Ground-Aware Automotive Radar Odometry

Daniel Casado Herraez* Franz Kaschner* Matthias Zeller Dominik Muhle
Jens Behley Michael Heidingsfeld Daniel Cremers Cyrill Stachniss

***Abstract*— Odometry is crucial for the navigation of autonomous vehicles in unknown environments. While cameras and LiDARs are commonly used to estimate the ego-motion of a vehicle, these sensors face limitations under bad lighting and severe weather conditions. Automotive radars overcome these challenges, but radar point clouds are generally sparse and noisy, making it difficult to identify useful features within a radar scan. In this paper, we address the problem of ego-motion estimation using a single automotive radar sensor. We propose a simple, yet effective, heuristic-based method to extract the ground plane from single radar scans and perform ground plane matching between consecutive scans. Additionally, we perform a windowed factor-graph optimization of the poses together with the ground plane, improving the accuracy of the pose estimation. We put our work to the test using the 4DRadarDataset. Our findings illustrate the state-of-the-art performance of our odometry approach compared to existing alternatives that use radar point clouds.**

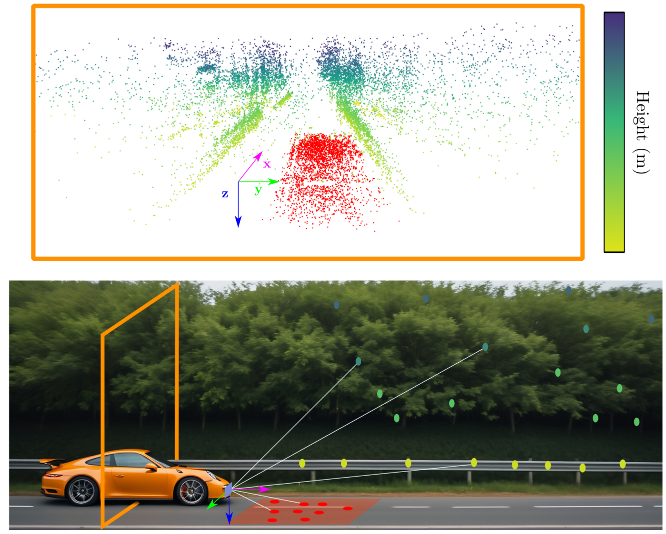


Fig. 1: Ground segmentation on sparse and noisy 3D radar scans employed in our radar odometry approach. The top image displays 50 aggregated scans for a clear visualization of the segmented ground plane. We use this information for odometry estimation of the vehicle. Below we show a graphic description of the environment captured by the radar, with the ground points in **red**.

I. Introduction

Odometry is a fundamental pillar of autonomous driving. It involves estimating the vehicle's ego-motion over time using onboard sensors. This is especially important in environments with challenging GNSS conditions, such as cities with tall skyscrapers, tunnels, and parking garages. Motion estimation is traditionally achieved using cameras or LiDARs, which are sensitive to lighting or can be affected by bad weather conditions, respectively. Furthermore, LiDARs are often challenging to fit in end-user vehicles and have a significantly higher price than cameras or radar sensors.

In this paper, we explore the task of automotive radar motion estimation, without relying on GNSS, wheel odometers, or other external sensors. Radars are compact, low-cost, and robust to adverse weather, making them increasingly common in consumer vehicles. Radars also provide Doppler velocity values and radar cross-section (RCS) information, with RCS being related to material properties and the angle of reflection. However, the sparse and noisy nature of radar scans presents significant challenges for radar odometry.

Early radar ego-motion estimation approaches rely on geometric relationships between the radar measurements and the vehicle [19]. Other approaches use positional information of the point measurements [1] [13] or a combination of both, velocity and positional information [7], leading to improvements of the estimation accuracy. To mitigate the sparsity of radar point clouds, some approaches try to transform radar scans to resemble LiDAR point clouds [22] [39]. Additionally, due to the sparse nature of radar scans, extracting scene features like planes and edges becomes a challenge, which is not the case with LiDAR [42]. Although some researchers group radar point measurements into clusters [45], little work has been done leveraging information from the ground plane, a feature commonly present in automotive radar measurements, see Fig. 1.

The main contribution of this paper is an effective radar odometry approach that exploits the ground plane within individual scans of a single automotive radar sensor, without relying on any additional sensors. We propose a simple, yet effective, heuristic-based ground segmentation method, used in our system to improve the vehicle's relative pose estimation. We achieve this by removing points that do not belong to a feasible ground region, leveraging the RCS information provided by the radar, extracting the ground plane from the scans, and matching it across consecutive frames in a point-to-ground-plane manner. We also use windowed optimization within a factor graph including a ground plane vertex, which leads to an improvement in our radar-only odometry results.

*Equal contribution. D. Casado Herraez and M. Zeller are with CARIAD SE and the University of Bonn, Germany. F. Kaschner is with AUDI AG, and the Technical University of Munich, Germany. D. Muhle and D. Cremers are with the Technical University of Munich, Germany. J. Behley is with the Center for Robotics, University of Bonn, Germany. M. Heidingsfeld is with CARIAD SE, Germany. C. Stachniss is with the Center for Robotics, University of Bonn and the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

In sum, we make three key claims: Our approach (i) achieves state-of-the-art performance in automotive radar odometry; (ii) exploits the RCS feature of automotive radars to estimate the ground plane from single radar scans; (iii) improves odometry accuracy by leveraging the ground plane through scan-to-map matching during scan registration, and across multiple scans via pose graph optimization.

