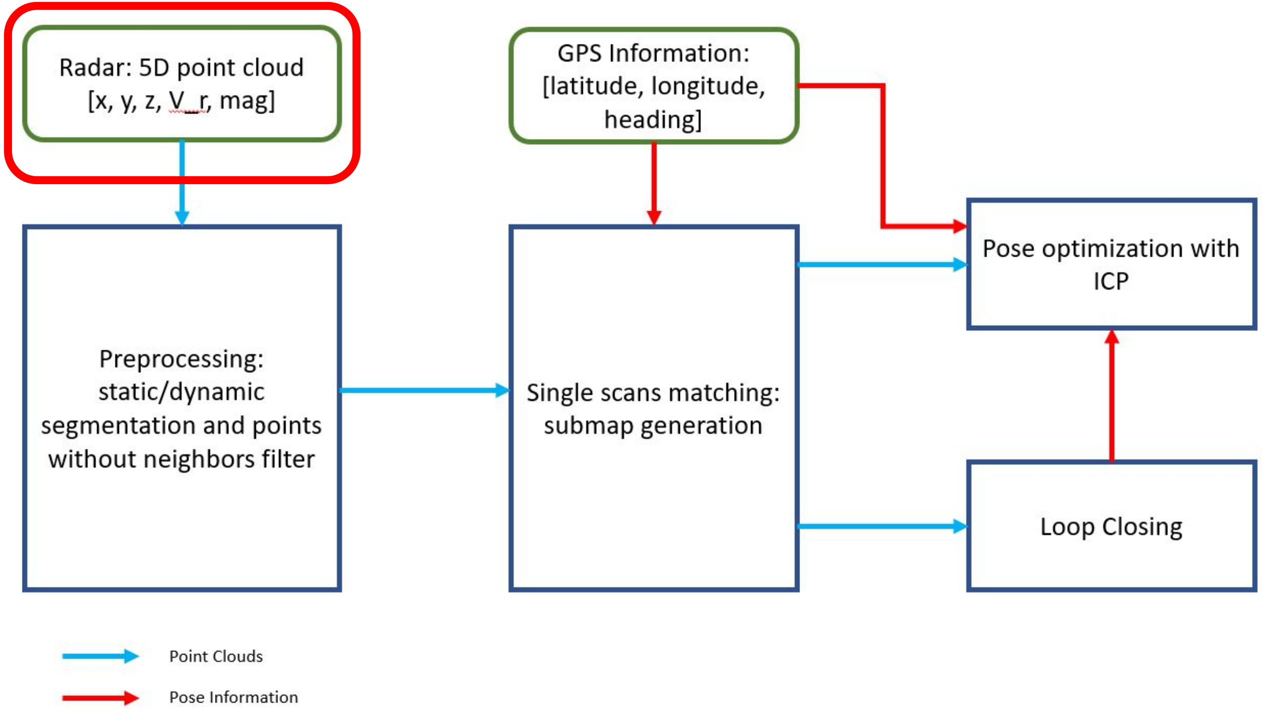
Content

1. SLAM
2. Datasets
3. Classical approaches
4. Deep learning approaches
5. Conclusion

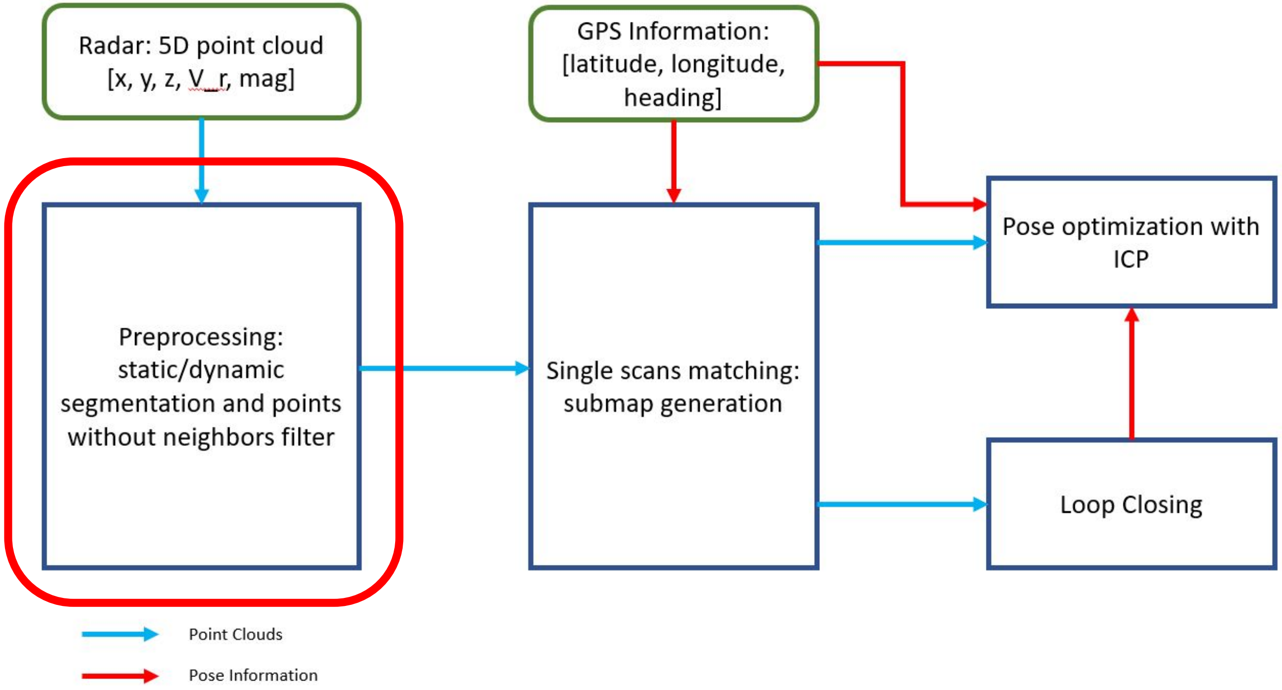
Classical methods



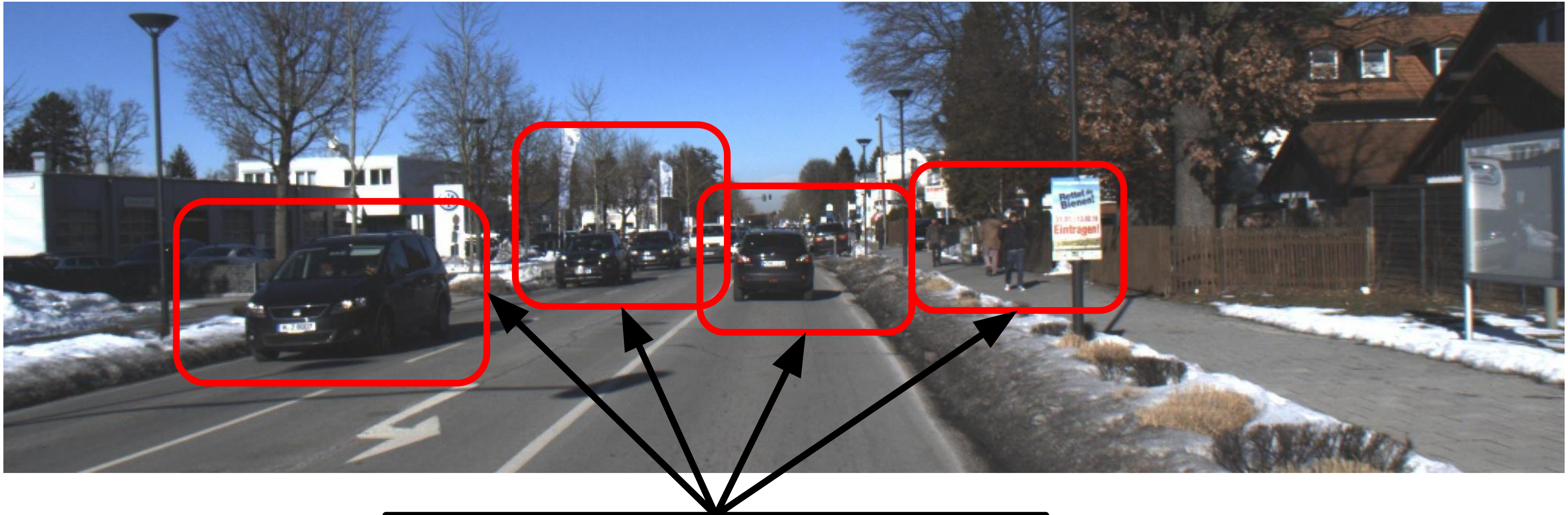
Classical methods



Classical methods

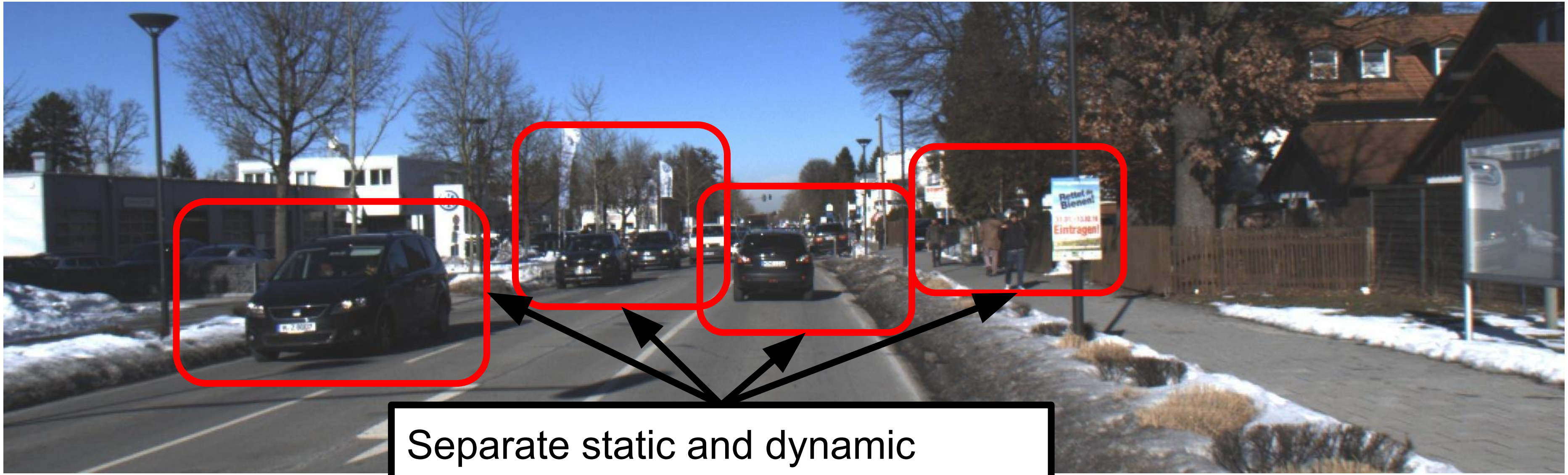


Classical methods



Dynamic detections sources in the scene

Classical methods



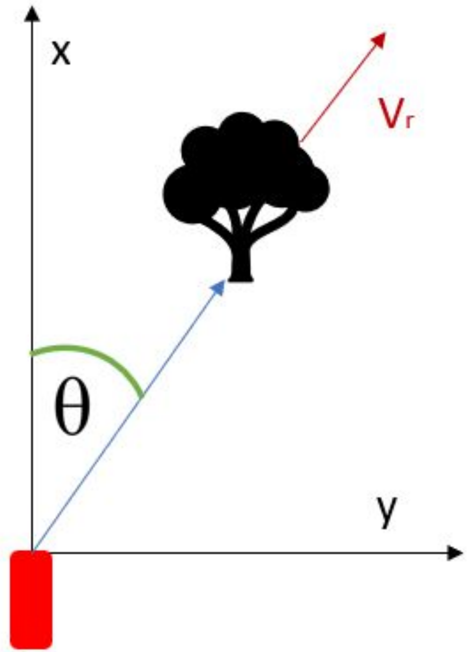
Separate static and dynamic detections using random sample consensus (RANSAC).

