

# CFEAR-Teach-and-Repeat: Fast and Accurate Radar-only Localization

Maximilian Hilger<sup>1</sup>, Daniel Adolfsson<sup>2</sup>, Ralf Becker<sup>3</sup>, Henrik Andreasson<sup>2</sup>, and Achim J. Lilienthal<sup>1,2</sup>

**Abstract**—Reliable localization in prior maps is essential for autonomous navigation, particularly under adverse weather, where optical sensors may fail. We present CFEAR-TR, a teach-and-repeat localization pipeline using a single spinning radar, which is designed for easily deployable, lightweight, and robust navigation in adverse conditions. Our method localizes by jointly aligning live scans to both stored scans from the *teach* mapping pass, and to a sliding window of recent live keyframes. This ensures accurate and robust pose estimation across different seasons and weather phenomena. Radar scans are represented using a sparse set of oriented surface points, computed from Doppler-compensated measurements. The map is stored in a pose graph that is traversed during localization. Experiments on the held-out test sequences from the Boreas dataset show that CFEAR-TR can localize with an accuracy as low as 0.117 m and 0.096°, corresponding to improvements of up to 63% over the previous state of the art, while running efficiently at 29 Hz. These results substantially narrow the gap to lidar-level localization, particularly in heading estimation. We make the C++ implementation of our work available to the community.

## I. INTRODUCTION

In the autonomous navigation of mobile robots, accurate information about the current position is crucial. This is of increased importance, especially in degraded settings, where GNSS reception might be blocked and sensors operating in the optical wavelength, such as cameras and lidars, fail. Spinning radar recently emerged as a suitable sensing modality in autonomous navigation, with lidar-approaching accuracies in odometry [1], Simultaneous Localization and Mapping (SLAM) [2], and autonomous path following [3].

While spinning radar has been widely studied for odometry and SLAM, comparatively little attention has been given to metric localization. In localization, the objective is to estimate the robot’s pose relative to a prior map. Current radar localization approaches achieve an accuracy of 11.9 cm in translation and 0.27° in rotation [4]. This performance lags behind lidar, which typically localizes within 5.0 cm and 0.04°, with the gap especially pronounced in heading estimation [4]. A common strategy to improve heading estimation, particularly in radar-based odometry, is to fuse gyroscope measurements [5]–[7]. While this approach is effective, such multi-sensor systems pose additional requirements on extrinsic calibration, time synchronization, and bias handling. These requirements increase the setup complexity of radar-based localization in real-world deployments.

<sup>1</sup>Chair of Perception for Intelligent Systems, Munich Institute of Robotics and Machine Intelligence, Technical University of Munich, Munich, Germany. Corresponding author: maximilian.hilger@tum.de

<sup>2</sup>Robot Navigation and Perception Lab of the AASS Research Center, Örebro University, Örebro, Sweden

<sup>3</sup>Bosch Rexroth, Germany

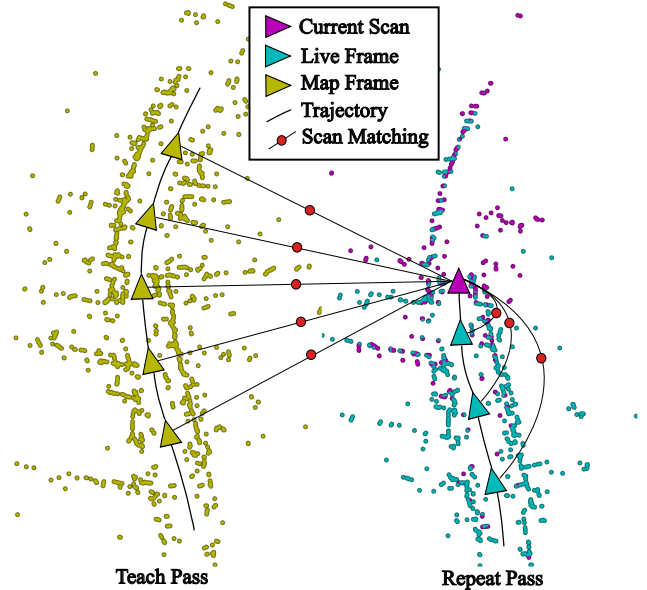


Fig. 1: Visualization of the easily deployable, radar-only CFEAR-Teach-and-Repeat localization pipeline. The current scan (magenta) is simultaneously registered to keyframe scans from the teach phase (yellow) and recent live keyframes from the repeat phase (cyan). By matching sparse feature sets extracted from carefully undistorted radar scans, the system achieves both efficient and accurate localization without the need for additional sensors.

An alternative is to rely on radar-only localization, which offers simpler deployment and avoids the need for additional sensors. However, existing radar-only methods remain limited. Prior work has applied Iterative Closest Points (ICP)-based teach-and-repeat localization [4]. Research in odometry estimation has shown that using sparse oriented surface points outperforms ICP both in terms of computational efficiency and drift [1]. Oriented surface points provide a compact and robust representation: they capture local surface geometry while supporting fast scan-to-submap registration with rich scene information. This makes them particularly well-suited for radar localization, yet they have not previously extended into a full localization approach.

Our approach, named CFEAR-TR (CFEAR-Teach-and-Repeat), inspired by prior work of Burnett et al. [4], implements a teach-and-repeat localization framework based on the radar-only odometry approach CFEAR radar odometry [1]. By leveraging sparse oriented surface points and a pose graph-based map structure, our system achieves accurate localization without requiring extrinsic calibration

or multi-sensor fusion. Fig. 1 shows the pose graph structure along with its contained oriented surface points. Our contributions are:

- An efficient radar-only teach-and-repeat localization system that allows for accurate real-time route-following from sparse scan representations.
- A dual registration formulation combining map frames and live frames. Unlike prior work that uses odometry and localization consecutively, we optimize both against the map and recent scans simultaneously to ensure robustness to changes and reduce noise in consecutive pose estimates.
- An evaluation of our improved version of CFEAR radar odometry, which includes radar encoder reading, range bias, and Doppler range compensation. The presented adaptations result in a new state of the art for radar-only odometry at 0.42% translation drift, while running at 29 Hz.
- A systematic evaluation of parameter sensitivity and map refinement strategies.