II. Related Work

We provide an overview of current odometry methods and ground plane segmentation techniques. First, we discuss the latest strategies for camera and LiDAR odometry, highlighting their applicability and limitations in the radar domain. Then, we present odometry approaches based on automotive radars. Lastly, we summarize existing LiDAR ground plane segmentation and detection methods and analyze their shortcomings when applied to radar data.

Sensor-based odometry approaches estimate the relative pose of the vehicle over time using onboard sensors, often employing cameras or LiDARs. Visual information from cameras can be used with keypoint feature extraction and local bundle adjustment [6]. This keypoint-landmark concept was adopted by Huang et al. [17] and applied to radar sensors. In contrast, LiDARs generate a 3D point cloud containing depth information of each measured point [9] [31]. Numerous LiDAR odometry approaches [28] [42] [44] extract multiple surface plane and/or edge features from the point clouds. This is only feasible in sparse radar point clouds if the surface is large enough, like a flat ground plane. Other techniques project the LiDAR point cloud into a range image before performing feature extraction [35] [36]. LeGO-LOAM [36] additionally extracts the ground plane from the range image to perform matching over multiple frames. However, sparse radar point clouds would lead to a range image with very few occupied pixels. Other LiDAR techniques directly minimize the point-to-point error [38] to obtain an accurate pose estimate. In our approach, we employ the point-to-point matching strategy used in LiDAR systems [38] and combine it with a point-to-ground-plane error metric that includes the ground plane information present in the radar scans.

Radar-based odometry estimates the relative pose of the vehicle over time using onboard radars. These sensors are resilient to environmental conditions and can be divided into two categories: scanning / spinning radars, and automotive radars. Scanning radars output a 2D image representation of the environment, where each pixel indicates the intensity of the measurement. While some odometry techniques using these sensors extract keypoints from the images [2] [3] [25] [27], others perform signal processing directly on the image [32]. However, the dimensions and cost of spinning radars make them not well-suited for self-driving applications. Automotive radars are compact, inexpensive, and can be mounted behind the bumpers of consumer vehicles. They provide a point cloud with hundreds of points representing the scene, including the velocity and radar cross-section for each measurement. The earliest approaches to automotive radar odometry [12] [19] [21] exploit Doppler velocities and geometric constraints of the vehicle. Improvements in the field have resulted in scan matching strategies that also exploit positional information of the points augmenting it with their velocity [7] [40]. Furthermore, combinations of automotive radars with other sensors have also been investigated, including the usage of IMUs [4] [5] [16] [45] or cameras [10]. Others leverage semantically labeled scans to perform scan registration [18] or extract features like point clusters [45] for pose estimation.

The ground plane, however, is a feature consistently present in most radar scans when mounted on a vehicle and can be leveraged as a valuable source of information. Chen et al. [8], who combine radar with an IMU, exploit the velocity of the ground points to estimate the vehicle's ego-motion. In our radar-only approach, we propose a heuristic that leverages the radar's RCS information to extract the ground plane and match it across scans. We also incorporate the detected ground plane into a pose graph, matching it against a ground plane node. This leads to an improved accuracy with less trajectory drift over time compared to existing odometry methods.

Ground plane segmentation and detection identifies the points in a scan that belong to the ground and estimates the ground plane parameters, commonly using heuristics. Himmelsbach et al. [15] and Steinhauser et al. [37] fit lines to point sets within the scan, classifying points as ground or non-ground based on the properties of the line segments. More recent work [41] proposes using principal component analysis (PCA) [11] to estimate the final ground plane. However, they rely on the assumption that the lowest points within a scan belong to the ground, which is not true in radar point clouds due to the high amount of noise and multipath propagation. Based on the cylindrical geometry of LiDAR point clouds, other approaches divide the scans into multiple concentric regions that are segmented separately [23] [26]. Narski et al. [29] leverage the LiDAR ring properties for their segmentation method. Moreover, Koide et al. [20] present a RANSAC [11] fitting approach where they include a horizontal ground plane within a pose graph. Nevertheless, these methods have been developed for LiDAR sensors, relying on minimal noise, little multi-path propagation below the ground, and concentric point clouds from spinning laser devices. In our work, we adapt concepts from the LiDAR domain and observe how the ground plane region detected by radars is limited to a bounded small region in front of the sensor. Our heuristic method also exploits the RCS property from radar measurements to estimate the ground points.

In contrast to prior work, we propose an odometry approach that leverages automotive radars for ground segmentation. Our heuristic, yet effective, segmentation strategy extracts a ground plane that is matched across frames. We enhance odometry accuracy by including the ground plane information into a pose graph optimized using windowed optimization. This results in an accuracy improvement, achieving state-of-the-art results comparable to LiDAR systems.

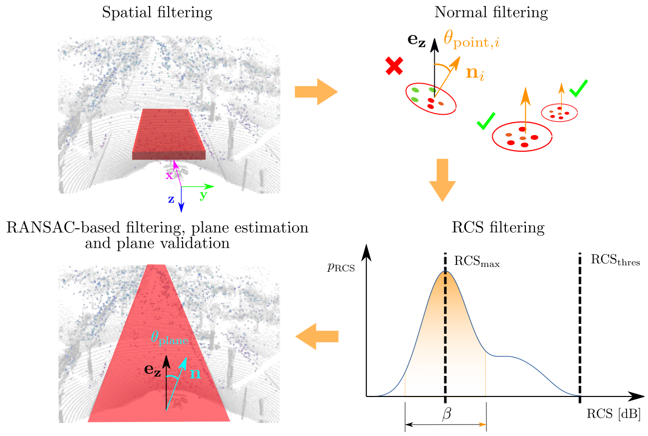


Fig. 2: Steps of our ground plane estimation approach. First, spatial filtering discards points outside of predefined physical boundaries. Then, a normal filtering step keeps individual points that form a horizontal plane with its neighbors, followed by an RCS filter that removes points with RCS values outside of an estimated bandwidth β. Finally, outliers not belonging to the ground plane are discarded and the ground plane is estimated and validated.

