DRIO: Robust Radar-Inertial Odometry in Dynamic Environments

Hongyu Chen, Yimin Liu, *Member, IEEE*, and Yuwei Cheng

***Abstract*—Accurate and robust localization is essential for mobile robots. Recently, millimeter wave (mmWave) radars have been widely used for odometry, owing to their robustness to all-weather conditions, lightweight and low cost. However, existing radar-based odometry methods degrade severely in high-dynamic environments. In this letter, we propose a robust radar-inertial odometry method for high-dynamic environments (DRIO) by exploiting the ground, an ever-present static target that is unaffected by the dynamic environments. The points of the ground surface were traditionally treated as clutter points in previous works due to their unstable distribution. We overcome this limitation by detecting ground points using both Doppler and geometric characteristics. During the detection process, accurate radar velocity is jointly estimated, which is then fused with inertial data to obtain the odometry. The real-world evaluations indicate that the proposed method achieves robust and Lidar-level localization in complex dynamic environments. In addition to odometry, our method can effectively improve the quality of radar point clouds for subsequent perception tasks.**

***Index Terms*—Dynamic environments, localization, millimeter wave radar, odometry.**

I. Introduction

ACCURATE and robust odometry is critical for autonomous vehicles and mobile robots to navigate in unknown and complex environments. Over the years, visual odometry (VO) [1], [2], [3], [4], [5] and Lidar odometry (LO) [6], [7] have attracted great interest and achieved encouraging outcomes. However, both VO and LO struggle to work under adverse weather conditions and VO is sensitive to poor lighting conditions. By contrast, millimeter wave (mmWave) radar is less affected by adverse weather and is indifferent to lighting conditions. Owing to its robustness, lightweight design, and low cost, mmWave radar has emerged as a widely-used alternative for mobile robots in recent years [8].

Several works have studied odometry based on mmWave radars [9], [10], [11], [12]. Existing methods can achieve promising results when the environment is completely static or the static objects are in the majority. However, these methods degrade severely in high-dynamic environments where the number of detected dynamic objects is larger than that of static objects, as they cannot distinguish between dynamic targets and static targets. In practical applications, high-dynamic environments are very common for odometry, as shown in Fig. 1. Therefore, robust radar odometry methods in dynamic environments remain to be a challenging problem.

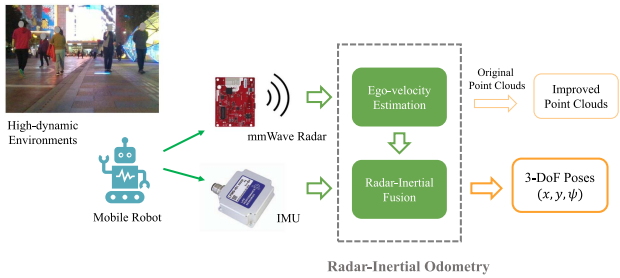


Fig. 1. Radar-inertial odometry in dynamic environments.

Existing odometry methods based on mmWave radars can be generally divided into two categories: scan matching methods [9], [10] and instantaneous methods [11], [12]. It is difficult to achieve reliable and precise matching results using radar scans because radar point clouds tend to be sparse and noisy [13]. By contrast, instantaneous methods use a single radar scan to estimate motion and thus are a more robust and efficient solution. Given this, an instantaneous method is employed in this study. However, it is infeasible to estimate the yaw rate using a single radar without any assumption of the motion model [12], as the Doppler velocity measured by radars is radial velocity. To address this problem, an Inertial Measurement Unit (IMU) is utilized to obtain the orientation of the radar. Similar to mmWave radars, IMU is indifferent to external environments. Hence, the fusion of mmWave radars and IMU is a robust solution to odometry, which has gained great attention recently [13], [14], [15], [16], [17], [18].

In applications for Unmanned Ground Vehicles (UGVs), mmWave radars usually detect the point clouds generated by the ground. Due to the limited angular resolution of mmWave radars and the large elevation angle of the ground surface, the height of ground points measured by the radar is prone to inaccuracy. Consequently, the distribution of ground points in radar point clouds is unstable and discontinuous, as shown in Fig. 3(a). Thus, ground points were usually treated as clutter points in previous applications, which were removed during preprocessing [8]. However, for odometry, the ground is an ever-present static target, which is unaffected by the dynamic environments. Given this characteristic of the ground, we aim

Manuscript received 30 May 2023; accepted 26 July 2023. Date of publication 3 August 2023; date of current version 9 August 2023. This letter was recommended for publication by Associate Editor J. Zhang and Editor J. Civera upon evaluation of the reviewers' comments. *(Corresponding author: Yuwei Cheng.)*

The authors are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: chenhy20@mails.tsinghua.edu.cn; yiminliu@tsinghua.edu.cn; chengyw18@tsinghua.org.cn).