We demonstrate that using our approach, a localization accuracy of 11.7 cm and  $0.096^\circ$  can be achieved on hold-out test sequences on a public benchmark, cutting the previously published values in orientation by up to 63%. We make the C++ implementation of our work available online<sup>1</sup>.

## II. RELATED WORK

We review the related work in radar odometry and mapping (Sec. II-A) and in radar localization (Sec. II-B), which together frame the context of our teach-and-repeat pipeline.

### A. Radar Odometry and Mapping

Spinning radar odometry incrementally estimates motion from spinning radar data. Most approaches align live scans against local submaps to compute the motion increments. These approaches can be divided into registration-based pipelines [1], [5], [8], [9] and direct approaches working on image correlation [7], [10]–[12]. Accurate Doppler and motion compensation is a recurring requirement across methods [13]. The current state-of-the-art in radar-only odometry is held by CFEAR radar odometry [1], which uses sparse oriented surface points in a scan-to-submap registration approach to estimate the sensor movement. We build upon this work and store the submaps in a *teach* pass for usage as the reference map in the later *repeat* pass.

If submaps are obtained from odometry, the poses are subject to odometry drift. A possible way to alleviate the drift, especially in orientation, is the use of loop closure detection. This extends odometry pipelines to full SLAM [2], [14], [15] that correct odometry for global consistency. However, as a consequence, the local accuracy of the map might reduce – particularly in case of incorrect loop closures, or over- or under-confident covariance estimates. To mitigate incorrect loopclosures, Adolfsson et al. [2] use an introspective loop closure detection approach that uses a

combination of alignment verification, appearance similarity, and an odometry prior to reject incorrect loop closures. We explore whether their loop closure detection applied to the pose graph in the mapping stage improves teach-and-repeat localization if multiple frames from the teach pass are used.

Another way to reduce drift, particularly in orientation, is radar-inertial fusion. These reduce drift by integrating gyroscope measurements into odometry pipelines. Both Burnett et al. [5] and Li et al. [6] use gyroscope readings to support registration-based odometry pipelines. Le Gentil et al. [7] use dense image correlation to calculate the Doppler velocity and the translation between scan and a submap. While achieving state-of-the-art results in radar-inertial odometry, their approach cannot maintain the same accuracy without gyroscope measurements. In general, methods using both the radar and an IMU require sensor calibration, time synchronization, and IMU bias handling, which complicates deployment.

Our work leverages CFEAR radar odometry [1] as the basis for odometry and mapping. We enhance preprocessing by compensating for Doppler distortion, correcting range bias, and utilizing antenna angle encoder measurements. Keyframe scans derived from odometry are stored in a pose-graph structure as a map for repeat-phase localization.

### B. Radar Localization

Radar localization operates at different scales, including place recognition to metric localization. Place recognition methods aim to identify previously visited areas without the need for sub-meter accuracy. Approaches for place recognition in spinning radar encompass handcrafted descriptors [16]–[19] or learning-based pipelines [20]–[25].

Metric localization, in contrast, seeks to align the live radar scan to a previously recorded map representation. Recent methods can be categorized into homogeneous radar-radar methods or heterogeneous radar-to-lidar approaches that make use of accurate lidar maps.

1) *Radar-to-lidar localization*: In radar-to-lidar localization, the map is generated using lidar point clouds. Yin et al. [26] propose an end-to-end localization framework that directly aligns raw radar measurements to pre-built lidar maps using a differentiable measurement model. Ma et al. [27] use scan projection to compare radar scans to lidar scans. After obtaining a rough initial guess, they perform ICP to get a more accurate transformation estimate. RaLF proposes the use of radar- and lidar feature encoders, a place recognition head, and a metric localization head to achieve global localization in a lidar map [28]. All of the aforementioned methods quantify their errors relative to a global reference frame. Burnett et al. [4] take a different approach to evaluate their setup: They implement their localization pipeline in a teach-and-repeat framework [29], which evaluates the localization with regard to the closest available submap of the map. In their work, they first compute radar odometry to create an initial guess for scan-to-map registration. Their framework employs a motion-compensated continuous-time radar-to-lidar ICP and is able to achieve translational accuracies in the range of 12 cm. In

<sup>1</sup>[www.github.com/TUM-PercInS/CFEAR-TR](http://www.github.com/TUM-PercInS/CFEAR-TR)

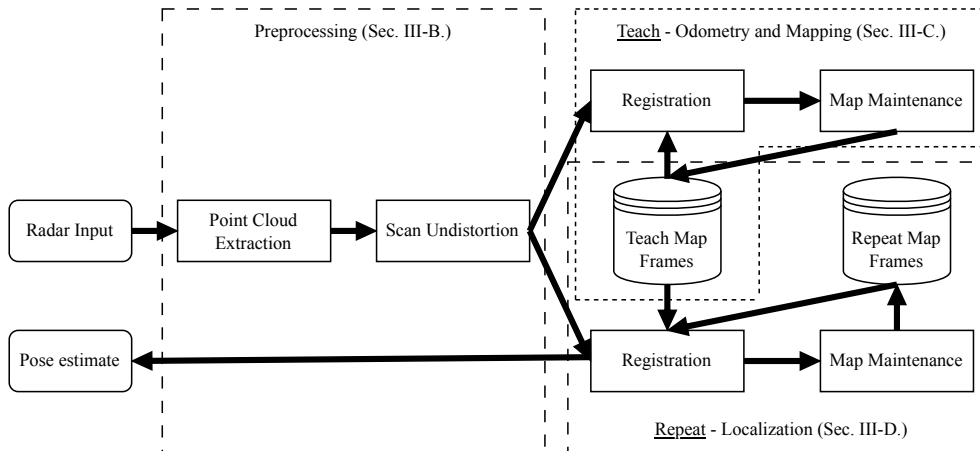


Fig. 2: Architecture of our teach-and-repeat localization framework. Incoming radar scans are filtered and undistorted. In the teach pass (odometry and mapping), the scan is registered to a set of recent keyframes. In the repeat pass (localization), the scan is jointly registered with both map frames from the teach-phase and recent scans from the repeat-phase.

a recent follow-up, Lisus et al. [30] improve radar-to-lidar registration using a learnt weight mask and a differentiable ICP implementation. While they do not employ registration in a full localization pipeline, they demonstrate improved accuracy in their scan matching.