Martin A. Fischler and Robert C. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: Commun. ACM 24.6 (June 1981)

Classical methods

Each detection point in the point cloud gets a direction of arrival angle to the sensor

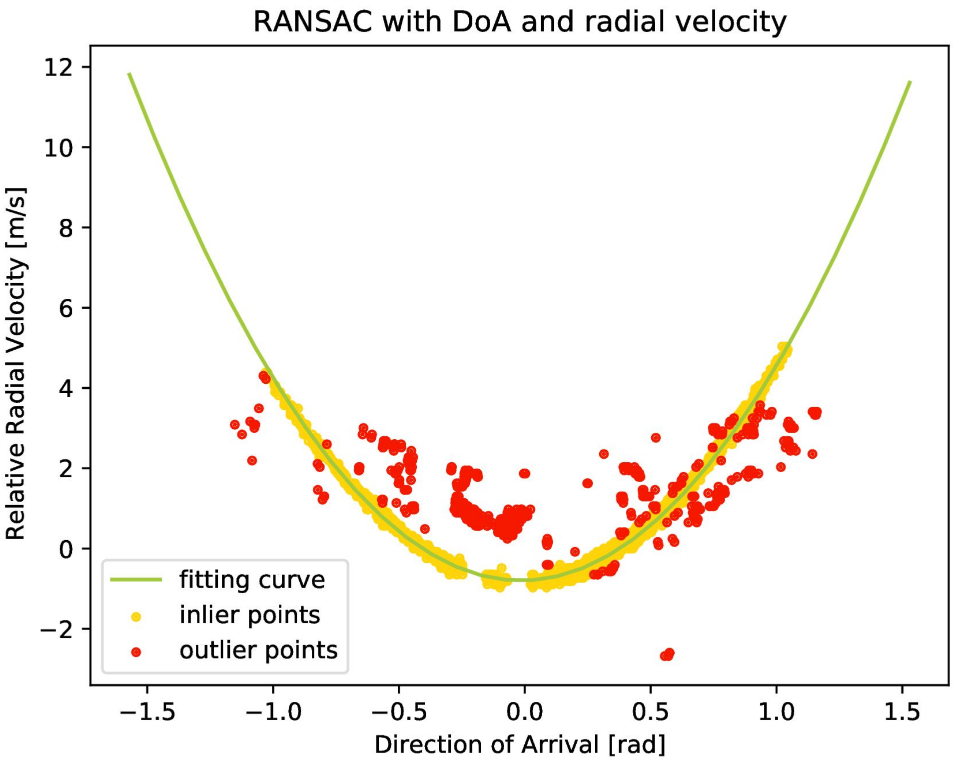
Based on relation between vehicle speed, radial velocity and angle of arrival, dynamic objects can be separated as static or dynamic.



Classical methods

The separation is based on the relation between angle of arrival and radial velocity.

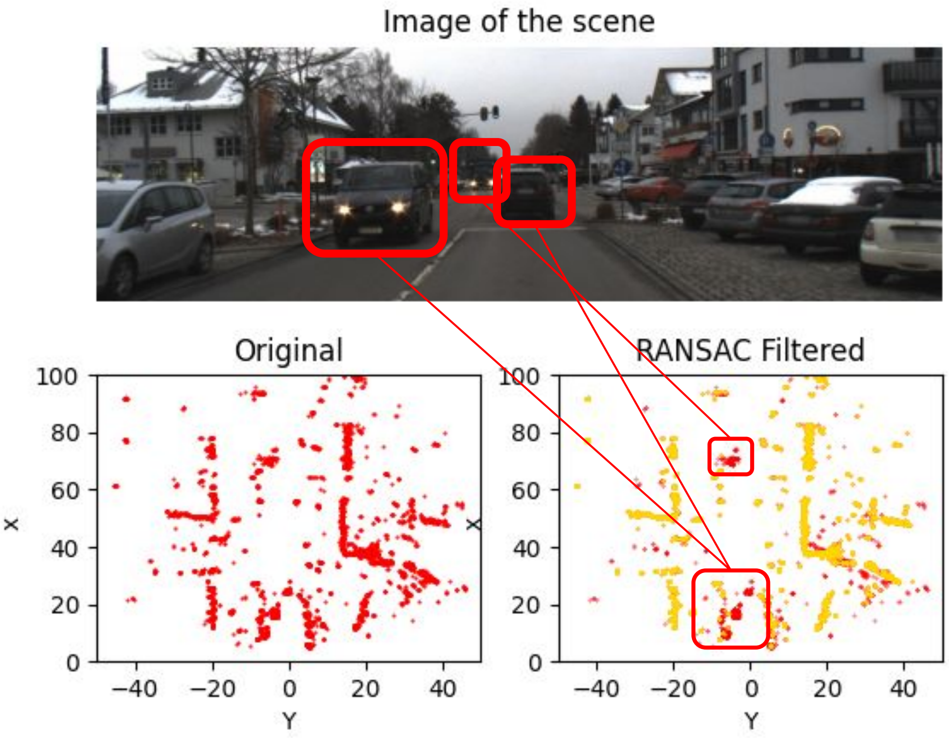
The relation will be approximated with parabola .



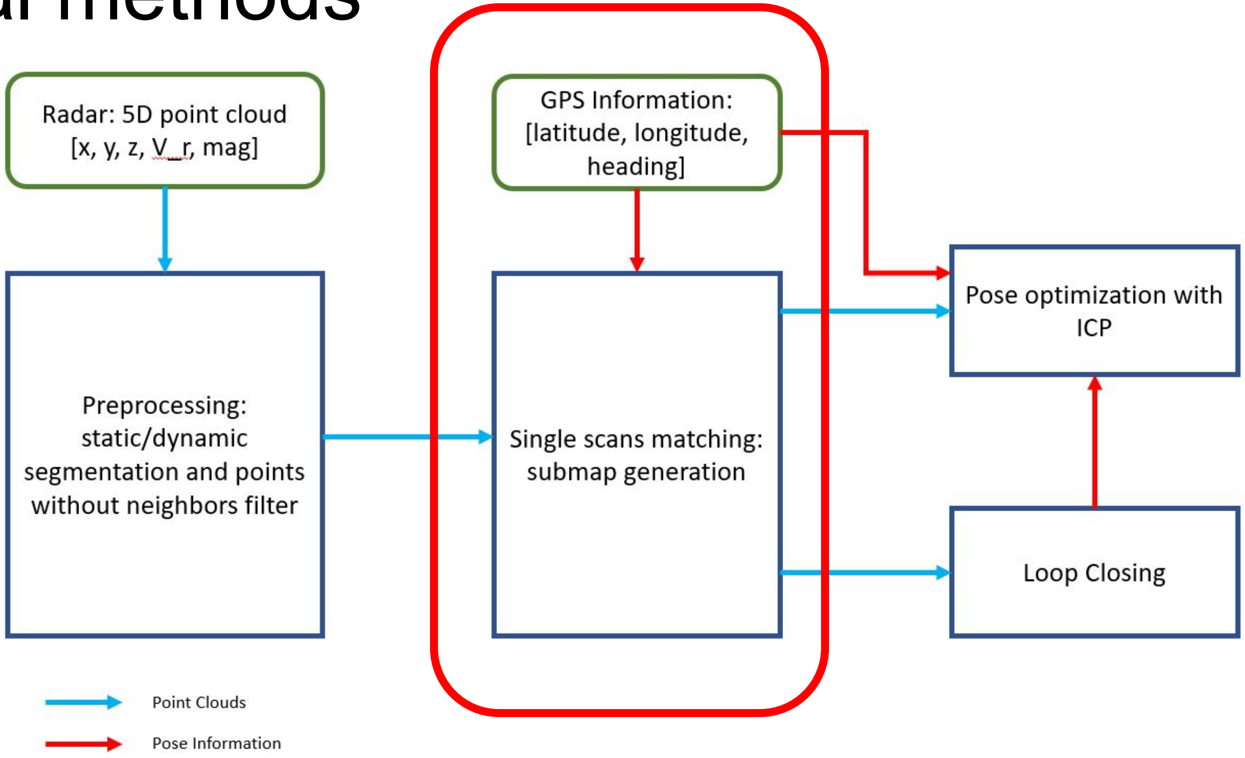
Martin A. Fischler and Robert C. Bolles. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography”. In: Commun. ACM 24.6 (June 1981)

Classical methods

The RANSAC reduces the noise and separate most of the moving objects from the point cloud.



Classical methods

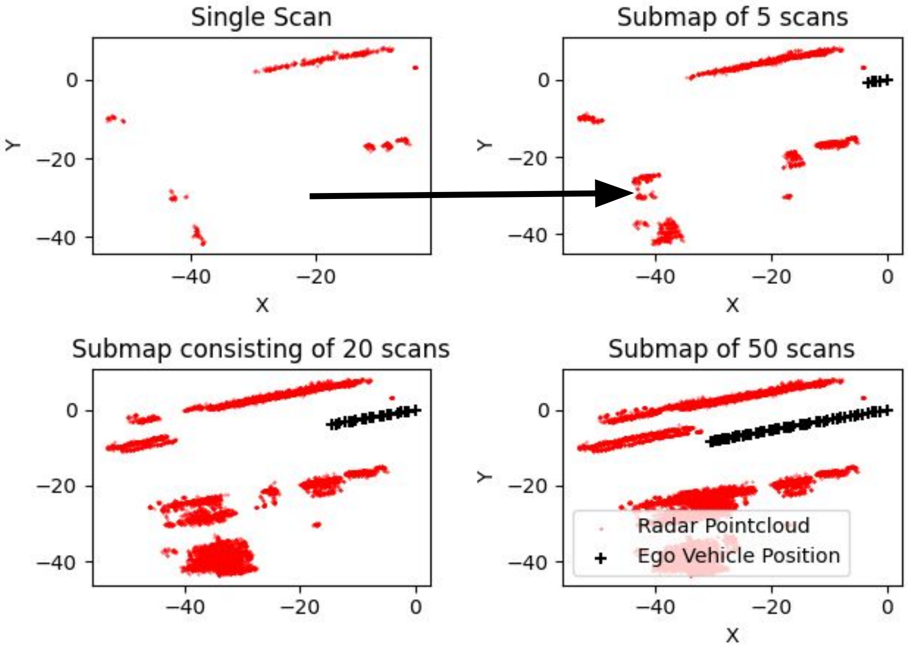


Classical methods

Use translation and rotation information from GPS

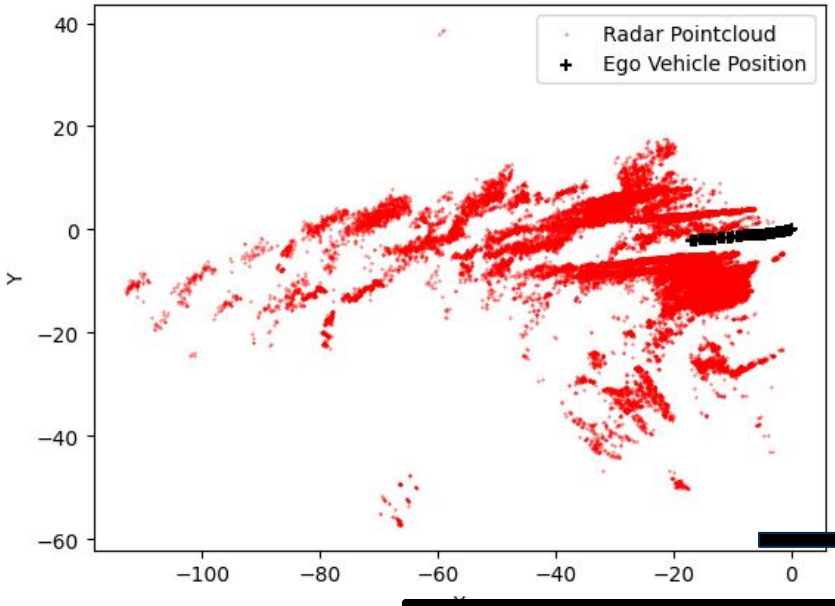
Submap generation:

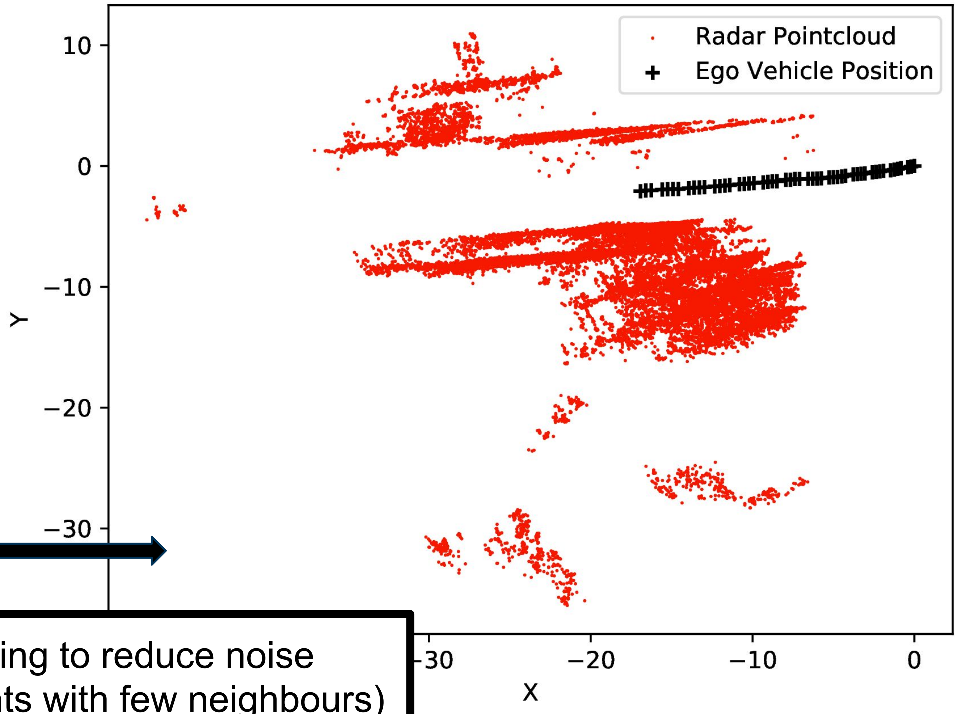
- Merge scans together into same cartesian coordinates (vs. new origin in every scan)
- Position Information from GPS
- ICP on single scans fails due to sparsity and irregularity → ICP on submaps



