Doppler Correspondence: Non-Iterative Scan Matching With Doppler Velocity-Based Correspondence

Jiwoo Kim[*,†,1], Geunsik Bae[*,1], Changseung Kim[1], Jinwoo Lee[1], Woojae Shin[1], and Hyondong Oh[††,1]

Abstract—Achieving successful scan matching is essential for LiDAR odometry. However, in challenging environments with adverse weather conditions or repetitive geometric patterns, LiDAR odometry performance is degraded due to incorrect scan matching. Recently, the emergence of frequency-modulated continuous wave 4D LiDAR and 4D radar technologies have provided the potential to address these unfavorable conditions. The term 4D refers to point cloud data characterized by range, azimuth, and elevation along with Doppler velocity. Although 4D data is available, most scan matching methods for 4D LiDAR and 4D radar still establish correspondence by repeatedly identifying the closest points between consecutive scans, overlooking the Doppler information. This paper introduces, for the first time, a simple Doppler velocity-based correspondence—*Doppler Correspondence*—that is invariant to translation and small rotation of the sensor, with its geometric and kinematic foundations. Extensive experiments demonstrate that the proposed method enables the direct matching of consecutive point clouds without an iterative process, making it computationally efficient. Additionally, it provides a more robust correspondence estimation in environments with repetitive geometric patterns. The implementation of our proposed method is publicly available at https://github.com/Tars0523/Doppler_Correspondence.

I. INTRODUCTION

State estimation is a fundamental component in autonomous systems, enabling robots to achieve tasks accurately and safely [3, 7, 20]. Accurate odometry plays a critical role in reliable state estimation by tracking the robot's movement and position. Therefore, LiDAR odometry has been extensively researched to achieve high accuracy and real-time performance, leveraging its superior precision and detailed environmental sensing capabilities. However, scan matching, which is essential for LiDAR odometry, is degraded in adverse weather conditions such as rain, smoke, and fog [15]. Additionally, most scan matching methods, which depend on the geometry of the environment, often fail to accurately estimate sensor motion in repetitive geometric structures such as tunnels and highways [7].

Recently, the frequency-modulated continuous wave (FMCW) 4D LiDAR and 4D radar have shown great potential in alleviating these limitations. Unlike conventional LiDAR, FMCW-based 4D ranging sensors are more resilient to weather conditions [10, 15]. Moreover, they provide 4D point cloud information (range, azimuth, elevation, and Doppler velocity), which has potential [7, 15] for robust scan matching in scenarios with repetitive geometric structures.

Existing approaches related to correspondence estimation for 4D point cloud scan matching either rely on heuristic utilization

*Equal contributions. †Project lead. ††Corresponding author.
[1]Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, Republic of Korea.
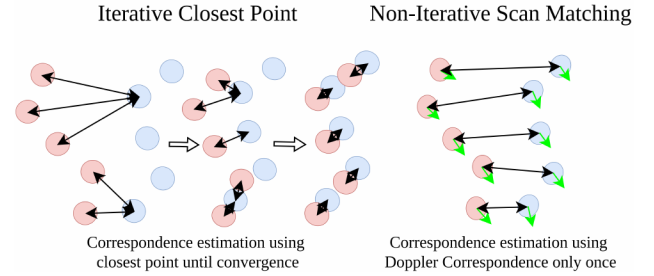Emails: {tars0523, baegs94, pon02124, jinwoolee2021, oj7987, h.oh}@unist.ac.kr

Fig. 1: The red circle denotes source points, the blue circle represents target points, and the green arrow indicates the Doppler velocity of each point. While the ICP method requires multiple iterations to refine correspondences, the proposed method utilizes the direct scan matching approach based on Doppler Correspondence.
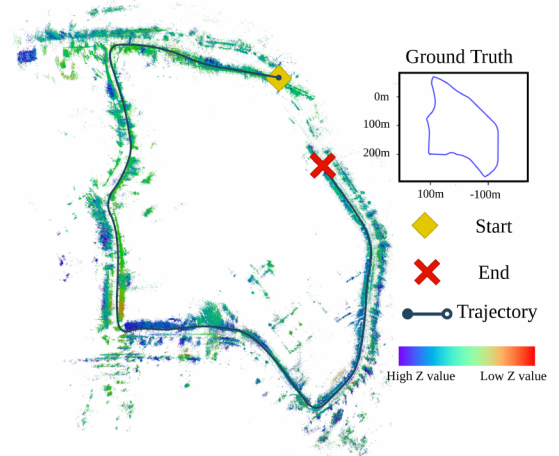

Fig. 2: The result of the proposed non-iterative scan matching method using the *nyl* trajectory in the NTU4DRadLM dataset [22].

of radar cross section (RCS) [8] or leverage the uncertainty inherent in point clouds derived from 3D information [20]. Although FMCW-based 4D ranging sensors provide Doppler velocity but still the correspondence estimation methods typically rely solely on 3D information [6, 7, 11, 13, 14, 23].

In this paper, for the first time, — *Doppler Correspondence*— a simple correspondence that leverages Doppler velocity is introduced. This correspondence is based on the range and Doppler velocity of each point. Unlike the iterative closest point (ICP), scan-matching utilizing Doppler Correspondence directly matches point clouds without an iterative process, as illustrated in Fig. 1, while preserving a reasonable level of odometry

accuracy, as shown in Fig. 2. The key contributions of this work are summarized as follows:

- A simple and novel correspondence utilizing Doppler velocity is proposed for the first time. Geometric and kinematic foundations of this correspondence are derived;
- Doppler Correspondence enables a non-iterative scan matching. Consequently, it significantly reduces time consumption compared with traditional ICP methods;
- Since the proposed correspondence does not depend on geometric cues alone, it remains effective even in the presence of repetitive geometric structures; and
- Scan matching based on Doppler Correspondence achieves performance comparable to ICP methods across diverse scenarios and sensor configurations with less computation time.

II. RELATED WORK

A. 4D Radar Odometry

4D radar odometry and simultaneous localization and mapping (SLAM) methods have gained significant attention due to their robustness in challenging environments, such as visually degraded or bad weather conditions. Early work by Doer and Trommer [3, 4] demonstrated the feasibility of fusing 4D radar data with inertial measurements to achieve 4D radar-inertial odometry (RIO). Using extended kalman filter (EKF)-based approaches, they highlighted improvements in pose estimation through techniques such as RANSAC-based ego-velocity estimation, barometric height fusion, and online 4D radar extrinsic calibration, which removed the need for tedious pre-calibration processes. Building on this foundation, Michalczyk et al. [13, 14] extended RIO methods, incorporating persistent landmark detection and multi-state estimation frameworks to improve performance.

More recently, as 4D radar point clouds become more dense, researchers have focused on optimization-based approaches. Unlike earlier Kalman filter-based methods, which rely on sequential state estimation, optimization-based methods aim to solve the sub-trajectory by considering the global consistency of sensor data. Li et al. [11] developed a pose graph optimization-based 4D radar SLAM framework that utilizes ego-velocity pre-integration factors to improve robustness in noisy 4D radar data. Huang et al. [8] introduced an enhanced RIO that integrates Doppler velocity and RCS information to filter noisy point cloud, adhering to the *less is more* principle. Additionally, Nissov et al. [15] tackled the challenges of LiDAR degeneracy by proposing a LiDAR-4D radar-inertial fusion, leveraging 4D radar's robustness in the fog-filled hallway, while preserving LiDAR accuracy in well-conditioned environments. Herraez et al. [6] presented methods for pose estimation and map creation, highlighting the benefits of velocity-aided odometry and map filtering to enhance accuracy. Zhang et al. [23] and Xu et al. [20] further advanced 4D radar odometry by modeling point uncertainty in polar coordinates, integrating into data association and motion estimation for improved performance in adverse conditions.

B. 4D LiDAR Odometry

4D LiDAR is an emerging technology that enables Doppler velocity measurements, offering advantages in environments with geometric degeneracy or dynamic objects. Several recent studies have explored the potential of this novel sensing modality for odometry and SLAM tasks. Wu et al. [19] introduced the first continuous-time 4D LiDAR-only odometry method leveraging Doppler velocity measurements. Their method employs Gaussian process regression to estimate vehicle trajectories and correct motion distortion caused by scanning, significantly outperforming existing methods. Similarly, Hexsel et al. [7] proposed DICP, which integrates Doppler velocity measurements into the ICP framework for robust point cloud registration. By jointly optimizing Doppler and geometric objective functions, their method demonstrated improved registration accuracy and convergence speed, particularly in featureless environments such as tunnels and hallways. Yoon et al. [21] and Lisus et al. [12] developed correspondence-free methods that estimate 6-DOF velocity directly from Doppler information without requiring explicit scan matching and incorporate multiple sensors or complementary modules such as IMUs or gyroscopes for full motion estimation.

C. Summary and Limitations of 4D Radar / LiDAR Odometry

The 4D information of FMCW 4D LiDAR and 4D radar has leveraged its advantages in three main directions. First, Doppler velocity has been effectively incorporated into optimization frameworks to improve accuracy and robustness in challenging environments, such as repetitive geometric settings, foggy conditions, or a significant number of dynamic objects [3, 4, 6, 7, 11, 13, 14, 15, 19, 23]. Second, scan matching methods have been developed by refining point correspondences, using RCS information [8] or the uncertainty of point clouds [20]. Third, direct motion estimation methods [21, 12] attempt to recover full 6-DOF motion from Doppler information without relying on point correspondences.

