

Dr-PoGO: Direct Radar Pose-Graph Optimization

Cedric Le Gentil¹, Weican Li¹, Leonardo Brizi², Timothy D. Barfoot¹

Abstract—This paper introduces Dr-PoGO, a method for Simultaneous Localization And Mapping (SLAM) using a 2D spinning radar. Unlike cameras or lidars that require line-of-sight, millimetre-wave radars can ‘see’ through dust, falling snow, rain, etc. Accordingly, it is a great modality for robust perception regardless of the weather conditions. While most existing radar-based SLAM methods rely on the extraction of point clouds or features to perform ego-motion estimation, Dr-PoGO leverages direct registration techniques for odometry (DRO) and loop-closure registration. An off-the-shelf radar-focused place recognition algorithm, RaPlace, provides loop-closure candidates. As RaPlace does not provide relative transformations, Dr-PoGO introduces a coarse-to-fine registration that uses visual features and descriptors to obtain an initial guess for the direct transformation refinement. The global trajectory is optimized in a pose-graph optimization. Dr-PoGO demonstrates state-of-the-art performance over 300 km of data in various real-world automotive environments. Our implementation is publicly available: https://github.com/utiasASRL/dr_pogo.

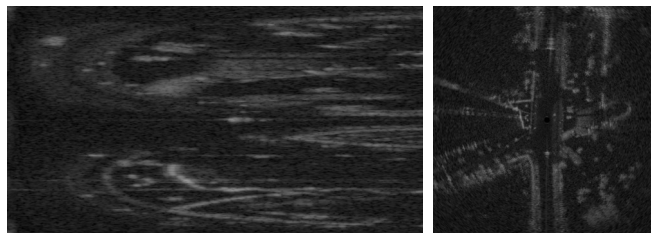
I. INTRODUCTION

At the forefront of any autonomous stack, robust perception and state estimation are required to guarantee reliable robotic operations regardless of the environment and weather conditions. Radar-based sensing was the core of early landmark-based Simultaneous Localization And Mapping (SLAM) works [1]. With the popularization of millimetre-wave radars in the automotive industry for advanced driver assistance systems, there has been a renewed interest in this modality among the robotics community [2]. While radar data are generally less accurate than lidar, they can ‘see’ through dust, smoke, heavy precipitation, etc., providing the required robustness for all-weather navigation [3]. This paper introduces Direct Radar Pose-Graph Optimization (Dr-PoGO), a method for 2D trajectory estimation using Frequency-Modulated Continuous-Wave (FMCW) radars.

Radar-based state estimation is commonly performed by first extracting point clouds or features from the raw data before leveraging ICP-like registration techniques. While lightening the computational burden, the feature extraction process disregards a non-negligible amount of information originally contained in the raw data. Direct Radar Odometry (DRO) [4] demonstrated superior accuracy by performing direct scan-to-local-map registration through continuous cross-correlation using all the scan’s information. In this work, we leverage DRO for odometry before full trajectory optimiza-



(a) Radar-inertial-lidar-visual automotive collection platform



(b) Raw radar polar image

(c) Cartesian image

Fig. 1. Dr-PoGO performs SLAM using 2D radar data. It is validated with 225 km of real-world data. (a) is the platform used to record data (in front of the Skyway environment). (b) and (c) are a sample of radar data in polar and Cartesian forms, respectively.

tion, and we extend the use of direct methods to local-map-to-local-map registration.

Whether it is exteroceptive or proprioceptive-based, or both, odometry is bound to drift due to sensor noise, outlier associations, numerical precision, etc. Pose-graph optimization provides a way to correct for the accumulated drift by explicitly integrating *loop closures* in a global optimization that estimates the whole system’s trajectory, not only the current pose. Accordingly, loop-closure detection, detecting when the robot revisits previously observed areas, has been a strong focus to improve state estimation accuracy. Radar-based perception is no exception, and multiple works have addressed this problem. Dr-PoGO leverages RaPlace [5] to detect loop closures by matching together pairs of radar local-maps built with DRO. Unfortunately, RaPlace does not provide the metric relative transformation between matching local maps. To address this gap, we extend DRO’s abilities to perform local-map-to-local-map registration, as aforementioned. However, unlike in the odometry scenario, where consecutive scans are collected almost from the same location and thus where the identity transformation is a sufficient initial guess, potential loop-closure pairs can correspond to quite different robot poses. Dr-PoGO extracts keypoints

¹ Robotics Institute University of Toronto. Corresponding author: cedric.legentil@robotics.utias.utoronto.ca

² Department of Computer, Control, and Management Engineering “Antonio Ruberti”, Sapienza University of Rome.

and their associated descriptors to perform RANSAC-based coarse registration before a refinement with a direct approach. The use of descriptor-based associations alleviates the need for an initial guess.

This work (i) introduces a novel pipeline mixing feature-based and direct methods for radar data registration without initial guess; (ii) integrates direct odometry, loop-closure detection, and registration in a radar SLAM formulation built upon pose-graph optimization; (iii) demonstrates state-of-the-art performance over more than 300 km of public and self-collected real-world data. Our platform and a data sample are shown in Fig. 1.

II. RELATED WORK

Radars used for robotics perception can generally be split into two main categories: phased arrays for 3D/4D data collection and mechanically spinning antennas for 360° 2D sensing. As Dr-PoGO relies on the latter, this section focuses on 2D radar methods and state estimation over the Special Euclidean group in two dimensions ($SE(2)$). For more details on 3D/4D radar perception, we invite the reader to refer to [2], [6], and [7]. The data collected with FMCW 2D spinning radars consist of polar images where each row corresponds to an azimuth and each column to a different range. As shown in Fig. 1, the data can be converted into *Cartesian images* for visualization or processing. The intensity of the pixels corresponds to the reflections of the radar-emitted radio waves on elements of the environment.