2) *Radar-to-radar localization*: Less work exists in the area of radar-to-radar localization. Burnett et al.’s teach-and-repeat localization [4] is also implemented for radar-to-radar localization using continuous-time ICP. They achieve translational deviations in the scale of 12 cm, similar to radar-to-lidar localization. Recently, Zhang et al. [9] proposed SDRP, where they generate a monolithic radar map based on ground truth poses. They propose a denoising filter and employ Normal Distributions Transform (NDT)-based scan matching to localize the radar scans against the map. They achieve meter-scale localization errors using their framework.

Our work presented in this paper is most similar to Burnett et al.’s radar teach-and-repeat localization [4], but differs in two key aspects: We do not use a denser set of radar detections directly, but instead, make use of a sparser set of oriented surface points to retain efficiency. Additionally, we extend the formulation with a dual registration strategy that aligns both to map frames and recent live frames simultaneously, ensuring robustness under environmental change.

### III. METHODOLOGY

#### A. Radar Teach-and-Repeat Overview

Our pipeline follows the teach-and-repeat paradigm: In the *teach* pass, the robot traverses a previously unseen environment while generating a map. In the *repeat* pass, the robot localizes relative to this map in order to autonomously retrace the path [29]. The map is stored as a pose graph, where each node contains a keyframe of radar data in the form of oriented surface points.

CFEAR-Teach-and-Repeat builds on the CFEAR radar odometry framework [1]. To adapt it for teach-and-repeat

localization, we make two adjustments: First, we adapt the preprocessing by integrating Doppler compensation and using the sensor’s encoder readings to convert radar images to point clouds. While this is already done in other works [4], [5], [13], it was not considered in CFEAR previously. The inclusion ensures unbiased, deskewed point clouds. Also, we extend the odometry approach, introducing a dual registration strategy that jointly aligns live scans to (i) nearby map frames from the *teach* pass and (ii) a sliding window of recent live frames, obtained during the *repeat* pass. This aims to consistent with the taught path while preserving local consistency when the environment has changed. Before the repeat pass, an optional global refinement step using loop closure detection and pose graph optimization can further increase global consistency of map. Our pipeline can be subdivided into three distinct phases: radar preprocessing (Sec. III-B), odometry and mapping (Sec. III-C), and localization (Sec. III-D). An overview is given in Fig. 2

#### B. Radar Preprocessing

The radar preprocessing converts raw polar scans  $Z_{N_a \times N_r}$  with  $N_a$  azimuth bins and  $N_r$  range bins into Cartesian point clouds  $\mathcal{P}$ . Its purpose is to suppress noise, correct radar-specific distortions, and produce a compact representation suitable for registration.

1) *Peak extraction*: Typical radar noise sources include multipath reflections, speckle noise, and receiver saturation. To suppress these, we use the  $k$ -strongest filter [1], which, for each azimuth, selects the  $k$  returns with the highest intensity exceeding a threshold  $z_{min} = 60$ . Using only  $k$  returns filters out multipath and receiver saturation, while the intensity threshold aims to remove the speckle noise. The  $k$ -strongest filter has demonstrated excellent performance in various odometry estimation pipelines [1], [31]. The output of the filter is a set of reflections.

2) *Range correction and Cartesian conversion*: We observe two main effects, inhibiting the range readings of the

spinning radar: Doppler distortion and a static range offset. Doppler distortion is caused by the relative velocity during chirp acquisition and is increasingly influential with higher velocities. We compensate for the Doppler distortion using the velocity-dependent offset

$$\Delta r_d(\theta_a) = \beta(v_x \cos \theta_a + v_y \sin \theta_a), \quad (1)$$

where the correction factor  $\beta$  depends on the slope and mean of the radar's frequency ramp [13], and the velocity  $\mathbf{v} = [v_x \ v_y \ \omega]^T$  is estimated from the recent history using the constant velocity assumption. In contrast to previous implementations of CFEAR [1], we do not assume equally spaced azimuth angles in the radar image, but use the encoder reading of the antenna's angle during the azimuth acquisition  $\theta_a$ . The static range bias is corrected by an empirically tuned offset  $\Delta r_r = -0.31\text{m}$ . The resulting polar-to-cartesian mapping can then be expressed as

$$\tilde{\mathbf{p}} = \begin{bmatrix} \tilde{p}_x \\ \tilde{p}_y \end{bmatrix} = (d\gamma + \Delta r_d(\theta_a) + \Delta r_r) \begin{bmatrix} \cos \theta_a \\ \sin \theta_a \end{bmatrix}, \quad (2)$$

with  $d$  and  $a$  describing the range and azimuth indices and  $\gamma$  expressing the radar's resolution.

3) *Motion compensation*: Because the Navtech CIR radar spins at only 4Hz, intra-scan motion causes distortion. We perform motion compensation as described in CFEAR [1] to remove the motion distortion, assuming constant velocity between scans.

The output of the preprocessing is a deskewed, Doppler- and range-compensated point cloud suitable for mapping and localization. Compared to the original CFEAR [1], our preprocessing explicitly integrates Doppler compensation and uses the encoder readings of the antenna for the conversion to Cartesian space. While these techniques are used in other radar pipelines [13], these were not included in the previous evaluations of CFEAR, the impact of these improvements is discussed in Sec. IV.