While recent advances have shown promising results, most existing correspondence-based methods still rely solely on 3D geometric information, overlooking the rich Doppler velocity available in 4D LiDAR or radar data. Also, correspondence-free approaches estimate 6-DOF motion directly from Doppler measurements but suffer from limited rotational observability, requiring multiple LiDAR sensors or an additional gyroscope to compensate. In this work, we address these limitations by proposing a novel correspondence formulation that explicitly leverages Doppler information. Unlike prior methods, our approach associates Doppler measurements not only with translational motion but also with rotational motion from a single 4D sensor.

The remaining sections are organized as follows. In Section III, Doppler Correspondence is introduced in detail, including its geometric derivation and the rationale behind its robustness in repetitive geometric environments. Section IV presents the experimental results, demonstrating the effectiveness of the proposed method across various datasets and comparing its performance against point-to-point ICP and DICP. In Section V, the limitations of the approach are introduced, addressing scenarios where the method may face challenges. In Section VI, Doppler Correspondence is integrated with ICP to compare its performance with ICP and DICP, demonstrating its potential applicability. Finally, Section VII provides the conclusion, summarizing the key contributions of the work and

its potential for the field of 4D ranging sensor-based odometry.

III. METHODOLOGY

A. Problem Definition

ICP is a widely used method for point cloud registration, aiming to estimate the relative transformation between two consecutive scans. Let us consider two consecutive point cloud scans, $\mathcal{P} = \{p_1, p_2, \ldots, p_N\}$ and $\mathcal{Q} = \{q_1, q_2, \ldots, q_M\}$, where $\mathcal{P}$ and $\mathcal{Q}$ contain N and M points, respectively. Assume the sensor undergoes a rotation $\Delta R \in SO(3)$ and a translation $\Delta t \in \mathbb{R}^3$ between scans. To estimate ΔR and Δt, the three-step process is applied iteratively as below.

1) **Initialization**: The initial predictions for the rotation $\Delta \hat{R}$ and translation $\Delta \hat{t}$ are set to $I_{3 \times 3}$ and $\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$, respectively.

2) **Correspondence Estimation**: Identify correspondences $\mathcal{C}$ by determining the closest points between the two point cloud scans.

$$\mathcal{C} = \left\{ (p, q) \,\middle|\, q = \arg\min_{q \in \mathcal{Q}} \|p - q\|_2, \, \forall p \in \mathcal{P} \right\},$$

where p and $q \in \mathbb{R}^3$.

3) **Optimization**: Compute δR and δt that minimize the following loss function:

$$(\delta R, \delta t) = \arg\min_{\delta R, \delta t} \sum_{(p,q) \in \mathcal{C}} \|q - (\delta R p + \delta t)\|_2.$$

4) **Update**: Update the $\Delta \hat{R}$, $\Delta \hat{t}$ and source point cloud $\mathcal{P}$:

$$\Delta \hat{R} \leftarrow \delta R \Delta \hat{R}, \quad \Delta \hat{t} \leftarrow \delta t + \Delta \hat{t},$$

$$p \leftarrow \delta R p + \delta t, \quad \forall p \in \mathcal{P}.$$

Correspondence Estimation, **Optimization** and **Update** are repeated until convergence is achieved, typically defined by a threshold on the change in the loss function $\|q - (\delta R\, p + \delta t)\|^2$ between iterations.

Mostly, the **Correspondence Estimation** step solely relies on the spatial information of each point cloud. Relying only on the spatial information in **Correspondence Estimation** makes registration prone to degradation in environments with repetitive geometric structures [7, 15] or when consecutive scans fail to capture the same spatial points due to its sparse and noisy characteristics [8]. To address these limitations, this study proposes Doppler Correspondence for the **Correspondence Estimation**, which is explained from the following in detail.

B. Geometric Derivation of Doppler Correspondence

Let us denote points $p_i \in \mathcal{P}$ and $q_j \in \mathcal{Q}$ are the true corresponding points, represented as:

$$p_i = \begin{bmatrix} x_{p,i} & y_{p,i} & z_{p,i} \end{bmatrix}^\top \quad \text{and} \quad q_j = \begin{bmatrix} x_{q,j} & y_{q,j} & z_{q,j} \end{bmatrix}^\top.$$

Here, (x, y, z) are the spatial coordinates of the point. Besides, let $v_{p,i}$ and $v_{q,j}$ be the Doppler velocity of each point. Assuming that, during a short time interval ΔT between two consecutive scans, the trajectory of the point can be approximated by the vector $\overrightarrow{PQ}$, as shown in Fig. 3. Decomposing $\overrightarrow{PQ}$ into radial (r) and tangential (θ) components yields:

$$\overrightarrow{PQ} = \overrightarrow{PR} + \overrightarrow{RQ},$$
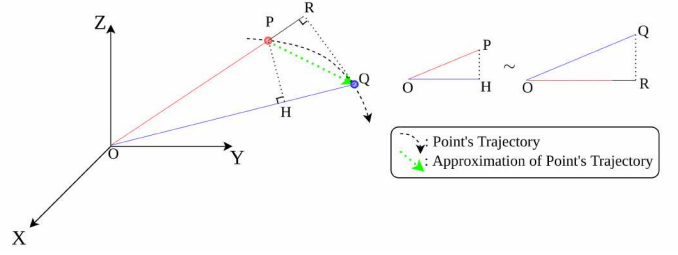

Fig. 3: Two corresponding points across consecutive scans. The radial and tangential velocity components form the basis of our Doppler Correspondence derivation.

where $\overrightarrow{PR}$ and $\overrightarrow{RQ}$ represent the radial and tangential components, respectively. Expressing this in terms of velocity:

$$\overrightarrow{V} \Delta T = \overrightarrow{V}_{p,r} \Delta T + \overrightarrow{V}_{p,\theta} \Delta T,$$

where $\overrightarrow{V}$ is the average velocity over $\overrightarrow{PQ}$, and $\overrightarrow{V}_{p,r}$ and $\overrightarrow{V}_{p,\theta}$ are the radial and tangential velocities at point p_i, respectively. Similarly, for q_i:

$$\overrightarrow{PQ} = \overrightarrow{HQ} + \overrightarrow{PH},$$

and:

$$\overrightarrow{V} \Delta T = \overrightarrow{V}_{q,r} \Delta T + \overrightarrow{V}_{q,\theta} \Delta T.$$

Using the similarity condition $\triangle POH \sim \triangle QOR$ (i.e., side-angle-side similarity), the following proportional relationship holds:

$$\frac{|\overrightarrow{OR}|}{|\overrightarrow{OQ}|} = \frac{|\overrightarrow{OH}|}{|\overrightarrow{OP}|}.$$

Expanding this gives:

$$\frac{|\overrightarrow{OP}| + |\overrightarrow{PR}|}{|\overrightarrow{OQ}|} = \frac{|\overrightarrow{OQ}| - |\overrightarrow{HQ}|}{|\overrightarrow{OP}|}. \tag{1}$$

Substituting $|\overrightarrow{OP}| = r_{p,i}$, $|\overrightarrow{OQ}| = r_{q,j}$, $|\overrightarrow{PR}| = v_{p,i} \Delta T$, and $|\overrightarrow{HQ}| = v_{q,j} \Delta T$, where $r_{p,i}$ and $r_{q,j}$ are the ranges of p_i and q_j respectively, and $v_{p,i}$ and $v_{q,j}$ are their respective Doppler velocities, into Eq. (1) provides:

$$\frac{r_{p,i} + v_{p,i} \Delta T}{r_{q,j}} = \frac{r_{q,j} - v_{q,j} \Delta T}{r_{p,i}}.$$

Rearranging and simplifying the above equation yields:

$$r_{p,i}^2 + r_{p,i} v_{p,i} \Delta T = r_{q,j}^2 - r_{q,j} v_{q,j} \Delta T. \tag{2}$$

This can be expressed as:

$$f(p_i, v_{p,i}) = g(q_j, v_{q,j}). \tag{3}$$

Note that the above relationship, termed as *Doppler Correspondence*, is formulated such that the left-hand side, $f(p_i, v_{p,i})$, depends only on the information from point p_i, while the right-hand side, $g(q_j, v_{q,j})$, depends only on the information from point q_j. Furthermore, since there are no terms involving the Δt, this Doppler Correspondence is inherently invariant to translation of the sensor. Moreover, unlike traditional correspondence that relies only on spatial (x, y, z) information, this method additionally utilizes Doppler information. This enables a more robust and reliable matching process in repetitive geometric environments, as it relies more on the flow of the point cloud [2] rather than the spatial information from the environments.

C. Kinematic Derivation of Doppler Correspondence

Alternative to the previous geometric derivation of Doppler Correspondence, this section utilizes manipulation of simple matrices and vectors. To establish Doppler Correspondence, we start with the kinematic relationship from [2, 25]:

$$\frac{q_j - p_i}{\Delta T} \cdot \frac{p_i}{\|p_i\|_2} \approx v_{p,i}, \tag{4}$$

$$\frac{q_j - p_i}{\Delta T} \cdot \frac{q_j}{\|q_j\|_2} \approx v_{q,j}. \tag{5}$$

Here, each left-hand side of Eqs. (4) and (5), represents the radial velocity computed relative to p_i and q_j, respectively, while each right-hand side corresponds to the Doppler velocity measured by the sensor. By clearing denominators, we obtain:

$$(q_j - p_i) \cdot p_i = v_{p,i} \Delta T \|p_i\|_2, \tag{6}$$

$$(q_j - p_i) \cdot q_j = v_{q,j} \Delta T \|q_j\|_2. \tag{7}$$

Adding Eqs. (6) and (7) yields:

$$(q_j - p_i) \cdot (q_j + p_i) = v_{p,i} \Delta T \|p_i\|_2 + v_{q,j} \Delta T \|q_j\|_2.$$

Since $(q_j - p_i) \cdot (q_j + p_i) = \|q_j\|_2^2 - \|p_i\|_2^2$, $\|p_i\|_2 = r_{p,i}$, and $\|q_j\|_2 = r_{q,j}$, where $r_{p,i}$ and $r_{q,j}$ are the ranges of p_i and q_j, respectively, substituting these into the above equation leads to:

$$r_{q,j}^2 - r_{p,i}^2 = v_{p,i} r_{p,i} \Delta T + v_{q,j} r_{q,j} \Delta T.$$

Rearranging terms, we obtain the final form:

$$r_{p,i}^2 + r_{p,i} v_{p,i} \Delta T = r_{q,j}^2 - r_{q,j} v_{q,j} \Delta T.$$

which matches Eq. (2).