@Astyx GmbH / Cruise

Classical methods

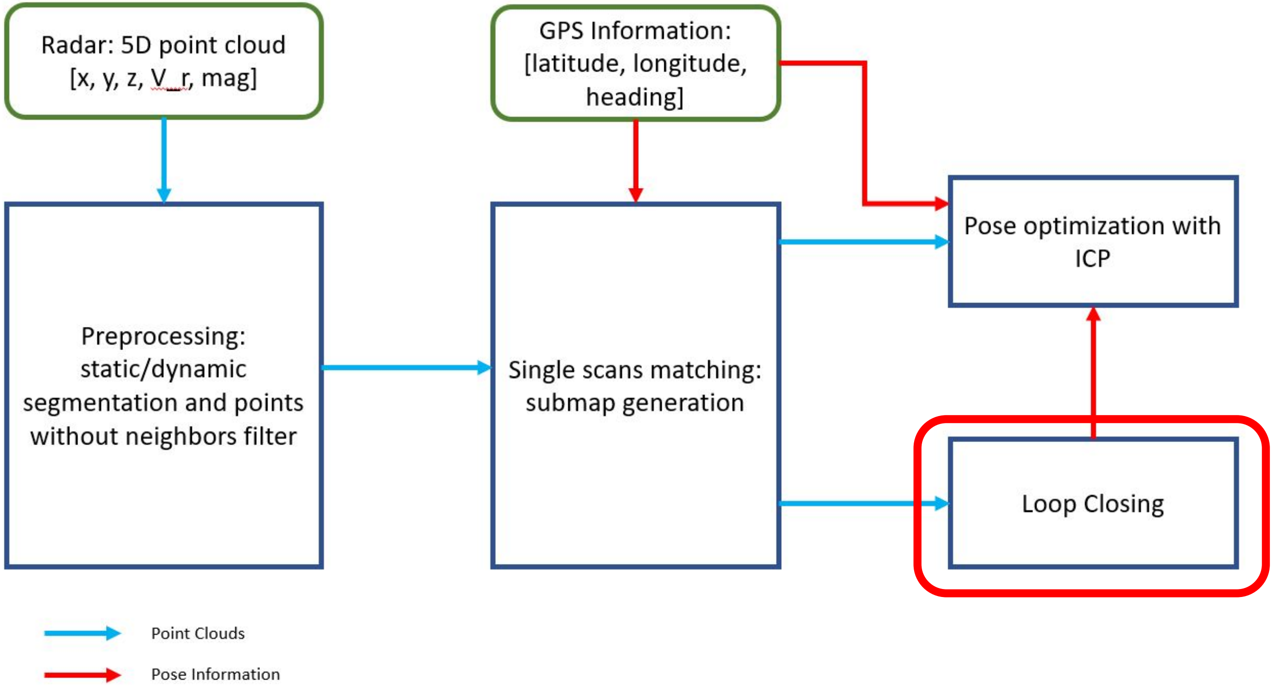




@Astyx GmbH / Cruise

Radius based outlier filtering to reduce noise inside the submaps. (points with few neighbours)

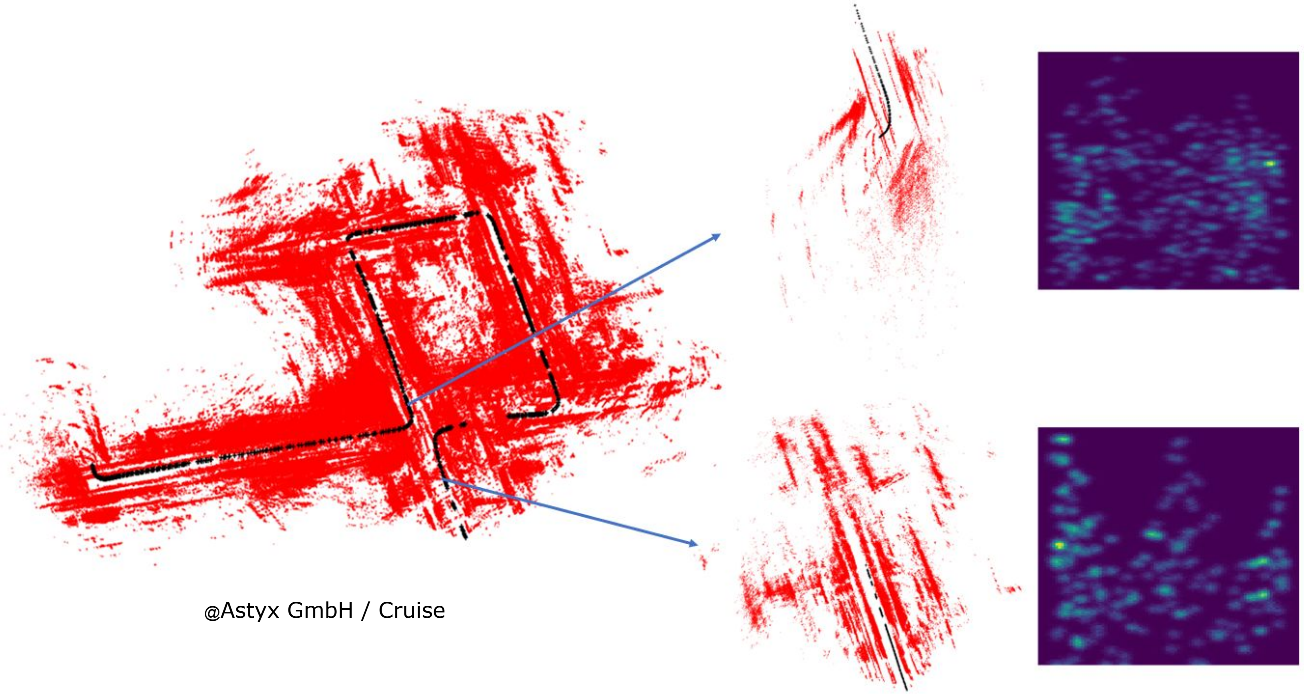
Classical methods



Classical methods

Loop closure with Geometrical Landmark Relations (GLARE)

- Brute force search quickly unfeasible
- False alarm loop detection throws whole map off

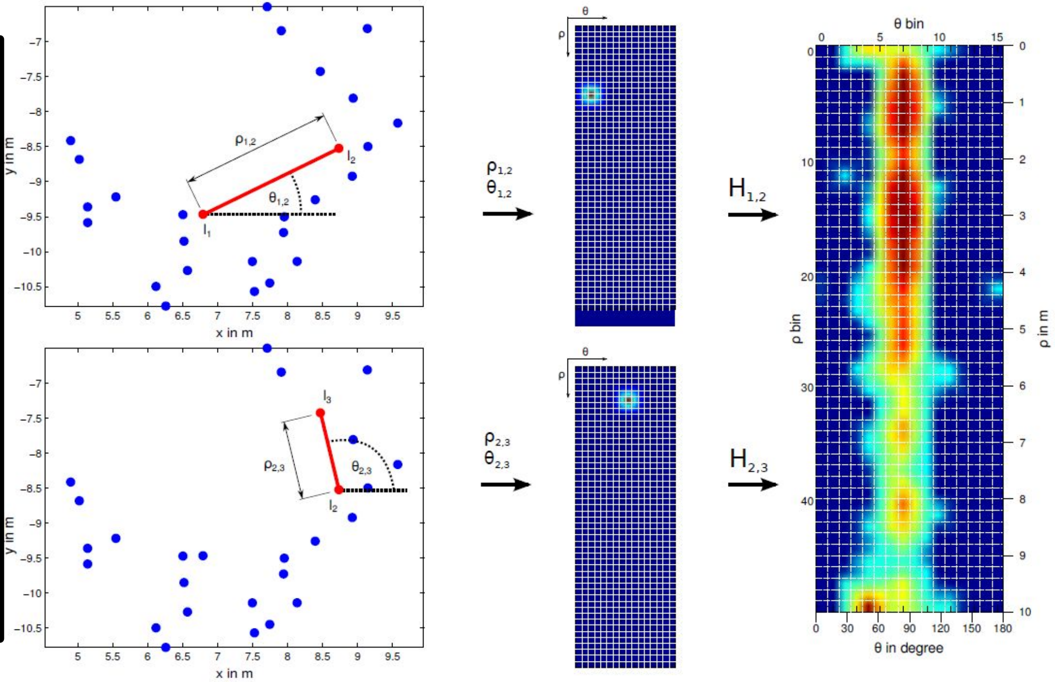


@Astyx GmbH / Cruise

Classical methods

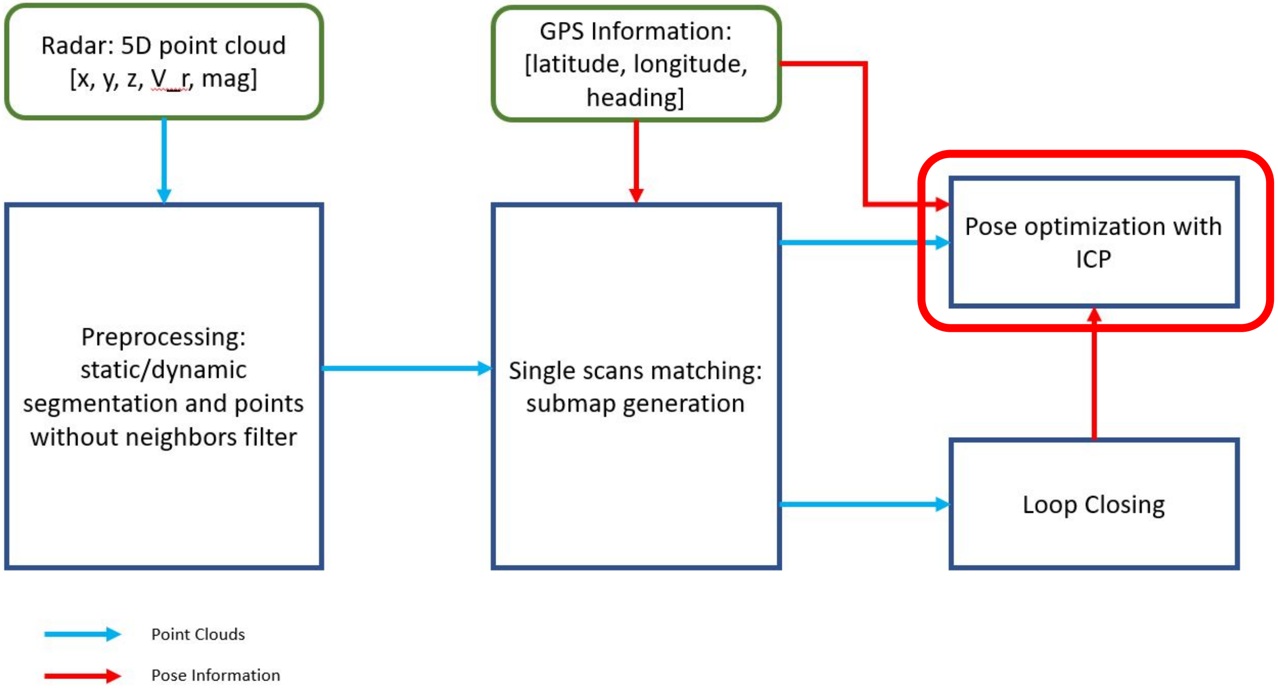
generation of one GLARE signature:

1. feature extraction on submap as image
2. plot angle and distances in histogram
3. add multivariate gaussian
4. sum for signature of submap



Himstedt, Marian & Frost, Jan & Hellbach, Sven & Bohme, Hans-Joachim & Maehle, Erik. (2014). "Large scale place recognition in 2D LIDAR scans using Geometrical Landmark Relations. " in IEEE International Conference on Intelligent Robots and Systems.