Modern radar-based methods have mostly relied on the extraction of point clouds or features from the raw data. Early works used standard computer-vision keypoints and descriptors [8] over Cartesian images [9]. Feature extractors such as CFAR [10], K-strongest [11], or CFEAR [12] are used to extract points for each azimuth from the polar images. The authors of [13] evaluated the impact of feature extraction on radar-based odometry [13]. It is important to note that radar data is not collected as a snapshot. As the radar’s antenna sweeps its radio-wave beam around (generally at around 4 Hz), any motion of the robot will create motion distortion in the collected scans. Atop the simple fact that the sensor is physically at two different locations when collecting two consecutive azimuths, the relative velocity between the sensor and the environment leads to further distortion due to the Doppler effect. These sources of distortion can be accounted for in a ‘one-off’ undistortion step based on previous state estimates, as in [12]. Another approach is to use continuous-time state estimation to elegantly undistort radar-extracted points at the same time as estimating the platform’s ego-motion, such as [14], [15], and [16]). Other methods leverage deep-learned features [17] or weights [18] to robustify the scan registration process.

Another branch of radar-based works uses *direct* methods for scan registration. The most common approach is to compute cross-correlation scores between the current scan and the previous one for a discrete set of potential state updates. The hypothesis with the largest score is used to update the state. Works such as [19], [20], and [21] rely

on this principle. Note that these methods do not model motion distortion and the Doppler effect. The authors of [22] used a radar with a triangular modulation pattern and demonstrated that estimating the radial velocities from the observed Doppler distortion provided a reliable source of odometry, even in feature-deprived environments, when associated with a yaw-gyroscope. Recently, [4] introduced DRO, the first continuous direct method that rigorously accounts for motion and Doppler distortions by maximizing a continuous cross-correlation score between the current scan and a local map that is updated ‘on-the-fly’.

Full-batch trajectory optimization with loop-closure constraints is a way to mitigate the inherent drift of odometry. In its simplest form, the trajectory is estimated by optimizing a pose graph where each link is a constraint on the relative pose of the sensor at two different timestamps. Pose-graph optimization is the backbone of radar SLAM works such as [23], [24], and [25]. These rely on loop-closure detection approaches originally designed for lidar data ([26], [27], and [28], respectively). Dr-PoGO also builds upon a pose graph for full-batch trajectory estimation, but it leverages a radar-specific place-recognition algorithm: RaPlace [5]. RaPlace employs the Radon Transform over Cartesian radar scans and Fast Fourier Transforms to compute rotation- and translation-invariant scan descriptors. Loop candidates are selected by thresholding the cross-correlation between descriptors. To the best of our knowledge, Dr-PoGO is the first radar-based SLAM framework built using direct methods for relative pose estimation, both for odometry and loop-closure constraints.

III. METHOD

A. Overview

Let us consider a 2D spinning FMCW radar rigidly mounted to a ground robotic platform. The radar’s spinning axis is orthogonal to the locally planar motion of the platform. Optionally, a yaw gyroscope is also rigidly mounted to the platform with its axis parallel to the radar’s spinning axis. The proposed method, Dr-PoGO, estimates the trajectory of the radar $\mathbf{T}_W^m \in SE(2)$, with $m = 1, \dots, N$ and N the total number of radar scans in a given sequence.

Fig. 2 shows the different blocks of Dr-PoGO. First, direct radar odometry is performed with DRO [4]. The latter estimates the system’s ego-motion $\tilde{\mathbf{T}}_{t_m}^{t_{m+1}}$ through radar-based scan-to-local-map registration and updates the local map for each registered scan. Dr-PoGO leverages both DRO’s scan-to-scan motion estimates $\tilde{\mathbf{T}}_{t_m}^{t_{m+1}}$ and local maps \mathcal{M}_m . Loop closures are detected using RaPlace [5] between local maps in Cartesian form. Succinctly, RaPlace computes the Radon transform of each incoming scan (or local map) before evaluating its cross-correlation with past scans (or local maps). It results in a set of matching scores, one per past scan (or local map). Dr-PoGO selects the past local map with the highest score to form a pair $\{\mathcal{M}_i, \mathcal{M}_j\}$ that represents a potential loop closure without thresholding the matching score. The goal is to maximize the number of true-positive matches that are carried to the next step. For each pair $\{\mathcal{M}_i, \mathcal{M}_j\}$,

Dr-PoGO attempts local-map-to-local-map registration in a coarse-to-fine manner as detailed in Section III-B. Finally, successful loop-closure transformations $\tilde{\mathbf{T}}_{t_i}^{t_j}$ are integrated in a pose-graph batch optimization problem that estimates the full trajectory $\hat{\mathcal{S}} = \{\tilde{\mathbf{T}}_W^{t_1}, \dots, \tilde{\mathbf{T}}_W^{t_N}\}$. The motivation to perform Dr-PoGO using DRO’s on-the-fly local maps instead of radar scans is their robustness to noise and to the presence of dynamic objects in the environment. Built using per-pixel low-pass filters, the local maps reliably capture the static elements of the scene, thus enabling high-quality loop-closure detection and registration in real-world driving scenarios where inconsistent traffic can hinder scan-based approaches.

B. Loop-closure direct registration

DRO [4] demonstrated that direct radar data registration outperformed feature-based methods in terms of registration accuracy. However, as DRO’s cross-correlation maximization is not performed with a global solver, it requires a decent initial guess to converge toward the true transformation between two local maps. While it is easily obtained for consecutive scan registration (odometry), this is not the case for loop-closure registration due to the accumulated drift along the sensor’s trajectory. Accordingly, Dr-PoGO first uses a coarse feature-based registration step before refining the relative pose estimate with a direct approach similar to DRO’s.

1) *Feature-based coarse registration*: Given two local maps \mathcal{M}_i and \mathcal{M}_j as Cartesian intensity images, SIFT features and descriptors [8] are extracted independently in both images. Using a brute-force matcher between the two descriptor sets, a list of candidate matches is obtained. The coarse transformation between \mathcal{M}_i and \mathcal{M}_j is estimate through RANSAC-based [29] SE(2) registration. Note that no motion-distortion and Doppler-shift corrections are needed, as these are already accounted for during the local-map generation in DRO. The number of inliers and a user-defined threshold are used to reject unlikely loop closures.