D. Outlier Rejection

A pair of points $p_i \in \mathcal{P}$ and $q_j \in \mathcal{Q}$ that satisfy Eq. (3) can be considered as a correspondence. However, ambiguity could arise when multiple candidate points, such as q_j and its symmetric counterpart $q_j' \in \mathcal{Q}$ with respect to the sensor frame, satisfy Doppler Correspondence for a single p_i. In this case, both points yield $f(p_i, v_{p,i}) = g(q_j, v_{q,j}) = g(q_j', v_{q',j})$, as shown in Fig. 4(a).

This occurs because $g(q_j, v_{q,j})$ is identical to $g(q_j', v_{q',j})$, as the range and radial velocity of each point is the same ($r_{q_j} = r_{q_j'}$ and $v_{q_j} = v_{q_j'}$). Such ambiguity can result in incorrect correspondence estimation. To address this issue, it is assumed that correctly matched points p_i and q_j lie within similar spatial regions between scans. This assumption enables the use of a distance-based rejection mechanism to filter out mismatched candidates, as shown in Fig. 4(b). Similar to the maximum correspondence distance used in [17], a threshold is applied to eliminate unlikely matches. Specifically, both the spatial distance $d_{\text{spatial}} = \|p_i - q_j\|_2$ and the Doppler distance $d_{\text{doppler}} = |f(p_i, v_{p,i}) - g(q_j, v_{q,j})|$ should satisfy:

$$d_{\text{spatial}} \leq \tau_{\text{spatial}} \quad \text{and} \quad d_{\text{doppler}} \leq \tau_{\text{doppler}}.$$

Any pair of points that significantly deviates from these constraints is considered an outlier and is removed from the set of correspondences. By enforcing these thresholds, the impact of false correspondence estimation is mitigated.

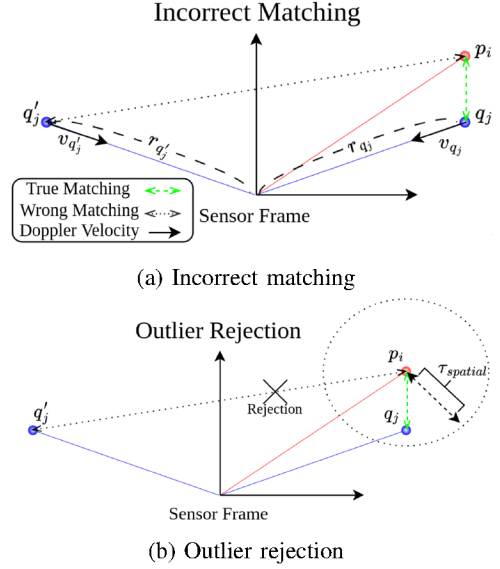(a) Incorrect matching

(b) Outlier rejection

Fig. 4: (a) illustrates multiple candidates due to symmetry in range and Doppler velocity, and (b) shows that the distance-based outlier rejection method filters out mismatched pairs.

E. Non-iterative Scan Matching Algorithm

In this section, a simple registration method leveraging Doppler Correspondence is introduced. This method is designed to evaluate the feasibility of the proposed correspondence for odometry. The framework is intentionally designed to remain lightweight to isolate and emphasize the impact of the proposed Doppler Correspondence. To focus solely on demonstrating the effectiveness of this correspondence, techniques such as robust kernels and RANSAC are deliberately omitted from the algorithm. Further details are provided in Algorithm 1.

The process begins by receiving two consecutive scans, $\mathcal{P}$ and $\mathcal{Q}$, and calculating $f(p_i, v_{p,i})$ and $g(q_j, v_{q,j})$. FindClosest is then performed to estimate correspondences. Unlike the conventional ICP explained in Algorithm 2, which relies on 3D-based correspondence, our method operates on 1D-based correspondence. This could reduce computational complexity, even though $f(p_i, v_{p,i})$ and $g(q_j, v_{q,j})$ need to be calculated. While this approach may be slightly less accurate than 3D-based methods due to less information, it offers faster computations. Furthermore, unlike the closest point correspondence methods [1, 7, 16, 17], which iteratively refine correspondences through multiple iterations, Algorithm 1 performs a direct correspondence estimation in a single step. This eliminates the iterative search process, significantly reducing the computational load, as illustrated in Fig. 1. Next section, the feasibility of Doppler Correspondence is validated by comparing the proposed method against ICP-based approaches.

IV. EXPERIMENT AND DISCUSSION

A. Dataset

To evaluate the effectiveness of Doppler Correspondence for odometry, experiments are conducted on datasets with varying point cloud densities. The datasets include sparse 4D radar, semi-dense 4D radar, and dense 4D LiDAR point clouds. This diversity allows us to examine the performance of our approach across different sensing conditions.

Algorithm 1 Non-Iterative Scan Matching Algorithm Based on Doppler Correspondence

---

Require: $\mathcal{P}$ (source point cloud), $\mathcal{Q}$ (target point cloud), τ_{spatial} (spatial distance threshold), τ_{doppler} (doppler distance threshold)

Ensure: $\hat{R}, \hat{t}$

 Compute $f(\mathcal{P}), g(\mathcal{Q})$
 $\mathcal{Q}' \leftarrow \texttt{FindClosest}(f(\mathcal{P}), g(\mathcal{Q}))$
 Masking Outliers:
  $\mathcal{M} \leftarrow \|\mathcal{P} - \mathcal{Q}'\|_2 \leq \tau_{\text{spatial}}$ and $|f(\mathcal{P}) - g(\mathcal{Q}')| \leq \tau_{\text{doppler}}$
  $\mathcal{C} \leftarrow \mathcal{P}[\mathcal{M}], \mathcal{Q}'[\mathcal{M}]$
 Predict Transformation:
  $\hat{R}, \hat{t} \leftarrow \underset{R,t}{\arg\min} \sum_{(p,q) \in \mathcal{C}} \|q - (Rp + t)\|_2$
  return $\hat{R}, \hat{t}$

Algorithm 2 Iterative Closest Point Algorithm

Require: $\mathcal{P}$ (source point cloud), $\mathcal{Q}$ (target point cloud), τ_{spatial} (spatial distance threshold)

Ensure: $\hat{R}, \hat{t}$

 while not converged **do**
  $\mathcal{Q}' \leftarrow \texttt{FindClosest}(\mathcal{P}, \mathcal{Q})$
  Masking Outliers:
   $\mathcal{M} \leftarrow \|\mathcal{P} - \mathcal{Q}'\|_2 \leq \tau_{\text{spatial}}$
   $\mathcal{C} \leftarrow \mathcal{P}[\mathcal{M}], \mathcal{Q}'[\mathcal{M}]$
  Predict Transformation:
   $\delta R, \delta t \leftarrow \underset{R,t}{\arg\min} \sum_{(p,q) \in \mathcal{C}} \|q - (Rp + t)\|_2$
  Update:
   $\mathcal{P} \leftarrow \texttt{TransformSourcePoint}(\delta R, \delta t, \mathcal{P})$
   $\hat{R}, \hat{t} \leftarrow \texttt{UpdateTransformation}(\delta R, \delta t, \hat{R}, \hat{t})$
 end while
  return $\hat{R}, \hat{t}$

---

1) Sparse 4D Radar: The *Campus* dataset [11] is utilized to evaluate sparse 4D radar point clouds. The data were collected using a ZF FRGen21 4D radar sensor mounted on the front bumper of a vehicle. Each 4D radar scan generates a sparse and noisy point cloud, containing approximately 400 to 1,400 points every 60 ms. The *Campus* dataset encompasses outdoor environments featuring trees, vehicles, and pedestrians. These diverse settings challenge methods to operate effectively under low-density radar point cloud conditions.

2) Semi-Dense 4D Radar: The *Loop 1* dataset from the NTU4DRadLM dataset [22] is used to evaluate semi-dense 4D radar. The data were collected using a car platform with an Oculii Eagle 4D radar, traversing the NTU campus main roads, resulting in a 6.95 km trajectory. The car moved at an average speed of 25–30 km/h, and the 4D radar scans exhibit semi-dense point clouds with approximately 1,000 to 2,000 points per 80ms. The environment features a mix of structured and semi-structured regions.

3) Dense 4D LiDAR: The *Straight Wall, Curved Wall,* and *Baker-Barry Tunnel* datasets from [7] are utilized to evaluate dense 4D LiDAR. The *Straight Wall* and *Curved Wall* datasets, generated in the CARLA simulator, simulate featureless environments by placing large parallel walls along the trajectory. These datasets provide an effective platform for evaluating

odometry performance, particularly in environments with limited geometric features and repetitive structures. The *Baker-Barry Tunnel* dataset was collected using an Aeva Aeries I FMCW 4D LiDAR sensor. The point clouds from 4D LiDAR are significantly denser than 4D radar data, containing approximately 40,000 to 120,000 points per 100ms, depending on the sequence.

