2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)
August 28 - September 1, 2024. Bari, Italy

2024 IEEE 20th International Conference on Automation Science and Engineering (CASE) | 979-8-3503-5851-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/CASE59546.2024.10711511

RIV-SLAM: Radar-Inertial-Velocity optimization based graph SLAM

Dong Wang [1], Stefan May[2] and Andreas Nuechter[1]

***Abstract*— 4D imaging radars, commonly known as 4D radars, deliver comprehensive point cloud data that encapsulates range, azimuth, elevation, and Doppler velocity information even in harsh environmental conditions, such as rain, snow, smoke, and fog. However, 4D radar data also suffers from high noise and sparsity, which poses great challenges for SLAM applications. This paper presents RIV-SLAM, a complete radar-inertial-velocity optimization-based graph SLAM system designed to exploit the full potential of 4D imaging radar technology. RIV-SLAM consists of four integral components: front-end, loop closure, IMU pre-integration and graph optimization, each optimized to effectively leverage the unique attributes of radar data and tightly coupled with IMU data. This is also the first SLAM system known to us that outputs an optimized ego velocity. This capability ensures reliable ego motion estimation under extreme conditions (e.g., wheel odometry fails). Furthermore, we develop a new ground extraction approach, specifically adapted for the 4D imaging radar, which substantially improves the system's z-axis accuracy. Comprehensive evaluations of the RIV-SLAM system on a variety of datasets demonstrate its superior performance, significantly surpassing existing state-of-the-art Radar-SLAM frameworks. The code of RIV-SLAM will be released at: RIV-SLAM**

I. INTRODUCTION

The development of autonomous robots and autonomous vehicles has attracted considerable interest. Fundamental to the autonomy of these machines is their ability to perform state estimation, localization, and mapping with both precision and efficiency, especially in complex environments. Over the years, numerous Simultaneous Localization and Mapping (SLAM) algorithms have been presented for different sensors, such as Light Detection and Ranging (LiDARs) and cameras. However, it is widely recognized that the performance of these sensors is significantly impacted by various factors such as weather, lighting conditions, and interference, including airborne particles, which can challenge their effectiveness in harsh environments [1][2]. To address this limitation, the focus has turned to the mmWave Radars (radio detection and ranging), as recent improvements in electronics and materials science have enabled the radar sensor to be housed in a smaller package than previous products, and provide consistent reliability in diverse environments compared to cameras or LiDARs [3][4]. The latest mmWave radar sensors are often referred to as 4D imaging radars because they provide a denser 3D point cloud with richer information such as range, azimuth, elevation, and Doppler velocity. With more spatial information and increased resolution, 4D radars enable new possibilities for SLAM applications.

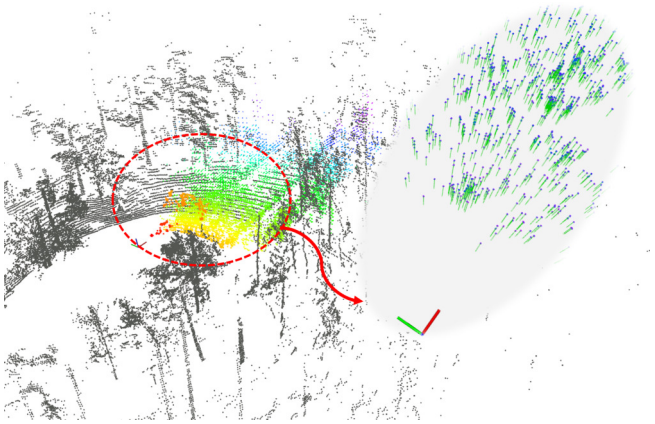


Fig. 1: 4D Radar Data aligned with LiDAR Scan in forest environment (colored: radar scan, black: LiDAR scan). The magnitude and direction of Doppler's velocity in radar scan are illustrated by green arrows

Any opportunity, however, comes with risks. The deployment of 4D radar poses two major challenges: *1)* As shown in Fig. 1, the data points captured by 4D radar tend to be more sparse compared to 3D LiDAR, complicating the extraction of reliable geometric features like edges and planes. *2)* The intrinsic physical properties of multichannel 4D radar systems can lead to the phenomenon of interference waveforms. These are often misinterpreted as ghost objects [5], which introduce larger errors in the scan matching method widely used in LiDAR and camera-based SLAM.

In this context, several 4D imaging-radar-based SLAM frameworks were proposed [6][7][8]. Inspired by these approaches, we present RIV-SLAM, a novel SLAM system for 4D imaging radar based on Radar-Inertial-Velocity graph optimization. In comparison to the previous work, the key contributions of our work are as follows.

- RIV-SLAM is a complete system that makes full use of 4D radar data characteristics and consists of four parts: front-end, loop closure, IMU pre-integration, and graph optimization.
- We design a method to optimize radar Doppler velocity with IMU data, which ultimately achieves an accurate and stable estimation of ego velocity.
- We propose a ground extraction method tailored for 4D imaging-radar data to increase the accuracy of the system

[1]Dong Wang (dong.wang@uni-wuerzburg.de) and Andreas Nuechter are with the Department of Informatics XVII Robotics, Julius-Maximilians-University Wuerzburg, Wuerzburg, 97074, Germany.

[2]Stefan May is with the Faculty of Electrical Engineering, Precision Engineering, Information Technology, Nuremberg Institute of Technology Georg Simon Ohm, Nuremberg, 90489, Germany

in the z-axis and a hybrid scan registration approach based on tight coupling of scan-to-scan and scan-to-map registration.
- The proposed RIV-SLAM system is thoroughly evaluated on various datasets, and the results obtained significantly outperform the current state-of-the-art Radar-SLAM frameworks.