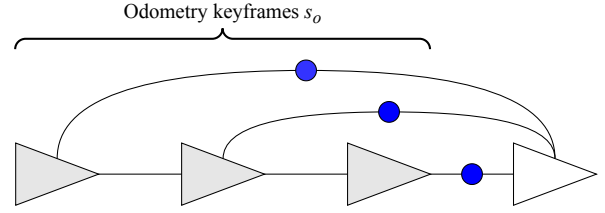
### C. Teach - Odometry and Mapping

The purpose of odometry and mapping in the *teach* pass is to provide (i) incremental pose estimates for building a compact map and (ii) a set of keyframes later used in localization. We build on CFEAR radar odometry [1], which represents scans as a sparse set of oriented surface points and performs scan-to-submap registration. The representation  $\mathcal{M}^l$  of each scan is aligned against a submap consisting of the most recent  $s_o$  keyframes  $\{\mathcal{M}^k\}$ . The registration cost  $f_{s2k}$  between the scan and a keyframe is given by

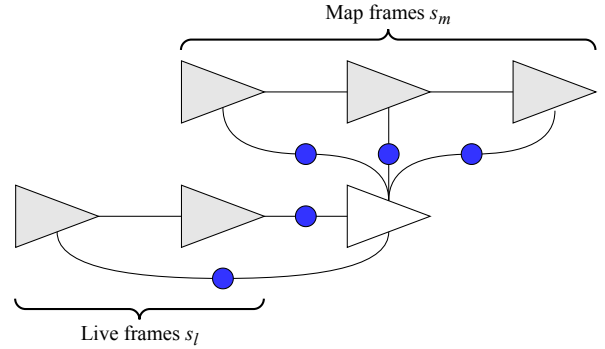
$$f_{s2k}(\mathcal{M}^k, \mathcal{M}^l, \mathbf{x}^l) = \sum_{\forall \{i,j\} \in \mathcal{C}} w_{i,j} \mathcal{L}(g_{p2d}(m_j^k, m_i^l, \mathbf{x}^l)), \quad (3)$$

where  $\mathcal{C}$  denotes the set of correspondences between surface points,  $w_{i,j}$  are weights scoring the similarity between surface normals,  $\mathcal{L}$  the Cauchy robust loss function [32], and  $g_{p2d}$  the point-to-distribution residual. For implementation details, we refer the reader to [1]. We obtain the pose estimate as

$$\mathbf{x}^l = \arg \min_{\mathbf{x}^l} \sum_{k=1}^{s_o} f_{s2k}(\mathcal{M}^k, \mathcal{M}^l, \mathbf{x}^l), \quad (4)$$



(a) *Teach* pass: Odometry graph using  $s_o = 3$  keyframes.



(b) *Repeat* pass: Localization graph using  $s_m = 3$  map frames and  $s_l = 2$  live frames.

Fig. 3: Optimization problems in (a) *teach* and (b) *repeat* phase, depicted as factor graphs. Grey triangles denote fixed scans; the white triangle is the optimized state. Blue dots show the scan-to-keyframe residuals.

solved using the Ceres nonlinear least-squares solver [33]. The optimization problem is depicted graphically in Fig. 3a.

As described in [1], keyframes are created whenever the robot moves more than 1.5 m or rotates more than  $5^\circ$ , ensuring sufficient geometric variation. During the *teach* pass, all keyframes are stored in a pose graph, with each node containing its pose estimate  $\mathbf{x}^m$  and oriented surface points  $\mathcal{M}^m$ . This pose graph forms the map used for *repeat* pass localization.

As the *repeat* stage uses multiple consecutive map keyframes, inaccurate transformations between keyframes due to odometry drift might inhibit localization performance. To reduce the drift between keyframes, we, optionally, allow for optimizing the pose graph using loop closure detection. This is done via the state-of-the-art introspective radar-only SLAM, TBV SLAM [2]. We evaluate whether this drift reduction on the global scale will improve localization if using multiple map keyframes in Sec. IV-E.

### D. Repeat - Localization in Prior Map

The objective of the trajectory estimation in the *repeat* pass is twofold: The estimated trajectory needs to (i) remain drift-free with respect to the map recorded in the *teach* pass, and (ii) accurately maintain local consistency when the environment has changed (e.g., due to seasonal or dynamic effects). To achieve this, we propose a dual registration approach. Each incoming scan  $\mathcal{M}^l$  is jointly registered against both map frames  $\{\mathcal{M}^m\}$  and recent live frames  $\{\mathcal{M}^l\}$ . The map

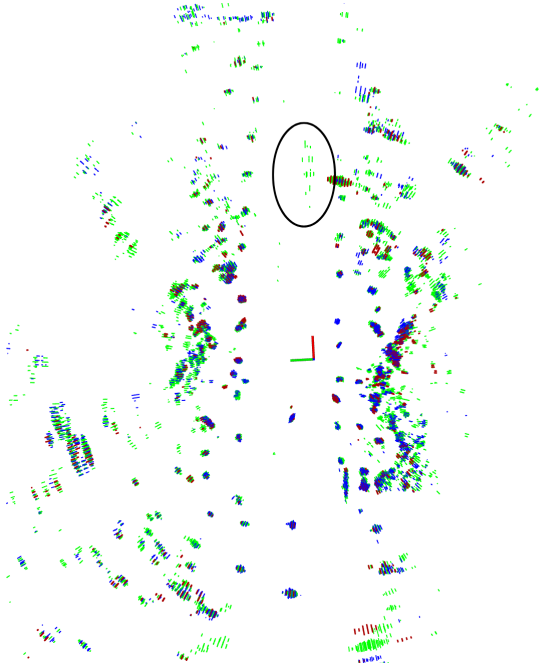


Fig. 4: The live robot scan (red) is registered with map frames (green) and recent live frames (blue). Despite the presence of a dynamic object in the map (circled), the localization successfully aligns with the static environment.

frames include the closest node in the pose graph of the *teach* pass and its neighbors, and anchor the trajectory to the taught path. The live frames are a sliding window of the most recent keyframes from the *repeat* pass, which preserve local consistency. Both the number of map frames  $s_m = |\{\mathcal{M}^m\}|$  and the number of live frames  $s_l = |\{\mathcal{M}^l\}|$  can be tuned by the user. An illustration of our formulation is presented in Fig. 3b.

The resulting cost function combines both terms:

$$\mathbf{x}^t = \arg \min_{\mathbf{x}^t} \underbrace{\sum_{m=1}^{s_m} f_{s2k}(\mathcal{M}^m, \mathcal{M}^t, \mathbf{x}^t)}_{\text{scan to map frames}} + \underbrace{\sum_{l=1}^{s_l} f_{s2k}(\mathcal{M}^l, \mathcal{M}^t, \mathbf{x}^t)}_{\text{scan to live frames}}, \quad (5)$$

where  $f_{s2k}$  denotes the robust scan-to-submap cost introduced in Eq. 3. This optimization is solved with a non-linear least squares solver. We visualize how the live point cloud is registered with the map frames and live frames in Fig. 4.