III. Our Approach to Radar Odometry

Our approach aims to achieve relative pose estimation of a vehicle by relying solely on automotive radar sensors. The process involves leveraging the radar properties to estimate the ground plane within a scan (Sec. III-A). We use the plane as a constraint during scan matching (Sec. III-B) and as an element in pose graph optimization (Sec. III-C).

A. Ground Plane Segmentation and Detection

The sparse nature of radar point clouds poses significant challenges to extracting meaningful features from radar scans. However, the combination of the mounting position and the radar field of view can return multiple points belonging to the ground plane. Although LiDAR ground plane extraction algorithms exist [26] [30], these methods often do not account for the characteristics of radar scans, which are noisier, sparser, and contain fewer ground points than LiDAR scans. We propose a simple, yet effective, strategy that exploits the properties of the measured ground points, segmenting them from the rest and extracting the parameters of the ground plane at the current time, as illustrated in Fig. 2. Our approach consists of the following five steps:

1) ***Spatial filtering*** is dependent on the sensor mounting height h_{sensor} and discards points that are outside a predefined region where the ground plane can be detected. Some LiDAR approaches [41] rely on the assumption that the points with minimum height within the scan most likely belong to the ground. Other approaches consider scans that provide a full 360-degree view around the sensor [26].

Radar scans, like those in the 4DRadarDataset [24], often contain reflections below the ground level due to multi-path propagation, with most points concentrated in front of the sensor. Thus, the lowest points cannot be used for ground plane estimation. To capture ground points, we propose to use a bounding box that extends within a range $[d_{\text{min},x}, d_{\text{max},x}]$ to the front in the x-direction, a symmetric range to the sides in the y-direction $[-d_y, d_y]$ and a height tolerance range in the vertical direction $[-d_z - h_{\text{sensor}}, d_z - h_{\text{sensor}}]$.

2) ***Normal estimation and filtering*** computes the normal vector of the remaining points. The goal is to remove points created by noise, clutter, or surrounding objects, and thus, whose normals are not approximately vertical. We estimate the normal vector $\mathbf{n}_i$ of each point by computing the covariance of all neighboring points within a radius and performing PCA, similar to Zermas et al. [41]. A point is retained when the angle $\theta_{\text{point},i}$ between the normal and the z-axis is below a predefined threshold $\theta_{\text{point,max}}$. Hence, we keep points where

$$\mathbf{n}_i^\top \mathbf{e}_z > \cos(\theta_{\text{point,max}}), \tag{1}$$

with $\mathbf{e}_z$ denoting the unit vector in vertical direction.

3) ***RCS filtering*** leverages the radar cross-section (RCS) property measured by the sensor. The RCS is a feature returned by automotive radars for each point that measures how detectable an object is by the sensor [14]. It depends on various properties of the target, including its material, the incidence and reflection angle, and the size of the target. During our experiments, we observed that the RCS of the ground points differs from the RCS of most other objects. Consequently, we can filter out potentially remaining non-ground points based on their RCS values. We keep the points where the RCS value $\mathbf{p}_{i,\text{RCS}}$ of a point $\mathbf{p}_i$ is below a threshold $\text{RCS}_{\text{thres}}$ such that

$$\mathbf{p}_{i,\text{RCS}} < \text{RCS}_{\text{thres}}. \tag{2}$$

To further refine this filtering process, we first remove all points outside a range centered on an estimated RCS_{max}. Manually tuning the parameter RCS_{max} would reduce robustness against RCS variations between scans. Second, using histograms for estimating the maximum RCS would require manually defining the bin size and the number of bins. Instead, we take a third approach and employ Gaussian kernel density estimation to compute the continuous probability density function of the RCS values following

$$\mathbf{p}_{\text{RCS}}(x) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mathbf{p}_{i,\text{RCS}})^2}{2\sigma^2}\right). \tag{3}$$

Eq. (3) adds N Gaussians, each one centered at $\mathbf{p}_{i,\text{RCS}}$, to obtain a continuous distribution of the RCS values. We use Scott's rule [34] to estimate a suitable standard deviation $\sigma = N^{-\frac{1}{5}}$. We then determine the maximum of the probability density function $\text{RCS}_{\text{max}} = \max(\mathbf{p}_{\text{RCS}}(x))$ and remove all points outside of a manually set bandwidth β.

4) ***RANSAC-based filtering and plane estimation*** are the final steps to identify a plane within a radar scan. We first use RANSAC [11] to fit a plane and discard those points that are over a certain distance γ from the estimation, resulting in a subset $\mathcal{P}_G \subset \mathcal{P}$ to which we refer to as ground points. However, in contrast to other approaches [20] [24], we do not use the parameters estimated with RANSAC to determine the ground plane, as we found that it is less suitable for radar data. Because of the low number of ground points covering only a small area, RANSAC tends to estimate the tilt of the ground plane incorrectly if the threshold γ is not tuned perfectly. Instead, we use PCA on the remaining point inliers $\mathcal{P}_G$ to estimate the normal vector $\mathbf{n}_t$ of the ground plane at the current timestep, defined as $\boldsymbol{\pi}_t = (\mathbf{n}_t, d_t)$. The distance of the plane to the origin d_t is given by

$$d_t = -\mathbf{n}_t^\top \overline{\mathbf{p}}, \tag{4}$$

where $\overline{\mathbf{p}}$ is the centroid of $\mathcal{P}_G$.