Classical methods

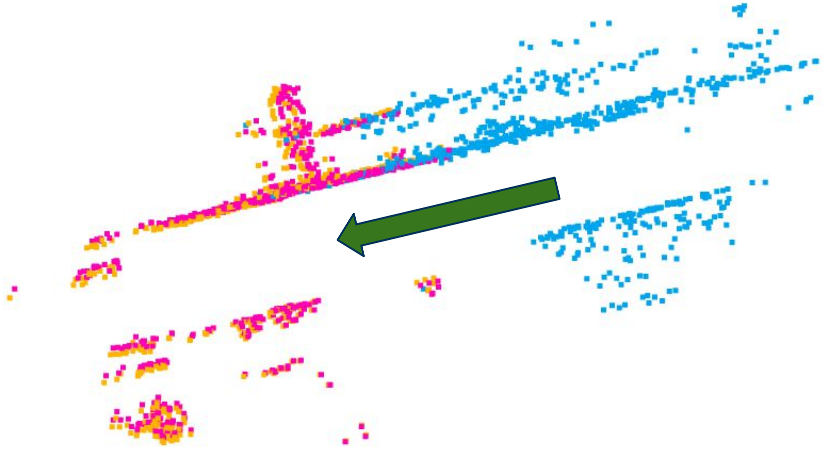


Classical methods

Iterative Closest Point (ICP) optimization

- Minimize difference between two point clouds (here submaps)
- Find transformation matrix (rotation and translation) that transforms starting submap to next submap with most overlap.
- Initial transformation guess from GPS
- Iterative process until convergence

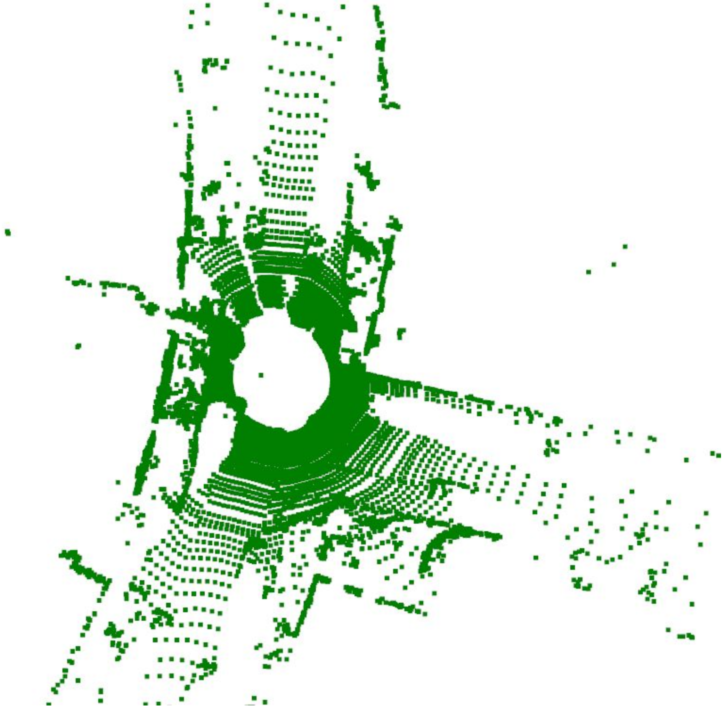
Reduces inaccuracies from using only GPS

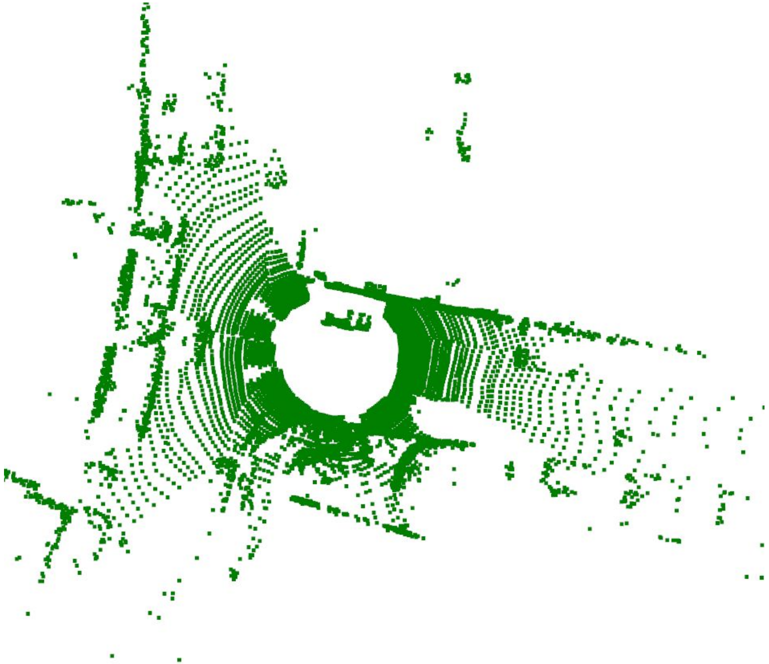


P. J. Besl and N. D. McKay. “A method for registration of 3-D shapes”. In: IEEE Transactions on Pattern Analysis and Machine Intelligence 14.2 (1992)

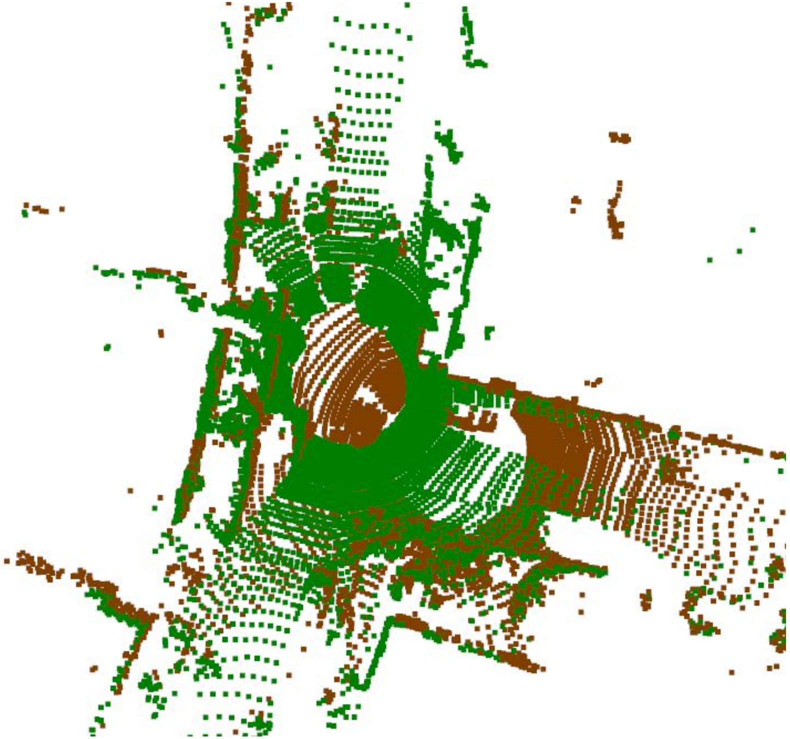
Content

Deep learning approaches

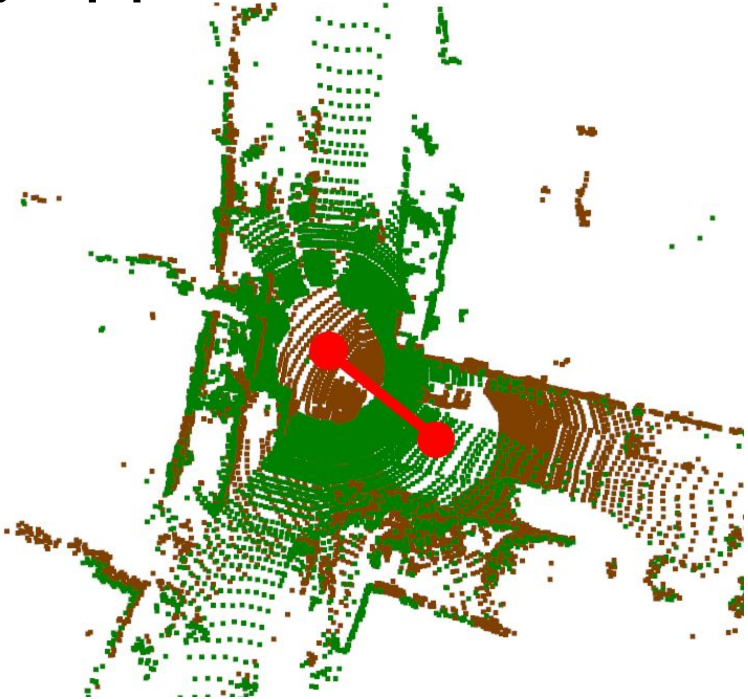




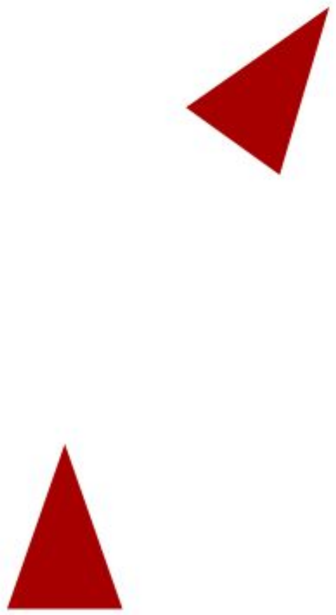
Deep learning approaches



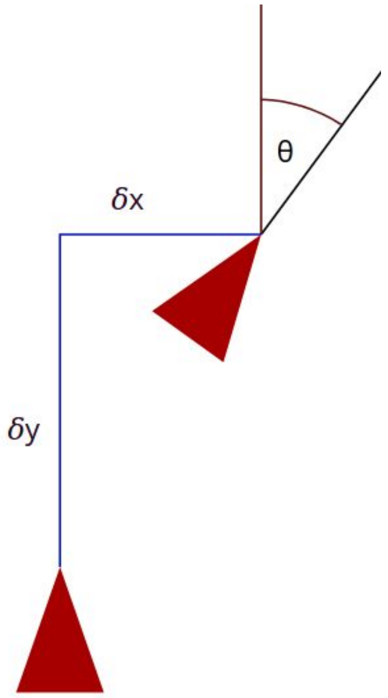
Deep learning approaches



Deep learning approaches

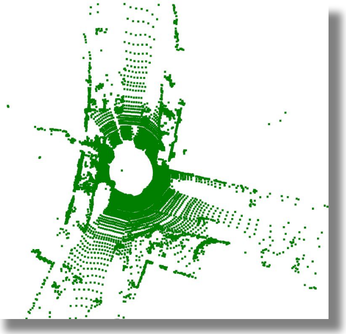


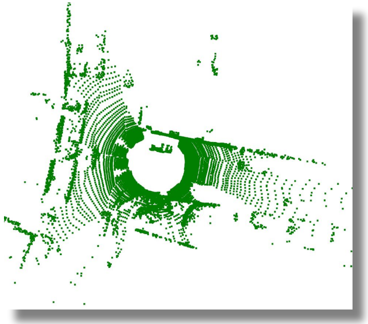
Deep learning approaches



- 2D relative pose instead of 3D pose

Deep learning approaches

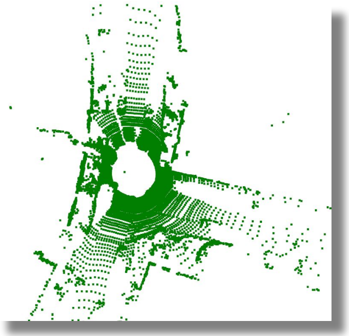




Deep neural network

δ**x,** δ**y, θ**

Deep neural networks for point clouds

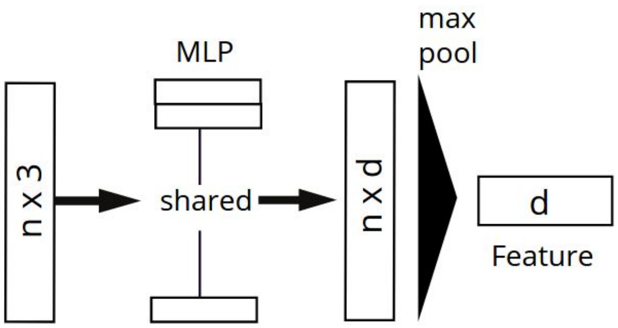


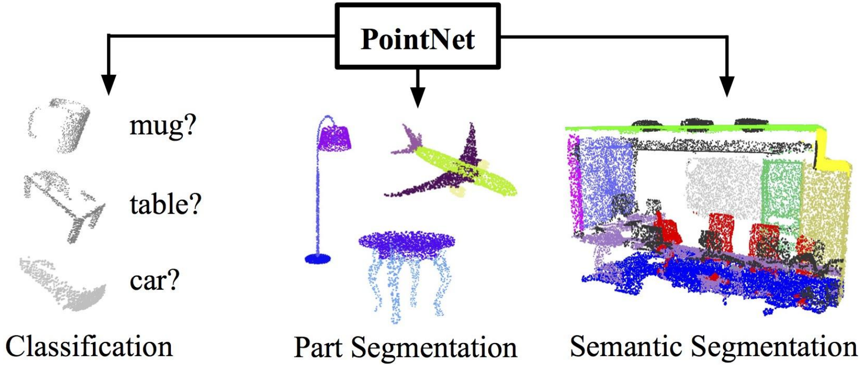
= {p1, p2,...,pn}

Point clouds:

- Are unordered
- Have variable size

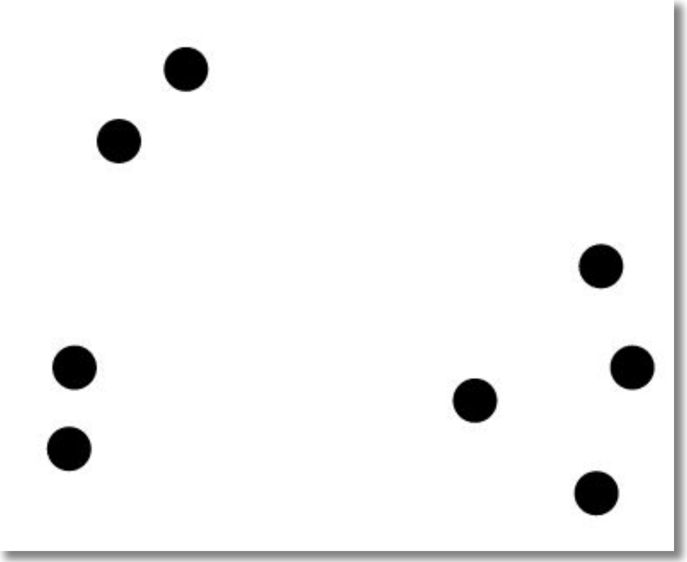
PointNet

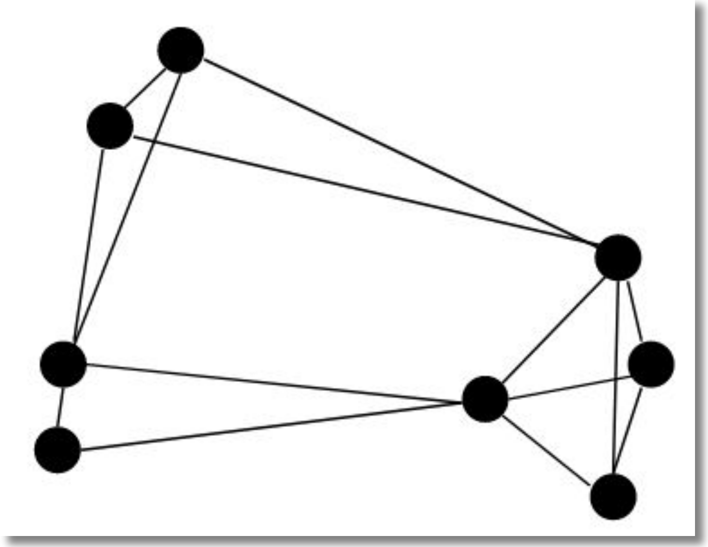




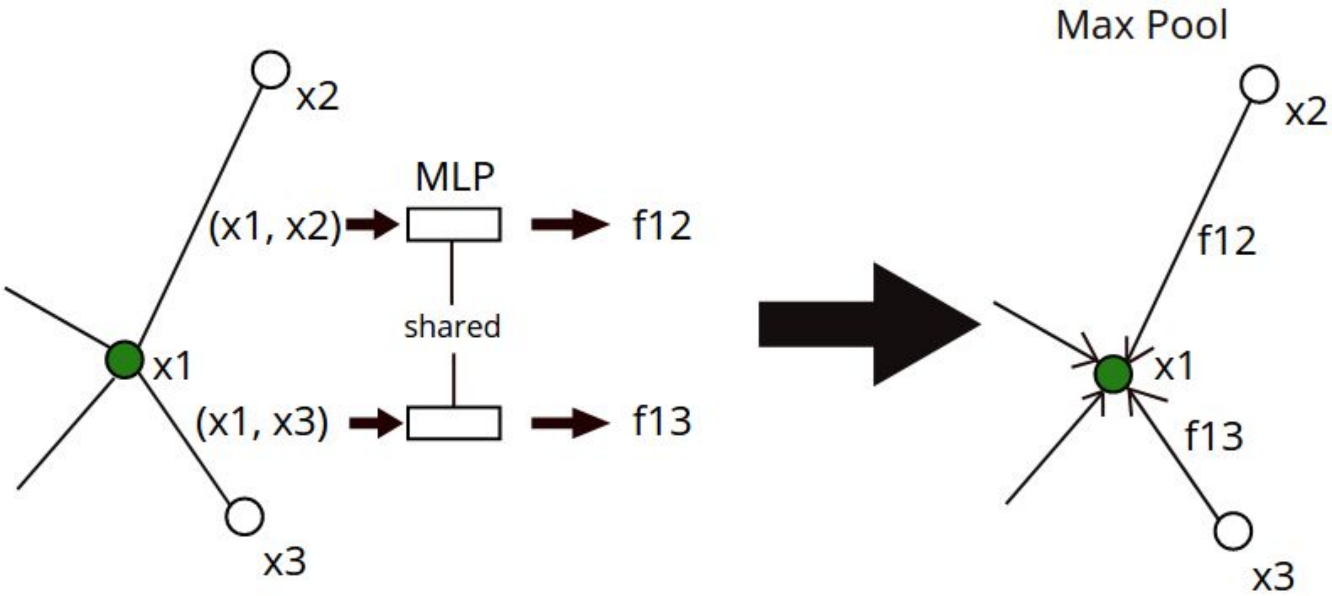
R. Q. Charles et al. “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation”. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017, pp. 77–85.

From point clouds to graphs

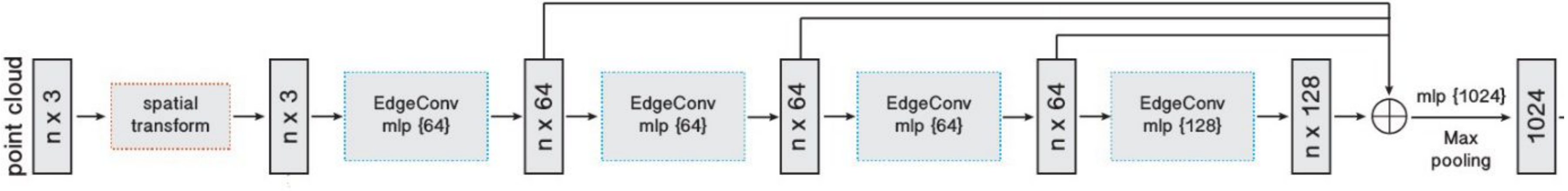




Graph convolutions

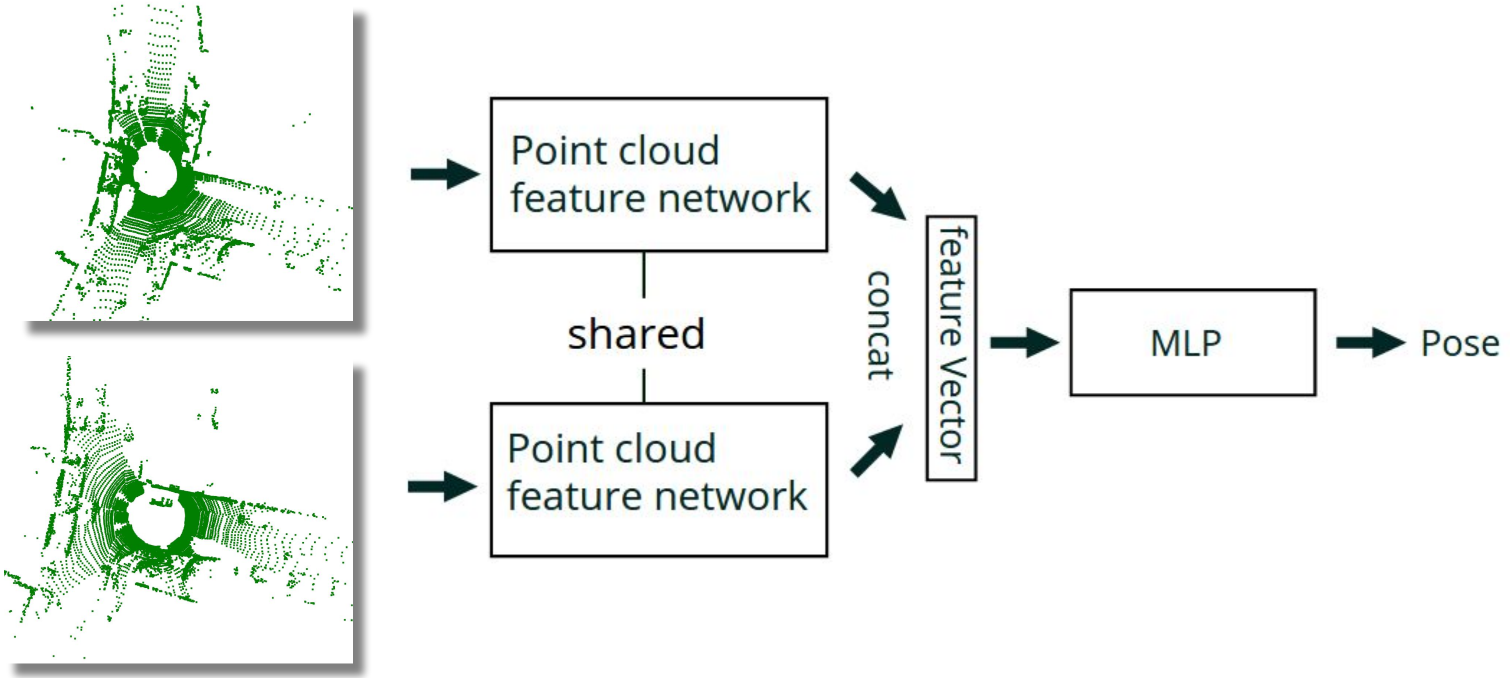


Graph convolutions



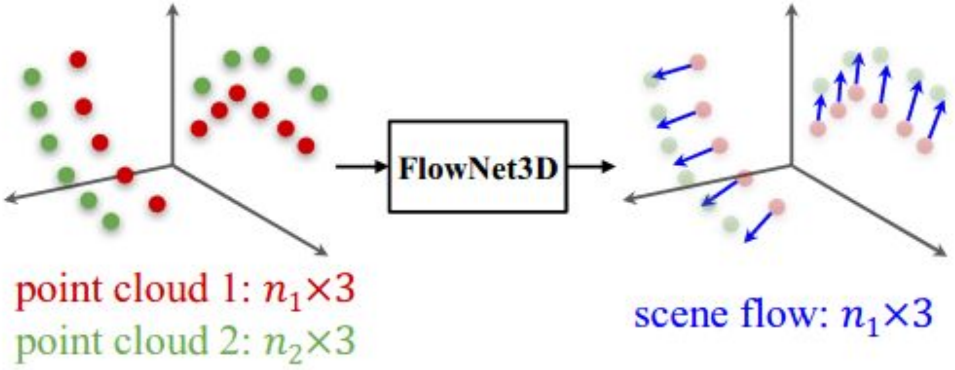
Yue Wang et al. “Dynamic Graph CNN for Learning on Point Clouds”. In: ACM Trans. Graph. 38 (2019), 146:1–146:12.

Pose prediction



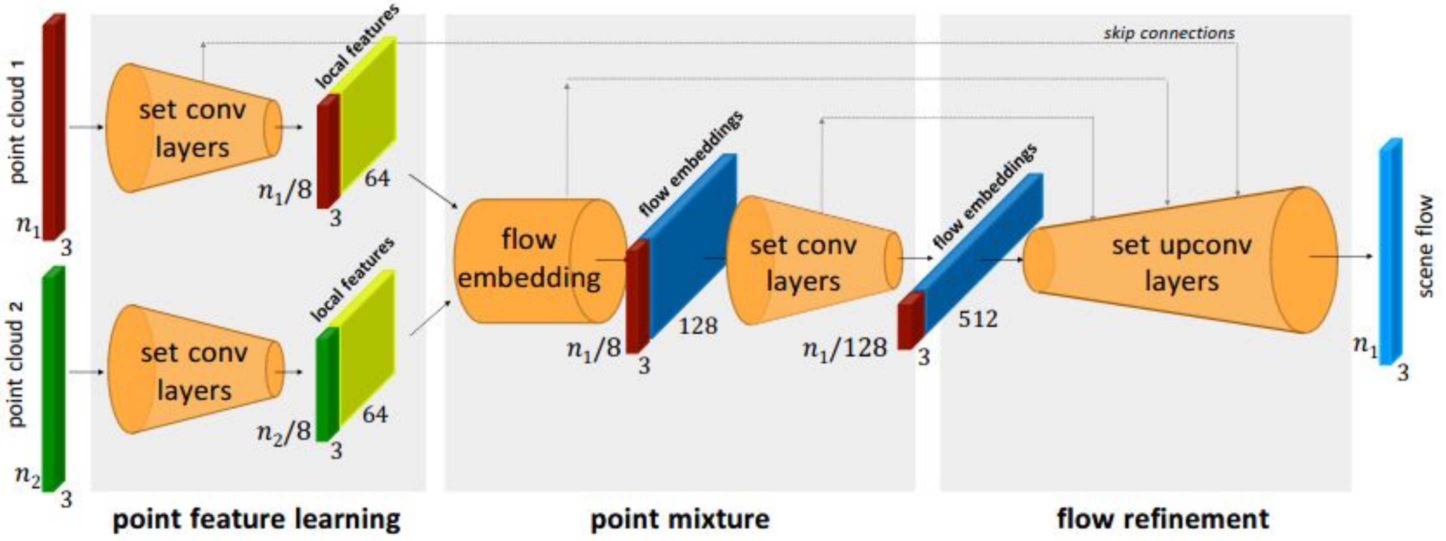
Vinit Sarode et al. One Framework to Register Them All: PointNet Encoding for Point Cloud Alignment. 2019.