2) *Direct fine registration*: To refine the aforementioned coarse transformation, Dr-PoGO performs direct registration via continuous cross-correlation maximization:

$$\begin{aligned} \tilde{\mathbf{T}}_{t_i}^{t_j} &= \operatorname{argmax}_{\mathbf{T}_{t_i}^{t_j}} g(\mathcal{M}_i, \mathcal{M}_j, \mathbf{T}_{t_i}^{t_j}), \text{ with} \\ g(\mathcal{M}_i, \mathcal{M}_j, \mathbf{T}_{t_i}^{t_j}) &= \sum_{x,y} \beta \left(\mathcal{M}_i, \mathbf{T}_{t_i}^{t_j} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \right) \mathcal{M}_j(x, y), \end{aligned} \quad (1)$$

x and y some Cartesian coordinates in \mathcal{M}_j , $\mathcal{M}_j(x, y)$ the local-map intensity at coordinates x, y , and $\beta(\mathcal{M}_i, [\hat{x} \ \hat{y} \ 1]^\top)$ the bilinear interpolation of \mathcal{M}_i ’s intensity at coordinates \hat{x}, \hat{y} . While being a good relative metric for registration, the absolute cross-correlation value g depends on the environment and the total ‘amount of intensity’ in \mathcal{M}_j . As illustrated in Fig. 3, g cannot be directly thresholded to classify a potential loop-closure registration as successful. Thus, Dr-PoGO applies a user-defined threshold

over the scaled cross-correlation $s = \frac{g(\mathcal{M}_i, \mathcal{M}_j, \tilde{\mathbf{T}}_{t_i}^{t_j})}{\sum_{x,y} \mathcal{M}_j^2(x,y)}$ to reject unlikely loop closures. Intuitively, this score is a proxy for the overlap ratio between the non-zero-intensity data in \mathcal{M}_i and \mathcal{M}_j . In our experiments, the threshold is 0.5.

C. Pose-graph optimization

After coarse-to-fine registration of loop-closure candidates, the set of successful pairs \mathcal{L} and their associated relative transformations $\tilde{\mathbf{T}}_{t_i}^{t_j}$ are integrated in an SE(2) pose graph as binary factors with relative pose residuals:

$$\mathbf{r}_i^j = \operatorname{Log} \left((\tilde{\mathbf{T}}_{t_i}^{t_j})^{-1} (\mathbf{T}_W^{t_i})^{-1} \mathbf{T}_W^{t_j} \right), \quad (2)$$

where Log converts an SE(2) matrix into its x - y - θ vector representation. The DRO scan-to-scan estimates are used to constrain consecutive poses with residuals \mathbf{r}_m^{m+1} . If a gyroscope is used, DRO estimates angular velocity biases b_m using a simple heuristic when the vehicle is stopped. While improving over the scenario of no bias estimation, these bias estimates do not leverage radar information. We propose to integrate bias corrections of DRO’s biases in our pose-graph optimization by modifying DRO’s scan-to-scan poses as $\tilde{\mathbf{T}}_{t_m}^{t_{m+1}} \mathbf{T}_{b_m}$ with

$$\mathbf{T}_{b_m} = \begin{bmatrix} \cos(\phi) & \sin(\phi) & 0 \\ -\sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \phi = (b_m - \tilde{b}_m) \Delta t.$$

In such a scenario, the state \mathcal{S} to estimate is $\{\mathbf{T}_W^{t_1}, b_1, \dots, \mathbf{T}_W^{t_N}, b_N\}$. Eventually, Dr-PoGO’s pose-graph optimization is

$$\hat{\mathcal{S}} = \operatorname{argmax}_{\mathcal{S}} \sum_{i=1}^{N-1} \left(\|\mathbf{r}_i^{i+1}\|_{\Omega_i^{i+1}}^2 + \|r_{b_i}\|_{\Omega_{b_i}}^2 \right) + \sum_{\{i,j\} \in \mathcal{L}} \|\mathbf{r}_i^j\|_{\Omega_i^j}^2,$$

with $\|\mathbf{r}_\bullet\|_{\Omega_\bullet} = \mathbf{r}_\bullet^\top \Omega_\bullet^{-1} \mathbf{r}_\bullet$, Ω_\bullet the covariance matrix associated with \mathbf{r}_\bullet , and the residual $r_{b_i} = b_i - b_{i+1}$ enforces the slow varying nature of gyroscope biases via a Brownian motion process. To accommodate the presence of outliers in the loop-closure constraints, Dr-PoGO applies a Cauchy loss function to the associated residuals.

IV. EXPERIMENTS

A. Implementation details

The original implementation of RaPlace computes the Radon transform for every incoming scan and attempts detection by computing a similarity score between the current scan and every previous frame. However, the Radon transform is a slow operation (≈ 1 s), and the $\mathcal{O}(N)$ matching strategy does not scale to very long trajectories. To reduce the computational load, we only perform the Radon transform for keyframes and attempt loop-closure detection solely with past keyframes that are within a certain radius of the current estimate. To maximize the number of loop-closure detections while allowing real-time operations, we dedicate one CPU core to RaPlace computations. Accordingly, a local map is considered to be a new keyframe only if, upon reception, the RaPlace-dedicated CPU core has finished the previous

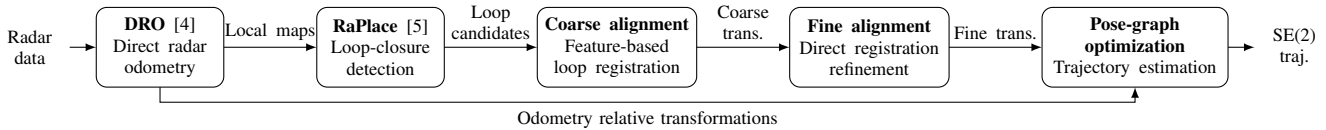


Fig. 2. Block-diagram overview of Dr-PoGO. First, DRO processes the raw radar data (and optional yaw-gyroscope measurements) to generate per-scan local maps and relative SE(2) transformations. Leveraging the local maps, RaPlace proposes loop-closure candidates that are later used for coarse-to-fine registration. Finally, a pose-graph optimization estimates the whole trajectory with DRO’s relative poses and the refined loop-closure transformations.