5) ***Ground plane validation*** verifies that the estimated ground plane normal remains within a range with respect to the vertical axis. Due to the vehicle's pitch and roll angles and the road's slope, small deviations from the vertical direction are possible. We define the ground plane as valid if the angle θ_{plane} between its normal $\mathbf{n}_t$ and the unit vector in z-direction $\mathbf{e}_z$ is below a threshold $\theta_{\text{plane,max}}$, i.e.,

$$\mathbf{n}_t^\top \mathbf{e}_z > \cos(\theta_{\text{plane,max}}). \tag{5}$$

When this condition is not met, we determine that the ground plane cannot be successfully estimated and exclude it from scan registration. This may occur due to a low number of measurements or steep slopes. Note that while in the normal filtering step described above, the verticality check was applied to each point, here the final ground plane parameters are being checked.

B. Ground-ICP: Exploiting the Ground Plane

Rather than relying on the flat world assumption during scan matching, we leverage the extracted ground plane in ICP optimization. Our ground-based scan-matching module combines a point-to-point error metric with a point-to-ground error to find the transformation that aligns the current radar scan to the previous radar scans in the local map.

For point-to-point ICP, we adopt the position error residual from our prior work Radar-ICP [7]. The goal is to obtain the transformation $\mathsf{T} \in SE(3)$ between the current scan $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_M\}$ and a local map $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_N\}$. We do this iteratively by extracting the closest point correspondences $\mathcal{M} = \{(\mathbf{p}, \mathbf{q})_1, (\mathbf{p}, \mathbf{q})_2, ..., (\mathbf{p}, \mathbf{q})_N\}$ from the transformed current scan to the local map and estimate T by minimizing the point error function

$$E_{\text{P2P}}(\mathsf{T}) = \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{M}} \rho(||\mathbf{q} - \mathsf{T}\mathbf{p}||), \tag{6}$$

with the Geman McClure kernel ρ to potentially reduce the effect of outliers [38].

Our point-to-ground ICP error function E_{P2G} differs from traditional methods that establish point correspondences [33]. Instead, it minimizes the distance between a ground point $\mathbf{p}_G \in \mathcal{P}_G$ from the current scan, and the ground plane of the previous scan $\boldsymbol{\pi}_{t-1} = (\mathbf{n}_{t-1}, d_{t-1})$ following

$$E_{\text{P2G}}(\mathsf{T}) = \sum_{\mathbf{p}_G \in \mathcal{P}_G} \rho(||\mathbf{n}_{t-1}^\top(\mathsf{T}\mathbf{p}_G) + d_{t-1}||). \tag{7}$$

The final Ground-ICP objective function is defined as

$$E(\mathsf{T}) = (1 - \alpha) \cdot E_{\text{P2P}}(\mathsf{T}) + \alpha \cdot E_{\text{P2G}}(\mathsf{T}), \tag{8}$$

Spatial filtering	sensor height	h_{sensor}	0.663 m
	min. x-value	$d_{\text{min},x}$	0.5 m
	max. x-value	$d_{\text{max},x}$	9.0 m
	y-range	d_y	2.0 m
	z-range	d_z	0.2 m
Normal filtering	point vertical threshold	$\theta_{\text{point,max}}$	15.0°
RCS filtering	threshold	$\text{RCS}_{\text{thres}}$	-35.0 dB
	bandwidth	β	20.0 dB
Outlier removal	RANSAC threshold	γ	0.02 m
Ground plane validation	plane vertical threshold	$\theta_{\text{plane,max}}$	1.0°
Graph optimization	sliding window size	n	20
Ground-ICP	weighting factor	α	0.993

TABLE I: Hyperparameters of our ground segmentation and odometry method for the 4DRadarDataset [24].

and combines the point-to-point with the point-to-ground residual using a hand-tuned weighting parameter $\alpha \in [0, 1]$.

C. Ground Plane Pose Graph Optimization

The previous section explained how Ground-ICP reduces the error between ground points $\mathcal{P}_G$ in the current scan and the ground plane $\boldsymbol{\pi}_{t-1}$ in the previous scan. We now add an additional constraint to our framework that considers a global ground plane over an entire sequence, similar to the LiDAR approach by Koide et al. [20]. This turns out to be more reliable than having no ego-motion estimation in the vertical direction.

To achieve this, we create a pose graph that links the pose estimations using odometry edges and connect the estimated ground plane at each pose with a global horizontal ground plane node. We assume that the trajectory has no significant variations in slope and that the overall terrain is mostly flat. This assumption is reasonable for many man-made environments, such as parking lots. Consequently, the normal of the global ground plane coincides with the vertical z-direction $\mathbf{e}_z$ and the distance to the origin is the sensor mounting height h_{sensor}, such that

$$\boldsymbol{\pi}_G = (\mathbf{n}_G, d_G) = (\mathbf{e}_z, h_{\text{sensor}}). \tag{9}$$

The odometry edge residual of the pose graph, given between the current and previous pose $\mathsf{P}_t, \mathsf{P}_{t-1} \in \text{SE}(3)$, respectively, is defined as

$$\mathbf{e}_{\text{odom}}(\mathsf{P}_{t-1}, \mathsf{P}_t, \mathsf{T}) = \text{Log}(\mathsf{P}_t) - \text{Log}(\mathsf{T}\mathsf{P}_{t-1}), \tag{10}$$

with the logarithmic mapping Log: $\text{SE}(3) \rightarrow \mathbb{R}^6$.