B. Experiment Setup

The proposed method is evaluated by comparing its performance with established methods: point-to-point ICP and DICP [7]. DICP is a novel algorithm that integrates Doppler velocity into the **Optimization** in the ICP framework. It enhances scan matching by jointly optimizing both the closest point term and additional constraints derived from the Doppler velocity of each point. These methods are tested in two conditions: *No Seed* and *With Seed.* In the *No Seed* category, no initial pose estimate is provided, whereas the *With Seed* category uses the estimated pose from the previous scan pair under a constant-velocity motion model assumption [18].

For point-to-point ICP, the parameters are set to the following values: maximum iteration = 100, convergence threshold = 10^{-5}, downsampling factor = 2, and $\tau_{\text{spatial}} = 3$. The maximum iteration defines the maximum number of optimization iterations. The method is considered converged when the change in error falls below the convergence threshold. The point clouds are uniformly downsampled at intervals determined by the downsampling factor, reducing density and computational complexity.

DICP is evaluated using its original parameters from the open-source implementation. Each value is as follows: maximum iteration = 100, convergence threshold = 10^{-5}, downsampling factor = 2, and $\tau_{\text{spatial}} = 0.3$.

The proposed method is evaluated with parameters set to these values: downsampling factor = 1, $\tau_{\text{spatial}} = 3$, and $\tau_{\text{doppler}} = 5$. It does not use maximum iteration or convergence threshold, as it performs optimization in a single step. All experiments are conducted on a system equipped with an Intel Core i7-14700KF CPU.

C. Evaluation Metrics

The performance is evaluated using the relative error (RE) from [24]. This metric standardizes trajectory estimation accuracy, allowing fair comparisons across different methods. The evaluation measures odometry accuracy through segment-wise error analysis. This approach provides detailed insights into the local and global performance of the trajectory estimation. By using shorter sub-trajectories, RE captures local consistency and short-term accuracy, while longer sub-trajectories emphasize global consistency and long-term accuracy. This dual capability makes RE a versatile and informative metric for trajectory analysis.

The computation time for transformation estimation, excluding point cloud pre-processing, is evaluated. A fair assessment of computational efficiency could be ensured by excluding computation time for point cloud pre-processing.

D. Experiment Results

The performance evaluation on sparse 4D radar data compares the proposed method against the standard point-to-point ICP

TABLE I: Each cell shows rotation and translation errors in degrees and meters, respectively for different segment lengths. The final column highlights the average computation time per frame (ms). For each segment, the bold values represent the best performance.

Dataset	Method	Evaluation Units (rotation [deg] / translation [m])						
		8m	16m	24m	32m	40m	48m	Time (ms)
Sparse 4D Radar *Campus1*	(With Seed)point-to-point	3.776 / **0.937**	4.543 / **1.708**	5.223 / **2.452**	5.881 / **3.176**	6.519 / **3.914**	7.144 / **4.654**	5.4
	(No Seed)point-to-point	3.666 / 2.293	4.411 / 4.413	5.028 / 6.436	5.666 / 8.368	6.305 / 10.245	6.895 / 12.087	6.0
	Ours	**3.594** / 1.215	**4.149** / 2.189	**4.468** / 3.048	**4.811** / 3.828	**5.158** / 4.557	**5.422** / 5.285	**0.17**
Campus2	(With Seed)point-to-point	2.690 / 1.329	3.640 / 2.578	4.533 / 3.800	5.410 / 5.019	6.294 / 6.203	7.156 / 7.378	5.6
	(No Seed)point-to-point	2.526 / 2.516	3.314 / 4.918	4.068 / 7.323	4.798 / 9.720	5.534 / 12.128	6.236 / 14.526	5.9
	Ours	**2.379** / **0.769**	**2.957** / **1.271**	**3.505** / **1.707**	**4.090** / **2.078**	**4.712** / **2.401**	**5.278** / **2.680**	**0.16**
Semi-Dense 4D Radar *Loop1_0*	(With Seed)DICP	0.821 / **0.627**	1.201 / 1.110	1.434 / 1.606	1.600 / 2.084	1.771 / 2.559	1.949 / 3.024	4.7
	(No Seed)DICP	**0.813** / 0.631	**1.193** / 1.112	**1.419** / 1.605	**1.590** / 2.084	**1.763** / 2.566	**1.935** / 3.033	4.8
	Ours	1.172 / 0.646	1.750 / **1.071**	2.242 / **1.500**	2.649 / **1.906**	3.044 / **2.295**	3.421 / **2.734**	**1.2**
Loop1_2	(With Seed)DICP	0.625 / **0.620**	0.928 / **1.115**	1.196 / **1.633**	1.442 / 2.156	**1.670** / 2.682	**1.859** / 3.213	4.0
	(No Seed)DICP	0.625 / 0.623	0.932 / 1.124	1.207 / 1.647	1.461 / 2.177	1.698 / 2.709	1.897 / 3.247	4.4
	Ours	**0.617** / 0.650	**0.942** / 1.148	**1.216** / 1.651	**1.472** / **2.145**	1.707 / **2.631**	1.913 / **3.110**	**0.4**
Loop1_3	(With Seed)DICP	**0.841** / 0.631	1.161 / **1.117**	**1.413** / **1.617**	**1.619** / 2.106	**1.808** / 2.597	**1.997** / 3.095	4.5
	(No Seed)DICP	0.843 / **0.630**	**1.160** / 1.119	1.415 / 1.623	1.622 / 2.118	1.812 / 2.615	2.003 / 3.118	4.6
	Ours	1.396 / 0.704	2.138 / 1.204	2.650 / 1.687	3.044 / **2.106**	3.399 / **2.502**	3.768 / **2.882**	**1.2**
Loop1_4	(With Seed)DICP	1.318 / **0.660**	1.904 / 1.181	2.280 / 1.651	2.629 / 2.092	3.172 / 2.525	3.767 / 2.925	5.3
	(No Seed)DICP	**1.253** / 0.663	**1.807** / **1.170**	**2.160** / **1.633**	**2.452** / **2.065**	**2.963** / **2.477**	**3.540** / **2.827**	5.3
	Ours	3.123 / 0.880	5.398 / 1.760	7.349 / 2.700	9.169 / 3.782	11.949 / 5.227	15.604 / 6.883	**2.2**
Dense 4D LiDAR *Straight Wall*	(With Seed)DICP	0.050 / 0.028	0.111 / 0.059	0.128 / 0.073	0.154 / 0.086	**0.258** / 0.103	**0.303** / 0.112	29.1
	(No Seed)DICP	0.050 / 0.028	0.111 / 0.059	0.128 / 0.073	0.154 / 0.086	0.258 / 0.103	0.304 / 0.112	29.1
	Ours	0.486 / 0.068	0.896 / 0.173	1.420 / 0.291	1.882 / 0.525	2.213 / 0.789	2.627 / 1.091	**7.8**
Curved Wall	(With Seed)DICP	0.378 / 0.092	0.761 / 0.137	1.167 / 0.133	1.565 / 0.120	1.974 / 0.223	2.399 / 0.404	29.1
	(No Seed)DICP	0.371 / **0.090**	0.745 / **0.135**	1.143 / **0.131**	1.532 / **0.119**	1.932 / **0.220**	2.347 / **0.397**	29.2
	Ours	**0.327** / 0.183	**0.601** / 0.368	**0.856** / 0.552	**1.069** / 0.724	**1.261** / 0.888	**1.442** / 1.050	**8.6**
Baker-Barry Tunnel (Empty)	(With Seed)DICP	1.028 / **0.289**	1.652 / **0.490**	2.515 / **0.744**	3.313 / **0.963**	3.964 / **1.140**	4.794 / **1.342**	20.8
	(No Seed)DICP	**1.022** / 0.290	**1.634** / 0.492	**2.487** / 0.746	**3.269** / 0.965	**3.914** / 1.141	**4.730** / 1.341	26.1
	Ours	1.804 / 0.614	2.599 / 0.945	3.325 / 1.301	3.767 / 1.581	4.189 / 1.919	4.931 / 2.367	**5.8**
Baker-Barry Tunnel (Vehicles)	(With Seed)DICP	**0.778** / **0.179**	**1.435** / 0.339	**2.110** / 0.497	**2.525** / 0.621	**3.027** / 0.732	**3.479** / 0.833	20.1
	(No Seed)DICP	0.780 / 0.181	1.449 / 0.341	2.121 / 0.499	2.542 / 0.624	3.049 / 0.733	3.506 / **0.831**	24.1
	Ours	2.216 / 0.776	3.659 / 1.417	4.904 / 2.170	6.099 / 2.633	7.167 / 3.138	8.121 / 3.646	**3.6**