In summary, the dual registration formulation ensures that the repeat trajectory remains consistent with the taught map while adapting locally to environmental change.

#### IV. EVALUATION

We present our experiments to show the capabilities of our method. First, we present the experimental setup (cf. Sec. IV-A). Sec. IV-B evaluates the adapted odometry pipeline used during odometry and mapping, and compares it against relevant methods. In Sec. IV-C, we perform both *teach* and *repeat* phase experiments to investigate the importance of the presented techniques for undistorting radar scans. After that,

in Sec. IV-D, we investigate tuning of the newly introduced parameters. We evaluate the impact of pose graph optimization on localization performance in Sec. IV-E. Finally, in Sec. IV-F, we perform a comparative evaluation against the current state of the art in radar localization.

##### A. Experimental Setup

We evaluate our approach on the Boreas dataset [34], which was recorded in an automotive setting and contains multiple seasons, weather conditions, and lighting scenarios. The platform is equipped with a Navtech CIR 304-H spinning radar, a 128-beam Velodyne Alpha-Prime lidar, and an Applanix POS LV GNSS-INS. The radar has a maximum range of 200 m and an angular resolution of  $0.9^\circ$ . During the recording, the sensor's firmware was updated. Hence, in older sequences, the range resolution is  $\gamma = 0.0596\text{m}$ , while the range resolution of the newer sequences is a finer  $\gamma = 0.04381\text{m}$ . Ground truth trajectories are obtained from offline fusion of RTX-Global Navigation Satellite System (GNSS), Inertial Measurement Unit (IMU), and wheel odometry, with typical accuracy of 2 cm-4 cm.

For the teach pass, we select the sequence 2020-11-26, which is recorded with coarser resolution. We generate a map using radar odometry without the use of GNSS and IMU information. Our parameter settings for the mapping sequence are the same as those used in the entry to the radar in robotics competition [35], with the exception that we increase the number of peaks per azimuth to  $k = 40$  (instead of  $k = 12$ ).

In the repeat pass, we initialize the repeat path using the first ground truth pose. After that, we evaluate the relative transformation from the estimated live sensor pose to the sensor pose of the closest node in the teach pass. If not noted differently, we set the number of map frames in the registration to  $s_m = 5$  and the number of live frames to  $s_l = 3$ . Following the Boreas localization benchmark [34], we compute the Root Mean Squared Error (RMSE) of longitudinal displacements, lateral displacements, and the heading error. Furthermore, we keep track of the computational cost during the experiments. All experiments have been performed on a ThinkPad P16 with 64 GB of memory and an Intel i9-13980HX (5.6 GHz 32 cores) CPU, running a single thread. To save computation time, we perform up to 15 experiments simultaneously. In the following sections, we explain the different experiments we performed.

##### B. Odometry - Comparative Evaluation

We present an evaluation of CFEAR radar odometry with modifications presented in this paper. We follow the standard KITTI benchmark that computes drift between 100-800 m.

In the unseen test sequences of the Boreas dataset, we achieve a translation drift of 0.42%, and a rotation drift of  $0.136^\circ/100\text{m}$ . This cuts down the values reported in [35] by approximately 30% in both translation and rotation. Notably, we even outperform most gyro-aided pipelines in both translation and rotation, scoring second in both rotation and translational drift. Meanwhile, the odometry operates at

TABLE I: KITTI odometry benchmark metrics for various radar-only and gyroscope-aided approaches evaluated on test sequences from the Boreas dataset.. Translational error is given in %, rotational error in  $^{\circ}/100$  m. Best and second best results are **bold/underlined**.

| Method                 | radar | gyro. | accel. | transl.     | rot.         |
|------------------------|-------|-------|--------|-------------|--------------|
| VTR3 [4]               | ✓     |       |        | 1.97        | 0.565        |
| STEAM-RO [5]           | ✓     |       |        | 1.43        | 0.410        |
| CFEAR [35]             | ✓     |       |        | 0.61        | 0.206        |
| <b>CFEAR-TR (ours)</b> | ✓     |       |        | <b>0.42</b> | <b>0.136</b> |
| CFEAR++ [6]            | ✓     | ✓     |        | 0.51        | 0.140        |
| DRO-G [7]              | ✓     | ✓     |        | <b>0.26</b> | <b>0.049</b> |
| STEAM-RIO [5]          | ✓     | ✓     | ✓      | 0.95        | 0.266        |
| STEAM-RIO++ [5]        | ✓     | ✓     | ✓      | 0.62        | 0.175        |

TABLE II: KITTI odometry benchmark metrics on the training set, showing the effect of omitting different undistortion steps. Translational error is given in meter, and rotational error in  $^{\circ}/100$  m.

|                  | transl. error | rot. error |
|------------------|---------------|------------|
| full             | 0.46          | 0.14       |
| w/o doppler      | 0.47          | 0.15       |
| w/o range offset | 0.47          | 0.15       |
| w/o azimuth      | 0.49          | 0.16       |
| w/o all          | 0.62          | 0.19       |

a rate of 29.4 Hz, which allows execution seven times faster than the sensor rate. This highlights the effectiveness of the improvements. Detailed results are given in Tab. I.

### C. Ablation Study - Scan Undistortion

In CFEAR-TR, we added a range offset, Doppler compensation, and the use of encoder measurements for the radar’s azimuth (see Sec. III-B). In this experiment, we investigate how these techniques impact odometry and localization performance. For that, we perform an ablation study, where we compare leaving out each of the three techniques one by one. We compare the results to the full pipeline, and to a pipeline without all three adjustments, as described in [1]. We evaluate the teach-phase odometry on all training sequences, and store the keyframes of the mapping sequence 2020-11-26. The resulting odometry drift in KITTI metrics is given in Tab. II. We observe that range and Doppler offsets only moderately improve the trajectory drift. Using the azimuth measurements also leads to just minor improvements; however, the performance gap to the full pipeline is larger than for range and Doppler offsets. However, using all compensations together appears to have a major impact on the performance, which can be observed if comparing to the results without all adjustments.