This paper is structured as follows. Section II reviews the related work. Section III presents the proposed RIV-SLAM framework. The experimental evaluation is described in Section. IV. Section V summarizes the study and discusses future directions.

II. RELATED WORK

With advancements in mmWave sensor technology, researchers are turning to 4D imaging radar sensors for robot SLAM and autonomous vehicles. These sensors have many benefits, including their small size, low cost, robustness to environmental factors such as lighting conditions, and low power consumption. In this section, we will take an in-depth look at the state-of-the-art approaches to Radar Odometry and 4D Radar SLAM.

A. Radar Ego-motion Estimation

Recent improvements in electronics and materials science have allowed the radar (radio detection and ranging) sensor to be packaged in smaller sizes compared to earlier products and provide denser point cloud data. Besides this, the latest radar sensors can also measure radial velocity using the Doppler effect [9][4]. The advantage of radar odometry over LO (LiDAR Odometry) or VO (Visual Odometry) is that the Doppler radar can directly measure the relative velocity of stationary objects within a single frame, rather than deriving the relative velocity from the changing position of the stationary object in successive frames. Early studies, such as the work presented in [10], proposed an instantaneous approach for 2D radar ego velocity estimation using only one radar scan with Doppler radial velocity measurements. The authors utilized the RANdom SAmple Consensus (RANSAC) algorithm to filter out the moving objects in the environment and employed the Least-SQuares estimator (LSQ) to optimize the radial velocity of stationary objects relative to the radar to obtain radar ego velocity. In [11], the approach was extended to multiple radar sensors with joint spatial optimization. In our previous work [12], we adopted a multi-strategy weighted LSQ optimization approach to estimate the ego-motion of robots, which we further improve in this work. To achieve higher accuracy, the fusion of radar measurements with inertial data has shown impressive results [13][14][3]. However, without the help of yaw angle, the yaw drift increases with time due to changes in the yaw rate of the MEMS IMU. To compensate for this, the authors in [15][16] presented variants of Radar Odometry based on further sensor data fusion, such as Radar Visual Inertial Odometry, Radar Thermal Inertial Odometry, and GNSS-aided Radar Inertial Odometry.

B. 4D Radar SLAM

The exploration of 4D radar SLAM is still at a nascent stage, with existing research primarily centered around two categories.

1) Indirect Method: The indirect method refers to feature extraction and scan-matching, which have been extensively researched on LiDAR-SLAM [17][18]. Similarly, the radar point cloud is considered as an image and the transformation between successive scans is computed by extracting and matching specific features. A radar keypoint extraction and graph scan matching is proposed in [1] to enable a scanning-radar-based odometry. Hong *et al.* [19] presented a complete Radar-SLAM framework for 360° scanning radar, including SURF feature extraction and global pose graph optimization. However, feature extraction suffers from the noise of measurements and sparse radar point clouds, which makes it even harder to apply to a 4D solid-state radar with an effective FoV (Field of View) of 120°. Therefore, the most recent works for 4D radar SLAM are based on Point cloud registration. In [7], a scan-to-submap NDT (Normal distribution transform) is applied with point cloud registration, while velocity pre-integration is used to improve optimization performance. Zhang *et al.* [6] proposed an Adaptive Probability Distribution-GICP (APDGICP) to address radar measurement noise, considering the spatial probability distribution of each point in GICP [20]. Another approach to improving the matching quality of sparse radar data is 4DiRIOM [21]. Point matching here is expressed in terms of distribution-to-multiple-distribution constraints, which can be achieved by matching the current scan with a submap constructed by the mapping module, rather than scan-to-scan matching. Huang *et al.* [22] leverages the RCS (radar cross section) information to refine the point-to-point correspondence, thus improving the estimation of poses based on radar point matching.
2) Direct Method: Instead of using feature extraction and scan matching, the direct method takes the entire radar image as input and builds correspondence between consecutive scans in the Fourier domain. In [23] the Fourier Mellin Transformation (FMT) is utilized to estimate the transformation matrix taking advantage of phase correlation properties. In Fast-MbyM [24] a CNN was proposed to mask radar observations, and they decoupled the search for angle and translation by utilizing the translational invariance of the Fourier Transform.

C. Summary

Inspired by [6] [7] [8] [14] [21] [22], we propose our RIV-SLAM system, which significantly advances existing methods in several respects:

- RIV-SLAM is, to the best of our knowledge, the first radar SLAM framework to optimize both pose and ego-motion.
- We propose a ground extraction method tailored for 4D imaging-radar data to enhance the system's performance.

- A novel registration approach based on the anisotropy of radar measurements is introduced to overcome sparsity and noise in radar data.
- Extensive evaluations on a variety of datasets demonstrate the accuracy, robustness, and real-time performance of RIV-SLAM.

III. METHODOLOGY

In this paper, we employ the following conventions to represent the various mathematical and physical quantities used in our research:

- 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, like $\mathbf{b}$.
- Subscripts and superscripts are used to denote different frames of reference. For example, a vector $\mathbf{b}$ in radar frame $\{\}^r$ will be denoted as $\mathbf{b}^r$, and the rotation from frame $\{\}^r$ to frame $\{\}^w$ 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 $\{\}^w$ or $\{\}^W$.

By using this formalism, we aim to provide a clear and consistent notation that facilitates communication and understanding of our mathematical models and results.

A. Framework Overview

Fig. 2 shows the overview of the proposed system consisting of four components: Outlier Removal, Ground Extraction, Loop Detection, and Graph Optimization.

