

# Doppler-SLAM: Doppler-Aided Radar-Inertial and LiDAR-Inertial Simultaneous Localization and Mapping

Dong Wang<sup>ID</sup>, *Graduate Student Member, IEEE*, Hannes Haag,

Daniel Casado Herraez<sup>ID</sup>, *Graduate Student Member, IEEE*, Stefan May, Cyrill Stachniss<sup>ID</sup>, *Member, IEEE*, and Andreas Nüchter<sup>ID</sup>, *Senior Member, IEEE*

**Abstract**—Simultaneous localization and mapping is a critical capability for autonomous systems. Traditional SLAM approaches often rely on visual or LiDAR sensors and face significant challenges in adverse conditions such as low light or featureless environments. To overcome these limitations, we propose a novel Doppler-aided radar-inertial and LiDAR-inertial SLAM framework that leverages the complementary strengths of 4D radar, FMCW LiDAR, and inertial measurement units. Our system integrates Doppler velocity measurements and spatial data into a tightly-coupled front-end and graph optimization back-end to provide enhanced ego velocity estimation, accurate odometry, and robust mapping. We also introduce a Doppler-based scan-matching technique to improve front-end odometry in dynamic environments. In addition, our framework incorporates an innovative online extrinsic calibration mechanism, utilizing Doppler velocity and loop closure to dynamically maintain sensor alignment. Extensive evaluations on both public and proprietary datasets show that our system significantly outperforms state-of-the-art radar-SLAM and LiDAR-SLAM frameworks in terms of accuracy and robustness.

**Index Terms**—Odometry, Mapping, Localization, SLAM, Autonomous Vehicle Navigation.

Received 14 April 2025; accepted 20 July 2025. Date of publication 4 August 2025; date of current version 12 August 2025. This article was recommended for publication by Associate Editor M. Camurri and Editor J. Civera upon evaluation of the reviewers' comments. This work was supported by the Federal Ministry for Economic Affairs and Climate Action (BMWK) on the basis of a decision by the German Bundestag under Grant KK5150106RL4. (*Corresponding author: Dong Wang.*)

Dong Wang is with the Julius-Maximilians-Universität Würzburg, 97070 Würzburg, Germany (e-mail: dong.wang@uni-wuerzburg.de).

Hannes Haag and Stefan May are with the Nuremberg Institute of Technology Georg Simon Ohm, 90489 Nürnberg, Germany.

Daniel Casado Herraez is with the CARIAD SE and the University of Bonn, 53113 Bonn, Germany.

Cyrill Stachniss is with the Center for Robotics, University of Bonn, 53113 Bonn, Germany, and also with the Lamarr Institute for Machine Learning and Artificial Intelligence, 44227 Dortmund, Germany.

Andreas Nüchter is with the Julius-Maximilians-Universität Würzburg, 97070 Würzburg, Germany, and also with the Zentrum für Telematik e.V., Würzburg and currently International Visiting Chair at U2IS, ENSTA, Institut Polytechnique de Paris 91762 Palaiseau Cedex, France.

To encourage further research, the code of our Doppler-SLAM and our dataset are available at: <https://github.com/Wayne-DWA/Doppler-SLAM>.

This article has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2025.3595071>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3595071

## I. INTRODUCTION

**I**N THE pursuit of robust and reliable navigation solutions, simultaneous localization and mapping (SLAM) has emerged as a cornerstone technology that enables autonomous systems to perceive and interpret their environment while estimating their own position. Traditional SLAM methods often rely on visual or LiDAR sensors. Those modalities, however, can be susceptible to poor lighting, extreme weather, or featureless terrain. SLAM using radar sensing has the potential to increase robustness to environmental variability and operate effectively under challenging conditions.

In recent years, integrating radar and LiDAR sensors with inertial measurement units (IMUs) has improved the accuracy and reliability of SLAM systems [1], [2], [3]. Doppler velocity from 4D radars and frequency-modulated continuous wave (FMCW) LiDARs provides direct motion information that, when fused with inertial data, enhances motion estimation and mapping [4], [5]. However, reliable SLAM in complex or dynamic environments remains challenging, and most methods either focus on a single sensor type or fail to fully exploit Doppler information in a unified framework.

This paper presents a novel approach to Doppler-aided radar-inertial and FMCW LiDAR-inertial SLAM, leveraging the complementary capabilities of radar, FMCW LiDAR, and IMU. By incorporating Doppler velocity measurements and spatial data into the SLAM framework, we aim to achieve enhanced odometry estimation and a more robust mapping process. The proposed methodology is designed to operate in complex and dynamic environments, offering a reliable solution for ground vehicles.

The main contribution of this paper is a novel, unified SLAM approach that combines a tightly-coupled front-end [1] with a graph optimization back-end [2], seamlessly integrating IMU, radar or FMCW LiDAR, and Doppler velocity measurements. Additionally, we propose an innovative online extrinsic calibration mechanism between radar-IMU or FMCW LiDAR-IMU, aided by Doppler velocity and loop closure [6], to ensure consistent sensor alignment during operation, and a novel Doppler-based scan-matching method for front-end odometry, significantly improving accuracy in dynamic scenarios. Our proposed SLAM system is thoroughly evaluated on various open-source datasets alongside our dataset. The results significantly outperform current state-of-the-art radar-SLAM

and FMCW LiDAR-SLAM frameworks. Finally, we make our Doppler-SLAM system open-source to foster further research and development within the community.

## II. RELATED WORK

In this section, we review state-of-the-art SLAM approaches based on LiDAR and radar, including traditional frameworks and recent methods that leverage Doppler information. We also discuss techniques that integrate these sensors with IMU measurements and present our proposed method in the context of Doppler-aided SLAM for radar and FMCW LiDAR.

**LiDAR-based Odometry and SLAM** have significantly evolved with the development of various registration and optimization techniques. One of the fundamental methods for point cloud registration is the iterative closest point (ICP) algorithm, which aligns 3D shapes by minimizing point-to-point distance [7]. Generalized-ICP (GICP) [8] extends this by combining point-to-plane and point-to-point constraints, improving registration robustness. Building on these foundations, LOAM introduces a two-thread approach where one thread estimates motion in real-time while another refines the map [9]. KISS-ICP has recently been proposed as a point-to-point scan-to-map matching and keyframe-based LiDAR odometry technique that leads to high accuracy while maintaining computational efficiency [10]. However, LiDAR-only approaches struggle with featureless environments and motion distortion in highly dynamic platforms. The integration of an IMU improves the accuracy and robustness of LiDAR odometry by providing a reliable scan distortion and an initial pose for ICP. In addition, the high-frequency IMU measurements help to correct for motion distortion within LiDAR scans, which is particularly beneficial in dynamic environments where motion blur degrades scan quality. Tightly-coupling LiDAR and IMU data has also been explored through direct LiDAR-inertial fusion methods. Methods like tightly-coupled 3D LiDAR-inertial odometry [11] and LINS [12] demonstrate improved state estimation accuracy through real-time optimization techniques. Graph-based approaches such as LIO-SAM also integrate LiDAR and inertial data for more accurate and globally consistent odometry [2]. To enhance computational efficiency and real-time performance, FAST-LIO [13] introduces a tightly-coupled iterated Kalman filter framework for robust LiDAR-inertial odometry, which is later improved with FAST-LIO2 by reducing computational complexity while maintaining accuracy [1]. Recent improvements in FMCW technology have paved the way for a novel iteration of LiDAR, namely FMCW-LiDAR, which has the additional capability of measuring the relative radial velocity (Doppler velocity) of each point [14], [15]. Doppler iterative closest point extends ICP by leveraging Doppler information to improve robustness in high-speed scenarios [16]. FMCW-LIO [17] and Doppler-Odom [18] incorporate Doppler LiDAR measurements with IMU to refine motion estimation and mitigate drift in dynamic environments.