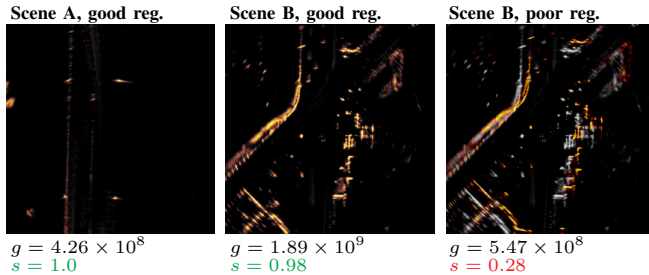


Fig. 3. Example of local-map registration. Alone, the cross-correlation score g does not allow for discrimination between good and poor registrations across all environments. The scaled score s is a better proxy as it is less impacted by the scene’s average intensity return.

detection. For the match search, the radius is proportional to the distance travelled using the typical maximum drift of the odometry (arbitrarily set to 1%).

Our coarse, feature-based alignment is performed using OpenCV for feature extraction, matching (brute force), and SE(2) registration. The native registration method also estimates the scale alongside the SE(2) transformation. If the resulting scale differs from 1 ($\pm 5\%$), the pair of local maps is rejected. The proposed fine registration extends DRO’s GPU implementation using PyTorch. Note that an additional coarse grid search around the estimate of the feature-based alignment step is performed to further avoid unwanted local minima. Finally, the whole trajectory is estimated via pose-graph optimization every time a new loop-closure detection is validated by the direct registration refinement. To accommodate long trajectory loops (which result in significant odometry drift), the optimization is performed once with a large scale for the Cauchy loss, then again with a smaller one.

B. Datasets, baselines, and metrics

1) *Datasets*: We have evaluated our method using both the Boreas [30] and Boreas-RT [31] datasets. The former focuses on multi-season driving through a wide range of weather conditions on the same 8 km suburban route over a year, while the latter is collected along various routes with challenging roads and environments. Both have been collected with an automotive platform equipped with a Navtech RAS6 radar and a gyroscope, among other sensors. A notable difference is that the radar firmware used for Boreas-RT has a triangular modulation pattern that enables Doppler velocity extraction, as shown in [22]. For the evaluation with Boreas data, we randomly selected 13 se-

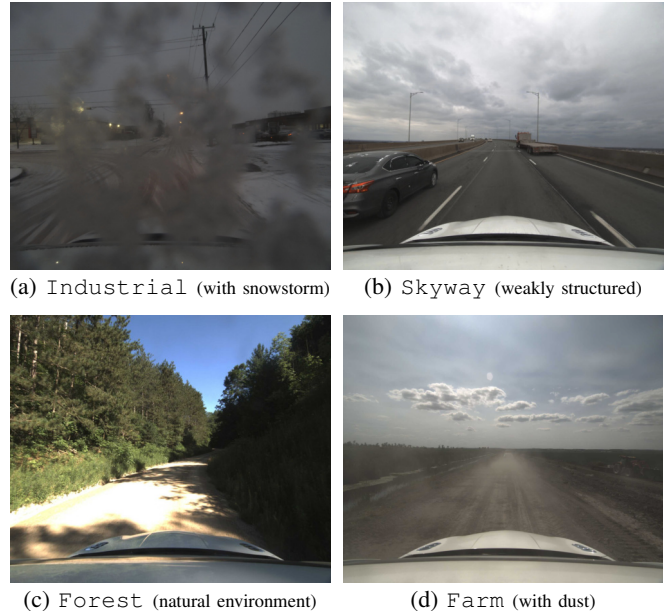


Fig. 4. Illustration of the challenging conditions in Boreas-RT

quences.¹ With Boreas-RT, we leverage 4 sequences of each of the following routes: Suburbs (7.9 km), Industrial (5.4 km), Skyway (11.1 km), Forest (16.4 km), and Farm (10.8 km). Both the Suburbs and Skyway routes offer numerous opportunities for loop-closure detections as the second half of each sequence corresponds to the same itinerary as the first half, but in the other direction. The other sequences form large loops with a fairly limited overlap between the beginning and end of the trajectories. As illustrated in Fig. 4, the selected environments span a wide range of challenging driving conditions with snowstorms, weakly structured environments, and natural scenes. Additionally, the Forest and Farm sequences are collected on uneven dirt roads, inducing significant roll and pitch of the vehicle.

2) *Baselines*: We benchmark Dr-PoGO against two recent open-source radar SLAM implementations. The first is Navtech-SLAM [32], which relies on ORORA [33] for odometry, Scan Context [34] for loop-closure detection and ICP for loop-closure registration. The second is TBV-SLAM, with CFAR-3 [12] as the odometry method and Scan Context [34] with CorAI [35] for loop-closure detec-

¹2020-{11-26-13-58, 12-18-13-44}, 2021-{01-26-11-22, 02-02-14-07, 03-02-13-38, 03-30-14-23, 04-20-14-11, 05-13-16-11, 07-20-17-33, 09-02-11-42, 10-15-12-35, 11-14-09-47, 11-23-14-27}.

tion/validation. Note that these two methods also rely on a pose-graph optimization for full-batch trajectory estimation.

3) *Metrics*: The Absolute Trajectory Error (ATE) is used to evaluate the overall accuracy of Dr-PoGO and the different baselines for trajectory estimation. After aligning the ground-truth trajectory and the estimated one with SVD-based SE(2) registration, the ATE is computed as $\sqrt{\frac{1}{N} \sum_{i=1}^N \|\hat{\mathbf{p}}_W^{t_i} - \mathbf{p}_{GT}^{t_i}\|^2}$, with $\hat{\mathbf{p}}_{\blacksquare}$ the translation component of $\hat{\mathbf{T}}_{\blacksquare}$. Since all the sequences finish close to the starting location, we use the End-Pose Error (EPE) as a proxy to assess the effectiveness of loop-closure constraints in each method. The EPE is defined as the norm of the translation component of $(\mathbf{T}_{GT}^{t_N})^{-1} \mathbf{T}_{GT}^{t_1} (\hat{\mathbf{T}}_W^{t_1})^{-1} \hat{\mathbf{T}}_W^{t_N}$.