B. Front-end

1) Ego Motion Estimation: 4D radar captures a series of targets, detailing their three-dimensional positions $\mathbf{p}^r$, corresponding Doppler radial velocities v_d^r, and the signal-to-noise ratio (SNR) s for every target. The radar coordinate system is defined as $\{\}^r$. At any given moment, the radar's velocity is represented by $\mathbf{v}^r$. The Doppler velocity v_d^r is calculated by taking the magnitude of the projection of the relative velocity vector between the target and radar onto the ray connecting the target and the radar, as illustrated in Fig. 1. This calculation involves the dot product of the target's velocity v_d^r in the radar frame $\{\}^r$ and the unit vector pointing from the radar towards the target:

$$-v_d^r = \frac{\mathbf{p}^r}{\|\mathbf{p}^r\|} \cdot \mathbf{v}^r = \mathbf{r}^r \cdot \mathbf{v}^r = r_x^r v_x^r + r_y^r v_y^r + r_z^r v_z^r \quad (1)$$

If we assume that the targets within the scene remain stationary and only the sensor platform is in motion, each detected target acts as a constraint on the estimated velocity of the radar. Thus, we obtain a set of N detections in a radar measurement and write (1) in matrix notation, we get (2), and the residual $\mathbf{e}$ can be derived as (3).

$$\underbrace{\begin{bmatrix} -v_{d,1}^r \\ -v_{d,2}^r \\ \vdots \\ -v_{d,N}^r \end{bmatrix} = \begin{bmatrix} r_{x,1}^r & r_{y,1}^r & r_{z,1}^r \\ r_{x,2}^r & r_{y,2}^r & r_{z,2}^r \\ \vdots & \vdots & \vdots \\ r_{x,N}^r & r_{y,N}^r & r_{z,N}^r \end{bmatrix} \begin{bmatrix} v_x^r \\ v_y^r \\ v_z^r \end{bmatrix}}_{\text{-}\mathbf{v}_d^r = \mathbf{H}\mathbf{v}^r} \quad (2)$$

$$\mathbf{e} = \mathbf{H}\mathbf{v}^r + \mathbf{v}_d^r \quad (3)$$

Taking into account the anisotropy of the radar measurement errors, i.e., these errors appearing in (1) are not only velocity-dependent, but also position-dependent, and in particular, are related to the estimated angle of the object. Therefore, we extend the multi-strategy weighting LSQ optimization approach proposed in our previous work [12] to achieve accurate and robust ego motion estimation. The correspondence between residuals, azimuth contribution and SNR consistency is defined as $w_i^{residual}$, $w_i^{azimuth}$ and w_i^{snr}. Applying the residual definition of (3) yields the residual with multi-strategy weighting LSQ optimization:

$$\mathbf{e}^m = (w^{residual} \cdot w^{azimuth} \cdot w^{snr})(\mathbf{H}\mathbf{v}^r + \mathbf{v}_d^r) \quad (4)$$

Remarkably, in practice, most targets in a 4D radar point cloud are static in the world frame, and all static points fulfill the (1). To account for the dynamic environment, a three-point RANdom SAmple Consensus (RANSAC) [15] approach is employed to eliminate dynamic outliers and extract static inliers for (2). This also aids subsequent point cloud registration by filtering out dynamic points and improving the accuracy of scan matching. Since the angular velocity $\boldsymbol{\omega}_m^b$ of the base frame $\{\}^b$ leads to an additional velocity in the radar frame $\{\}^r$, we apply rigid body motion to determine $\boldsymbol{\omega}_m^b$ as (5).

$$\boldsymbol{v}^r = \boldsymbol{\omega}_m^b \times \boldsymbol{l}_{br}^b + \boldsymbol{v}^b \quad (5)$$

2) Ground Extraction: Ground extraction from 4D radar point clouds is a significant challenge, especially when compared to much less noisy and dense point clouds captured by LiDAR. This fundamental difference in data quality means that the sophisticated algorithms developed for feature extraction in LiDAR SLAM systems are often unsuitable for direct application to 4D radar data. Therefore, we propose a ground extraction method tailored for 4D radar data to overcome the problems of sparsity, anisotropic noise and feature obscurity inherent in radar point clouds. At the beginning, when the system is completely stationary, we use the IMU measurements to initialize the system and estimate the gravity. The z-axis of the first radar frame is then aligned with the gravity direction. Because in most cases, the ground plane is nearly perpendicular to the direction of gravity. Following the initial setup, all subsequent gravity vectors within the sliding window are aligned with the initial gravity. Next, the known radar mounting height and direction of gravity are used to determine the reference ground plane. Then we search for potential ground points in the altitude