(a) *Campus1* (b) *Campus2* (c) *Loop1_0* (d) *Loop1_2* (e) *Loop1_3*

(f) *Loop1_4* (g) *Straight Wall* (h) *Curved Wall* (i) *Baker-Barry Tunnel (Empty)* (j) *Baker-Barry Tunnel (Vehicles)*
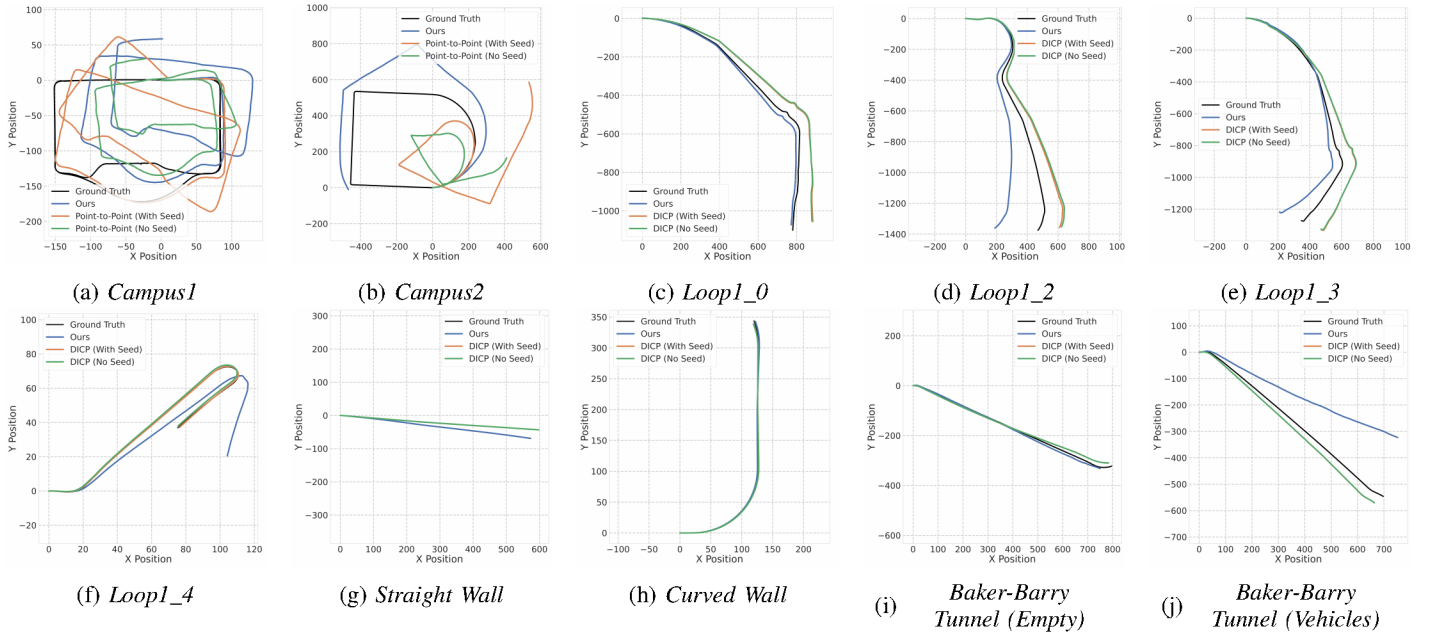
Fig. 5: Trajectory results on multiple sequences. Each plot's horizontal axis (x-axis) and vertical axis (y-axis) are in meters.

approach, while the open-source implementation of DICP is excluded due to its failure to produce reliable results on sparse 4D radar data. Under these conditions, closest point correspondence, which relies on the spatial overlap of consecutive scans, becomes unreliable [8]. By contrast, the proposed correspondence leverages the flow of points [2] rather than explicit spatial positions, offering robustness against sparse and noisy data. As shown in Table I, Figs. 5(a) and (b), and Fig. 6(a), the proposed method attains substantially higher accuracy than that of point-to-point ICP. Moreover, unlike point-to-point ICP, which requires repeated correspondence estimation in 3D space,

our approach simplifies the process to a single-step and 1D correspondence estimation.

For semi-dense 4D radar point clouds, the proposed method was tested and compared with DICP. The proposed method's translation errors exhibited a lower median but higher variance than those of DICP, as illustrated in Fig. 6(b). Additionally, rotation errors demonstrated higher median and variance compared with DICP, particularly in sequences with significant rotational motion (*Loop1_4*). Despite these limitations, the proposed method achieves comparable performance to DICP in other scenarios. Furthermore, it offers computational advantages due

(a) Sparse 4D Radar   (b) Semi-Dense 4D Radar   (c) Dense 4D LiDAR
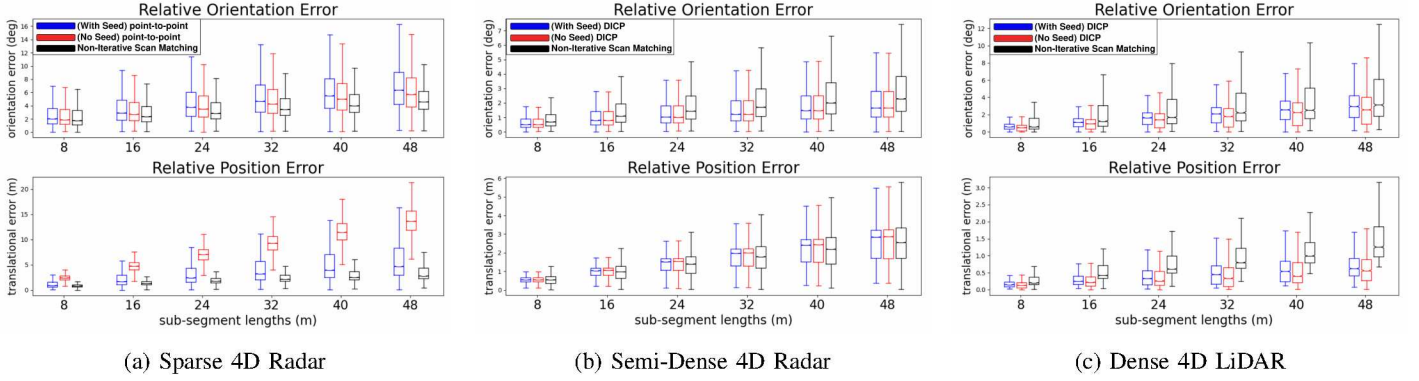
Fig. 6: Error comparison of different methods, ICP, DICP, and Non-Iterative Scan Matching (Ours), across various point cloud densities. Each figure represents the averaged results for the corresponding dataset.

(a) ICP, *Straight Wall*   (b) ICP, *Curved Wall*   (c) ICP, *Baker-Barry Tunnel (Empty)*

(d) Non-Iterative Scan Matching, *Straight Wall*   (e) Non-Iterative Scan Matching, *Curved Wall*   (f) Non-Iterative Scan Matching, *Baker-Barry Tunnel (Empty)*
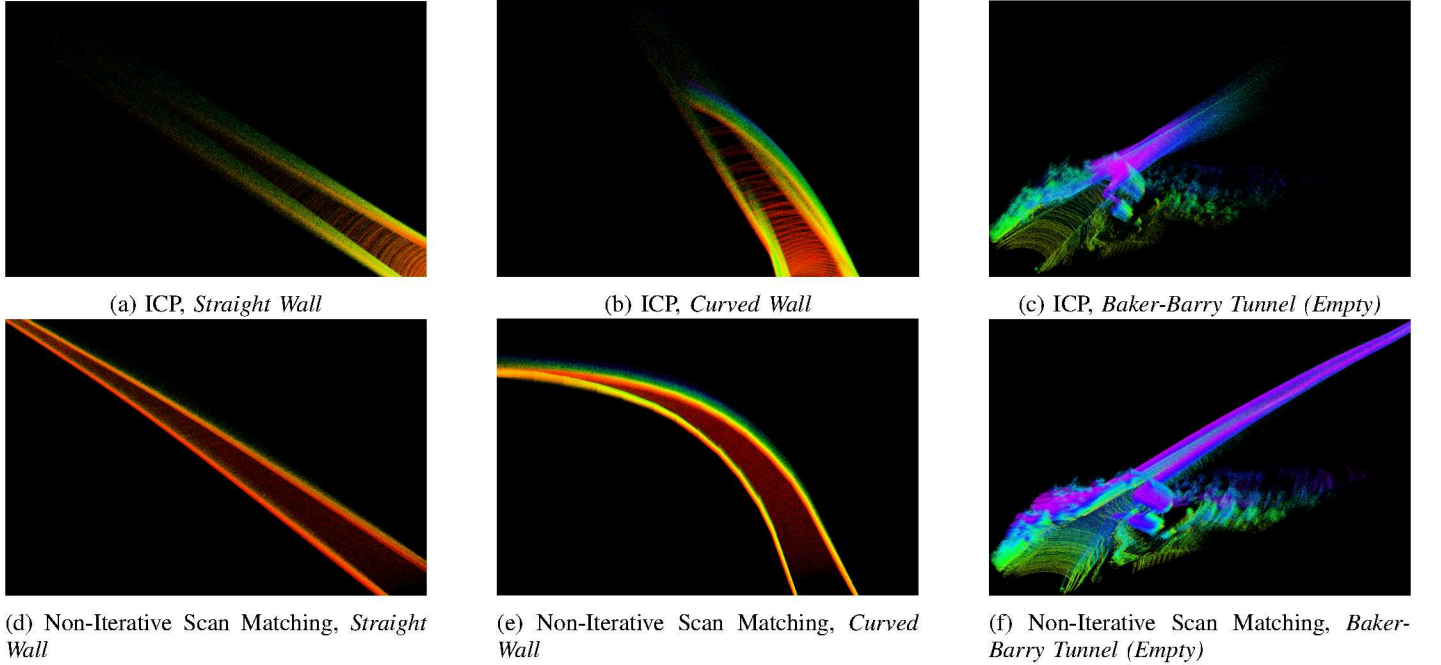
Fig. 7: The top row shows results using point-to-point ICP, while the bottom row shows those from the proposed method. Our method alleviates matching ambiguity caused by repetitive geometry.

to its efficient matching process.