**4D-radar-based Odometry and SLAM** have gained remarkable attention due to their robustness in adverse environmental conditions, such as fog, rain, and low-light scenarios. Several approaches leverage radar-inertial fusion, probabilistic estimation techniques, and deep learning-based feature extraction to enhance odometry performance. Since 4D radar is capable of estimating the 3D ego velocity from a single scan [19], [20], radar-only SLAM approaches can utilize the estimation of ego velocity

to increase the accuracy of scan-matching [21], [22], [23]. Casado Herraez et al. [24] propose a point-to-point ICP with Doppler velocity constraint technique specifically designed to harness the velocity information provided by radar sensors. Their approach has recently been extended to integrate IMU information, enabling fusion and optimization through global and local factor graphs [25]. A scan-to-submap Normal distribution transform is presented for radar point cloud registration, while velocity pre-integration is used to improve optimization performance [26]. Zhang et al. [3] proposes an adaptive probability distribution-GICP to address radar measurement noise, considering the spatial probability distribution of each point in GICP [8]. Another approach to improving the matching quality of sparse radar data is 4DIRIOM [5]. Here, point matching is expressed in terms of distribution-to-multiple-distribution constraints, which is achieved by matching the current scan with a sub-map constructed by the mapping module, rather than scan-to-scan matching. Huang et al. [27] leverage the radar cross section information to refine the point-to-point correspondence, thus improving the estimation of poses based on radar point matching. However, radar-only SLAM methods often struggle with low spatial resolution and cluttered environments, making robust feature extraction and scan-matching challenging. Integrating 4D radar data with inertial measurements has been shown to enhance odometry accuracy and robustness. Tightly-coupled radar-inertial odometry methods, such as DGRO [28], DRIO [29], and multi-state EKF-based radar-inertial odometry [30], integrate Doppler velocity measurements and IMU readings to provide accurate motion estimation. These methods leverage persistent landmarks and extended Kalman filtering to reduce drift in odometry estimation. Additionally, a tightly coupled factor graph formulation for radar inertial odometry [31] has been proposed to enhance global consistency through optimization-based approaches. Despite their advantages, radar-inertial SLAM techniques remain susceptible to sensor noise and require accurate calibration to ensure robustness. Recent studies in deep learning and probabilistic estimation have led to novel radar odometry techniques. Zhou et al. [32] leverage deep neural networks to extract robust features from radar scans for odometry estimation. Additionally, methods such as Auto-Place [33] and SPR [34] focus on extracting point-wise features and generating scene descriptors to improve place recognition. However, these methods rely on large training datasets and may generalize poorly to unseen environments, limiting their adaptability in dynamic conditions.

Inspired by previous works [1], [5], [16], [17], we propose our Doppler-SLAM approach, which significantly enhances existing methods by unifying 4D radar-inertial and FMCW LiDAR-inertial SLAM into a single framework that directly incorporates Doppler velocity for distortion correction, scan-matching and online calibration, enabling robust SLAM in dynamic environments.

## III. DOPPLER-SLAM

To represent the various mathematical and physical quantities used in our research, we use the following conventions in this paper. Scalars will be printed as lowercase, non-bold letters (e.g.,  $b$ ), and constants will be printed as uppercase, non-bold letters (e.g.,  $B$ ). Matrices will be printed as bold upper case letters, like  $\mathbf{B}$ . Vectors will be represented by bold lowercase letters,

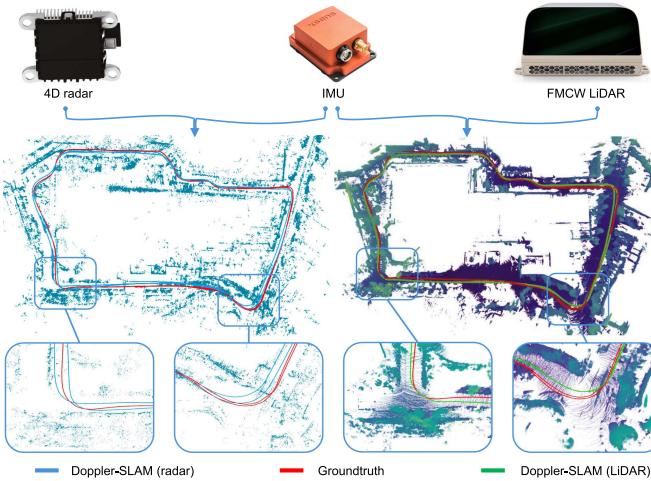


Fig. 1. Generalizability of our proposed Doppler-SLAM on the HeRCULES dataset “Sports Complex”. Left: radar map and trajectory (blue) generated with Doppler-SLAM. Right: FMCW LiDAR map and trajectory (green) generated with Doppler-SLAM.

like  $\mathbf{b}$ . Subscripts and superscripts are used to denote different frames of reference. For example, a vector  $\mathbf{b}$  in the radar frame  $\{\cdot\}^r$  will be denoted as  $\mathbf{b}^r$ , and the rotation from frame  $\{\cdot\}^r$  to frame  $\{\cdot\}^w$  will be represented by either the matrix  $\mathbf{B}_r^w$  or the quaternion  $\mathbf{b}_r^w$ . The global world frame is represented by  $\{\cdot\}^W$ . To simplify the exposition, the subsequent references to radar in this paper refer to 4D radar. Similarly, FMCW-LiDAR with Doppler measurements is abbreviated as LiDAR .

### A. Framework Overview

Fig. 2 illustrates the overall system architecture, highlighting four key modules: (i) velocity filter, (ii) motion compensation, (iii) state estimation, and (iv) online calibration with graph optimization. The following subsections describe the design and implementation of each module in detail.

### B. Front-End

1) *System Input and State:* The primary input of our system is provided by an IMU and a LiDAR or a radar sensor. The measurement  $\mathbf{u}$  from an IMU is defined as (1):

$$\begin{aligned}\mathbf{u} &\triangleq [\hat{\omega}_t \quad \hat{\mathbf{a}}_t], \quad \hat{\omega}_t = \omega_t + \mathbf{b}_\omega + \mathbf{n}_\omega, \\ \hat{\mathbf{a}}_t &= \mathbf{R}_t^{WB}(\mathbf{a}_t - \mathbf{g}) + \mathbf{b}_a + \mathbf{n}_a,\end{aligned}\quad (1)$$

where  $\hat{\omega}_t$  and  $\hat{\mathbf{a}}_t$  are the raw IMU measurements in IMU frame  $\{\cdot\}^B$  at time  $t$ .  $\hat{\omega}_t$  and  $\hat{\mathbf{a}}_t$  are influenced by the slowly varying bias  $\mathbf{b}$  and white noise  $\mathbf{n}$ .  $\mathbf{R}_t^{WB}$  is the rotation matrix from world frame  $\{\cdot\}^W$  to IMU (body) frame  $\{\cdot\}^B$ . The term  $\mathbf{g}$  refers to the constant gravity vector in the world frame. Although the measurement principles of 4D radar and FMCW-LiDAR are somewhat different, their output data is reformulated into a unified representation. Let  $i$  denote the index of radar or LiDAR scans and let the points  $\hat{\mathbf{m}}$  in the scan be represented as  $\mathbf{s}_i^S = \{\hat{\mathbf{m}}_0, \hat{\mathbf{m}}_1, \hat{\mathbf{m}}_2, \dots\}$ , which are sampled at the local radar or LiDAR coordinate frame  $\{\cdot\}^S$  at the end of the scan. Each point  $\hat{\mathbf{m}}_j^S$  provides three-dimensional geometric point  $[x, y, z]$  and radial Doppler velocity  $v_j^S$  along the point’s direction. The measured point  $\hat{\mathbf{m}}_j^S$  is typically affected by noise terms  $\mathbf{n}^S$

and  $\eta^S$ , which account for geometric noise and velocity noise, respectively. Eliminating this noise recovers the true location and Doppler velocity of the point in the local sensor coordinate frame.