For plane-to-plane edges, the plane-to-plane residual between the global ground plane $\boldsymbol{\pi}_G$ and the local ground plane observation $\boldsymbol{\pi}_t$ is given by

$$\mathbf{e}_{\text{ground}}(\boldsymbol{\pi}_G, \boldsymbol{\pi}_t) = \boldsymbol{\tau}(\boldsymbol{\pi}_G) - \boldsymbol{\tau}(\boldsymbol{\pi}_t), \tag{11}$$

where $\boldsymbol{\tau}(\boldsymbol{\pi}) = \left(\arctan\left(\frac{\mathbf{n}_y}{\mathbf{n}_x}\right), \arctan\left(\frac{\mathbf{n}_z}{\|\mathbf{n}_x\|_2}\right), d\right)^\top$.

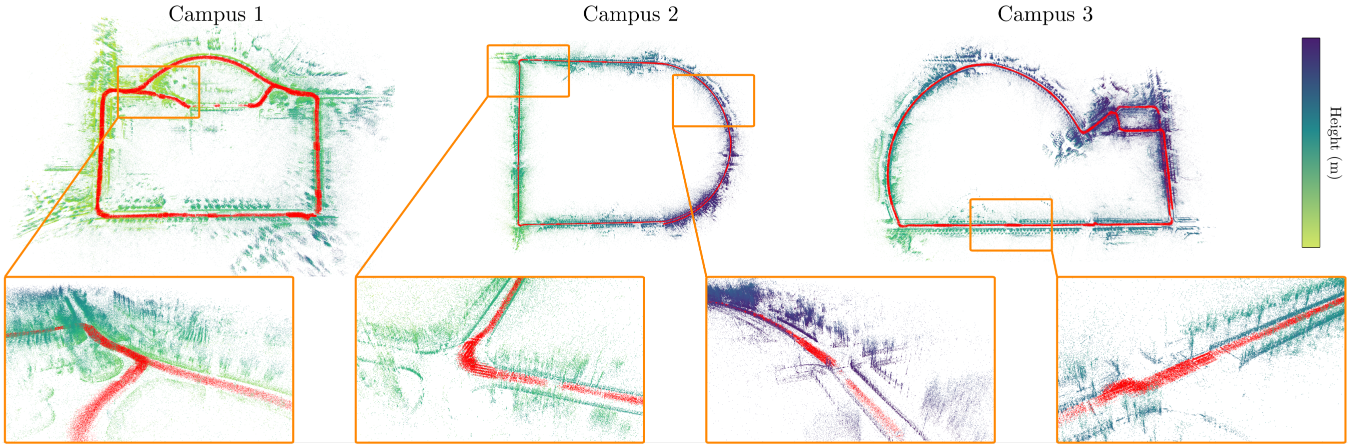


Fig. 3: Ground segmentation (shown in **red**) results for "Campus 1", "Campus 2" and "Campus 3" of the 4DRadarDataset [24]. Points below the ground plane have been removed for better visualization.

	Campus 1			Campus 2			Campus 3			Campus 4		
	RE_{trans} [%]	RE_{rot} [°/m]	ATE [m]	RE_{trans} [%]	RE_{rot} [°/m]	ATE [m]	RE_{trans} [%]	RE_{rot} [°/m]	ATE [m]	RE_{trans} [%]	RE_{rot} [ ]	ATE [m]
LeGO-LOAM (LiDAR) [36]	1.96	0.020	2.27	3.19	0.012	2.57	2.68	0.010	3.33	-	-	-
4DRadarSLAM [43]	16.0	0.103	60.5	39.1	0.115	351.5	36.0	0.096	205.7	42.0	0.127	949.3
4DRaSLAM [24]	2.32	0.021	2.28	3.13	0.020	**3.79**	3.06	0.024	3.83	-	-	-
4DRaSLAM (Odom.) [24]	2.52	0.025	4.14	3.84	0.019	10.2	3.29	0.027	9.08	-	-	-
Radar-ICP [7]	1.56	**0.013**	6.50	2.80	0.012	40.0	2.86	0.017	16.9	8.91	0.041	118.3
KISS-ICP (Radar) [38]	1.55	0.014	3.71	2.00	0.010	11.6	2.06	**0.013**	14.7	6.95	**0.039**	71.6
Ours	**1.52**	0.014	**1.78**	**1.92**	**0.009**	7.29	**1.91**	**0.013**	**1.69**	**6.86**	**0.039**	**13.4**

TABLE II: Comparison of our system to state-of-the-art radar and LiDAR odometry and SLAM approaches on the 4DRadarDataset [24].

We use windowed optimization considering the last n scans to optimize the pose graph with a cost function:

$$E(\boldsymbol{\mathcal{X}}) = \sum_{k=t-n+1}^{t-1} \|\mathbf{e}_{\text{odom}}(\mathsf{P}_{k-1}, \mathsf{P}_k, \mathsf{T})\|^2_{\boldsymbol{\Omega}_{\text{odom}}} + \sum_{k=t-n+1}^{t} \|\mathbf{e}_{\text{ground}}(\boldsymbol{\pi}_G, \boldsymbol{\pi}_k)\|^2_{\boldsymbol{\Omega}_{\text{ground}}}, \tag{12}$$

where $\boldsymbol{\mathcal{X}}$ is the set of all vertices, and $\boldsymbol{\Omega}_{\text{odom}}$ and $\boldsymbol{\Omega}_{\text{ground}}$ refer to the odometry and ground plane estimation information matrices. In the cases when the estimated ground plane at time t is invalid, the ground plane edge is not considered and only the odometry edge is added to the pose graph.