C. Trajectory error

1) *Boreas dataset*: Fig. 5 shows the per-sequence ATE and EPE for Dr-PoGO, Navtech-SLAM and TBV-SLAM on the Boreas dataset [30]. Dr-PoGO achieves the lowest ATE, with an average of 0.82 m and a maximum of 1.32 m. In terms of EPE, Dr-PoGO also outperforms the different baselines. TBV-SLAM attains moderate ATEs on average (9.61 m) but most sequences are between 2 and 3 m ATE. There is a strong correlation between the ATE and EPE in TBV-SLAM’s results, suggesting that the odometry (CFEAR [12]) is fairly reliable, but occasional failures in loop-closure detection hinder the system’s global performance. The presence of traffic in the data and the conservative nature of CorAl [35] loop-closure validation are possible explanations for missing loop closures. The inconsistency in Navtech-SLAM’s results and the absence of a clear correlation between ATE and EPE indicate a lack of robustness both in the odometry (ORORA [33]) and the loop-closure detection mechanism. Indeed, ORORA relies on a feature-matching front end that is sensitive to sparse or degraded radar scans, as it reduces both the number and the spatial distribution of correspondences. This negatively impacts the decoupled rotation estimates and subsequent anisotropic translations.

Overall, we attribute Dr-PoGO’s performance to a more reliable odometry source (DRO [4]) and our coarse-to-fine loop-closure registration and validation using local maps. The use of local maps improves robustness in real-world scenarios where dynamic elements, such as traffic, impede scan-based operations. Additionally, using the local-map scaled cross-correlation to validate or not a loop-closure candidate provides more information than a descriptor-based similarity score. Quantitative results of an ablation study in Section IV-E support these claims.

2) *Boreas-RT dataset*: Unfortunately, when using radar data collected with a triangular modulation pattern, the odometry component of Navtech-SLAM fails. To obtain quantitative results for Navtech-SLAM, we preprocessed the radar data by removing the Doppler distortion based on the ground-truth velocities. Table I presents both the average ATE and EPE for each method in each environment. The figures in this table only account for successful runs. Defining success by an ATE under 1% of the travelled distance,

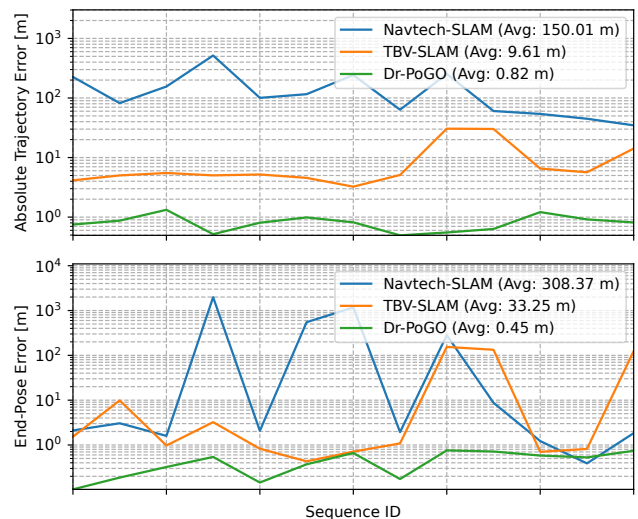


Fig. 5. Absolute Trajectory Error and End-Pose Error (log scale) for each of the 13 sequences used for benchmarking Dr-PoGO and the baselines on the Boreas dataset.

Navtech-SLAM and TBV-SLAM fail in some sequences. Omitting these, all the SLAM methods correct large odometry drifts thanks to successful loop-closure detections in *Suburbs* and *Industrial* environments, as shown by the low EPE. As expected, the average EPE obtained with DRO is large due to the absence of a loop-closure mechanism. Thanks to DRO’s robustness, Dr-PoGO is the only SLAM method to be successful on the highly challenging *Skyway*, *Forest*, and *Farm* sequences.

When considering the ATE, the different approaches span a wider range of results. Navtech-SLAM performs the worst. As illustrated in Fig. 6, it suggests that ORORA odometry introduces a drift/warp of the trajectory that is too large to be fully corrected with loop-closure constraints. TBV-SLAM performs well with the second-best results. The estimated trajectories are more consistent with the true motion when compared to Navtech-SLAM. Dr-PoGO significantly outperforms all the other methods with GPS-level ATE (< 5 m) throughout all setups. Comparing Dr-PoGO’s results (ATE and EPE) with DRO’s demonstrates the soundness of our approach and its ability to reach new levels of robust and accurate trajectory estimation.

The trajectory accuracy displayed by Dr-PoGO rivals that of lidar-based frameworks. To support that claim, we have selected Fast-LIO2 [36] and 2Fast-2Lamaa [37] as lidar-inertial baselines. These two frameworks do not perform explicit loop closure but estimate their state in a scan-to-map manner. Accordingly, for trajectories where the vehicle retraces its way to the starting location, it can be seen as loop-closure localization for every lidar scan. Unfortunately, when the trajectory forms a large loop, these methods fail due to odometry drift. Thus, we only benchmark Fast-LIO2 and 2Fast-2Lamaa on the *Suburbs* sequences and obtain

ATEs² of 13.82³ and 2.26 m, respectively. Dr-PoGO offers a four-fold improvement over the best lidar baseline.

D. Loop-closure detection and registration

In this section, we analyze the accuracy of the proposed coarse-to-fine loop-closure registration on the three most structured sequences of the Boreas-RT dataset. Fig. 7 shows strip plots of the alignment errors for the 15 self-collected sequences, and Table II provides numerical values per environment type: average per-sequence number of matches, average number of inliers (position and rotation errors under 1.5 m and 1°, respectively), position and rotation Root Mean Squared Error (RMSE) for the inliers. The RMSE values suggest that direct refinement significantly improves registration accuracy for all sequence types compared to feature-based alignment. One can see that after the coarse registration, there is a large number of outliers. It is important to remember that Dr-PoGO does not threshold RaPlace’s similarity score to keep a maximum number of true-positive matches and prevent having too many parameters to tune. Thus, it is normal to find many outliers after the coarse registration step. When applying the direct pose refinement with our proposed scaled-cross-correlation score threshold, the number of outliers is drastically reduced with a very limited impact on the number of inliers, thus greatly improving the inlier ratio.