FlowNet3D



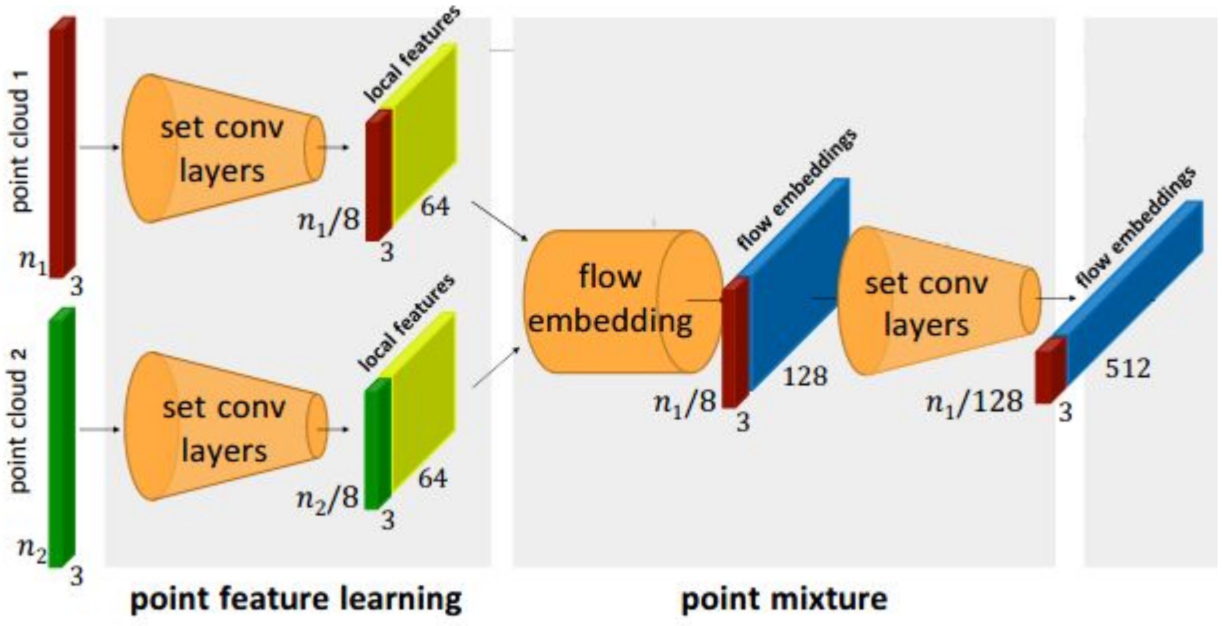
Xingyu Liu, Charles R Qi, and Leonidas J Guibas. “FlowNet3D: Learning Scene Flow in 3D Point Clouds”. In: CVPR (2019).

FlowNet3D



Xingyu Liu, Charles R Qi, and Leonidas J Guibas. "FlowNet3D: Learning Scene Flow in 3D Point Clouds". In: CVPR (2019).

FlowNet3D

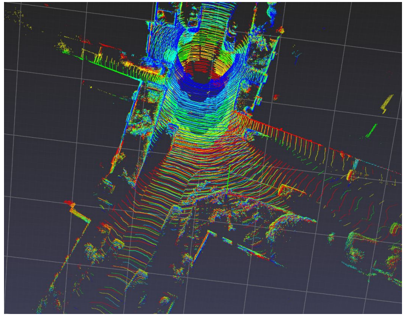


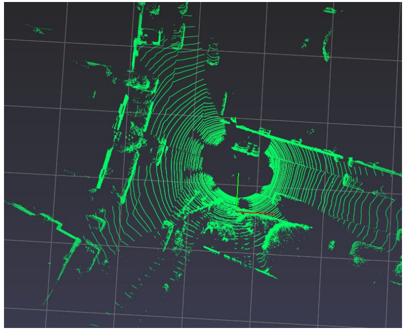
Xingyu Liu, Charles R Qi, and Leonidas J Guibas. “FlowNet3D: Learning Scene Flow in 3D Point Clouds”. In: CVPR (2019).

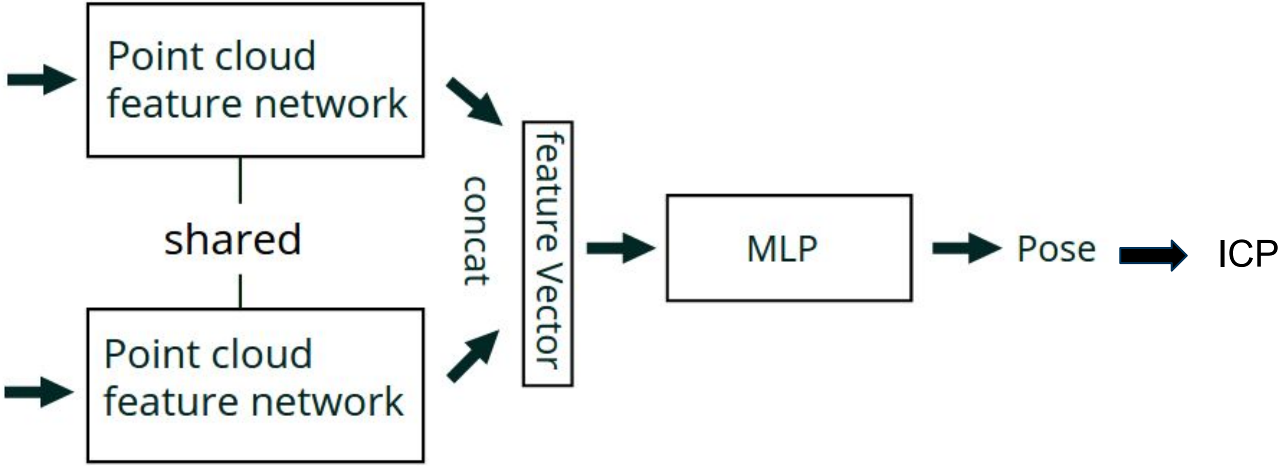
Possible improvements

- Use submaps for the first point cloud
- Refine pose estimate with ICP

Possible improvements

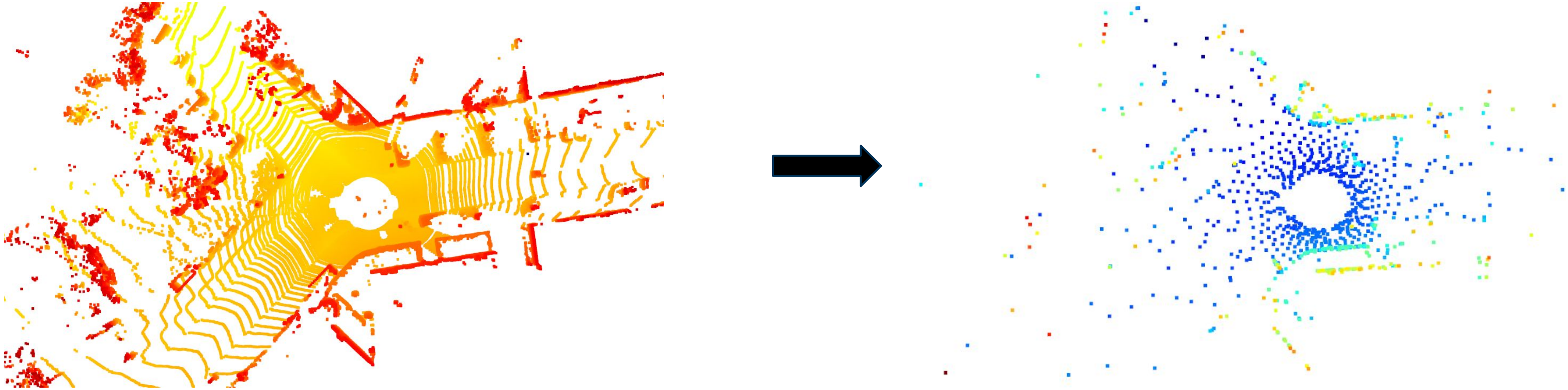




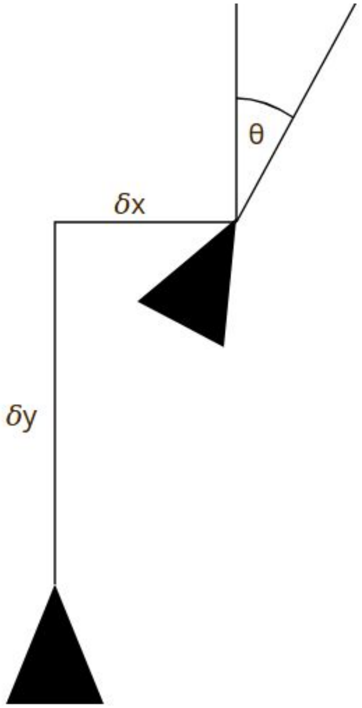


Training on KITTI dataset

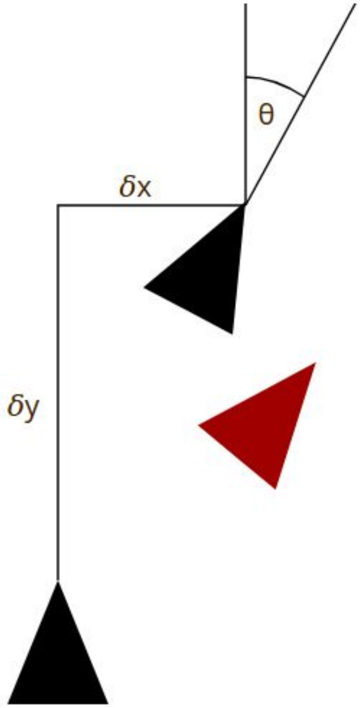
- Use sequences 00-07 for training
- Use sequences 08-10 for evaluation
- Compare PointNet, DGCNN, FlowNet3D as feature networks
- Point Clouds are randomly subsampled (1000 points) to simulate sparsity



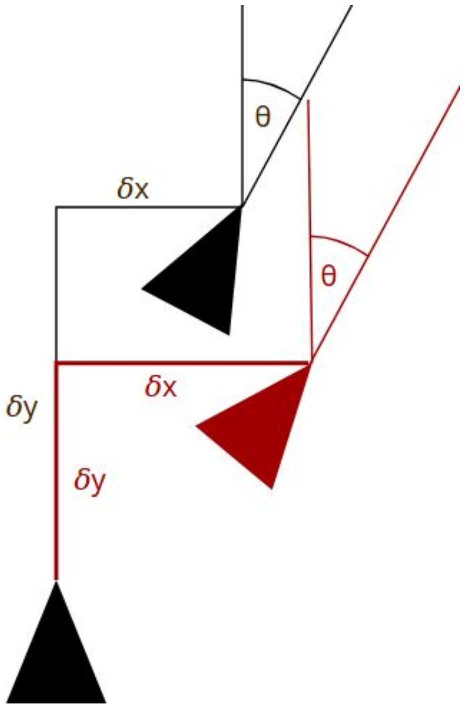
Results



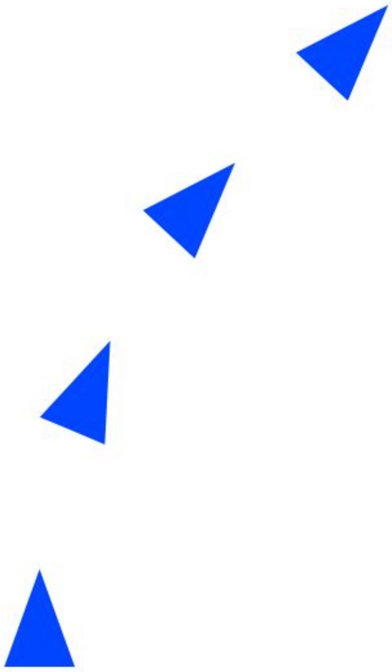
Results



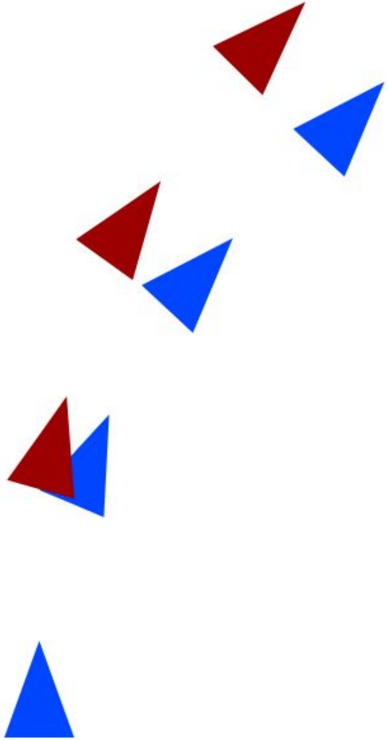
Results



Results



Results



Own Results

Table 1: Rotational and translational errors on sequence 8 of the KITTI odometry dataset