We, furthermore, use the stored mapping sequence in the repeat phase to localize. The resulting metrics are given in Tab. III. Here, we see a different impact: Leaving out Doppler compensation increases the longitudinal error by 165 %. We observe that the increase in lateral error can almost exclusively be attributed to the Doppler distortion. Furthermore, assuming a uniform azimuth distribution leads to a three times higher heading estimation error. The range offset has comparatively only a small influence. Leaving

TABLE III: Localization errors on Boreas training sequences when different undistortion steps are omitted.

|                  | lateral [m] | longitudinal [m] | heading [ $^{\circ}$ ] |
|------------------|-------------|------------------|------------------------|
| full             | 0.053       | 0.080            | 0.085                  |
| w/o doppler      | 0.067       | 0.212            | 0.093                  |
| w/o range offset | 0.063       | 0.079            | 0.088                  |
| w/o azimuth      | 0.088       | 0.087            | 0.275                  |
| w/o all          | 0.099       | 0.213            | 0.277                  |

TABLE IV: Localization errors for different amounts of map frames  $s_m$  and live frames  $s_l$ . The RSME is given as (lateral [m] / longitudinal [m] / heading [ $^{\circ}$ ]). Best and second best results are **bold/underlined**.

|           | $s_m = 1$         | $s_m = 3$                | $s_m = 5$                |
|-----------|-------------------|--------------------------|--------------------------|
| $s_l = 1$ | 0.057/0.109/0.110 | 0.055/0.080/0.089        | <u>0.054/0.081/0.087</u> |
| $s_l = 2$ | 0.065/0.081/0.105 | <u>0.054/0.079/0.087</u> | <u>0.054/0.081/0.087</u> |
| $s_l = 3$ | 0.058/0.080/0.095 | <u>0.054/0.079/0.087</u> | <b>0.053/0.080/0.085</b> |
| $s_l = 4$ | 0.072/0.081/0.105 | 0.067/0.080/0.098        | <u>0.054/0.079/0.085</u> |
| $s_l = 5$ | 0.081/0.082/0.108 | <u>0.054/0.079/0.085</u> | <u>0.054/0.078/0.087</u> |

it out increases the lateral error by 1 cm. This highlights the importance of proper scan undistortion for localization, which aligns with the findings explained by Burnett et al. [13] with regards to Doppler compensation.

### D. Repeat-phase Registration - Parameter Study

Here, we present a parameter study of the registration component within the repeat phase. Specifically, we investigate the number of live scans  $s_l$  and the number of map frames  $s_m$  used in the registration. For that, we perform a grid search using  $s_m \in \{1, 3, 5\}$  and  $s_l \in \{1, 2, 3, 4, 5\}$ . We only test odd numbers for the number of map frames, as this ensures symmetric spacing around the closest node. Tab. IV displays the resulting localization errors. We observe that using a single map frame from the teach graph is not sufficient for accurate localization. However, adding more than three map frames does not yield better results. This hints that with three map frames already, sufficient environmental features are available and that additional frames do not further constrain the registration. Similarly, the number of live frames only has a modest influence on the results. However, the number of live frames should not be larger than the number of map frames. We can observe that in this case, errors can grow larger, which might happen due to overreliance on the drifting odometry. All in all, our parameter study shows that already with a low number of used scans (e.g.,  $s_m = 3$  and  $s_l = 1$ ), accurate overall performance is achieved. For the remaining experiments, we select the configuration with the lowest longitudinal and heading error ( $s_m = 5, s_l = 3$ ).

### E. Pose Graph Optimization in the Teach Pass

As shown in the previous section, using multiple keyframes from the teach pass as map frames improves localization accuracy. As these are susceptible to odometric drift, we analyze whether pose graph optimization can be used to alleviate the drift and improve localization further. For that, we optimize the map used in the previous experiments by applying TBV SLAM [2], and localize the live

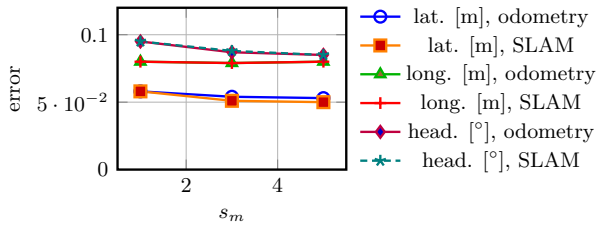


Fig. 5: Localization errors in a teach graph generated using odometry and in a graph optimized with SLAM, depending on the used number of map frames  $s_m$ . Lateral (lat.) error is reduced, Longitudinal (long.) and Heading (head.) are not influenced.

radar scans within the optimized pose graph. We evaluate our localization pipeline for varying sizes  $s_m \in \{1, 3, 5\}$  and plot the results in Fig. 5, compared to the results without pose graph optimization. We observe that the lateral localization error reduces slightly, while the longitudinal and heading errors remain at the same level. We hypothesize that the loop closure faces the same challenge with accurate registration along the longitudinal direction. In that case, longitudinal errors can hardly be alleviated by loop closure. For the improved lateral accuracy, we proceed with the remaining experiments using the optimized pose graph as a reference.

### F. Comparative Evaluation

Finally, we aim to evaluate the competitiveness of our approach compared to the state of the art. As we showed that pose graph optimization improves the localization performance, we chose to use it as well for the comparative evaluation. The results are shown in Tab. V, and we visualize the resulting trajectory and error distributions of an example sequence in Fig. 6 and Fig. 7, respectively.

Our approach outperforms the state of the art in both lateral positioning and heading estimation. Overall localization error was measured at 0.117 m, with a lateral error as low as 0.054 m. We observe a substantial improvement in heading estimation, cutting down the previously reported values by 63%. This is in line with the results in the odometry estimation (see Tab. I). We think that the oriented surface points, due to accumulating the points, are producing a smoother representation of the structures at the sides of the car compared to raw point clouds extracted directly from the radar scan. The smoother representation leads to better convergence in orientation. In longitudinal direction, we observe two small peaks in the error histogram given in Fig. 7 at  $-0.15$  m and  $-0.1$  m. We hypothesize that this is caused by incorrect registration results, sticking to vehicles next to the car that are moving at a similar speed. Future work can investigate this bias in more depth.