E. Ablation study

To provide insight into the impact of each component of the proposed method, we have conducted an ablation study using the Suburbs, Industrial, and Skyway sequences of the Boreas-RT dataset. Table III shows the corresponding ATEs and EPEs. Note that all variants share the same parameters. The first observation is the importance of using a gyroscope; without it, Dr-PoGO’s performance significantly drops both in terms of ATE and EPE, and some failures occur. It suggests that the accuracy of gyro-less odometry is not sufficient to provide high-quality trajectory priors and, thus, impedes loop-closure detections, leading to missing or erroneous loop-closure constraints. Further analysis of gyro-less operations is part of future work. When using a gyroscope, the use of local maps (both for odometry and loop-closure detection/registration) and the direct refinement are Dr-PoGO’s most critical features. Not leveraging local maps or relying solely on coarse loop-closure registration results in errors up to four times higher. Finally, thanks to the relatively high quality of the gyroscope present on our data collection platform, the impact of gyroscope bias is fairly limited. Accordingly, ignoring the bias does not have a significant impact on Dr-PoGO’s overall performance.

F. Computation time

Table IV shows the breakdown of Dr-PoGO’s computation time using a laptop equipped with an Intel i7-13850HX CPU

²To convert the SE(3) estimates into SE(2), we first fit a plane to the estimated trajectories and project all the poses on that plane.

³Note that Fast-LIO2’s results are computed on 3 out of the 5 sequences due to dramatic failures, probably caused by unhandled dynamic objects.

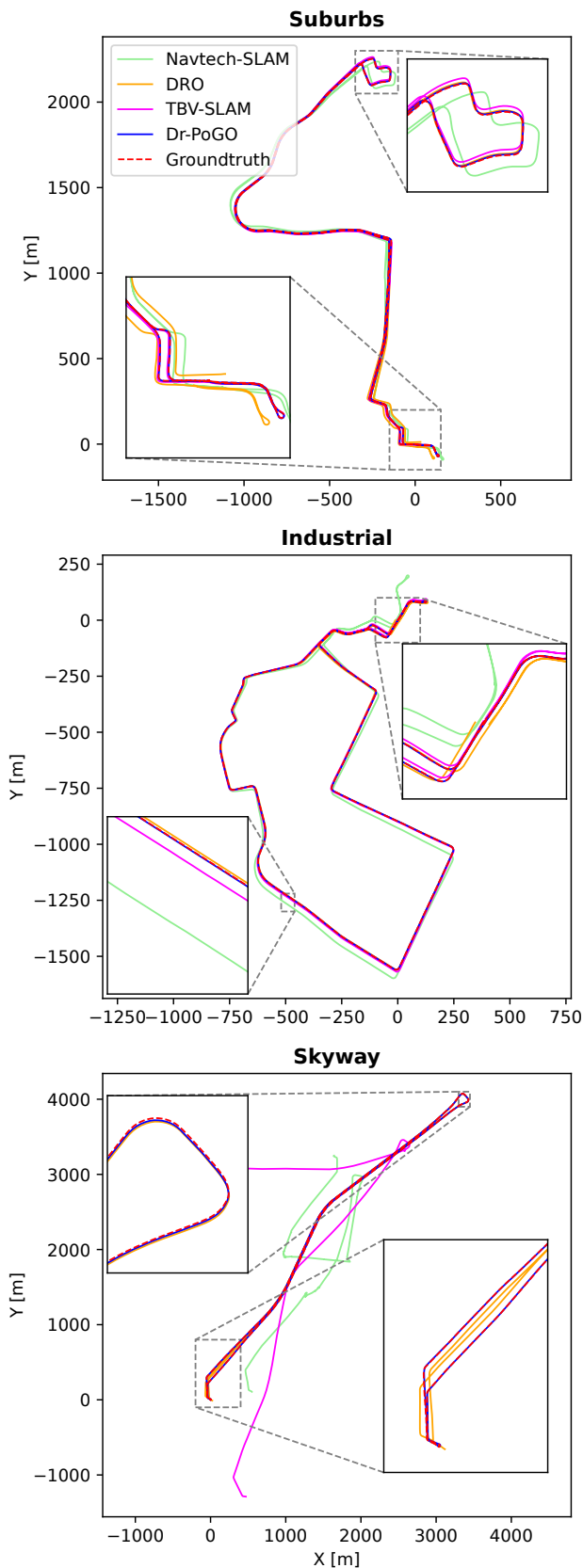


Fig. 6. Trajectory examples over the three environment types of the self-collected dataset. Estimated trajectories are aligned to the ground-truth as for the ATE computation.

TABLE I
AVERAGE ATE AND EPE [M] ON SELF-COLLECTED DATASET

Method	Suburbs	Industrial	Skyway	Forest	Farm
DRO [4]	9.53 / 38.9	6.32 / 17.2	18.6 / 60.7	25.8 / 80.5	38.8 / 170.4
Navtech-SLAM [32]*	36.4 / 0.65 ¹	28.9 / 1.46 ¹	- / - ⁴	- / - ⁴	- / - ⁴
TBV-SLAM [25]	6.11 / 0.55	4.72 / 0.82 ¹	- / - ⁴	- / - ⁴	- / - ⁴
Dr-PoGO (ours)	0.75 / 0.42	1.58 / 0.77	3.16 / 0.23	4.31 / 0.75	4.19 / 0.56

Results reported as XX / YY, with XX the ATE and YY the EPE.

¹ The superscript indicates the number of failed sequences (ATE above 1% of the sequence distance). * Navtech-SLAM required Doppler undistortion preprocessing.