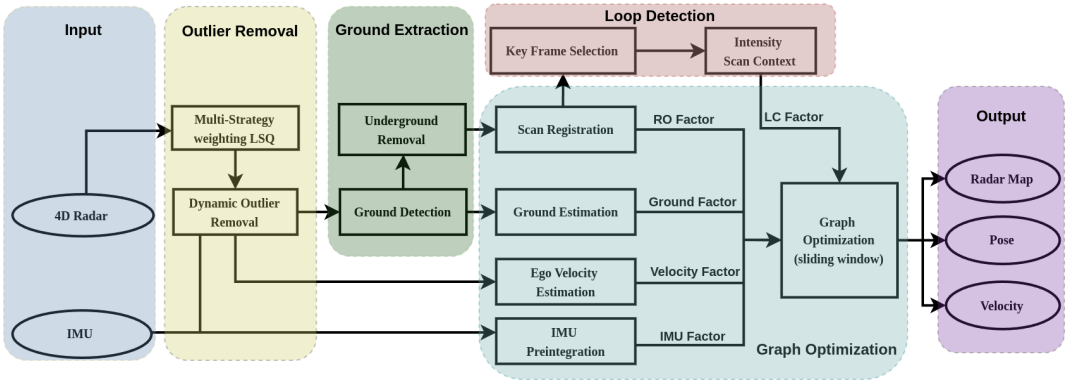


Fig. 2: Pipeline of RIV-SLAM

interval between $-\mathbf{z}_{ground}$ and $+\mathbf{z}_{ground}$ around the reference plane and calculate the point's normal vectors via Principal Component Analysis (PCA) [25]. Potential ground points are only classified as ground points if they meet two key criteria: first, the difference between the normal vector and the positive gravity vector must fall below a specified threshold θ_g; second, the signal-to-noise ratio of the point must fall within a range γ_{min} and γ_{max}. Last, the RANSAC algorithm is applied to refine the ground points. These criteria ensure that points identified as ground maintain both the expected orientation consistency with respect to gravity and the signal reflection characteristics normally expected from the ground. Once the exact ground points are obtained, we fit the entire ground plane and filter all the points below the ground plane, which are called ghost points, and the points above the ground plane are used for scan registration. The ground coefficients are added to the graph optimization as constraints on the z-axis. The ground extraction process is illustrated in Fig. 3. On the left side of the figure, an image captured from a front-view camera is displayed, providing a visual context of the environment. On the right side, the figure showcases the ground features that have been extracted from the radar frames, presented from a side view.

Fig. 3: Ground Extraction. left: camera view, right side from top to down: radar inliers, ground detection and fitting, non-ground filter

3) Scan Registration: Radar-based scan registration typically utilizes two matching strategies: scan-to-scan or scan-to-submap. The scan-to-scan approach aligns two consecutive radar scans to determine point correspondences and has the advantage of requiring less computation. However, this method tends to be less reliable, especially with sparse radar data or significant changes in the environment, such as during turns or uphill travel, leading to error accumulation. On the other hand, the scan-to-submap method involves comparing the current scan against a locally maintained map, enhancing robustness and accuracy. But this method also introduces a large computational load, which reduces the real-time performance of the system. Based on the above two approaches and inspired by [26], we propose a hybrid registration approach, i.e., we use scan-to-scan to compute the real-time radar odometry, and then use scan-to-submap for map-based odometry correction. As we in [12] mentioned, the Angle of Arrival (AoA) φ of 4D radar signal can be mathematically calculated as: $\varphi = sin^{-1}(\frac{\lambda \cdot \Delta\phi}{2\pi r})$, where $\Delta\phi$ is the phase change of the FFT peak, r represents the distance between consecutive antennas and λ is the wavelength. It should be noted that $\Delta\phi$ depends on $sin(\varphi)$, which exhibits a non-linear dependency. The approximation of $sin(\varphi)$ as a linear function is only valid when φ is small in magnitude. Thus, radar measurements exhibit anisotropic uncertainties [8], meaning the precision varies for different AoAs. Specifically, the uncertainty associated with each radar point is greater in the azimuthal direction compared to the radial direction. And when the azimuth angle is smaller, the measurement is more accurate. Following these observations, we modify the Adaptive Probability Distribution-GICP (APDGICP) [6] to incorporates the spatial probability distribution of each point into the Generalized Iterative Closest Point (GICP) framework in accordance with the anisotropy of the 4D radar measurements. The covariance matrix of the point in the local frame is $S = \begin{bmatrix} \sigma_r & 0 & 0 \\ 0 & \sigma_a & 0 \\ 0 & 0 & \sigma_e \end{bmatrix}$ with $\sigma_a \approx \frac{sin(\alpha)r}{cos(\varphi)}$ and $\sigma_e \approx \frac{sin(\beta)r}{cos(\varphi)}$. σ_r, σ_a and σ_e represent the probability distributions related to range, azimuth, and elevation respectively. In contrast to the original APDGICP, the covariance matrix is not only related to the radial distance but also to the angle of arrival. By adding another term $\frac{1}{cos(\varphi)}$, the anisotropy of the radar measurement can be modelled in scan registration.

C. IMU Pre-integration

In our framework, the estimation of the continuous state of the radar is achieved by propagating the IMU data. Should

there be any changes in the previous state, repropagation becomes imperative. However, repropagation over the entire sliding window is computationally intensive. Inspired by [18], we adopt IMU pre-integration to improve the efficiency and effectiveness of state estimation. The measurements of angular velocity and acceleration from an IMU are defined using (6):

$$\hat{\boldsymbol{\omega}}_t = \boldsymbol{\omega}_t + \mathbf{b}_t^{\boldsymbol{\omega}} + \mathbf{n}_t^{\boldsymbol{\omega}} \tag{6}$$

$$\hat{\mathbf{a}}_t = \mathbf{R}_t^{\mathbf{wb}}(\mathbf{a}_t - \mathbf{g}) + \mathbf{b}_t^{\mathbf{a}} + \mathbf{n}_t^{\mathbf{a}} \tag{7}$$

where $\hat{\boldsymbol{\omega}}_t$ and $\hat{\mathbf{a}}_t$ are the raw IMU measurements in $\{\}^b$ at time t.$\hat{\boldsymbol{\omega}}_t$ and $\hat{\mathbf{a}}_t$ are influenced by the slowly varying bias $\mathbf{b}_t$ and white noise $\mathbf{n}_t \cdot \mathbf{R}_t^{\mathrm{wb}}$ is the rotation matrix from $\{\}^w$ to $\{\}^b$ and it is abbreviated here as $\mathbf{R}_t$. $\mathbf{g}$ is the constant gravity vector in $\{\}^w$. Assume that the angular velocity and acceleration remain constant within a small integration time Δt, the velocity, position and rotation of the robot at time $t + \Delta t$ can be easily derived from (6)(7):

$$\begin{aligned}
\mathbf{v}_{t+\Delta t} &= \mathbf{v}_t + \mathbf{g}\Delta t + \mathbf{R}_t\left(\hat{\mathbf{a}}_t - \mathbf{b}_t^{\mathbf{a}} - \mathbf{n}_t^{\mathbf{a}}\right)\Delta t \\
\mathbf{p}_{t+\Delta t} &= \mathbf{p}_t + \mathbf{v}_t\Delta t + \frac{1}{2}\mathbf{g}\Delta t^2 + \frac{1}{2}\mathbf{R}_t\left(\hat{\mathbf{a}}_t - \mathbf{b}_t^{\mathbf{a}} - \mathbf{n}_t^{\mathbf{a}}\right)\Delta t^2 \\
\mathbf{R}_{t+\Delta t} &= \mathbf{R}_t \exp\left(\left(\hat{\boldsymbol{\omega}}_t - \mathbf{b}_t^{\boldsymbol{\omega}} - \mathbf{n}_t^{\boldsymbol{\omega}}\right)\Delta t\right)
\end{aligned} \tag{8}$$