The proposed method is further evaluated on dense 4D Li-DAR, particularly in geometrically repetitive environments, and compared with DICP. In static scenarios, *Straight Wall, Curved Wall*, and *Baker-Barry Tunnel (Empty)*, the proposed method demonstrates comparable performance to DICP while being significantly faster than DICP. Moreover, the method achieves lower rotation error than DICP in *Curved Wall*, possibly due to the Doppler velocity term in DICP having a limited effect on constraining the rotational component during the optimization process. The performance of the proposed method in the *Baker-Barry Tunnel (Vehicles)* is degraded due to a significant number of dynamic objects.

E. Repetitive Geometric Environment

In contrast to traditional closest-point correspondence, which relies solely on spatial geometry, the proposed approach leverages the simple functions f and g defined in Eq. (3), which incorporate Doppler velocity. Because these Doppler-based

functions do not depend on geometric cues alone, they offer robust performance even when surfaces appear similar or repetitive. As illustrated in Fig. 7, this fundamental reliance on velocity information disambiguates correspondences, providing accurate alignment in settings where ICP may fail. While the **Optimization** process has no difference from other registration techniques, the core difference lies in how correspondences are established (**Correspondence Estimation**) via Doppler information, rather than through purely closest point matching.

F. Correspondence Visualization

The matching results of Doppler Correspondence and the closest point correspondence method are visualized on sparse, semi-dense, and dense point clouds. The correspondences established after the single iteration are performed, as shown in Fig. 8. With only a single iteration, Doppler Correspondence achieves more coherent and consistent matches than the closest point approach across all levels of point cloud density.

(a) Sparse, Closest Point Correspondence

(b) Semi-Dense, Closest Point Correspondence

(c) Dense, Closest Point Correspondence

(d) Sparse, Doppler Correspondence

(e) Semi-Dense, Doppler Correspondence

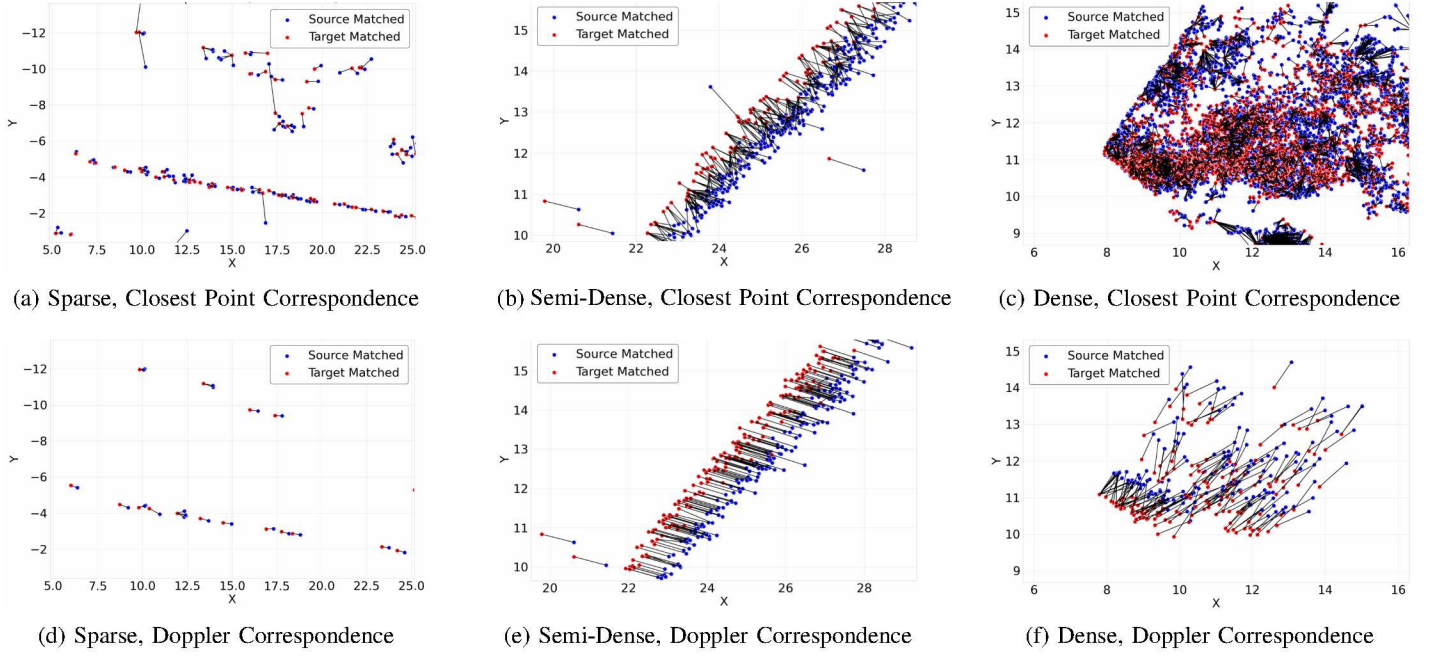(f) Dense, Doppler Correspondence

Fig. 8: Visualization of established correspondences after a single iteration in sparse, semi-dense, and dense point clouds.

In the sparse 4D radar point clouds, Doppler Correspondence generates significantly fewer correspondences than a number of points, as illustrated in Figs. 8(a) and (d). However, those correspondences are high quality, leading to better alignment performance compared with the closest-point method, as shown in Fig. 6(a). This result aligns with the *less is more* principle discussed in [8], where using fewer but more reliable correspondences can be advantageous in highly noisy data. For semi-dense 4D radar point cloud, while the closest point correspondence often fails to establish reliable matches, Doppler Correspondence succeeds in maintaining robust and consistent pairings, even when only a single iteration is performed, as illustrated in Figs. 8(b) and (e). For dense point clouds with repetitive geometric structures (*Baker-Barry Tunnel (Empty)*), Doppler Correspondence can produce clear and stable matches by effectively mitigating the ambiguity that may degrade the performance of ICP approaches, as depicted in Figs. 8(c) and (f).

V. LIMITATIONS

While Doppler Correspondence, introduced in Section III, offers notable advantages in terms of computational efficiency and robustness in repetitive geometric environments, it also comes with certain limitations. Specifically, the key scenarios where the method may fail are highlighted below. We also clarify the main assumptions underlying Eqs. (6) and (7). Furthermore, potential scenarios where these assumptions may fail in practice are discussed.

A. Key Assumptions and Their Breakdown

For identification of the case where Eqs. (6) and (7) hold, the motion of a point p_i can be considered as a combination of translation $\Delta t \in \mathbb{R}^{3 \times 1}$ and rotation $\Delta R \in SO(3)$ from the sensor between a time interval ΔT. Suppose $p_i(T)$ is the point p_i at time T. After the time interval ΔT, the point moves

according to:

$$p_i(T + \Delta T) = \Delta R \, p_i(T) + \Delta t, \qquad (8)$$

where $\Delta R = \exp(\Delta T \omega^{\times})$. Here, $\omega \in \mathbb{R}^3$ is the angular velocity, and $\omega^{\times}$ denotes its skew-symmetric matrix.

Assuming $\Delta T \, \|\omega\|_2 \ll 1$, ΔR can be linearized as:

$$\Delta R \approx I + \Delta T \omega^{\times}. \qquad (9)$$

Substituting this into Eq. (8) gives

$$p_i(T + \Delta T) - p_i(T) = (\Delta T \omega^{\times}) \, p_i(T) + \Delta t.$$

Dividing both terms by ΔT gives:

$$\frac{p_i(T + \Delta T) - p_i(T)}{\Delta T} = w^{\times} p_i(T) + \rho, \qquad (10)$$

where $\rho = \frac{\Delta t}{\Delta T}$. Assuming constant ω and ρ during a short consecutive scan, as $\Delta T \to 0$, then:

$$\dot{p}_i(T) = w^{\times} p_i(T) + \rho. \qquad (11)$$

Taking the inner product of both sides with the unit vector

$$n_{p_i}(T) = \frac{p_i(T)}{\|p_i(T)\|_2},$$

we obtain

$$v_{p_i}(T) = n_{p_i}(T) \cdot \rho, \qquad (12)$$

$$(\because n_{p_i}(T) \cdot (\omega \times p_i(T)) = 0).$$

Therefore, for any T between a short ΔT, ω and ρ do not significantly change, and Eq. (12) can be used.