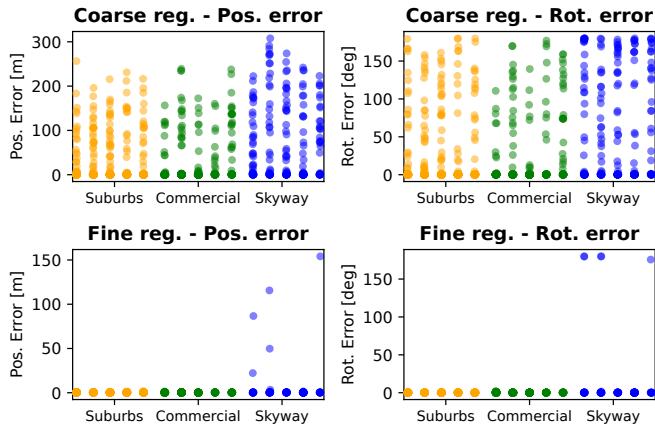


Fig. 7. Strip plots of the position (left column) and rotation (right column) alignment error after the coarse feature-based registration (top row) and the direct refinement (bottom row) using the 15 self-collected sequences.

TABLE II
LOOP-CLOSURE REGISTRATION ANALYSIS

Seq. type	Avg. matches	Avg. inliers	Inlier ratio	In. pos. RMSE	In. rot. RMSE
<i>Coarse registration</i>					
Suburbs	130	101	78%	0.50 m	0.26°
Industrial	72.6	54.8	75%	0.38 m	0.26°
Skyway	43.2	17.0	39%	0.51 m	0.26°
<i>Fine registration</i>					
Suburbs	91.0	90.6	99%	0.34 m	0.21 °
Industrial	55.4	54.2	98%	0.27 m	0.19 °
Skyway	19.4	18.2	94%	0.36 m	0.17 °

The best results between coarse and fine registration are shown in bold.

TABLE III
DR-POGO ABLATION STUDY (AVERAGE ATE AND EPE [M])

Variant	Suburbs	Industrial	Skyway
No local map	2.97 / 10.38	6.61 / 19.5	3.8 / 0.35
Coarse reg.	1.53 / 0.72 ¹	2.16 / 1.63	3.96 / 0.18 ¹
No bias	0.76 / 0.48	1.62 / 0.90	3.15 / 0.21
No gyro	7.05 / 4.09 ¹	9.21 / 1.59 ²	- / - ⁴
Dr-PoGO (Ours)	0.75 / 0.42	1.58 / 0.77	3.16 / 0.23

Results reported as XX / YY, with XX the ATE and YY the EPE.

¹ The superscript indicates the number of failed sequences.

TABLE IV
DR-POGO COMPUTATION TIME ANALYSIS

Component	Hardware	Time per operation [s]	Avg. per raw frame [s]
DRO-GD	GPU	0.104	0.104
RaPlace	CPU (1 core)	0.646	0.215
Coarse reg.	CPU (1 core)	0.170	0.024
Direct reg.	GPU	0.389	0.013
Pose graph	CPU (1 core)	0.093	0.002

and an Nvidia RTX 5000 Mobile GPU. The column ‘time per operation’ represents the marginal cost of each component. In contrast, the last column presents the average time per frame, calculated by dividing the total time by the number of radar scans in each sequence. Using a 4Hz radar as in the Boreas and self-collected datasets, we demonstrated that Dr-PoGO runs in real-time, as the sums of GPU and CPU times per frame are below the 250ms mark. Note that our implementation of RaPlace loop-closure detection attempts to maximize the usage of 1 CPU core. For lightweight operations, one can reduce the number of keyframes for place recognition and subsequent registrations and optimizations.

V. CONCLUSION

This paper introduced Dr-PoGO, a framework for radar-based 2D SLAM. It relies on DRO for odometry, RaPlace for loop-closure detection, and introduces a novel coarse-to-fine local-map alignment pipeline that leverages feature-based and direct registration techniques. The SE(2) trajectory is estimated through a full-batch pose-graph optimization. To the best of our knowledge, it is the first radar SLAM approach that leverages direct registration for odometry and loop-closure registration. We have benchmarked our framework over 300km of real-world automotive data (sequences up to 16 km-long) and demonstrated state-of-the-art trajectory accuracy. We even showed that Dr-PoGO can outperform state-of-the-art lidar-based frameworks. To enable direct localization with respect to a single global map, future work includes using Dr-PoGO’s output as the initial guess for a global bundle adjustment optimization that directly leverages radar data in a single optimization instead of SE(2) relative transformations.

VI. ACKNOWLEDGEMENT

The authors would like to thank Daniil Lisus and Katya M. Papais for leading the Boreas-RT collection campaign, and Raphael Falque for running the lidar baselines.