$$\hat{\mathbf{m}}_j^S = \mathbf{m}_j^S + \mathbf{n}_j^S, \quad \hat{v}_j^S = v_j^S + \eta_j^S. \quad (2)$$

The system state  $\mathbf{x}$  evolves on a 24-dimensional manifold and comprises the body frame’s rotation  $\mathbf{R}_b$ , position  $\mathbf{p}_b$ , and  $\mathbf{v}_b$  relative to the world frame (i.e., the initial body frame), the gyroscope and accelerometer bias  $\mathbf{b}_g$  and  $\mathbf{b}_a$ , as well as the radar-IMU or LiDAR-IMU extrinsic parameters  $\mathbf{R}_{sb}$  and  $\mathbf{p}_{sb}$ :

$$\mathbf{x} \triangleq [\mathbf{R}_b^\top \quad \mathbf{p}_b^\top \quad \mathbf{R}_{sb}^\top \quad \mathbf{p}_{sb}^\top \quad \mathbf{v}_b^\top \quad \mathbf{b}_g^\top \quad \mathbf{b}_a^\top \quad \mathbf{g}^\top]^\top. \quad (3)$$

2) *Velocity Filter:* We propose a velocity filter module that fuses Doppler velocities and IMU measurements to distinguish between dynamic and static points and effectively eliminate outliers. Assume that the optimal state estimate after fusing the last sensor scan is  $\mathbf{x}_i$  with the covariance matrix  $\xi_i$ . As proposed in FAST-LIO2 [1], forward propagation starts when an IMU measurement is received and stops upon receiving a new sensor scan. The continuous model is discretized at the IMU sampling period [35] based on the operation  $\boxplus$  and the derivative of the discrete model  $\mathbf{f}$  defined in FAST-LIO [13]. Let  $\Delta t$  denote the sampling period between two consecutive IMU measurements and  $w$  represent the noise. Then, the predicted state from the IMU is formulated as follows:

$$\begin{aligned}\mathbf{x}_{i+1} &= \mathbf{x}_i \boxplus (\mathbf{f}(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i) \Delta t) \\ \mathbf{w}_i &\triangleq [\mathbf{n}_\omega^\top \quad \mathbf{n}_a^\top \quad \mathbf{n}_{b\omega}^\top \quad \mathbf{n}_{ba}^\top]^\top.\end{aligned}\quad (4)$$

At any given moment, the ego velocity is represented by  $\mathbf{v}^S$ . The measured Doppler velocity  $v_j^S$  from each target is considered as taking the magnitude of the projection of the relative velocity vector between the target and the sensor onto the ray connecting the target and the sensor. This calculation involves the dot product of the target’s velocity  $\mathbf{v}^S$  in the sensor frame  $\{\cdot\}^S$  and the unit vector pointing from the sensor towards the target:

$$-v_j^S = \frac{\mathbf{m}^S}{\|\mathbf{m}^S\|} \cdot \mathbf{v}^S = \mathbf{r}^S \cdot \mathbf{v}^S = r_x^S v_x^S + r_y^S v_y^S + r_z^S v_z^S. \quad (5)$$

Considering the rigid body transformation detailed in [5], the velocity  $\mathbf{v}_b$  obtained from IMU forward propagation is transformed into the LiDAR or radar coordinate system  $\{\cdot\}^S$  by:

$$\hat{\mathbf{v}}^S = \mathbf{R}_{sb}^\top (\mathbf{R}^\top \mathbf{v}_b + (\omega_t - \mathbf{b}_\omega) \times \mathbf{p}_{sb}^\top). \quad (6)$$

By substituting (6) into (5), we obtain the predicted Doppler velocity for each point:

$$\hat{v}_j^S = \mathbf{r}^S \cdot \hat{\mathbf{v}}^S = r_x^S \hat{v}_x^S + r_y^S \hat{v}_y^S + r_z^S \hat{v}_z^S. \quad (7)$$

Introducing a predetermined threshold  $\Upsilon$  to mitigate low-amplitude fluctuations caused by sensor noise, the velocity filter is defined by the condition  $|\hat{v}_j^S - v_j^S| \leq \Upsilon$ . This formulation ensures that a point  $\hat{\mathbf{m}}_j^S$  is considered valid only if the discrepancy between its predicted Doppler velocity  $\hat{v}_j^S$  and measured Doppler velocity  $v_j^S$  is below  $\Upsilon$ , effectively filtering out dynamic outliers and ghost points. A key advantage of our proposed velocity filter over traditional least-squares methods [36] is that

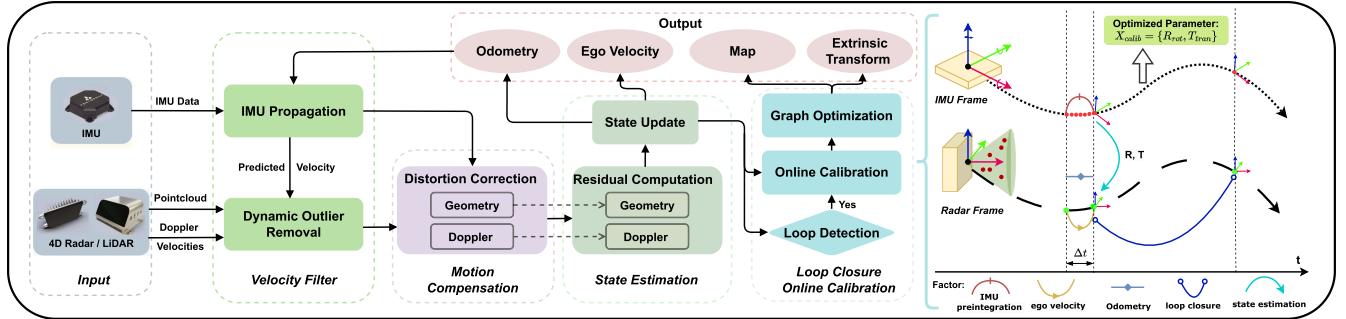


Fig. 2. Pipeline of Doppler-SLAM consists of four main modules: velocity filter (Section III-B2), motion compensation (Section III-B3), state estimation (Section III-B4), and loop closure with graph optimization (Section III-C). The graph on the right illustrates the workflow of online extrinsic calibration between the IMU and either radar or LiDAR using graph optimization. In this approach, we combine the IMU pre-integration factor, odometry factor, and ego velocity factor to construct a factor graph. Once a loop closure factor is detected, additional optimization refines the extrinsic transformation.