Digital Object Identifier 10.1109/LRA.2023.3301290

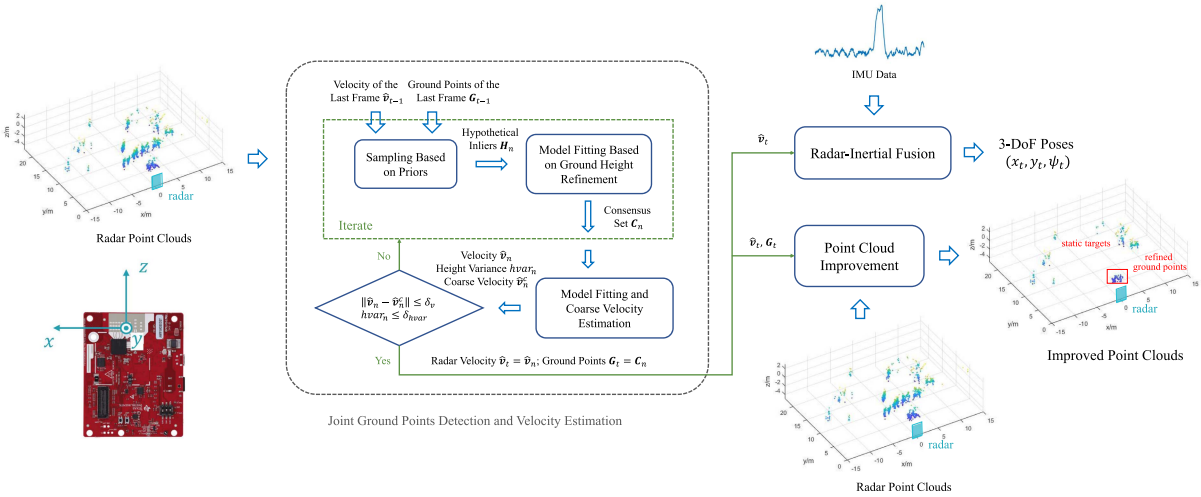
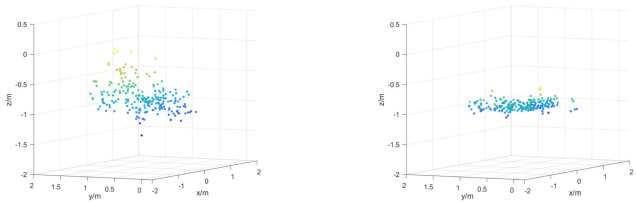


Fig. 2. Overview of the proposed DRIO method.



(a) Original ground points (b) Refined ground points

Fig. 3. Ground points with original height and refined height.

to make the most of ground points to robustly estimate motion in complex dynamic environments.

In this letter, we propose a robust radar-inertial odometry method in dynamic environments (DRIO) by exploiting the ever-present ground. To this end, we utilize both Doppler and geometric characteristics of the ground to robustly detect the ground points from radar point clouds. During the detection process, accurate radar velocity is estimated jointly. The evaluations on real-world datasets indicate that our method outperforms the existing radar-inertial odometry (RIO) methods in high-dynamic environments. In addition to reliable odometry, our method can effectively improve the quality of radar point clouds by removing the clutter points, which is significant to subsequent perception tasks.

To sum up, in this letter, our main contributions lie in the following aspects:

- We propose a novel radar-inertial odometry method by using the ground points, which can achieve robust and accurate localization in high-dynamic environments.
- We propose a reliable ground detection algorithm that can jointly estimate accurate radar velocity. In addition, our method can effectively improve the quality of radar point clouds for subsequent perception tasks.
- We build a dataset[1] based on mmWave radar point clouds containing different dynamic environments. The dataset is released to facilitate future research on radar odometry in dynamic environments.

The rest of this letter is structured as follows. Section II reviews the related work on mmWave radars and odometry methods in mobile robots. Section III introduces our robust radar-inertial odometry method based on ground points. Then, we evaluate the proposed method in real-world datasets and analyze the results in Section IV. Finally, we summarize the letter and present conclusions in Section V.