Computation-wise, our system can operate at a rate of 29 Hz on a single thread, which is many times faster than the sensor operating rate. This demonstrates that our system can operate in real-time with low resource consumption. Overall, due to its fast runtime and registration accuracy,

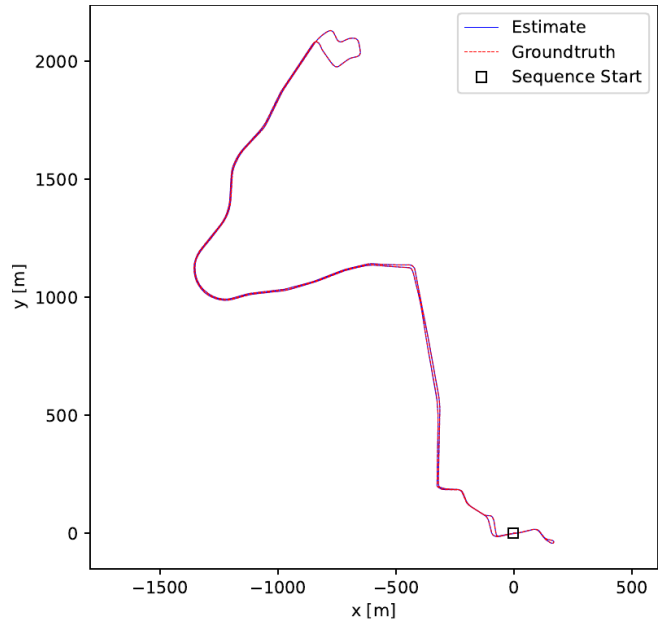


Fig. 6: Trajectory estimate of CFEAR-TR on the Boreas dataset, sequence 2021-03-09.

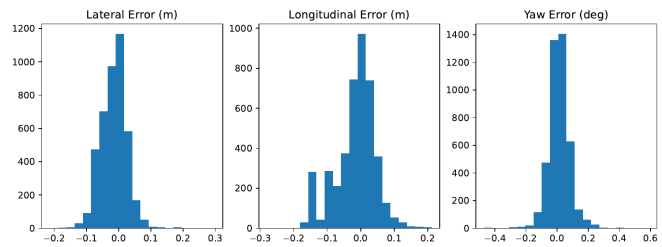


Fig. 7: Distribution of localization errors using CFEAR-TR on the Boreas dataset, sequence 2021-03-09.

TABLE V: Comparison with the state of the art approach VTR3 [4]. The RMSE is given as (lateral [m] / longitudinal [m] / heading [°]).

|                | CFEAR-TR (ours) |              |              | VTR3 [4]     |              |              |
|----------------|-----------------|--------------|--------------|--------------|--------------|--------------|
|                | lat.            | long.        | head.        | lat.         | long.        | head.        |
| 2020-12-04     | 0.070           | 0.088        | 0.088        | 0.072        | 0.082        | 0.211        |
| 2021-01-26     | 0.044           | 0.050        | 0.102        | 0.048        | 0.055        | 0.227        |
| 2021-02-09     | 0.063           | 0.056        | 0.105        | 0.053        | 0.051        | 0.235        |
| 2021-03-09     | 0.043           | 0.061        | 0.072        | 0.051        | 0.053        | 0.233        |
| 2021-06-29     | 0.052           | 0.094        | 0.083        | 0.069        | 0.095        | 0.246        |
| 2021-09-08     | 0.056           | 0.107        | 0.119        | 0.067        | 0.110        | 0.269        |
| 2021-10-05     | 0.052           | 0.119        | 0.093        | 0.069        | 0.109        | 0.288        |
| 2021-10-26     | 0.053           | 0.124        | 0.095        | 0.060        | 0.119        | 0.283        |
| 2021-11-06     | 0.055           | 0.157        | 0.093        | 0.062        | 0.155        | 0.256        |
| 2021-11-28     | 0.049           | 0.182        | 0.109        | 0.058        | 0.190        | 0.436        |
| <b>mean</b>    | <b>0.054</b>    | <b>0.104</b> | <b>0.096</b> | <b>0.061</b> | <b>0.102</b> | <b>0.268</b> |
| exec. time [s] | 0.034           |              |              | 0.075        |              |              |

we believe that CFEAR-TR forms an important step towards real-world deployment of radar-only localization.

## V. CONCLUSION AND FUTURE WORK

In this work, we presented CFEAR-TR, a fast and efficient radar-only teach-and-repeat localization system that builds on CFEAR radar odometry. Compared to the original formula-

tion, we enhanced preprocessing with Doppler compensation and encoder-based azimuth readings, yielding more consistent detection sets. Our localization further employs a dual registration strategy that jointly registers the current scan to nearby map frames from the teach pass and to a sliding window of recent live keyframes. Using pose graph optimization as a global refinement further increases the consistency of the map, as demonstrated by our experiments. Evaluation on the Boreas dataset indicates that the system achieves a localization accuracy as low as 0.117 m and 0.096°, corresponding to an improvement of 63% in heading estimation over the previous state of the art in radar-only localization. These results substantially narrow the gap to lidar-level localization. Future work aims to investigate whether Doppler-enabled spinning radar can be used to close the performance gap to lidar in the longitudinal direction, given its resilience in geometrically unconstrained environments. In addition, our method is computationally efficient, making it well-suited for radars with higher spin and measurement rates, which could further help close the gap.