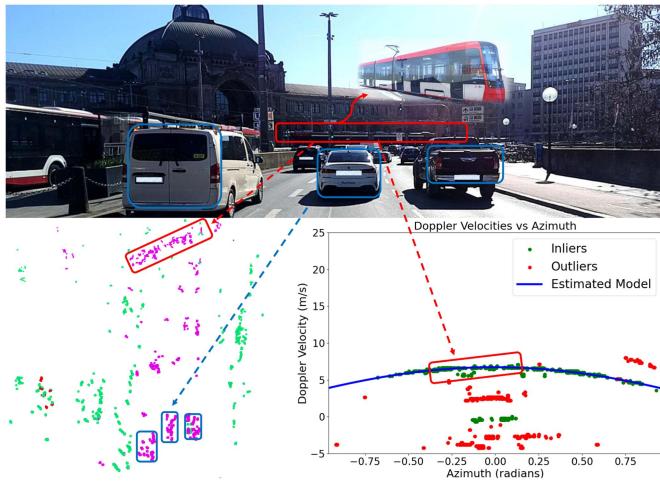


Fig. 3. Velocity Filter in a highly dynamic scenario with a moving tram. The top panel presents the camera view, the left panel shows the radar point cloud after processing by our proposed velocity filter, where purple indicates dynamic points and green indicates static points, and the right panel illustrates the traditional least-squares method, in which green points are inliers (static objects) detected by the method and red points are outliers (dynamic objects). The least-squares method relies on the Doppler velocities of all inliers to fit the ego velocity profile (blue curve) but struggles in highly dynamic environments because it incorrectly incorporates Doppler measurements from moving objects (tram). In contrast, our IMU-based velocity filter effectively distinguishes between dynamic and static points, yielding more accurate ego velocity estimates and robust performance in complex, real-world scenarios.

it does not require the assumption that most targets in the environment are stationary, which is often an unrealistic assumption, as illustrated in Fig. 3. This benefit is especially evident in highly dynamic outdoor environments, where least-squares approaches often fail.

3) *Motion Compensation:* To mitigate motion-induced distortion, our method performs a two-stage compensation of the LiDAR measurements. For radar, there is no motion distortion since its data acquisition method captures point cloud and Doppler velocities instantaneously, effectively bypassing the temporal distortions inherent in sequential LiDAR scanning.

*Geometry Compensation:* For a point  $\hat{\mathbf{m}}_j^S$  sampled at timestamp  $t_i^j$  in the scan  $s^S$  with scan-end time  $t_i$ , we adopt the backward propagation method from FAST-LIO [13] to compensate for the geometric motion distortion. Equation (4) is backward

propagated as  $\hat{\mathbf{x}}_{i-1} = \hat{\mathbf{x}}_i \boxplus (-\mathbf{f}(\hat{\mathbf{x}}_i, \mathbf{u}_i, 0)\Delta t)$ . The earlier IMU measurement is used as the input for all points sampled between two consecutive IMU measurements. Subsequently, we use the relative pose between  $t_i^j$  and  $t_i$  to transform the local measurement  $\hat{\mathbf{m}}_j^S$  into its corresponding scan-end measurement. In this way, the transformed points in the scan are considered to have all been scanned simultaneously at the scan-end time  $t_i$ .

*Doppler Compensation:* The measured Doppler velocity  $v_i^S$  for each point is influenced by both the target's motion and the sensor's motion. Dynamic targets are first filtered out by the velocity filter. Inspired by FMCW-LIO [17], we remove the sensor's motion over a scan period by subtracting its projected velocity change from the measured Doppler value. The relative velocity between  $t_i^j$  and  $t_i$  is also calculated by the backward propagation with IMU measurements.

4) *State Estimation:* To estimate the state vector given in (3), we employ an iterated extended Kalman filter (IEKF). The IEKF iteratively linearizes the nonlinear system around the most recent state estimate, thereby refining the estimate and enhancing the overall accuracy of the state estimation process. Keeping the first-order terms from (4) and setting the noise term  $w$  to zero, the error state  $\delta \mathbf{x}$  and the covariance  $\hat{\mathbf{P}}$  evolve according to the following linear model:

$$\begin{aligned}\delta \mathbf{x}_{i+1} &= \mathbf{F}_{\delta \mathbf{x}_i} \delta \mathbf{x}_i + \mathbf{F}_{w_i} \mathbf{w}_i, \\ \hat{\mathbf{P}}_{i+1} &= \mathbf{F}_{\delta \mathbf{x}_i} \hat{\mathbf{P}}_i \mathbf{F}_{\delta \mathbf{x}_i}^\top + \mathbf{F}_{w_i} \mathbf{Q}_i \mathbf{F}_{w_i}^\top.\end{aligned}\quad (8)$$

Here,  $\mathbf{F}_{\delta \mathbf{x}_i}$  and  $\mathbf{F}_{w_i}$  denote the transition matrix and noise Jacobian matrix, respectively, both linearized at  $\hat{\mathbf{x}}_i$  [17]. The noise covariance  $\mathbf{Q}_i$  is obtained from IMU calibration. Assuming that the system state and covariance are denoted as  $\hat{\mathbf{x}}_i$  and  $\hat{\mathbf{P}}_i$  when a new scan of radar or LIDAR arrives, the iteration of the system state is described below.

*Geometry Residual:* We first project each measured point  $\hat{\mathbf{m}}_j^S$  in the new scan to the global frame  $\hat{\mathbf{m}}_j^W = \hat{\mathbf{T}}_{wb} \hat{\mathbf{T}}_{bs} (\mathbf{m}_j^S + \mathbf{n}_j^S)$ , where  $\hat{\mathbf{T}}_{wb}$  consisting of  $\mathbf{R}_b$  and  $\mathbf{p}_b$  represents the body frame's pose relative to the world frame and  $\hat{\mathbf{T}}_{bs} = \text{inv}(\hat{\mathbf{T}}_{sb})$  represents radar-IMU or LiDAR-IMU extrinsic transform, which are all contained in the predicted state  $\hat{\mathbf{x}}_i$  from (8). The five nearest neighbors of the transformed point  $\hat{\mathbf{m}}_j^W$  are selected in the map using the *ikd-Tree* [1]. These neighboring points are then utilized to fit a local planar patch characterized by the normal vector  $\mathbf{u}_j$

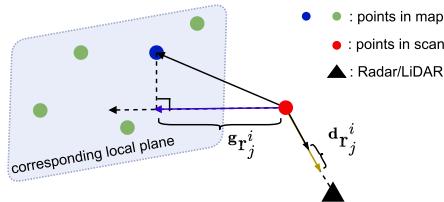


Fig. 4. Geometry and Doppler Residual.

and centroid  $\mathbf{q}_j^W$ . Ideally,  $\hat{\mathbf{m}}_j^W$  should lie exactly in the fitted plane. This leads to the following equation:

$$\mathbf{0} = \mathbf{u}_j^\top (\hat{\mathbf{T}}_{wb} \hat{\mathbf{T}}_{bs} (\mathbf{m}_j^S + \mathbf{n}_j^S) - \mathbf{q}_j^W). \quad (9)$$