In our framework, if we set $p_i \in \mathcal{P}$ and $q_j \in \mathcal{Q}$ as true correspondences from a small time interval ΔT, Eq. (10) can be written as:

$$\frac{q_j - p_i}{\Delta T} = w^{\times} p_i + \rho. \qquad (13)$$

Applying Eq. (12) at the points where p_i and q_j are measured:

$$v_{p,i} = n_{p_i} \cdot \rho, \qquad (14)$$

$$v_{q,j} = n_{q_j} \cdot \rho. \qquad (15)$$

Taking inner product of both sides $n_{p,i}$ to Eq. (13), Eq. (4) can be derived:

$$\frac{q_j - p_i}{\Delta T} \cdot n_{p_i} = (w^\times p_i + \rho) \cdot n_{p_i} = \rho \cdot n_{p_i} = v_{p_i}.$$

Multiplying both sides by ΔT and $\|p_i\|_2$, we obtain Eq. (6):

$$(q_j - p_i) \cdot p_i = v_{p,i} \Delta T \|p_i\|_2.$$

Derivation of Eq. (7) starts from the relationship $q_j = \Delta R p_i + \Delta t$, which can be expressed as:

$$\Delta R^T q_j - \Delta R^T \Delta t = p_i.$$

Thus, $p_i - q_j$ can be formulated as:

$$p_i - q_j = (I - \Delta T \omega^\times) q_j - (I - \Delta T \omega^\times) \Delta t - q_j$$
$$= -\Delta T \omega^\times q_j - (I - \Delta T \omega^\times) \Delta t.$$

Divide the above with ΔT:

$$\frac{p_i - q_j}{\Delta T} = -\omega^\times q_j - (I - \Delta T \omega^\times) \rho,$$

and take inner product with n_{q_j}:

$$\frac{p_i - q_j}{\Delta T} \cdot n_{q_j} = -(I - \Delta T \omega^\times) \rho \cdot n_{q_j}$$
$$= -\rho \cdot n_{q_j} + (\Delta T \omega^\times \rho) \cdot n_{q_j}$$
$$= -v_{q,j} + (\Delta T \omega^\times \rho) \cdot n_{q_j}.$$

Multiplying both sides by ΔT and $\|q_j\|_2$, we obtain:

$$(q_j - p_i) \cdot q_j = \underbrace{\Delta T\, v_{q_j} \|q_j\|_2}_{\text{Doppler velocity term}} - \underbrace{\Delta T^2 (\omega^\times \rho) \cdot q_j}_{\text{rotation term}}.$$

Since $\Delta T^2 \approx 0$, the *rotation term* can be neglected, leading to the Eq. (7):

$$(q_j - p_i) \cdot q_j \approx v_{q_j} \Delta T \|q_j\|_2.$$

Therefore, the *sufficient conditions* required to derive Eqs. (6) and (7) are that $\boldsymbol{\Delta T \|\omega\|_2 \ll 1}$, as *assumed in* Eq. (9), and that both $\boldsymbol{\omega}$ and $\boldsymbol{\rho}$ remain **constant over a short consecutive scan**, as *assumed in* Eq. (11).

The first condition, $\Delta T \|\omega\|_2 \ll 1$, is necessary to linearize the rotation matrix $\Delta R = \exp(\Delta T \omega^\times)$. This simplification enables the relative motion between p_i and q_j to be approximated using linear vector operations, which is essential for establishing a direct relationship between spatial displacement and Doppler velocity.

The second condition, assuming that both ω and ρ remain constant during a short interval ΔT, is used in Eq. (11). This implies that the velocity of a point p_i can be written as a time-invariant expression, ensuring that Doppler velocity measurements are consistent across a short time window. If ω or ρ varies significantly, the radial velocity observed from Doppler measurements would also change, making it difficult to formulate a consistent correspondence constraint between p_i and q_j.

Violating this condition may lead to degradation or failure of the proposed method. The following examples illustrate the cases where the assumption holds and it does not:
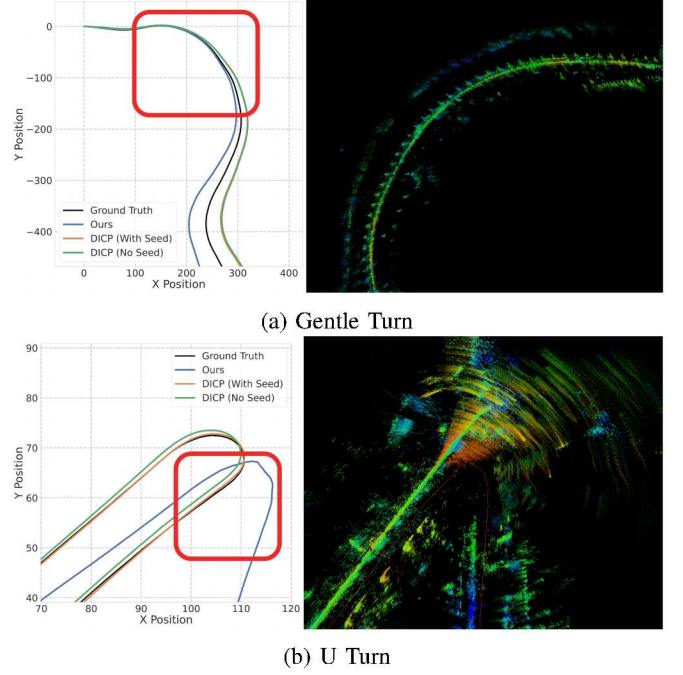

(a) Gentle Turn

(b) U Turn

Fig. 9: (a) corresponds to the gentle turn example from *Loop1_2*, and (b) corresponds to the U-turn example from *Loop1_4*. Each left figure shows the trajectories produced by each method, and the right figure illustrates the map constructed in the red area highlighted in the left figure.

1) Gentle Turn (Assumption Holds): The scenario where the assumption remains valid is illustrated in Fig. 9(a). The vehicle travels at a moderate speed $(v \approx 8.3\,\text{m/s})$ with a small angular velocity $(\|\omega\|_2 \approx 0.1\,\text{rad/s})$, and the sensor operates at $\Delta T = 0.083\,\text{s}$. In this case,

$$\Delta T \|\omega\|_2 = 0.083 \times 0.1 = 0.0083 \ll 1.$$

In this scenario, both Eqs. (6) and (7) hold, resulting in accurate Doppler-based odometry with minimal drift.

2) U-Turn (Assumption Fails): By contrast, Fig. 9(b) shows a scenario where the assumption is invalid. The angular velocity of the vehicle is significantly higher than that of *Gentle Turn* scenario $(\|\omega\|_2 \approx 0.56\,\text{rad/s})$. Even though ΔT remains $0.083\,\text{s}$, the product

$$\Delta T \|\omega\|_2 = 0.083 \times 0.56 = 0.04648,$$

is no longer "small." As a result, the odometry output can deviate significantly when sharp U-turns occur.

VI. PRACTICAL USE OF DOPPLER CORRESPONDENCE

Doppler Correspondence may offer versatile applications due to its simplicity. A straightforward implementation is its integration with ICP in a weighted form, referred to as Doppler 4D-ICP, as described in Algorithm 3. The process begins by computing $f(\mathcal{P})$ and $g(\mathcal{Q})$ from the point clouds and establishing Doppler Correspondence. In this setup, correspondences estimated from Doppler Correspondence remain fixed throughout the ICP iterations, while closest point correspondences are dynamically updated. This combination enables the optimization process to leverage two distinct correspondence characteristics: spatial

Algorithm 3 Doppler Correspondence-Based Iterative Closest Point for 4D Ranging Sensors (Doppler 4D-ICP)

Require: $\mathcal{P}$ (source point cloud), $\mathcal{Q}$ (target point cloud), τ_{spatial} (spatial distance threshold), τ_{doppler} (doppler distance threshold)

Ensure: $\hat{R}, \hat{t}$

    Compute $f(\mathcal{P}), g(\mathcal{Q})$

    $\mathcal{Q}' \leftarrow \texttt{FindClosest}(f(\mathcal{P}), g(\mathcal{Q}))$

    Masking Outliers:

        $\mathcal{M} \leftarrow \|\mathcal{P} - \mathcal{Q}'\|_2 \leq \tau_{\text{spatial}}$ and $|f(\mathcal{P}) - g(\mathcal{Q}')| \leq \tau_{\text{doppler}}$

        $\mathcal{C}_{\text{doppler}} \leftarrow \mathcal{P}[\mathcal{M}], \mathcal{Q}'[\mathcal{M}]$

    while not converged **do**

        $\mathcal{Q}' \leftarrow \texttt{FindClosest}(\mathcal{P}, \mathcal{Q})$

        Masking Outliers:

            $\mathcal{M} \leftarrow \|\mathcal{P} - \mathcal{Q}'\|_2 \leq \tau_{\text{spatial}}$

            $\mathcal{C}_{\text{spatial}} \leftarrow \mathcal{P}[\mathcal{M}], \mathcal{Q}'[\mathcal{M}]$

        Predict Transformation:

            $\delta R, \delta t \leftarrow \underset{R,t}{\arg\min} \; (1-\alpha) \sum_{(p,q) \in \mathcal{C}_{\text{spatial}}} \rho(\|q - (Rp + t)\|_2) + \alpha \sum_{(p,q) \in \mathcal{C}_{\text{doppler}}} \rho(\|q - (Rp + t)\|_2)$

        Update:

            $\mathcal{P} \leftarrow \texttt{TransformSourcePoint}(\delta R, \delta t, \mathcal{P})$

            $\hat{R}, \hat{t} \leftarrow \texttt{UpdateTransformation}(\delta R, \delta t, \hat{R}, \hat{t})$

    end while

        return $\hat{R}, \hat{t}$

TABLE II: Odometry performance on the semi-dense 4D radar dataset. Each cell reports the absolute pose errors [5] (in meters). Best results are in bold, second-best are underlined.

Method	Sequence			
	loop1_0	*loop1_1*	*loop1_3*	*loop1_4*
ICP point-to-point	61.72	<u>38.71</u>	146.65	3.15
ICP point-to-plane	55.18	43.40	127.52	<u>2.89</u>
Generalized-ICP [17]	64.21	57.28	138.41	3.30
VGICP [9]	**50.86**	61.86	141.37	4.09
DICP [7]	60.15	68.60	143.24	2.94
Doppler Correspondence (Ours)	<u>54.87</u>	50.04	**101.74**	7.49
Doppler 4D-ICP (Ours)	63.25	**37.74**	<u>114.76</u>	**2.80**

TABLE III: Odometry performance on the dense 4D LiDAR dataset. Each cell reports the absolute pose errors [5] (in meters). Best results are in bold, second-best are underlined. Note that Div. indicates the trajectory of the algorithm diverged.