IV. Experimental Evaluation

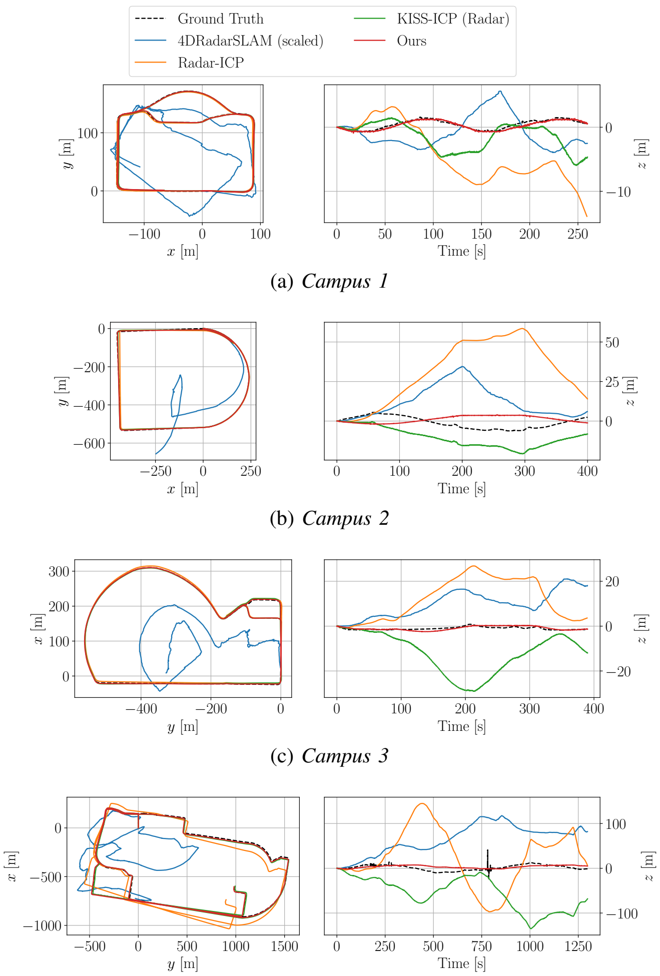
The main focus of this work is an automotive radar odometry method that exploits the ground plane information as an additional feature for pose estimation. We present our experiments to show the capabilities of our method. The results support our key claims that our approach (i) achieves state-of-the-art performance in automotive radar odometry; (ii) exploits the RCS feature of automotive radars to estimate the ground plane from single radar scans; (iii) improves odometry accuracy by leveraging the ground plane through scan-to-map matching during scan registration, and across multiple scans via pose graph optimization.

A. Implementation Details and Experimental Setup

For the evaluation of our approach, we run experiments on the publicly available 4DRadarDataset [24]. Its radar sensor is mounted on the front bumper of the car, where part of the output points belong to the ground. In our evaluation, rather than measuring the ground segmentation accuracy, we focus on the relative errors from the odometry task. Our approach for ground segmentation has multiple hyperparameters that also depend on the vehicle and sensor setup. The chosen parameters for the 4DRadarDataset [24] are listed in Tab. I. Note that a high value of α helps to compensate for the small amount of ground points in a single scan. In our experiments, we first compare our approach against the state-of-the-art radar and LiDAR odometry, defining as our ground truth the differential GNSS measurements provided by the dataset. We also show qualitative results of the estimated trajectories and the segmented ground planes within each sequence. Our second experiment performs an ablation study of our system, demonstrating how each component contributes to the final odometry estimation result and to the runtime.

B. Comparison with the State-of-the-Art

The first experiment evaluates the performance of our method and demonstrates that it achieves state-of-the-art results in automotive radar odometry comparable to LiDAR approaches. We compare our method against the LiDAR feature matching odometry method LeGO-LOAM [36] applied to LiDAR scans, the point-to-point matching technique KISS-ICP [38] on radar point clouds, two radar SLAM frameworks 4DRadarSLAM by Zhang et al. [43] and 4DRaSLAM by Li et al. [24] with and without loop closure, and the radar odometry approach Radar-ICP [7]. For LeGO-LOAM [36] and 4DRaSLAM [24] we get the results directly from Li et al.'s work [24]. We measure the relative errors using the relative translation ($\text{RTE}_{\text{trans}}$)



(a) *Campus 1*

(b) *Campus 2*

(c) *Campus 3*

(d) *Campus 4*

Fig. 4: Comparison of the trajectories in the x-y plane (left) and the z-movement over time (right) from the 4DRadarDataset [24]. In the right-hand plots, the trajectory of 4DRadarSLAM is scaled by 0.1 to improve visualization of the other methods along the z-axis.

and rotation (RTE_{rot}) KITTI metric. We also include the absolute trajectory error (ATE) over the entire sequence. The experimental evaluation is shown in Tab. II.

Our method improves the translation accuracy for all sequences and achieves comparable performance for relative rotation and absolute trajectory error. Notably, as shown in Fig. 4 we observe a big improvement in vertical drift thanks to leveraging the additional information obtained from the ground plane. In our experiments, conventional point-to-point ICP approaches employed in our method [7] [38] yield better results than NDT [24] and adaptive generalized ICP [43]. Our approach outperforms the other methods in terms of ATE for "Campus 1", "Campus 3", and "Campus 4". Likewise, our method shows comparable performance in most scenarios to using a 3D LiDAR sensor. We also show qualitative results of the ground plane segmentation in Fig. 3.