Summarizing (9) into a compact form and linearizing the measurement using a first-order Taylor expansion about  $\hat{\mathbf{x}}_i$  yields the following simplified form:

$$\begin{aligned} \mathbf{0} &= {}^g\mathbf{h}_j(\mathbf{x}_i, \mathbf{m}_j^S + \mathbf{n}_j^S) \simeq {}^g\mathbf{h}_j(\hat{\mathbf{x}}_i, 0) + {}^g\mathbf{H}_j^i \tilde{\mathbf{x}}_i + \mathbf{n}_i, \\ \mathbf{0} &= {}^g\mathbf{r}_j^i + {}^g\mathbf{H}_j^i \tilde{\mathbf{x}}_i + \mathbf{n}_i, \\ {}^g\mathbf{r}_j^i &= \mathbf{u}_j^T (\hat{\mathbf{T}}_{wb} \hat{\mathbf{T}}_{bs} (\mathbf{m}_j^S + \mathbf{n}_j^S) - \mathbf{q}_j^W), \end{aligned} \quad (10)$$

where  $\mathbf{x}_i = \hat{\mathbf{x}}_i \boxplus \tilde{\mathbf{x}}_i$ ,  ${}^g\mathbf{H}_j^i$  denotes the Jacobian matrix of geometry measurement function  ${}^g\mathbf{h}_j$  with respect to error state  $\tilde{\mathbf{x}}_i$ . Furthermore,  $\mathbf{n}_i$  models the raw measurement noise associated with  $\mathbf{n}^S$  and  ${}^g\mathbf{r}_j^i$  is defined as the geometry residual.

*Doppler Residual:* One of the key innovations of this paper is our dual matching strategy, which leverages both the observed 3D geometry and the Doppler velocity residuals for each point, as shown in Fig. 4. The geometric residuals ensure precise alignment of the 3D point cloud, while the Doppler velocity residuals provide critical information about the motion state, enhancing overall matching accuracy. From (6) and (7), we easily get the Doppler residual  ${}^d\mathbf{r}_j^i$  for each point  $\mathbf{m}_j^S$  in the new scan:

$$\begin{aligned} \mathbf{0} &= \sigma_i(\hat{v}_j^S - v_j^S) \simeq {}^d\mathbf{h}_j(\hat{\mathbf{x}}_i, 0) + {}^d\mathbf{H}_j^i \tilde{\mathbf{x}}_i + \boldsymbol{\eta}_i, \\ {}^d\mathbf{r}_j^i &= {}^d\mathbf{H}_j^i \tilde{\mathbf{x}}_i + \boldsymbol{\eta}_i \\ &= \sigma_i(\mathbf{r}^S \mathbf{R}_{sb}^\top \mathbf{R}^\top \mathbf{v}_b + (\omega_t - \mathbf{b}_\omega) \times \mathbf{p}_{sb}^\top) - v_j^S, \end{aligned} \quad (11)$$

where  $\sigma_i$  is the time interval between this frame and the previous one. Then, combining the prior distribution with the likelihoods derived from all geometric and Doppler observations, we obtain an equivalent maximum a posteriori (MAP) estimate [37] given by

$$\min_{\tilde{\mathbf{x}}_i} \left( \|\mathbf{x}_i \boxminus \hat{\mathbf{x}}_i\|_{\hat{\mathbf{P}}_i}^2 + \sum_{j=1}^m \|{}^g\mathbf{r}_j^i + {}^g\mathbf{H}_j^i \tilde{\mathbf{x}}_i + {}^d\mathbf{r}_j^i + {}^d\mathbf{H}_j^i \tilde{\mathbf{x}}_i\|_{\mathbf{R}_j}^2 \right) \quad (12)$$

where  $\hat{\mathbf{P}}_i$ ,  $\mathbf{R}_j$  represent the covariance of error state and the measurement noise, respectively.  $\boxminus$  is defined in FAST-LIO [13]. Let  $\mathbf{H} = [({}^d\mathbf{H}_1^i + {}^g\mathbf{H}_1^i), \dots, ({}^d\mathbf{H}_m^i + {}^g\mathbf{H}_m^i)]^\top$ ,  $\mathbf{R} = \text{diag}(\mathbf{R}_1, \dots, \mathbf{R}_m)$ , this MAP problem is solved using an IEKF with Kalman gain  $\mathbf{K}$  and partial differentiation of error state  $\mathbf{J}$ , as below:

$$\mathbf{K} = (\mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} + \mathbf{P}^{-1})^{-1} \mathbf{H}^\top \mathbf{R}^{-1},$$



Fig. 5. Experiment setup (left: sensor platform mounted on a car, right: CAD Model of the platform).

$$\mathbf{P} = \mathbf{J}^{-1} \hat{\mathbf{P}} \mathbf{J}^{-\top}. \quad (13)$$

Finally, after the IEKF converges, the optimal state  $\bar{\mathbf{x}}_i$  and its corresponding covariance  $\bar{\mathbf{P}}_i$  are given by:

$$\bar{\mathbf{x}}_i = \hat{\mathbf{x}}_i^{i+1}, \quad \bar{\mathbf{P}}_i = \mathbf{I} - (\mathbf{K}\mathbf{H}) \mathbf{P}. \quad (14)$$

### C. Graph Optimization

Online calibration and back-end optimization incorporate five principal components: IMU pre-integration, odometry, ego velocity, extrinsic transform, and loop closure factors. Among them, the IMU pre-integration, ego velocity, and extrinsic transform factors are only needed when online calibration is activated. Consequently, if the extrinsic transform between IMU and radar or LiDAR is already known, the back-end optimization is simplified to the classical loop closure optimization. The odometry and extrinsic transform factors are derived from (14), while the ego velocity factor is calculated using the least squares method combined with our proposed velocity filter. The IMU pre-integration factor connects keyframes to assist pose prediction and maintain graph constraints. Loop closure, based on ScanContext [6], encodes relative poses to reduce drift, and also constrains the extrinsic estimation globally.

## IV. EXPERIMENTAL EVALUATION

### A. Hardware Setup and Dataset Collection

As displayed in Fig. 5, our experimental platform consists of a 4D Altos V2 radar sensor operating at 77 GHz, two Livox Mid-70 LiDAR, and a Spatial Phidget IMU. Near ground truth data was obtained using U-Blox F9 RTK-GNSS combined with inertial navigation systems, providing centimeter-level positioning accuracy and enabling precise evaluation of our method. All sensors are time-synchronized and rigidly mounted on a roof rack on top of the test vehicle to ensure accurate spatial alignment. We collect data in diverse scenarios, including urban, suburban, and highway, under varying weather and lighting conditions. LiDAR-to-IMU extrinsic calibration is performed using the method proposed in LI\_Init [38], and radar-to-IMU extrinsic calibration is computed through the online calibration approach introduced in this paper.