Model	Rotation error (in °)			Translational error (in m)		
	25%	Median	75%	25%	Median	75%
PointNet	0.1947	0.4008	**0.7159**	0.0791	0.1606	0.2733
DGCNN	**0.1580**	**0.3718**	0.7449	0.0775	0.1557	0.2684
FlowNet3D	0.1840	0.4033	0.7781	**0.0758**	**0.1526**	**0.2652**

Own Results

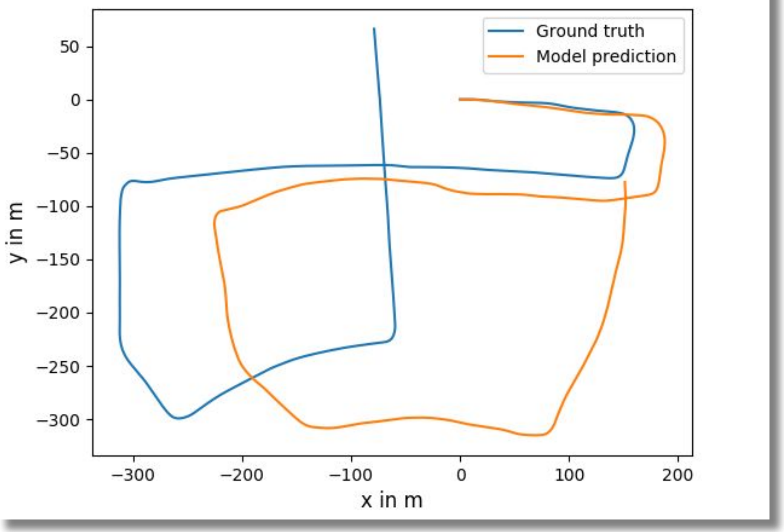
Table 1: Rotational and translational errors on sequence 8 of the KITTI odometry dataset

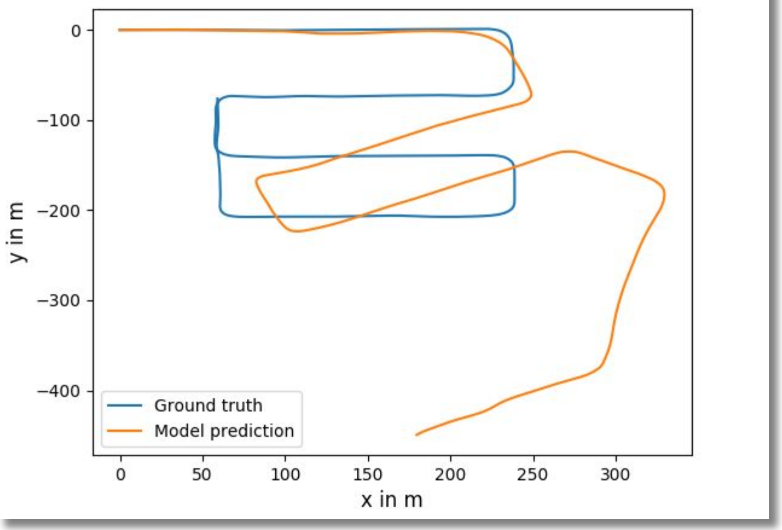
Model	Rotation error (in °)			Translational error (in m)		
	25%	Median	75%	25%	Median	75%
PointNet	0.1947	0.4008	**0.7159**	0.0791	0.1606	0.2733
DGCNN	**0.1580**	**0.3718**	0.7449	0.0775	0.1557	0.2684
FlowNet3D	0.1840	0.4033	0.7781	**0.0758**	**0.1526**	**0.2652**

Table 2: Rotational and translational errors on sequence 8 of the KITTI odometry dataset when trained with submaps

Model	Rotation error (in °)			Translational error (in m)		
	25%	Median	75%	25%	Median	75%
PointNet with submaps	0.1529	0.3394	0.6513	0.0694	0.1445	**0.2563**
PointNet with submaps & ICP	**0.1284**	**0.3130**	**0.6365**	**0.0652**	**0.1263**	0.2577

Evaluation on KITTI dataset





Training on Astyx/Cruise dataset

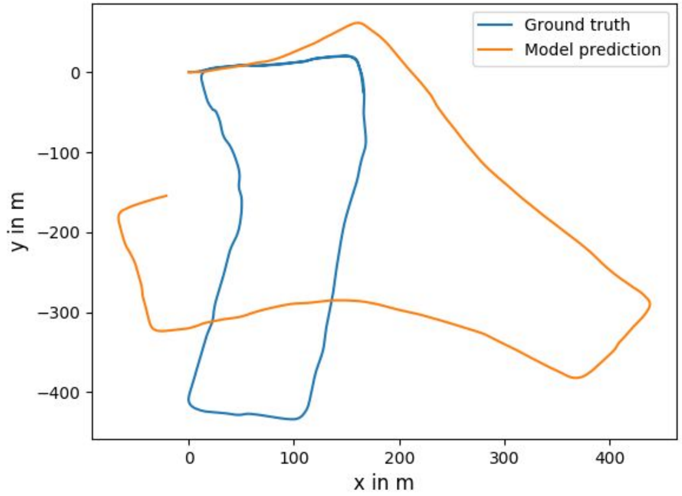
- Radar data includes radial velocity, magnitude
- Test, whether:
  - including this data has advantages
  - submaps bring improvement
  - using ICP is advantageous
- Evaluation has to be done on training data
  - Not enough data to split into training/evaluation

Own Results

Table 3: Rotational and translational errors on Astyx sequence B. p is the number of previous frames in the submap, f the number of features per point.

Model	p	f	Rotation error (in °)			Translational error (in m)		
			25%	Median	75%	25%	Median	75%
PointNet	0	3	0.1295	0.2612	0.4539	0.0501	0.1049	0.1815
PointNet	0	5	**0.1122**	**0.2296**	**0.3964**	**0.0379**	**0.0759**	**0.1332**
PointNet	5	3	0.1428	0.3083	0.5480	0.0617	0.1255	0.2109
PointNet	5	5	0.1377	0.3008	0.5411	0.0504	0.1009	0.1697

Evaluation on Astyx/Cruise dataset



Own Results

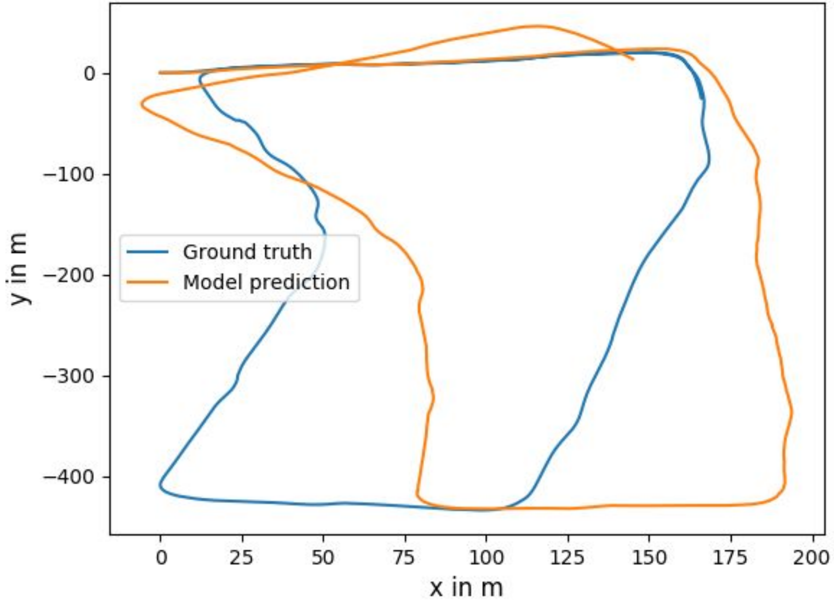
Table 3: Rotational and translational errors on Astyx sequence B. p is the number of previous frames in the submap, f the number of features per point.

Model	p	f	Rotation error (in °)			Translational error (in m)		
			25%	Median	75%	25%	Median	75%
PointNet	0	3	0.1295	0.2612	0.4539	0.0501	0.1049	0.1815
PointNet	0	5	**0.1122**	**0.2296**	**0.3964**	**0.0379**	**0.0759**	**0.1332**
PointNet	5	3	0.1428	0.3083	0.5480	0.0617	0.1255	0.2109
PointNet	5	5	0.1377	0.3008	0.5411	0.0504	0.1009	0.1697

Table 4: Rotational and translational errors on the astyx sequence B when refining the pose estimate with ICP

Model	p	f	Rotation error (in °)			Translational error (in m)		
			25%	Median	75%	25%	Median	75%
PointNet	0	5	0.0443	0.0919	0.1716	0.0208	0.0332	0.0493

Refinement with ICP



Thanks to LRZ for providing a GPU instance!

VCPUs	20
RAM	368GB
GPU	Nvidia V100 (16 GB Video memory)

Conclusion

- Pose estimation for SLAM is possible with Radar sensors.
- Point cloud registration even when the point clouds are very sparse and no further information from a GPS or IMU is available
- Unfortunately the frame drops of the astyx dataset prevent a proper result and evaluation of the methods

Thanks for your attention!

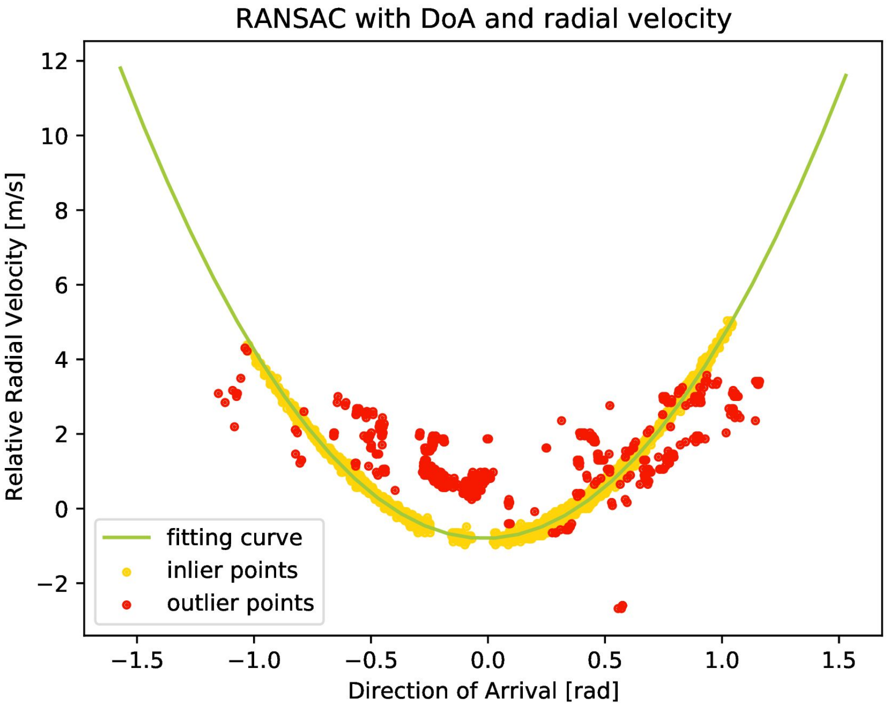
Backup

Classical methods

RANSAC steps:

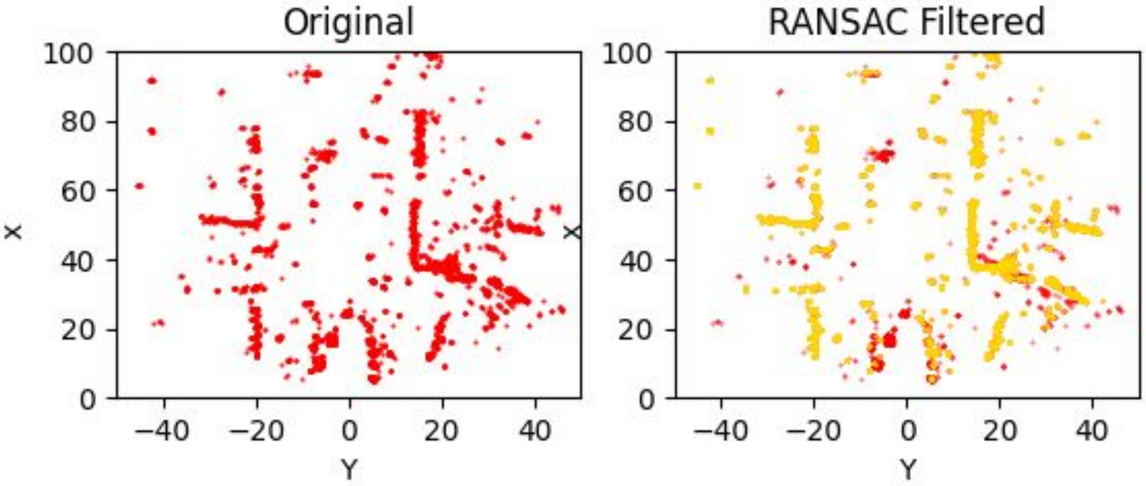
1. Randomly sampling points
2. Fit curve to sampled points
3. Evaluate curve on all points
4. If enough inliers
   a. Inlier points → Consensus set
   b. Optimizing fitting curve on complete Consensus set