We define the noise from t_i to t_j as $\delta\mathbf{v}_{ij}$,$\delta\mathbf{p}_{ij}$,$\delta\phi_{ij}$. The relative body motion between two timestamps can be then calculated by:

$$\begin{aligned}
\Delta\mathbf{v}_{ij} &= \mathbf{R}_i^\top\left(\mathbf{v}_j - \mathbf{v}_i - \mathbf{g}\Delta t_{ij}\right) + \delta\mathbf{v}_{ij} \\
\Delta\mathbf{p}_{ij} &= \mathbf{R}_i^\top\left(\mathbf{p}_j - \mathbf{p}_i - \mathbf{v}_i\Delta t_{ij} - \frac{1}{2}\mathbf{g}\Delta t_{ij}^2\right) + \delta\mathbf{p}_{ij} \\
\Delta\mathbf{R}_{ij} &= \mathbf{R}_i^\top\mathbf{R}_j\exp(\delta\phi_{ij})
\end{aligned} \tag{9}$$

The noise term from t_i to t_j is then normalized linearly:

$$\begin{aligned}
\delta\mathbf{v}_{ij} &= \delta\mathbf{v}_{ij-1} - \Delta\hat{\mathbf{R}}_{i,j-1}(\hat{\mathbf{a}}_{j-1} - \mathbf{b}_i^{\mathbf{a}})^\wedge\delta\phi_{i,j-1}\Delta t \\
&\quad + \Delta\hat{\mathbf{R}}_{i,j-1}\mathbf{n}_{j-1}^{\mathbf{a}}\Delta t \\
\delta\mathbf{p}_{ij} &= \delta\mathbf{p}_{ij-1} + \delta\mathbf{v}_{ij-1}\Delta t - \frac{1}{2}\Delta\hat{\mathbf{R}}_{i,j-1} \\
&\quad (\hat{\mathbf{a}}_{j-1} - \mathbf{b}_i^{\mathbf{a}})^\wedge\delta\phi_{ij-1}\Delta t^2 + \frac{1}{2}\hat{\mathbf{R}}_{i,j-1}\mathbf{n}_{j-1}^{\mathbf{a}}\Delta t^2 \\
\delta\phi_{ij} &= \Delta\hat{\mathbf{R}}_t^\top\delta\phi_{ij-1} + J_{r,j-1}\mathbf{n}_t^{\boldsymbol{\omega}}\Delta t
\end{aligned} \tag{10}$$

Applying the above IMU pre-integration model not only improves the computational efficiency of the system, but also adds the IMU pre-integration factor to graph optimization.

D. Loop Detection

Inspired by [7] [6], our approach modifies the ScanContext [27], originally designed for LiDAR SLAM, to facilitate radar-based loop detection. The original ScanContext methodology segments a LiDAR point cloud into bins, using the highest point in each bin to transform the entire cloud into a representative image. However, due to sparsity of 4D radar scans, we have adapted the method to utilize the maximum intensity of the radar points for encoding the point cloud, as high-intensity points are more likely to correspond to stable features in the environment. Upon identifying a loop, the system calculates the relative pose and covariance matrix between the current keyframe and the corresponding matching keyframe through scan registration. In addition, we use the ground feature as a higher-weight constraint to reduce false detection. This adjustment ensures more reliable and accurate loop detection in radar SLAM environments.

E. Graph Optimization

As shown in Fig. 4, the factor graph is deployed for back-end optimization, which incorporates six principal components: scan-to-scan registration factors, scan-to-submap correction factors, IMU pre-integration factors, ego velocity factors, ground factors and loop closure factors. The state of each graph node is denoted by the rotation $\mathbf{R}$, position $\mathbf{p}$ in $\mathbf{SE(3)}$, velocity $\mathbf{v^b}$, bias of acceleration $\mathbf{b^a}$, angular velocity $\mathbf{b}^\omega$.

- **scan-to-scan registration factors** are created by scan-to-scan matching with the rotation $\mathbf{R_{ss}}$ and position $\mathbf{p_{ss}}$.
- **scan-to-submap correction factors** are generated through scan-to-submap correction, characterized by the rotation $\mathbf{R_{sm}}$ and position $\mathbf{p_{sm}}$.
- **IMU pre-integration factors** are established between consecutive frames to assist in predicting the sensor pose and to maintain the factor graph well-constrained, especially under conditions of point cloud degeneration.
- **ego velocity factors** are created between continuous frames based on the radar ego motion estimation to constrain the IMU pre-integration and correct the bias.
- **ground factors** are obtained by detecting and fitting the ground as a plane equation $\mathbf{Ax} + \mathbf{By} + \mathbf{Cy} + \mathbf{D} = 0$ to minimize the error in z-axis.
- **loop closure factors**, determined by the intensity ScanContext as relative pose between two nodes, are added to the graph to reduce cumulative drift.