#### REFERENCES

- [1] D. Adolfsson, M. Magnusson, A. Alhashimi, A. J. Lilienthal, and H. Andreasson, "Lidar-level localization with radar? the cfar approach to accurate, fast, and robust large-scale radar odometry in diverse environments," *IEEE Trans. on Robotics*, vol. 39, no. 2, pp. 1476–1495, 2023.
- [2] D. Adolfsson, M. Karlsson, V. Kubelka, M. Magnusson, and H. Andreasson, "Tbv radar slam – trust but verify loop candidates," *IEEE Robotics and Automation Lett. (RA-L)*, vol. 8, no. 6, pp. 3613–3620, 2023.
- [3] X. Qiao, A. Krawciw, S. Lilje, and T. D. Barfoot, "Radar teach and repeat: Architecture and initial field testing," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2025.
- [4] 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 Lett. (RA-L)*, vol. 7, no. 4, pp. 10328–10335, 2022.
- [5] 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*, 2024.
- [6] S. Li, Z. Hong, Y. Chen, L. Hu, and J. Qin, "Get it for free: Radar segmentation without expert labels and its application in odometry and localization," *IEEE Robotics and Automation Lett. (RA-L)*, 2025.
- [7] C. L. Gentil, L. Brizi, D. Lissus, X. Qiao, G. Grisetti, and T. D. Barfoot, "Dro: Doppler-aware direct radar odometry," in *Robotics: Science and System (RSS)*, 2025.
- [8] P.-C. Kung, C.-C. Wang, and W.-C. Lin, "A normal distribution transform-based radar odometry designed for scanning and automotive radars," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2021, pp. 14417–14423.
- [9] R. Zhang, Y. Zhang, D. Fu, and K. Liu, "Scan denoising and normal distribution transform for accurate radar odometry and positioning," *IEEE Robotics and Automation Lett. (RA-L)*, vol. 8, no. 3, pp. 1199–1206, 2023.
- [10] 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)*. PMLR, 2020, pp. 303–316.
- [11] 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)*. IEEE, 2020, pp. 2617–2623.
- [12] R. Weston, M. Gadd, D. De Martini, P. Newman, and I. Posner, "Fastbym: Leveraging translational invariance of the fourier transform for efficient and accurate radar odometry," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2022, pp. 2186–2192.
- [13] 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 Lett. (RA-L)*, vol. 6, no. 2, pp. 771–778, 2021.
- [14] Z. Hong, Y. Petillot, A. Wallace, and S. Wang, "Radarslam: A robust simultaneous localization and mapping system for all weather conditions," *Intl. J. of Robotics Research (IJRR)*, vol. 41, no. 5, pp. 519–542, 2022.
- [15] M. Hilger, N. Mandischer, and B. Corves, "Randt slam: Radar slam based on intensity-augmented normal distributions transform," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024, pp. 7831–7838.
- [16] 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*, vol. 38, no. 3, pp. 1856–1874, 2021.
- [17] 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)*. IEEE, 2023, pp. 11194–11201.
- [18] M. Gadd and P. Newman, "Open-radvlad: Fast and robust radar place recognition," in *Proc. of the IEEE Radar Conf. (RadarConf)*. IEEE, 2024, pp. 1–6.
- [19] H. Kim, B. Choi, E. Choi, and Y. Cho, "Referee: radar-based lightweight and robust localization using feature and free space," *IEEE Robotics and Automation Lett. (RA-L)*, 2024.
- [20] 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)*. IEEE, 2020, pp. 9484–9490.
- [21] M. Gadd, D. De Martini, and P. Newman, "Look around you: Sequence-based radar place recognition with learned rotational invariance," in *Proc. of the IEEE/ION Position, Location and Navigation Symposium (PLANS)*. IEEE, 2020, pp. 270–276.
- [22] Ş. Săftescu, M. Gadd, D. De Martini, D. Barnes, and P. Newman, "Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2020, pp. 4358–4364.
- [23] J. Komorowski, M. Wysoczanska, and T. Trzcinski, "Large-scale topological radar localization using learned descriptors," in *Proc. of the APNNS Intl. Conf. on Neural Information Processing (ICONIP)*. Springer, 2021, pp. 451–462.
- [24] W. Wang, P. P. de Gusmão, B. Yang, A. Markham, and N. Trigoni, "Radarloc: Learning to relocalize in fmcw radar," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2021, pp. 5809–5815.
- [25] J. Yuan, P. Newman, and M. Gadd, "Off the radar: Variational uncertainty-aware unsupervised radar place recognition for introspective querying and map maintenance," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.
- [26] H. Yin, R. Chen, Y. Wang, and R. Xiong, "Rall: end-to-end radar localization on lidar map using differentiable measurement model," *IEEE Trans. on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6737–6750, 2021.
- [27] Y. Ma, X. Zhao, H. Li, Y. Gu, X. Lang, and Y. Liu, "Rolm: Radar on lidar map localization," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2023, pp. 3976–3982.
- [28] A. Nayak, D. Cattaneo, and A. Valada, "Ralf: Flow-based global and metric radar localization in lidar maps," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2024, pp. 5097–5103.
- [29] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *J. of Field Robotics*, vol. 27, no. 5, pp. 534–560, 2010.
- [30] D. Lissus, J. Laconte, K. Burnett, Z. Zhang, and T. D. Barfoot, "Pointing the way: Refining radar-lidar localization using learned icp weights," in *Proc. of the Conf. on Robots and Vision*, 2025.
- [31] E. Preston-Krebs, D. Lissus, and T. D. Barfoot, "The Finer Points: A Systematic Comparison of Point-Cloud Extractors for Radar Odometry," *Proc. of the Conf. on Robots and Vision*, 2025.
- [32] M. Bosse, G. Agamennoni, I. Gilitschenski, et al., "Robust estimation and applications in robotics," *Foundations and Trends in Robotics*, vol. 4, no. 4, pp. 225–269, 2016.
- [33] S. Agarwal, K. Mierle, and T. C. S. Team, "Ceres Solver," 10 2023. [Online]. Available: <https://github.com/ceres-solver/ceres-solver>
- [34] K. Burnett, D. J. Yoon, Y. Wu, A. Z. Li, H. Zhang, S. Lu, J. Qian, W.-K. Tseng, A. Lambert, K. Y. Leung, et al., "Boreas: A multi-season autonomous driving dataset," *Intl. J. of Robotics Research (IJRR)*, vol. 42, no. 1–2, pp. 33–42, 2023.
- [35] D. Adolfsson and M. Hilger, "An evaluation of cfar radar odometry," *arXiv preprint arXiv:2404.01781*, 2024.