We implement Doppler-SLAM in C++ with ROS1 and GT-SAM [39] and perform evaluations on a computer equipped with a 4.6 GHz AMD Ryzen 5600x CPU and 32 GB of RAM. Our evaluation metrics include absolute pose error (APE) and relative pose error (RPE) per frame. To thoroughly evaluate the performance of Doppler-SLAM, we conduct extensive experiments targeting accuracy, robustness, and generalization across diverse scenarios and sensor types, including various radar and LiDAR

TABLE I  
COMPARISON OF SLAM METHODS ON SNAIL-RADAR DATASET

Method	20240113/3 (4.6 km)			20240113/1 (0.5 km)			20240115/2 (6.6 km)			20240123/2 (8.5 km)			20240123/3 (2.2 km)		
	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]
FAST-LIO2-LC (LiDAR)	<b>0.007</b>	<b>0.011</b>	1.870	<b>0.010</b>	<b>0.013</b>	0.359	<b>0.007</b>	<b>0.008</b>	<b>1.183</b>	<b>0.022</b>	<b>0.005</b>	9.889	<b>0.009</b>	<b>0.008</b>	<b>0.693</b>
KISS-ICP (radar)	0.240	0.155	68.40	0.118	0.179	4.389	0.232	0.134	147.1	0.269	0.117	167.8	0.222	0.155	45.86
Radar-ICP	0.238	0.156	18.24	0.120	0.174	3.946	0.229	0.131	31.62	0.252	0.112	37.47	0.221	0.151	7.893
4DRadarSLAM	0.737	1.170	53.23	0.460	1.074	8.908	0.663	1.179	491.2	0.864	0.901	454.5	0.503	0.983	142.1
Graph-rio	-	-	-	0.169	0.172	9.523	0.195	0.172	763.4	-	-	-	0.266	0.168	497.0
RIV-SLAM	0.213	0.142	30.17	0.113	0.171	4.131	0.219	0.128	33.1	0.224	0.101	35.51	0.201	0.140	6.120
Doppler-Odometry (radar)	0.151	0.113	3.375	0.083	0.167	<b>0.285</b>	0.175	0.102	9.391	0.199	<b>0.095</b>	10.59	<b>0.156</b>	0.117	2.608
Doppler-SLAM (radar)	0.150	0.111	<b>1.532</b>	0.082	0.160	0.316	0.174	0.098	5.651	0.198	<b>0.095</b>	<b>5.810</b>	0.156	0.116	1.556

**Bold:** best results, underlined: best radar results.

sensors. The system is benchmarked on multiple publicly available datasets, including (i) Snail-Radar [40] using Continental ARS548 radar; (ii) NTU4DRadLM [41] employing the Oculii Eagle radar; (iii) HeRCULES [42] using Continental ARS548 radar paired with an Aeva FMCW LiDAR; and (iv) our newly introduced IMADAR dataset with the Altos V2 radar.

### B. Comparison to State-of-The-Art Methods

In the first experiment, we analyze the performance of our system and compare it to state-of-the-art methods. The results show that our proposed Doppler-SLAM achieves superior radar-SLAM performance compared to existing methods using the Snail-Radar dataset [40]. Next, we showcase the cross-modality generalization by comparing Doppler-SLAM with state-of-the-art radar- and LiDAR-SLAM approaches on the HeRCULES dataset [42], using both FMCW LiDAR and 4D radar data. All Snail-Radar and HeRCULES trajectories are evaluated in the plane, with some results adapted from [25]. We further validate Doppler-SLAM’s versatility on NTU4DRadLM [41] and our IMADAR dataset, which include both vehicle-mounted and handheld scenarios. For NTU4DRadLM and IMADAR, 3D pose accuracy is assessed by evaluating the vertical direction as well, confirming Doppler-SLAM’s robustness and effectiveness in diverse and dynamic environments.

We benchmark Doppler-SLAM against several state-of-the-art methods, including 4DRadarSLAM (Radar-only SLAM) [3], Graph-rio (Radar-inertial odometry) [43], Radar-ICP (Doppler velocity aided Radar-only odometry) [24], Go-rio (Doppler velocity aided Radar-inertial odometry) [21], KISS-ICP (LiDAR odometry approach on radar) [10], FAST-LIO2 (LiDAR-inertial odometry) [1], and RIV-SLAM (Radar-inertial SLAM) [44].

Quantitative results on the Snail-Radar dataset are presented in Table I. FAST-LIO2 with loop closure (FAST-LIO2-LC) on LiDAR data serves as the baseline, while all other methods operate on radar data. KISS-ICP provides good local accuracy on radar point clouds but suffers from increased drifts in large dynamic scenes. Both Radar-ICP and 4DRadarSLAM are radar-only, point-to-point matching methods that, without motion constraints, are prone to incorrect matches in large-scale or highly dynamic sequences such as 20240115/2 and 20240123/2. Radar-IMU-based methods, such as Graph-rio and RIV-SLAM, despite using IMU as motion constraints, also exhibit significant errors due to their failure to detect loop closure in dynamic environments. Doppler-Odometry (without loop closure) outperforms all other radar-based methods by utilizing Doppler velocities to improve motion estimation. Notably, Doppler-SLAM further enhances performance by incorporating loop closure, achieving accuracy on radar data that is comparable to LiDAR-based SLAM.

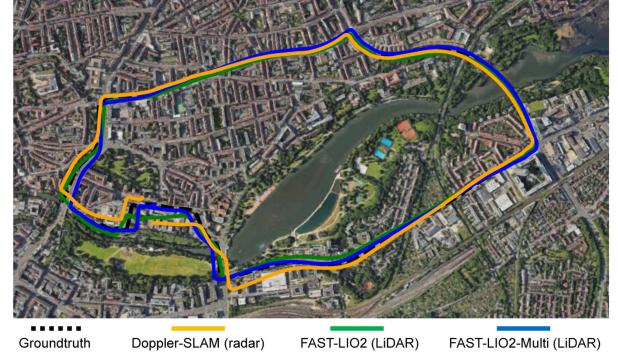


Fig. 6. Qualitative results on sequence “WoehrSee” from IMADAR dataset.

Table II presents the evaluation of Doppler-SLAM’s performance and generalizability on the HeRCULES dataset, examining both radar and FMCW LiDAR data, as shown in Fig. 1. To highlight the improvements of Doppler-SLAM over traditional LiDAR-SLAM methods on FMCW LiDAR, we provide a comparative analysis with FAST-LIO2 [1]. Thanks to the robustness of the velocity filter in dynamic scenarios and the tight coupling between IMU measurements and Doppler velocities, Doppler-SLAM consistently outperforms FAST-LIO2 in the sequences “Street Day”, characterized by highly dynamic conditions and rain, “Library Day”, and “Parking Lot”, notable for frequent sharp turns. These results demonstrate that Doppler-SLAM maintains remarkable robustness in harsh conditions where competing methods suffer significant performance degradation. These capabilities are further highlighted on the NTU4DRadLM dataset, as shown in Table III, which employs the Oculii Eagle radar with both handheld and vehicle-mounted data acquisition.

However, the datasets mentioned above are mostly low-speed (below 50 km/h) driving scenarios. To more comprehensively evaluate the performance of Doppler-SLAM in high-speed, long-distance scenarios, we conduct additional experiments using our own IMADAR dataset, benchmarked against FAST-LIO2 (FAST-LIO2-Multi for two LiDARs with asynchronous update) with loop closure for comparative analysis. Table IV presents the evaluation of three different sequences: “WoehrSee” and “HBF”, both representing urban traffic scenarios where the main challenges are complex dynamic conditions (as shown in Fig. 3 and Fig. 6) and degraded environments such as tunnels, and the “N4” sequence, representing high-speed and long-distance conditions with vehicle speeds up to 110 km/h. Benefiting from a higher frequency of asynchronous updates and a larger field-of-view angle, FAST-LIO2-Multi performs best

TABLE II  
COMPARISON OF SLAM METHODS ON HERCULES DATASET

Method	Mountain Day 1 (4 km, mountain)			Library Day 1 (1.6 km)			Sports Complex Day 1 (1.4 km)			Parking Lot 3 Night (0.5 km)			Street Day 1 (1 km, rain)		
	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]
FAST-LIO2-LC (LiDAR)	<b>0.085</b>	0.061	4.595	0.095	0.072	5.234	<b>0.084</b>	0.073	2.114	<b>0.091</b>	0.092	1.828	<b>0.028</b>	0.026	2.966
RIV-SLAM (radar)	0.077	0.084	206.9	0.014	0.064	4.175	-	-	-	0.020	0.075	2.380	0.010	0.042	10.74
Radar-ICP (radar)	<u>0.055</u>	<u>0.067</u>	118.6	0.049	<u>0.064</u>	10.23	0.049	0.071	7.125	0.058	0.089	3.414	0.022	<u>0.028</u>	11.66
Doppler-SLAM (radar)	0.059	0.068	<u>43.05</u>	0.012	0.094	13.24	0.081	0.065	2.718	0.015	0.078	0.717	0.013	0.039	7.448
Doppler-SLAM (LiDAR)	0.128	<b>0.054</b>	<u>4.498</u>	<u>0.085</u>	<b>0.061</b>	<u>3.365</u>	0.093	<u>0.067</u>	<b>2.069</b>	0.101	0.077	<u>1.642</u>	0.029	<b>0.024</b>	<u>2.813</u>

Bold: best LiDAR results, underlined: best radar results.