Method	Sequence	
	Baker-Barry Tunnel (Empty)	Baker-Barry Tunnel (Vehicles)
ICP point-to-point	Div.	Div.
ICP point-to-plane	Div.	Div.
Generalized-ICP [17]	Div.	Div.
VGICP [9]	Div.	Div.
DICP [7]	65.27	<u>37.12</u>
Doppler Correspondence (Ours)	<u>20.16</u>	47.54
Doppler 4D-ICP (Ours)	**14.50**	**24.06**

information and Doppler information. To further enhance robustness against outliers and noise, we additionally incorporate a Huber loss function $\rho(\cdot)$ as a robust kernel during optimization.

To evaluate the effectiveness of the proposed approach, experiments are conducted using a semi-dense 4D radar dataset, *Loop 1*, and dense 4D LiDAR datasets, *Baker-Barry Tunnel (Empty, Vehicles)*. The parameters are configured as follows: for the semi-dense 4D radar dataset, $\tau_{\text{spatial}} = 3$, $\tau_{\text{doppler}} = 10$, and $\alpha = 0.7$. For the dense 4D LiDAR datasets, the parameters are set to $\tau_{\text{spatial}} = 2$, $\tau_{\text{doppler}} = 5$, and $\alpha = 0.9$.

The experimental results are presented on the semi-dense 4D radar dataset in Table II and the dense 4D LiDAR dataset in Table III. Notably, in repetitive geometric environments, geometric ambiguity is difficult to resolve without incorporating Doppler information, leading to divergence in conventional scan-matching methods. By leveraging Doppler Correspondence, the proposed approach effectively fuses spatial and Doppler information, leading to more reliable and consistent trajectory estimation.

VII. Conclusion

Overall, the results indicate that Doppler Correspondence does not always guarantee better performance compared with the closest point method. Scenarios with high angular velocity may cause the violation of the aforementioned assumption and degradation of odometry performance. However, the proposed method offers distinct advantages, such as its weak dependence on the environment's geometry, which sets it apart from closest point correspondence. These features enable robust and efficient correspondence estimation, making it a valuable complement to traditional methods in specific applications. Furthermore, its simplicity makes it easy to use and integrate with other

methods. While further improvements are needed for broader applicability, the proposed method holds novelty as it leverages Doppler information in correspondence for the first time.

References

[1] Paul J Besl and Neil D McKay. Method for Registration of 3-D Shapes. In *Proc. Sensor fusion IV: Control Paradigms and Data Structures*, 1992. URL https://www.spiedigitallibrary.org/conference-proceedings-of-spie/1611/1/Method-for-registration-of-3-D-shapes/10.1117/12.57955.full.

[2] Fangqiang Ding, Zhijun Pan, Yimin Deng, Jianning Deng, and Chris Xiaoxuan Lu. Self-Supervised Scene Flow Estimation With 4-D Automotive Radar. *IEEE Robotics and Automation Letters*, 2022. URL https://ieeexplore.ieee.org/abstract/document/9810356.

[3] Christopher Doer and Gert F Trommer. An EKF Based Approach to Radar Inertial Odometry. In *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2020. URL https://ieeexplore.ieee.org/document/9235254.

[4] Christopher Doer and Gert F Trommer. Radar Inertial Odometry With Online Calibration. In *Proc. European Navigation Conference (ENC)*, 2020. URL https://ieeexplore.ieee.org/document/9317343.

[5] Michael Grupp. EVO: Python Package for the Evaluation of Odometry and SLAM. https://github.com/MichaelGrupp/evo, 2017.

[6] Daniel Casado Herraez, Matthias Zeller, Le Chang, Ignacio Vizzo, Michael Heidingsfeld, and Cyrill Stachniss. Radar-Only Odometry and Mapping for Autonomous Vehicles. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2024. URL https://ieeexplore.ieee.org/abstract/document/10610311.

[7] Bruno Hexsel, Heethesh Vhavle, and Yi Chen. DICP: Doppler Iterative Closest Point Algorithm. In *Proc. Robotics: Science and Systems*, 2022. URL https://www.roboticsproceedings.org/rss18/p015.pdf.

[8] Qiucan Huang, Yuchen Liang, Zhijian Qiao, Shaojie Shen, and Huan Yin. Less is More: Physical-Enhanced Radar-Inertial Odometry. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2024. URL https://ieeexplore.ieee.org/document/10611471.

[9] Kenji Koide, Masashi Yokozuka, Shuji Oishi, and Atsuhiko Banno. Voxelized GICP for fast and accurate 3D point cloud registration. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2021. URL https://ieeexplore.ieee.org/abstract/document/9560835.

[10] Jubong Lee, Jinseo Hong, and Kyihwan Park. Frequency Modulation Control of an FMCW LiDAR Using a Frequency-to-Voltage Converter. *Sensors*, 2023. URL https://www.mdpi.com/1424-8220/23/10/4981.

[11] Xingyi Li, Han Zhang, and Weidong Chen. 4D Radar-Based Pose Graph SLAM With Ego-Velocity Pre-Integration Factor. *IEEE Robotics and Automation Letters*, 2023. URL https://ieeexplore.ieee.org/document/10173499.

[12] Daniil Lisus, Keenan Burnett, David J Yoon, Richard Poulton, John Marshall, and Timothy D Barfoot. Are Doppler Velocity Measurements Useful for Spinning Radar Odometry? *IEEE Robotics and Automation Letters*, 2024. URL https://ieeexplore.ieee.org/abstract/document/10766427.

[13] Jan Michalczyk, Roland Jung, and Stephan Weiss. Tightly-Coupled EKF-Based Radar-Inertial Odometry. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. URL https://ieeexplore.ieee.org/document/9981396.

[14] Jan Michalczyk, Roland Jung, Christian Brommer, and Stephan Weiss. Multi-State Tightly-Coupled EKF-Based Radar-Inertial Odometry With Persistent Landmarks. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2023. URL https://ieeexplore.ieee.org/document/10160482.

[15] Morten Nissov, Nikhil Khedekar, and Kostas Alexis. Degradation Resilient LiDAR-Radar-Inertial Odometry. *arXiv preprint arXiv:2403.05332*, 2024. URL https://arxiv.org/abs/2403.05332.

[16] Szymon Rusinkiewicz and Marc Levoy. Efficient Variants of the ICP Algorithm. In *Proc. Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, 2001. URL https://ieeexplore.ieee.org/abstract/document/924423.

[17] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun. Generalized-ICP. In *Proc. Robotics: Science and Systems*, 2009. URL https://direct.mit.edu/books/edited-volume/2880/chapter-abstract/78093/Generalized-ICP?redirectedFrom=PDF.

[18] Ignacio Vizzo, Tiziano Guadagnino, Benedikt Mersch, Louis Wiesmann, Jens Behley, and Cyrill Stachniss. KISS-ICP: In Defense of Point-to-Point ICP – Simple, Accurate, and Robust Registration If Done the Right Way. *IEEE Robotics and Automation Letters*, 2023. URL https://ieeexplore.ieee.org/abstract/document/10015694/.

[19] Yuchen Wu, David J Yoon, Keenan Burnett, Soeren Kammel, Yi Chen, Heethesh Vhavle, and Timothy D Barfoot. Picking Up Speed: Continuous-Time Lidar-Only Odometry Using Doppler Velocity Measurements. *IEEE Robotics and Automation Letters*, 2022. URL https://ieeexplore.ieee.org/abstract/document/9968059.

[20] Yang Xu, Qiucan Huang, Shaojie Shen, and Huan Yin. Modeling Point Uncertainty in Radar SLAM. *arXiv preprint arXiv:2402.16082*, 2024. URL https://arxiv.org/abs/2402.16082.

[21] David J Yoon, Keenan Burnett, Johann Laconte, Yi Chen, Heethesh Vhavle, Soeren Kammel, James Reuther, and Timothy D Barfoot. Need for Speed: Fast Correspondence-Free LiDAR-Inertial Odometry Using Doppler Velocity. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023. URL https://ieeexplore.ieee.org/abstract/document/10341596.

[22] Jun Zhang, Huayang Zhuge, Yiyao Liu, Guohao Peng, Zhenyu Wu, Haoyuan Zhang, Qiyang Lyu, Heshan Li, Chunyang Zhao, Dogan Kircali, et al. NTU4DRadLM: 4D Radar-Centric Multi-Modal Dataset for Localization and Mapping. In *Proc. IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, 2023. URL https://ieeexplore.ieee.org/abstract/document/10422606.

[23] Jun Zhang, Huayang Zhuge, Zhenyu Wu, Guohao Peng, Mingxing Wen, Yiyao Liu, and Danwei Wang. 4DRadarSLAM: A 4D Imaging Radar SLAM System for Large-Scale Environments Based on Pose Graph Optimization. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2023. URL https://ieeexplore.ieee.org/abstract/document/10160670/.

[24] Zichao Zhang and Davide Scaramuzza. A Tutorial on Quantitative Trajectory Evaluation for Visual (-Inertial) Odometry. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018. URL https://ieeexplore.ieee.org/abstract/document/8593941.

[25] Huanyu Zhou, Shouyi Lu, and Guirong Zhuo. Self-Supervised 4-D Radar Odometry for Autonomous Vehicles. In *Proc. IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, 2023. URL https://ieeexplore.ieee.org/abstract/document/10422466.