REFERENCES

- [1] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.
- [2] K. Harlow, H. Jang, T. D. Barfoot, A. Kim, and C. Heckman, "A new wave in robotics: Survey on recent mmwave radar applications in robotics," *IEEE Trans. on Robotics (TRO)*, vol. 40, pp. 4544–4560, 2024.
- [3] C. Le Gentil, D. Lisus, and T. D. Barfoot, "Do we still need to work on odometry for autonomous driving?" in *ICRA 2025 Workshop on Field Robotics*, 2025.
- [4] C. Le Gentil, L. Brizi, D. Lisus, X. Qiao, G. Grisetti, and T. D. Barfoot, "Dro: Doppler-aware direct radar odometry," in *Proc. of Robotics: Science and Systems (RSS)*, 2025.
- [5] H. Jang, M. Jung, and A. Kim, "Raplac: Place recognition for imaging radar using radon transform and mutable threshold," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023, pp. 11 194–11 201.
- [6] N. J. Abu-Alrub and N. A. Rawashdeh, "Radar odometry for autonomous ground vehicles: A survey of methods and datasets," *IEEE Trans. on Intelligent Vehicles (T-IV)*, vol. 9, no. 3, pp. 4275–4291, 2024.
- [7] A. Venon, Y. Dupuis, P. Vasseur, and P. Merriaux, "Millimeter wave fmcw radars for perception, recognition and localization in automotive applications: A survey," *IEEE Trans. on Intelligent Vehicles (T-IV)*, vol. 7, pp. 1–1, 09 2022.
- [8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
- [9] J. Callmer, D. Törnqvist, F. Gustafsson, H. Svensson, and P. Carlbom, "Radar slam using visual features," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, 12 2011.
- [10] H. Rohling, "Radar cfar thresholding in clutter and multiple target situations," *IEEE Trans. on Aerospace and Electronic Systems (T-AES)*, vol. AES-19, no. 4, pp. 608–621, 1983.
- [11] D. Adolfsson, M. Magnusson, A. Alhashimi, A. Lilienthal, and H. Andreasson, "Cfear radarodometry - conservative filtering for efficient and accurate radar odometry," 09 2021, pp. 5462–5469.
- [12] D. Adolfsson, M. Magnusson, A. Alhashimi, A. J. Lilienthal, and H. Andreasson, "Lidar-level localization with radar? the cfear approach to accurate, fast, and robust large-scale radar odometry in diverse environments," *IEEE Trans. on Robotics (TRO)*, vol. 39, no. 2, pp. 1476–1495, 2023.
- [13] E. Preston-Krebs, D. Lisus, and T. D. Barfoot, "The Finer Points: A Systematic Comparison of Point-Cloud Extractors for Radar Odometry," *Proceedings of the Conference on Robots and Vision*, may 27 2025.
- [14] K. Burnett, A. P. Schoellig, and T. D. Barfoot, "Do we need to compensate for motion distortion and doppler effects in spinning radar navigation?" *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 771–778, 2021.
- [15] K. Burnett, Y. Wu, D. J. Yoon, A. P. Schoellig, and T. D. Barfoot, "Are We Ready for Radar to Replace Lidar in All-weather Mapping and Localization?" *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 4, pp. 10 328–10 335, 2022.
- [16] K. Burnett, A. P. Schoellig, and T. D. Barfoot, "Continuous-time radar-inertial and lidar-inertial odometry using a gaussian process motion prior," *IEEE Trans. on Robotics (TRO)*, vol. 41, p. 1059–1076, Jan. 2025.
- [17] K. Burnett, D. J. Yoon, A. P. Schoellig, and T. D. Barfoot, "Radar Odometry Combining Probabilistic Estimation and Unsupervised Feature Learning," in *Proc. of Robotics: Science and Systems (RSS)*, 2021.
- [18] Lisus, Daniil and Laconte, Johann and Burnett, Keenan and Zhang, Ziyu and Barfoot, Timothy D., "Pointing the Way: Refining Radar-Lidar Localization Using Learned ICP Weights," *Proceedings of the Conference on Robots and Vision*, may 27 2025.
- [19] P. Checchin, F. Gérossier, C. Blanc, R. Chapuis, and L. Trassoudaine, "Radar scan matching slam using the fourier-mellin transform," vol. 62, 01 2009, pp. 151–161.
- [20] D. Barnes, R. Weston, and I. Posner, "Masking by Moving: Learning Distraction-Free Radar Odometry from Pose Information," in *Conf. Robot Learn.*, 2019.
- [21] 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, pp. 2617–2623.
- [22] D. Lisus, K. Burnett, D. J. Yoon, R. Poulton, J. Marshall, and T. D. Barfoot, "Are doppler velocity measurements useful for spinning radar odometry?" *IEEE Robotics and Automation Letters (RA-L)*, vol. 10, no. 1, pp. 224–231, 2025.
- [23] M. Holder, S. Hellwig, and H. Winner, "Real-time pose graph slam based on radar," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1145–1151.
- [24] Z. Hong, Y. Petillot, A. Wallace, and S. Wang, "Radarslam: A robust simultaneous localization and mapping system for all weather conditions," *Intl. Journal of Robotics Research (IJRR)*, vol. 41, no. 5, pp. 519–542, 2022.
- [25] D. Adolfsson, M. Karlsson, V. Kubelka, M. Magnusson, and H. Andreasson, "Tbv radar slam – trust but verify loop candidates," *IEEE Robotics and Automation Letters (RA-L)*, vol. 8, no. 6, pp. 3613–3620, 2023.
- [26] M. Himstedt, J. Frost, S. Hellbach, H.-J. Böhme, and E. Maehle, "Large scale place recognition in 2d lidar scans using geometrical landmark relations," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 5030–5035.
- [27] L. He, X. Wang, and H. Zhang, "M2dp: A novel 3d point cloud descriptor and its application in loop closure detection," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 231–237.
- [28] G. Kim, S. Choi, and A. Kim, "Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments," *IEEE Trans. on Robotics (TRO)*, vol. 38, no. 3, pp. 1856–1874, 2022.
- [29] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [30] K. Burnett, D. J. Yoon, Y. Wu, A. Z. Li, H. Zhang, S. Lu, J. Qian, W.-K. Tseng, A. Lambert, K. Y. Leung, A. P. Schoellig, and T. D. Barfoot, "Boreas: A multi-season autonomous driving dataset," *Intl. Journal of Robotics Research (IJRR)*, vol. 42, no. 1-2, pp. 33–42, 2023.
- [31] D. Lisus, K. M. Papais, C. L. Gentil, E. Preston-Krebs, A. Lambert, K. Y. K. Leung, and T. D. Barfoot, "Boreas road trip: A multi-sensor autonomous driving dataset on challenging roads," 2026. [Online]. Available: <https://arxiv.org/abs/2602.16870>
- [32] Gisbi-Kim, "Github - gisbi-kim/navtech-radar-slam: Real-time radar slam: Orora + scancontext." [Online]. Available: <https://github.com/gisbi-kim/navtech-radar-slam>
- [33] 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, pp. 2046–2053.
- [34] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Madrid, Oct. 2018.
- [35] D. Adolfsson, M. Castellano-Quero, M. Magnusson, A. J. Lilienthal, and H. Andreasson, "Coral: Introspection for robust radar and lidar perception in diverse environments using differential entropy," *Journal on Robotics and Autonomous Systems (RAS)*, vol. 155, p. 104136, 2022.
- [36] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Trans. on Robotics (TRO)*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [37] C. Le Gentil, R. Falque, D. Lisus, and T. D. Barfoot, "2fast-2lamaa: Large-scale lidar-inertial localization and mapping with continuous distance fields," 2025. [Online]. Available: <https://arxiv.org/abs/2410.05433>