TABLE III  
COMPARISON OF SLAM METHODS ACROSS SEQUENCES ON  
NTU4DRADARLM DATASET

Method	cp (handcart, 0.25 km)				loop2 (car, 4.79 km)				loop3 (car, 4.23 km)			
	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]
4DRadarSLAM	0.129	0.255	<b>0.861</b>	1.337	0.308	<u>43.67</u>	1.216	0.455	334.7	-	-	-
Go-RIO	0.079	0.661	<u>1.035</u>	-	-	-	0.991	1.035	52.74	-	-	-
Doppler-Odometry	<b>0.022</b>	<b>0.109</b>	3.267	0.260	<u>0.179</u>	50.56	<u>0.209</u>	<b>0.162</b>	<u>48.34</u>	-	-	-
Doppler-SLAM	0.079	0.267	1.508	<u>0.261</u>	0.126	<u>4.278</u>	0.188	0.169	<u>8.182</u>	-	-	-

Bold: best results, underlined: best second best results.

TABLE IV  
COMPARISON OF SLAM METHODS ACROSS SEQUENCES ON IMADAR  
DATASET

Method	WoehrSee (7.1 km)		HBF (4.7 km)		N4 (15.4 km)	
	RPE [m]	APE [m]	RPE [m]	APE [m]	RPE [m]	APE [m]
FAST-LIO2 (LiDAR)	0.22	24.89	0.29	15.56	0.252	138.77
FAST-LIO2-Multi (LiDAR)	<b>0.115</b>	<b>18.65</b>	<b>0.104</b>	<b>2.50</b>	<b>0.153</b>	<b>58.75</b>
Doppler-SLAM (radar)	0.178	30.55	0.21	8.46	0.216	102.9

Bold: best results, underlined: best second best results.

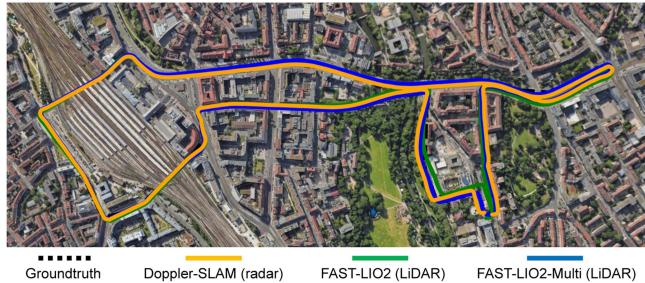


Fig. 7. Qualitative results on sequence “HBF” from IMADAR dataset.

on all three sequences. The quantitative results indicate that although both Doppler-SLAM and FAST-LIO2 show performance degradation over the three sequences, Doppler-SLAM using radar is still comparable to state-of-the-art LiDAR approaches and even outperforms FAST-LIO with LiDAR in highly dynamic environments on sequences “HBF” (as illustrated in Fig. 7) and “N4”. These results further validate our proposed online extrinsic calibration method, as it is consistently employed for radar-to-IMU calibration across all three sequences.

### C. Ablation Studies

We perform ablation studies to evaluate the contributions and computation time of individual components within the Doppler-SLAM framework. To isolate and eliminate sensor-related effects, we select sequence 20240113/3 from the Snail-Radar for the radar-based Doppler-SLAM and the sequence “Street Day” from HeRCULES for FMCW LiDAR-based ablation study. Both sequences represent dynamic and complex environments. As presented in Table V, the results clearly demonstrate that

TABLE V  
ABALION EVALUATION ON RADAR AND FMCW LiDAR SEQUENCES

Method	20240113/3 (radar)			Street Day (FMCW LiDAR)		
	RPE [m]	RPE [°]	APE [m]	RPE [m]	RPE [°]	APE [m]
w/o velocity filter	0.216	0.3730	16.07	0.112	0.102	11.06
w/o Doppler residual	<b>0.149</b>	<u>0.1111</u>	5.671	0.071	0.080	3.350
w/o online calibration	0.159	0.1112	4.140	0.052	<u>0.038</u>	3.476
w/o loop closure	0.151	0.1113	<u>3.378</u>	<u>0.046</u>	0.039	<u>2.940</u>
<b>Doppler-SLAM</b>	<u>0.150</u>	<b>0.1110</b>	<u>1.532</u>	<b>0.029</b>	<b>0.024</b>	<u>2.813</u>

Bold: best results, underlined: best second best results.

TABLE VI  
COMPUTATION TIME OF EACH MODULE (MS)

	velocity filter	Doppler residual	online calibration	loop closure
20240113/3 (radar)	0.024	1.877	0.036	37.72
Street Day (FMCW LiDAR)	0.175	10.8	0.097	137

the integration of Doppler velocity into both radar and FMCW LiDAR systems greatly improves accuracy and robustness by reducing false matches in dynamic scenes. The computation time for each module is summarized in Table VI. The velocity filter and online calibration modules are highly efficient, each contributing less than 0.2 ms per frame. Doppler residual computation is also lightweight, with slightly higher cost on LiDAR due to increased point cloud density. As expected, loop closure is the most computationally intensive component, but it remains within real-time capability for both sensor types. This confirms the efficiency and suitability of Doppler-SLAM for real-time radar and FMCW LiDAR applications.

### V. CONCLUSION

This paper proposes Doppler-SLAM, a novel Doppler-aided radar-inertial and LiDAR-inertial SLAM framework. By incorporating Doppler velocities into scan-matching, our approach unifies FMCW LiDAR- and 4D radar-based SLAM systems into a single framework, enabling robust SLAM performance under challenging dynamic conditions. This tightly integrated system fuses IMU data with either 4D radar or FMCW LiDAR to deliver odometry, ego velocity estimation, mapping, and extrinsic calibration between the IMU and the radar or LiDAR sensor. Our innovative online calibration technique, enhanced by Doppler velocity and loop closure, ensures consistent sensor alignment. Thorough evaluations demonstrate clear advantages over existing state-of-the-art radar-SLAM and FMCW LiDAR-SLAM frameworks, and by releasing Doppler-SLAM and our IMADAR dataset as open-source software, we encourage continued advancement and further research within the SLAM community.