The implementation of a sliding window mechanism serves to enhance both the accuracy and efficiency of the system. By constructing sub-maps from radar scans within this window, denser environmental information is obtained, facilitating more effective scan-to-submap matching. Simultaneously, the mechanism of continuously updating and optimizing a selection of the most recent data guarantees that the system remains both responsive and precise in its operations over time. This approach also ensures optimal use of computational resources, striking a balance between detailed environmental mapping and the efficient processing of data. The factor graph is optimized by the g2o library [28] to obtain a refined pose and velocity.

IV. EXPERIMENT EVALUATION

In order to evaluate the performance of RIV-SLAM on different scenarios and different sensor platforms, we conducted experiments on the following open-source datasets:

- **NTU4DRadLM Dataset** [29]: Two platforms are utilized for data collection, one handcart and one car, equipped with Oculii Eagle 4D Radar and VectorNav VN100 IMU. An external Ublox GPS sensor is also mounted on the

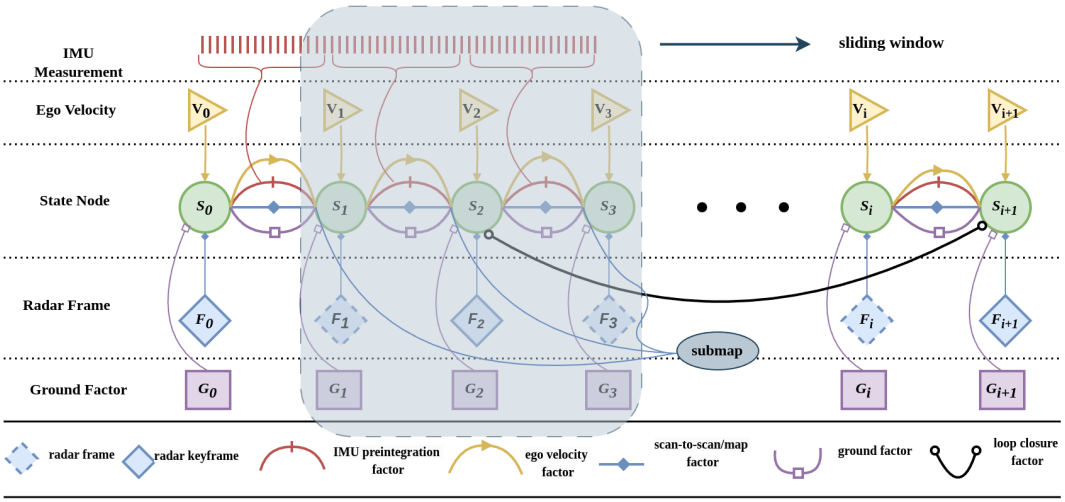


Fig. 4: Graph Optimization Structure of RIV-SLAM: Six types of factors are introduced to construct the factor graph in a sliding window: (a) scan-to-scan registration factor, (b) scan-to-submap correction factor, (c) IMU pre-integration factor, (d) ego velocity factor, (e) ground factor, (f) loop closure factor

sensor platform, which provides a reference value for ego velocity estimation.

- **Kvarntorp Dataset** [30]: Two sequences of data under harsh environmental conditions (one on underground mining and the other on forest) are captured via Sensrad Hugin Radar sensor and Xsens MTi-30 IMU.
- **4D Radar Dataset** [7]: This dataset is collected with a ZF FRGen21 4D Radar and a NovAtel GNSS on campus.

The toolbox *EVO* [31] is used for quantitative trajectory evaluation, i.e., Absolute Pose Error (APE) and Relative Pose Error (RPE).

A. Comparison With the State-of-the-Art Method

As shown in TABLE I, we mainly compare our system with APDGICP from [6], which is currently the only open-source full Radar SLAM system to the best of our knowledge. In the first dataset from NTU4DRadLM, data is collected using a handheld device moving at low speed (maximum speed of $1\,\mathrm{m\,s^{-1}}$) in a park lot at NTU university campus. Therefore, there are two main challenges for Radar SLAM, one is the removal of dynamic objects (pedestrians and vehicles), and the other is the drift along the z-axis. Fig. 5c showcases the estimated trajectories by APDGICP and the proposed system, alongside the ground truth generated by LiDAR-Visual-Inertial SLAM. As shown in Fig. 5e, the pose drift across the x, y, and z axes illustrates how the system's accuracy varies in three-dimensional space over time. Both methods are robust under dynamic environments, but the ground extraction of RIV-SLAM significantly improves the pose accuracy in the z-axis. Apart from that loop detection provides a great enhancement to both approaches. This observation is further supported by the results on the car platform. Accurate GPS data is also available in this sequence, so we compare the ego motion estimation produced by our system with the GPS data, and the average relative error is less than 2% for up to $4.5\,\mathrm{km}$ of urban driving, which also shows that our system is capable of outputting accurate ego velocity.

The results of the evaluation in the mine and forest datasets are presented in Fig. 5h and Fig. 5k. However, since APDGICP failed in both cases, only RIV-SLAM and ground truth were compared. The possible reasons for the failure of APDGICP in forest and mine environments are listed as follows: *a)* Lack of structured features in forest environments for scan matching estimation. *b)* The low density and high variance of radar data in forests leads to a rapid bias. *c)* Although structured features are provided in the mine environment, the repetition of monolithic features (rock walls and tunnels) results in a number of false matches. In contrast, RIV-SLAM with ground extraction, IMU constraints and velocity factors performs extremely well in both of these contexts. We also note that the z-axis pose is shifted considerably (nearly $30\,\mathrm{m}$) during the $4.5\,\mathrm{km}$ long mine dataset, and this will be the focus of our future work to further reduce the z-axis error using other sensor constraints.