C. Ablation Studies

The second experiment evaluates how each component of our system contributes to the final accuracy. We perform the evaluation on the sequences "Campus 1", "Campus 2" and "Campus 3", as "Campus 4" contains some jumps in the ground truth data. The results are presented in Tab. III. We additionally measure the impact on the runtime of each system component in Tab. IV.

	Campus 1		Campus 2		Campus 3	
	$\mathbf{RE_{trans}}$ [%]	**ATE** [m]	$\mathbf{RE_{trans}}$ [%]	**ATE** [m]	$\mathbf{RE_{trans}}$ [%]	**ATE** [m]
P2P-ICP	1.55	3.71	2.00	11.6	2.06	14.7
P2P-ICP with $z = 0$	1.81	1.97	2.27	7.83	3.32	3.11
Ground-ICP	**1.52**	3.09	2.11	8.70	2.07	8.22
P2P-ICP + Graph	**1.52**	**1.75**	1.95	7.62	1.92	3.31
Ground-ICP + Graph	**1.52**	1.78	**1.92**	**7.29**	**1.91**	**1.69**

TABLE III: Comparison of the influence of different components of our ground-based ICP using the 4DRadarDataset [24].

	P2P-ICP	Ground-ICP	Ground-ICP + Graph
Scan Matching	17.1 ms	16.5 ms	16.5 ms
Ground Plane Estimation	–	1.4 ms	1.4 ms
Pose Graph Construction & Optimization	–	–	0.7 ms
Total	17.1 ms	17.9 ms	18.6 ms

TABLE IV: Comparison of the runtimes per frame of our radar odometry system in different configurations. The measurements are averaged over all frames of the sequence "Campus 1" of the 4DRadarDataset [24]. The pose graph construction and optimization are only performed every 10 frames, so the corresponding runtimes in the table are averaged over these intervals.

The baseline method, point-to-point ICP (P2P-ICP), is based on KISS-ICP [38] applied to radar data. We also provide evaluation results when the poses are restricted to the horizontal plane (P2P-ICP with z = 0). When we add ground plane detection and optimization to ICP, we refer to this as "Ground-ICP". If we include the ground plane as a node in the pose graph for P2P-ICP, this method is labeled as "P2P-ICP + Graph". Combining the "Ground-ICP" with the ground plane as an additional pose graph node is termed "Ground-ICP + Graph".

We observe that removing the z-component from the poses leads to the worst performance, as it fails to account for minor movements in the vertical direction in areas where the ground is not perfectly planar. In contrast, the combination "Ground-ICP + Graph" achieves the highest average accuracy showing how the different components help to improve the final result. Moreover, our runtime analysis shows that these enhancements come with a minimal increase in the runtime of the system.

V. Conclusion

In this paper, we presented a novel approach to estimate the ego-motion of a vehicle using only a single automotive radar sensor. We proposed a method that leverages the sensor properties to segment ground points and use it for radar scan matching and pose graph optimization. We implemented and evaluated our approach on a real-world dataset and provided comparisons to other existing techniques, supporting all claims made in this paper. The experiments suggest that our ground-aware odometry approach enhances pose estimation performance, achieving LiDAR-level accuracy using a single automotive radar. In future work, we aim to evaluate our method in more diverse datasets with uneven terrains to assess its limitations in such scenarios.

REFERENCES

[1] R. Aldera, M. Gadd, D. De Martini, and P. Newman. What goes around: Leveraging a constant-curvature motion constraint in radar odometry. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7865–7872, 2022.

[2] D. Barnes and I. Posner. Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[3] D. Barnes, R. Weston, and I. Posner. Masking by moving: Learning distraction-free radar odometry from pose information. In *Proc. of the Conf. on Robot Learning (CoRL)*, 2019.

[4] T. Brühl, T.D. Eberhardt, R. Schwager, L. Ewecker, T.S. Sohn, and S. Hohmann. Odometry estimation by fusing multiple radar sensors and an inertial measurement unit. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[5] K. Burnett, A.P. Schoellig, and T.D. Barfoot. Continuous-time radar-inertial and lidar-inertial odometry using a gaussian process motion prior. *arXiv preprint arXiv:2402.06174*, 2024.

[6] C. Campos, R. Elvira, J.J.G. Rodríguez, J.M. Montiel, and J.D. Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. on Robotics (TRO)*, 37(6):1874–1890, 2021.

[7] D. Casado Herraez, M. Zeller, L. Chang, I. Vizzo, M. Heidingsfeld, and C. Stachniss. Radar-only odometry and mapping for autonomous vehicles. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[8] H. Chen, Y. Liu, and Y. Cheng. Drio: Robust radar-inertial odometry in dynamic environments. *IEEE Robotics and Automation Letters (RA-L)*, 8(9):5918 – 5925, 2023.

[9] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss. SuMa++: Efficient LiDAR-based Semantic SLAM. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[10] Y. Cheng, M. Jiang, and Y. Liu. Ms-vro: A multi-stage visual-millimeter wave radar fusion odometry. *IEEE Trans. on Robotics (TRO)*, 40:3004–3023, 2024.

[11] M. Fischler and R. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[12] A. Galeote-Luque, V. Kubelka, M. Magnusson, J.R. Ruiz-Sarmiento, and J. Gonzalez-Jimenez. Doppler-only single-scan 3d vehicle odometry. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[13] K. Haggag, S. Lange, T. Pfeifer, and P. Protzel. A credible and robust approach to ego-motion estimation using an automotive radar. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):6020–6027, 2022.