## REFERENCES

- [1] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, “FAST-LIO2: Fast direct LiDAR-inertial odometry,” *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.
- [2] 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. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.
- [3] J. Zhang et al., “4DRadarSLAM: A 4D imaging radar SLAM system for large-scale environments based on pose graph optimization,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 8333–8340.
- [4] C. Doer and G. F. Trommer, “An EKF based approach to radar inertial odometry,” in *Proc. IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, 2020, pp. 152–159.
- [5] Y. Zhuang, B. Wang, J. Huai, and M. Li, “4D iRIOM: 4D imaging radar inertial odometry and mapping,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 6, pp. 3246–3253, Jun. 2023.
- [6] H. Wang, C. Wang, and L. Xie, “Intensity scan context: Coding intensity and geometry relations for loop closure detection,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 2095–2101.
- [7] P. J. Besl and N. D. McKay, “Method for registration of 3-D shapes,” *Proc. SPIE*, vol. 1611, pp. 586–606, 1992.
- [8] A. Segal, D. Haehnel, and S. Thrun, “Generalized-ICP,” in *Proc. Robot.: Sci. Syst.*, 2009, pp. 435–442.
- [9] J. Zhang et al., “LOAM: Lidar odometry and mapping in real-time,” in *Proc. Robot.: Sci. Syst.*, 2014, vol. 2, no. 9, pp. 1–9.
- [10] 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 Robot. Automat. Lett.*, vol. 8, no. 2, pp. 1029–1036, Feb. 2023.
- [11] H. Ye, Y. Chen, and M. Liu, “Tightly coupled 3D lidar inertial odometry and mapping,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 3144–3150.
- [12] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, “LINS: A lidar-inertial state estimator for robust and efficient navigation,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 8899–8906.
- [13] W. Xu and F. Zhang, “FAST-LIO: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.
- [14] Y. Wu et al., “Picking up speed: Continuous-time lidar-only odometry using Doppler velocity measurements,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 1, pp. 264–271, Jan. 2023.
- [15] K. M. Papais, D. Lisus, D. J. Yoon, A. Lambert, K. Y. Leung, and T. D. Barfoot, “Balancing act: Trading off Doppler odometry and map registration for efficient lidar localization,” 2025, *arXiv:2503.02107*.
- [16] B. Hexsel, H. Vhayle, and Y. Chen, “DICP: Doppler iterative closest point algorithm,” 2022, *arXiv:2201.11944*.
- [17] M. Zhao, J. Wang, T. Gao, C. Xu, and H. Kong, “FMCW-LIO: A Doppler lidar-inertial odometry,” *IEEE Robot. Automat. Lett.*, vol. 9, no. 6, pp. 5727–5734, Jun. 2024.
- [18] D. J. Yoon et al., “Need for speed: Fast correspondence-free lidar-inertial odometry using Doppler velocity,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 5304–5310.
- [19] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, “Instantaneous ego-motion estimation using Doppler radar,” in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2013, pp. 869–874.
- [20] D. Wang, M. Masannek, S. May, and A. Nüchter, “Infradar-localization: Single-chip infrared-and-radar-based Monte Carlo localization,” in *Proc. Int. Conf. Automat. Sci. Eng.*, 2023, pp. 1–8.
- [21] W. Yang, H. Jang, and A. Kim, “Ground-optimized 4D radar-inertial odometry via continuous velocity integration using gaussian process,” 2025, *arXiv:2502.08093*.
- [22] Z. Hong, Y. Petillot, A. Wallace, and S. Wang, “Radarslam: A robust simultaneous localization and mapping system for all weather conditions,” *Int. J. Robot. Res.*, vol. 41, no. 5, pp. 519–542, 2022.
- [23] Y. S. Park, Y.-S. Shin, J. Kim, and A. Kim, “3D ego-motion estimation using low-cost mmWave radars via radar velocity factor for pose-graph SLAM,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7691–7698, Oct. 2021.
- [24] D. C. Herraez, M. Zeller, L. Chang, I. Vizzo, M. Heidingsfeld, and C. Stachniss, “Radar-only odometry and mapping for autonomous vehicles,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 10275–10282.
- [25] D. C. Herraez, M. Zeller, D. Wang, J. Behley, M. Heidingsfeld, and C. Stachniss, “RaI-SLAM: Radar-inertial SLAM for autonomous vehicles,” *IEEE Robot. Automat. Lett.*, vol. 10, no. 6, pp. 5257–5264, Jun. 2025.
- [26] X. Li, H. Zhang, and W. Chen, “4D radar-based pose graph SLAM with ego-velocity pre-integration factor,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 8, pp. 5124–5131, Aug. 2023.
- [27] Q. Huang, Y. Liang, Z. Qiao, S. Shen, and H. Yin, “Less is more: Physical-enhanced radar-inertial odometry,” *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2024.
- [28] C. Guo, B. Wei, B. Lan, L. Liang, and H. Liu, “DGRO: Doppler velocity and gyroscope-aided radar odometry,” *Sensors*, vol. 24, no. 20, 2024, Art. no. 6559.
- [29] H. Chen, Y. Liu, and Y. Cheng, “DRIO: Robust radar-inertial odometry in dynamic environments,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 9, pp. 5918–5925, Sep. 2023.
- [30] J. Michalczyk, R. Jung, and S. Weiss, “Tightly-coupled EKF-based radar-inertial odometry,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 12336–12343.
- [31] J. Michalczyk, J. Quell, F. Steidle, M. G. Müller, and S. Weiss, “Tightly-coupled factor graph formulation for radar-inertial odometry,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2024, pp. 3364–3370.
- [32] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue, “Towards deep radar perception for autonomous driving: Datasets, methods, and challenges,” *Sensors*, vol. 22, no. 11, 2022, Art. no. 4208.
- [33] K. Cai, B. Wang, and C. X. Lu, “AutoPlace: Robust place recognition with single-chip automotive radar,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 2222–2228.
- [34] D. C. Herraez et al., “SPR: Single-scan radar place recognition,” *IEEE Robot. Automat. Lett.*, vol. 9, no. 10, pp. 9079–9086, Oct. 2024.
- [35] D. He, W. Xu, and F. Zhang, “Embedding manifold structures into Kalman filters,” 2021, *arXiv:2102.03804*.
- [36] C. Doer and G. F. Trommer, “Radar visual inertial odometry and radar thermal inertial odometry: Robust navigation even in challenging visual conditions,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 331–338.
- [37] D. He, W. Xu, and F. Zhang, “Kalman filters on differentiable manifolds,” 2021, *arXiv:2102.03804*.
- [38] F. Zhu, Y. Ren, and F. Zhang, “Robust real-time lidar-inertial initialization,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 3948–3955.
- [39] F. Dellaert, “Factor graphs and GTSAM: A hands-on introduction,” Georgia Inst. Technol., Atlanta, GA, USA, Tech. Rep. 2.4, 2012.
- [40] J. Huai et al., “Snail-radar: A large-scale diverse dataset for the evaluation of 4D-radar-based SLAM systems,” 2024, *arXiv:2407.11705*.
- [41] J. Zhang et al., “NTU4DRadLM: 4D radar-centric multi-modal dataset for localization and mapping,” in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2023, pp. 4291–4296.
- [42] H. Kim et al., “HERCULES: Heterogeneous radar dataset in complex urban environment for multi-session radar SLAM,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2025.
- [43] R. Girod, M. Hauswirth, P. Pfreundschuh, M. Biasio, and R. Siegwart, “A robust baro-radar-inertial odometry M-estimator for multicopter navigation in cities and forests,” in *Proc. IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, 2024, pp. 1–8.
- [44] D. Wang, S. May, and A. Nüchter, “RIV-SLAM: Radar-inertial-velocity optimization based graph SLAM,” in *Proc. Int. Conf. Automat. Sci. Eng.*, 2024, pp. 774–781.