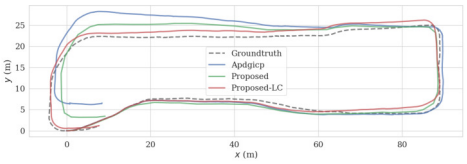
TABLE I: RPE and APE on different datasets

Approaches	Apdgicp			Proposed-LC		
Datasets	t_{rpe} (m)	t_{rpe} (%)	t_{ape} (m)	t_{rpe} (m)	t_{rpe} (%)	t_{ape} (m)
parking lot	4.26	0.0469	3.61	2.58	0.0342	**2.25**
urban road	4.89	**0.0397**	59.12	2.69	0.0456	**32.62**
forest	-	-	-	1.26	0.0125	4.52
mine	-	-	-	2.65	0.0762	58.62

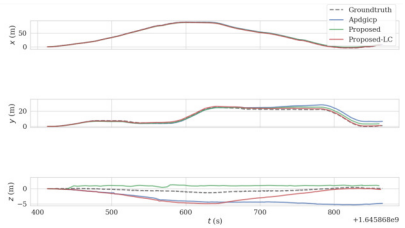
B. Real-Time Performance

The real-time performance of our proposed system is evaluated by comparing the mean computation time for each component in all datasets. In general, with the benefit of sliding window optimization efficiency, the complete system is able to output real-time pose, ego velocity estimation, and 3D radar maps at a rate of $4\,\mathrm{Hz}$ (the size of the sliding window is set to 20 with AMD R5-5600X CPU and 32GB

(a) top-down view of parking lot, figure adopted from [29]

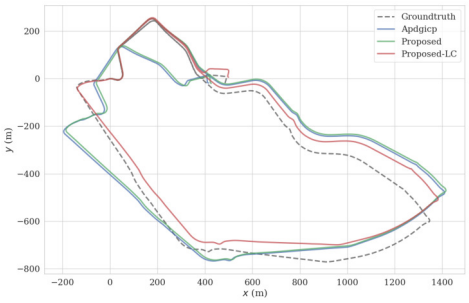


(b) estimated trajectories of proposed w/o Loop Closure, APDGICP, and ground truth

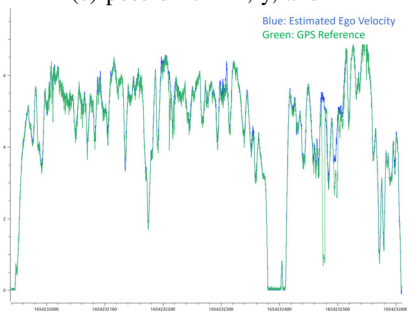


(c) pose drift in x, y, and z

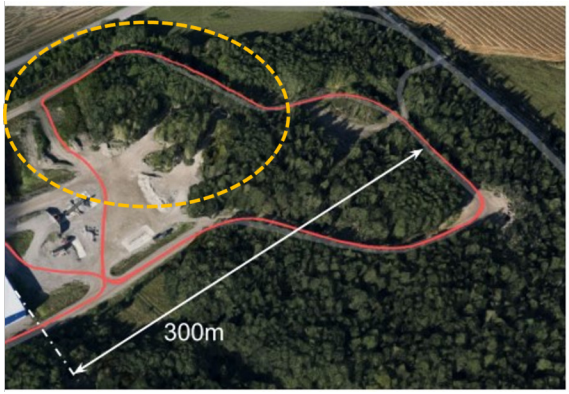
(d) top-down view of urban road, figure adopted from [29]



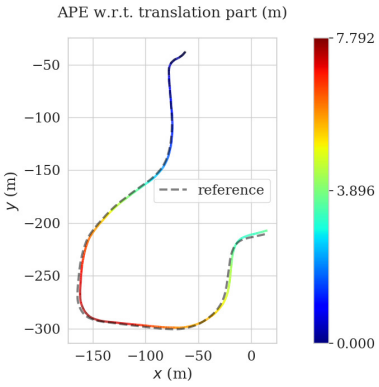
(e) estimated trajectories



(f) estimated ego velocity and GPS reference

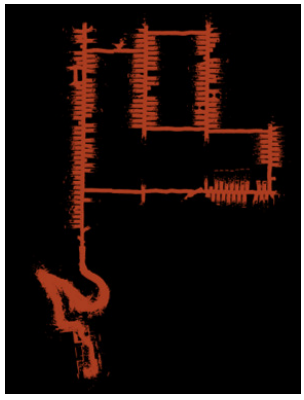


(g) top-down view of forest and sensor platform, figure adopted from [30]

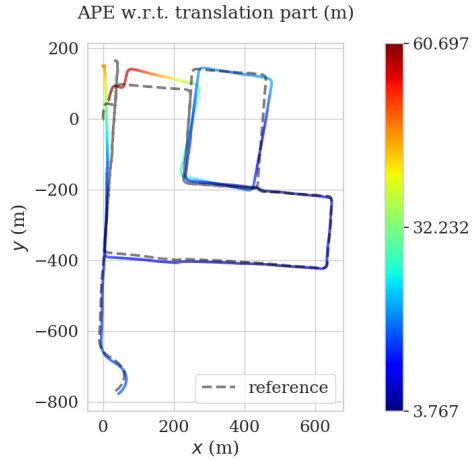


(h) absolute pose error with RIV-SLAM in forest environment

(i) pick-up truck driving through the Mine with the sensor platform, figure adopted from [30]



(j) LiDAR map of mine environment (rock walls and tunnels)



(k) absolute pose error with RIV-SLAM in mine environment

Fig. 5: Estimated trajectories of proposed RIV-SLAM, APDGICP, and ground truth

RAM). As the distance of the trajectory increases, the time taken for loop detection grows rapidly, and this is particularly obvious with long datasets.