[14] D.C. Herraez, L. Chang, M. Zeller, L. Wiesmann, J. Behley, M. Heidingsfeld, and C. Stachniss. Spr: Single-scan radar place recognition. *IEEE Robotics and Automation Letters (RA-L)*, 2024.

[15] M. Himmelsbach, F.V. Hundelshausen, and H.J. Wuensche. Fast segmentation of 3d point clouds for ground vehicles. In *Proc. of the IEEE Vehicles Symposium (IV)*, 2010.

[16] J.T. Huang, R. Xu, A. Hinduja, and M. Kaess. Multi-radar inertial odometry for 3d state estimation using mmwave imaging radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[17] Q. Huang, Y. Liang, Z. Qiao, S. Shen, and H. Yin. Less is more: Physical-enhanced radar-inertial odometry. *arXiv preprint arXiv:2402.02200*, 2024.

[18] S.T. Isele, F. Haas-Fickinger, and J.M. Zöllner. Seraloc: Slam on semantically annotated radar point-clouds. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2021.

[19] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer. Instantaneous ego-motion estimation using doppler radar. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2013.

[20] K. Koide, J. Miura, and E. Menegatti. A Portable Three-dimensional LIDAR-based System for Long-term and Wide-area People Behavior Measurement. *Intl. Journal of Advanced Robotic Systems*, 16(2), 2019.

[21] V. Kubelka, E. Fritz, and M. Magnusson. Do we need scan-matching in radar odometry? In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[22] P.C. Kung, C.C. Wang, and W.C. Lin. Radar occupancy prediction with lidar supervision while preserving long-range sensing and penetrating capabilities. *IEEE Robotics and Automation Letters (RA-L)*, 7(2):2637–2643, 2022.

[23] S. Lee, H. Lim, and H. Myung. Patchwork++: Fast and robust ground segmentation solving partial under-segmentation using 3d point cloud. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[24] X. Li, H. Zhang, and W. Chen. 4d radar-based pose graph slam with ego-velocity pre-integration factor. *IEEE Robotics and Automation Letters (RA-L)*, 8(8):5124–5131, 2023.

[25] H. Lim, K. Han, G. Shin, G. Kim, S. Hong, and H. Myung. Orora: Outlier-robust radar odometry. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[26] H. Lim, M. Oh, and H. Myung. Patchwork: Concentric zone-based region-wise ground segmentation with ground likelihood estimation using a 3d lidar sensor. *IEEE Robotics and Automation Letters (RA-L)*, 6(4):6458–6465, 2021.

[27] D. Lisus, K. Burnett, D.J. Yoon, and T.D. Barfoot. Are doppler velocity measurements useful for spinning radar odometry? *arXiv preprint arXiv:2404.01537*, 2024.

[28] Z. Liu and F. Zhang. Balm: Bundle adjustment for lidar mapping. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):3184–3191, 2021.

[29] P. Narksri, E. Takeuchi, Y. Ninomiya, Y. Morales, N. Akai, and N. Kawaguchi. A slope-robust cascaded ground segmentation in 3d point cloud for autonomous vehicles. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2018.

[30] Y. Pan, P. Xiao, Y. He, Z. Shao, and Z. Li. MULLS: Versatile LiDAR SLAM Via Multi-Metric Linear Least Square. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[31] Y. Pan, X. Zhong, L. Wiesmann, T. Posewsky, J. Behley, and C. Stachniss. PIN-SLAM: LiDAR SLAM Using a Point-Based Implicit Neural Representation for Achieving Global Map Consistency. *IEEE Transactions on Robotics (TRO)*, 2024.

[32] Y.S. Park, Y.S. Shin, and A. Kim. Pharao: Direct radar odometry using phase correlation. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[33] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Proc. of Intl. Conf. on 3-D Digital Imaging and Modeling*, 2001.

[34] D.W. Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[35] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[36] T. Shan and B. Englot. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[37] D. Steinhauser, O. Ruepp, and D. Burschka. Motion segmentation and scene classification from 3d lidar data. In *Proc. of the IEEE Vehicles Symposium (IV)*, 2008.

[38] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss. KISS-ICP: In Defense of Point-to-Point ICP – Simple, Accurate, and Robust Registration If Done the Right Way. *IEEE Robotics and Automation Letters (RA-L)*, 8(2):1029–1036, 2023.

[39] R. Weston, S. Cen, P. Newman, and I. Posner. Probably unknown: Deep inverse sensor modelling radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[40] X. Wu, Y. Chen, Z. Li, Z. Hong, and L. Hu. Efear-4d: Ego-velocity filtering for efficient and accurate 4d radar odometry. *arXiv preprint arXiv:2405.09780*, 2024.

[41] D. Zermas, I. Izzat, and N. Papanikolopoulos. Fast Segmentation of 3D Point Clouds: A Paradigm on Lidar Data for Autonomous Vehicle Applications. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.

[42] J. Zhang and S. Singh. LOAM: Lidar Odometry and Mapping in Real-time. In *Proc. of Robotics: Science and Systems (RSS)*, 2014.

[43] J. Zhang, H. Zhuge, Z. Wu, G. Peng, M. Wen, Y. Liu, and D. Wang. 4dradarslam: A 4d imaging radar slam system for large-scale environments based on pose graph optimization. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[44] L. Zhou, S. Wang, and M. Kaess. π-lsam: Lidar smoothing and mapping with planes. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[45] Y. Zhuang, B. Wang, J. Huai, and M. Li. 4d iriom: 4d imaging radar inertial odometry and mapping. *IEEE Robotics and Automation Letters (RA-L)*, 8(6):3246–3253, 2023.