Repeat process iteratively, then choose best found model. (here relation between V_r and DoA to considered static)

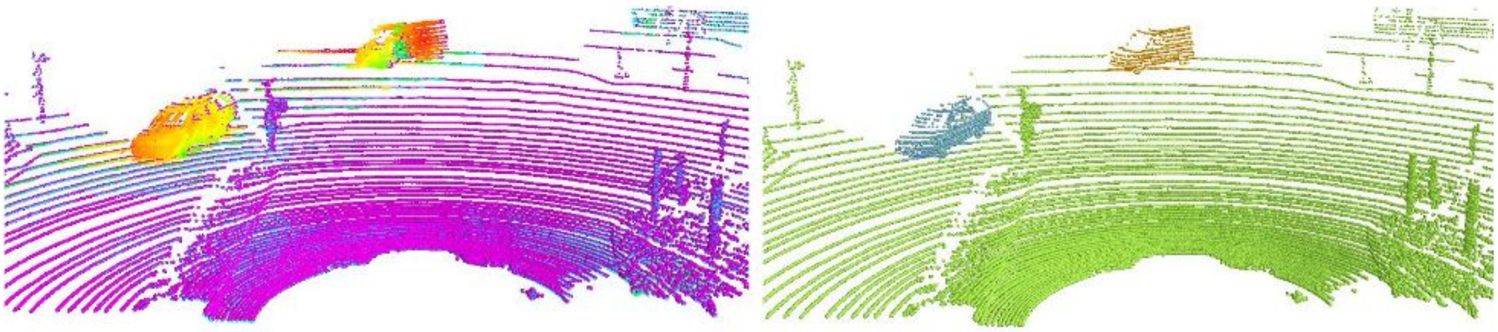


Martin A. Fischler and Robert C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: Commun. ACM 24.6 (June 1981)

Classical methods

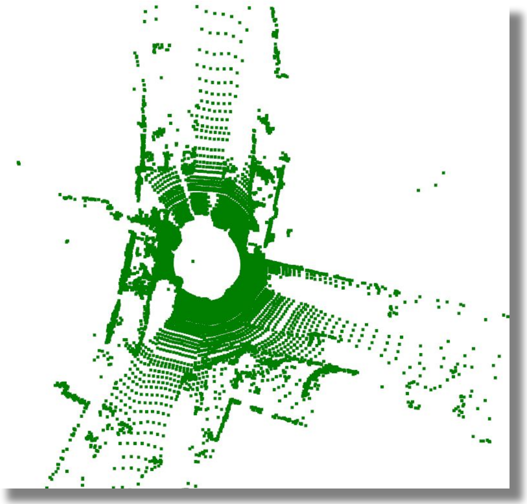


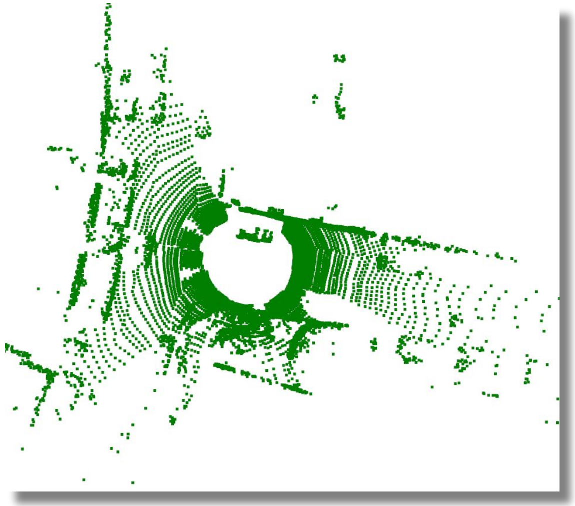
FlowNet3D



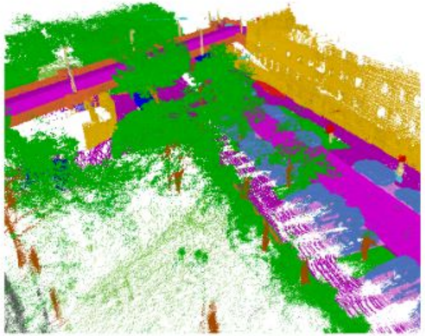
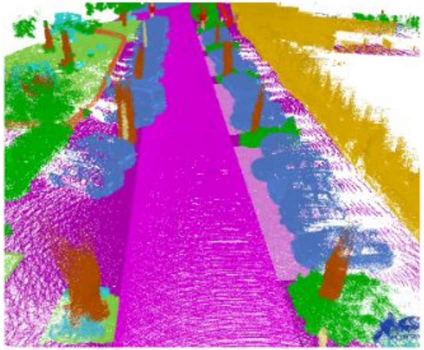
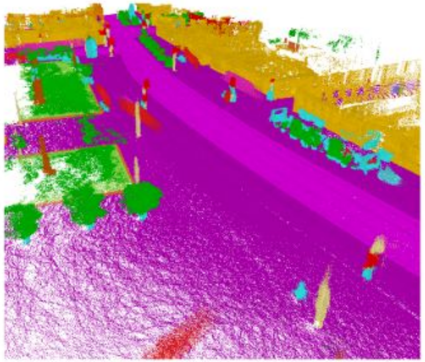
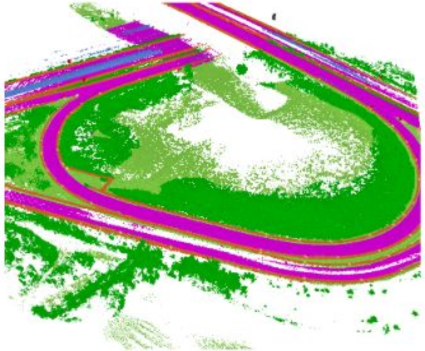
Xingyu Liu, Charles R Qi, and Leonidas J Guibas. “FlowNet3D: Learning Scene Flow in 3D Point Clouds”. In: CVPR (2019).

Static World assumption





Static World assumption

Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: Conference on Computer Vision and Pattern Recognition (CVPR). 2012.

Comparison KITTI - Astyx/Cruise

Table 2: Rotational and translational errors on sequence 8 of the KITTI odometry dataset when trained with submaps

Model	Rotation error (in °)			Translational error (in m)		
	25%	Median	75%	25%	Median	75%
PointNet with submaps	0.1529	0.3394	0.6513	0.0694	0.1445	**0.2563**
PointNet with submaps & ICP	**0.1284**	**0.3130**	**0.6365**	**0.0652**	**0.1263**	0.2577

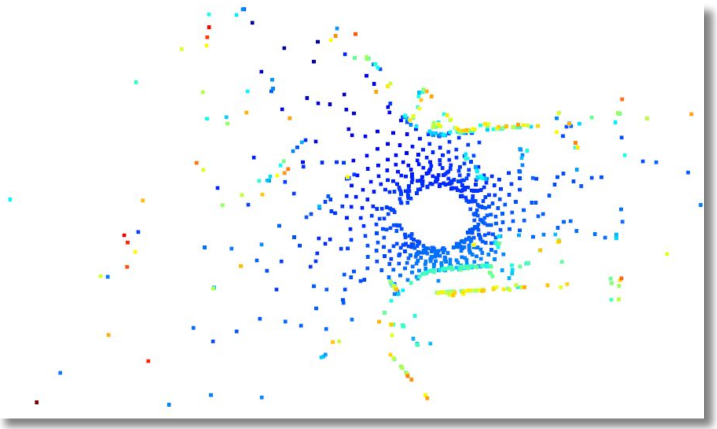
Table 4: Rotational and translational errors on the astyx sequence B when refining the pose estimate with ICP

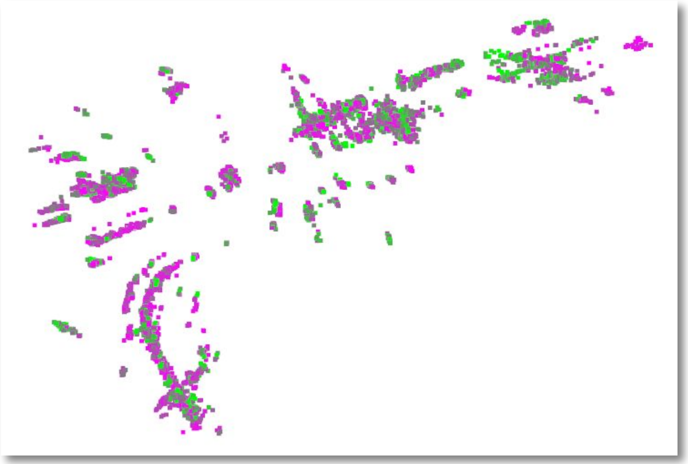
Model	p	f	Rotation error (in °)			Translational error (in m)		
			25%	Median	75%	25%	Median	75%
PointNet	0	5	0.0443	0.0919	0.1716	0.0208	0.0332	0.0493

Comparison KITTI - Astyx/Cruise

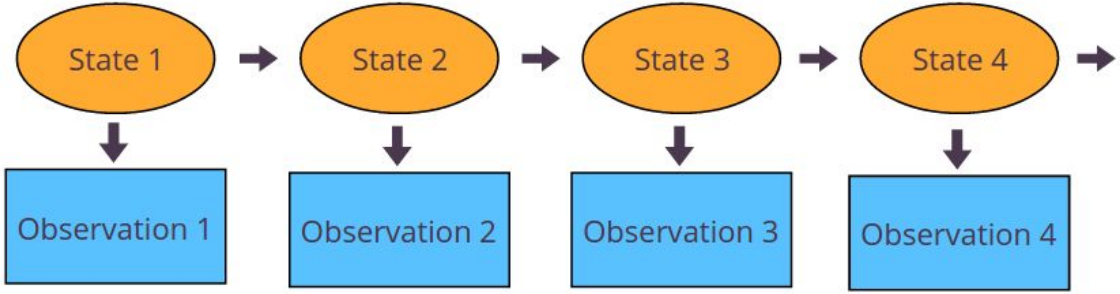
Reasons for better performance on Astyx/Cruise dataset:

- KITTI point clouds are randomly subsampled => no interesting features
- Astyx radar points are interesting features (selected by the sensor)
- Astyx data is evaluated on training dataset



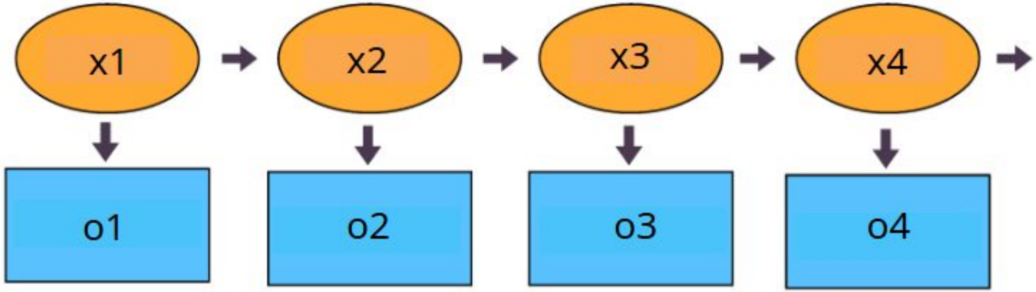


SLAM as MaP



State 1
State 2
State 3
State 4
Observation 1
Observation 2
Observation 3
Observation 4

SLAM as MaP



Filtering	$P(x_t \| o_1, o_2, ..., o_t)$
Smoothing	$P(x_k \| o_1, o_2, ..., o_t)$
MAP	$arg \max_{x_1, x_2, ..., x_t} P(x_1, x_2, ..., x_t \| o_1, o_2, ..., o_t)$

Division into Front-end / Back-end

- Front-end does Filtering
- Back-end fixes errors of Front-end to get the MAP estimate

Filtering	$P(x_t \mid o_1, o_2, ..., o_t)$
Smoothing	$P(x_k \mid o_1, o_2, ..., o_t)$
MAP	$arg \max_{x_1, x_2, .., x_t} P(x_1, x_2, ..., x_t \mid o_1, o_2, ..., o_t)$