V. CONCLUSIONS

In this paper, we introduce RIV-SLAM by effectively integrating 4D radar data, IMU measurement, and Doppler

velocity information into a cohesive graph-based framework. This combination is demonstrated to address some of the fundamental challenges associated with 4D radar-based SLAM, including handling sparse and noisy data, as well as improving z-axis accuracy and ego-motion estimation under adverse conditions. Our system's novel approach to optimizing radar Doppler velocity with IMU data has proven critical in achieving accurate and stable velocity estimations, a key differentiator from existing solutions. Furthermore, the introduction of a ground extraction method tailored for 4D radar data and a hybrid scan registration strategy significantly enhances the system's accuracy and robustness. Evaluations conducted on various datasets have demonstrated that the RIV-SLAM system substantially outperforms current state-of-the-art Radar-SLAM frameworks, particularly in challenging environments where the precision of movement and environmental understanding are paramount. The system's ability to produce optimized ego velocity outputs, even in scenarios like wheel tachometer or GPS failure, underscores its utility and innovation in the realm of autonomous navigation and mapping technologies. Future works will extend the proposed system to include more sensor modalities, such as cameras, GPS and so on.

REFERENCES

[1] S. H. Cen and P. Newman, "Precise ego-motion estimation with millimeter-wave radar under diverse and challenging conditions," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6045–6052.

[2] K. Burnett, Y. Wu, D. J. Yoon, A. P. Schoellig, and T. D. Barfoot, "Are we ready for radar to replace lidar in all-weather mapping and localization?" *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 328–10 335, 2022.

[3] 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 Robotics and Automation Letters*, vol. 6, no. 4, pp. 7691–7698, 2021.

[4] C.-Y. Ho, S.-C. Hsieh, M.-F. Jhong, C.-C. Wang, and C.-Y. Ting, "A 77ghz antenna-in-package with low-cost solution for automotive radar applications," in *2018 IEEE 68th Electronic Components and Technology Conference (ECTC)*. IEEE, 2018, pp. 191–196.

[5] Z. Hong, Y. Petillot, and S. Wang, "Radarslam: Radar based large-scale slam in all weathers," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5164–5170.

[6] J. Zhang, H. Zhuge, Z. Wu, G. Peng, M. Wen, Y. Liu, and D. Wang, "4dradarslam: A 4d imaging radar slam system for large-scale environments based on pose graph optimization," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8333–8340.

[7] X. Li, H. Zhang, and W. Chen, "4d radar-based pose graph slam with ego-velocity pre-integration factor," *IEEE Robotics and Automation Letters*, 2023.

[8] H. Lim, K. Han, G. Shin, G. Kim, S. Hong, and H. Myung, "Orora: Outlier-robust radar odometry," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2046–2053.

[9] A. Fischer, Z. Tong, A. Hamidipour, L. Maurer, and A. Stelzer, "77-ghz multi-channel radar transceiver with antenna in package," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 3, pp. 1386–1394, 2013.

[10] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using doppler radar," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 869–874.

[11] ——, "Instantaneous ego-motion estimation using multiple doppler radars," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1592–1597.

[12] D. Wang, M. Masannek, S. May, and A. Nüchter, "Infradar-localization: single-chip infrared-and radar-based monte carlo localization," in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2023, pp. 1–8.

[13] A. Kramer, C. Stahoviak, A. Santamaria-Navarro, A.-A. Agha-Mohammadi, and C. Heckman, "Radar-inertial ego-velocity estimation for visually degraded environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5739–5746.

[14] C. Doer and G. F. Trommer, "An ekf based approach to radar inertial odometry," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2020, pp. 152–159.

[15] ——, "Radar visual inertial odometry and radar thermal inertial odometry: Robust navigation even in challenging visual conditions," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 331–338.

[16] C. Doer, J. Atman, and G. F. Trnmmer, "Gnss aided radar inertial odometry for uas flights in challenging conditions," in *2022 IEEE Aerospace Conference (AERO)*. IEEE, 2022, pp. 1–10.

[17] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[18] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 5692–5698.

[19] Z. Hong, Y. Petillot, and S. Wang, "Radarslam: radar based large-scale slam in all weathers. in 2020 ieee," in *RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5164–5170.

[20] A. Segal, D. Haehnel, and S. Thrun, "Generalized-icp." in *Robotics: science and systems*, vol. 2, no. 4. Seattle, WA, 2009, p. 435.

[21] Y. Zhuang, B. Wang, J. Huai, and M. Li, "4d iriom: 4d imaging radar inertial odometry and mapping," *IEEE Robotics and Automation Letters*, 2023.

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

[23] Y. S. Park, Y.-S. Shin, and A. Kim, "Pharao: Direct radar odometry using phase correlation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2617–2623.

[24] R. Weston, M. Gadd, D. De Martini, P. Newman, and I. Posner, "Fastmbym: Leveraging translational invariance of the fourier transform for efficient and accurate radar odometry," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2186–2192.

[25] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[26] K. Koide, S. Oishi, M. Yokozuka, and A. Banno, "Tightly coupled range inertial localization on a 3d prior map based on sliding window factor graph optimization," *arXiv preprint arXiv:2402.05540*, 2024.

[27] H. Wang, C. Wang, and L. Xie, "Intensity scan context: Coding intensity and geometry relations for loop closure detection," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2095–2101.

[28] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g 2 o: A general framework for graph optimization," in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 3607–3613.

[29] J. Zhang, H. Zhuge, Y. Liu, G. Peng, Z. Wu, H. Zhang, Q. Lyu, H. Li, C. Zhao, D. Kircali *et al.*, "Ntu4dradlm: 4d radar-centric multi-modal dataset for localization and mapping," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 4291–4296.

[30] V. Kubelka, E. Fritz, and M. Magnusson, "Do we need scan-matching in radar odometry?" *arXiv preprint arXiv:2310.18117*, 2023.

[31] M. Grupp, "evo: Python package for the evaluation of odometry and slam." https://github.com/MichaelGrupp/evo, 2017.