II. Related Work

A. Odometry

Visual sensors and Lidars are two popular sensors used for odometry. Visual odometry can be generally classified into three types: feature-based methods, direct methods, and hybrid methods. Feature-based methods estimate ego-pose by extracting, matching, and tracking local image features [1], [2], while direct methods track the changes in pixel intensity directly [3], [4]. There are also hybrid methods that combine the advantages of both methods [5]. Lidar odometry utilizes geometric information contained in point clouds to estimate ego-motion and is generally treated as a registration problem [6], [7]. In recent decades, VO and LO have undergone great refinement and achieved high precision. However, due to the inherent weakness of optical sensors, vision-based and Lidar-based methods cannot meet the requirements for robustness in challenging environments, such as adverse weather conditions, poor lighting conditions, and high-dynamic environments.

B. Millimeter Wave Radar in Mobile Robots

In recent years, mmWave radars have been widely used for mobile robots owing to their robustness to challenging conditions. There are mainly two types of mmWave radars: mechanically scanning radars and single-chip radars. Mechanically scanning radars possess a high angular resolution and have been widely applied to various perception tasks, such as

[1][Online]. Available: https://github.com/chy525/DRIO-dataset.git

odometry [19] and place recognition [20]. However, due to its large size, high power consumption, and high cost, this type of radar is unsuitable for small mobile robots. By contrast, single-chip mmWave radars with lightweight design and low cost have become a popular choice for small robots. Compared to mechanically scanning radars, single-chip mmWave radars have a much lower angular resolution, leading to sparser point clouds. Hence, the methods proposed for scanning radars cannot be directly applied to single-chip radars. Recently, there has been an increasing interest in research on single-chip mmWave radars. In [21], a novel end-to-end neural network is proposed to generate lidar-like high-resolution point clouds from low-resolution radar input. Cheng et al. [22] use a single-chip mmWave radar to realize robust relocalization under visually degraded environments.

C. Millimeter Wave Radar Odometry

There have been several studies focusing on the odometry based on single-chip mmWave radars, which can be generally divided into two categories: scanning matching methods and instantaneous methods. Scan matching methods [9], [10] estimate motion by matching two consecutive radar scans. In [9], radar point clouds are represented with a Gaussian mixture model, and the motion is estimated by the joint optimization of spatial metric and Doppler metric. In [10], Lu et al. propose a deep neural network to match radar scans, where each radar scan is encoded by a convolutional neural network. Since radar point clouds are noisy due to clutter and multi-path effects, it is challenging and computationally demanding to achieve accurate and reliable matching results on radar scans [13].

Instantaneous methods [11], [12] estimate velocity using a single radar scan and thus are more robust and efficient in practical applications. [11] estimates the ego-velocity by analyzing the distribution of Doppler velocities of radar point clouds over their azimuth angles. However, since the Doppler velocity measured by radars is radial velocity, instantaneous methods are unable to estimate the yaw rate with a single radar without any assumption of the motion model [12]. An Ackerman model is assumed in [11] and multiple radars are used in [12] to address the problem. In addition, an extra IMU sensor can be utilized to obtain the orientation of the radar. The robustness of both the radar and IMU to external environments makes radar-inertial odometry a viable solution for challenging environments [13], [14], [15], [16], [17], [18].

In [14], an unscented Kalman filter is used to fuse inertial data with poses obtained by matching radar scans. Doer et al. [13] propose a radar-inertial odometry based on the extended Kalman filter (EKF), which fuses inertial data with the ego-velocity estimated by the radar. This work is further extended in [15] with yaw aiding based on a Manhattan world assumption. [16] uses a sliding window optimization method to fuse the inertial data with radar velocity. In [17], range and velocity measurements of the radar point clouds are fused with inertial data using an EKF. In [18], Zhuang et al. employ the graduated non-convexity method to estimate ego-velocity and perform scan-to-submap matching on radar scans. The velocity and matches are then fused with inertial data by an iterative EKF to get the pose. These existing RIO methods focus on the fusion of radar data and inertial data but do not consider the unique challenges in complex dynamic environments. Robust radar-inertial odometry in high-dynamic environments is still an unsolved problem.

III. Our Method

A. Overview

The overview of the proposed DRIO method is shown in Fig. 2. The inputs are time-aligned radar point clouds and inertial data. To deal with the challenges in complex high-dynamic environments, we rely on the ground points to estimate ego-velocity, as the ground is an ever-present static target for UGVs, which is not affected by the dynamic scenes. However, since the ground points have the following characteristics, the detection of ground points is needed.

- The height error of ground points may exceed 1 m, as shown in Fig. 3(a). Thus, the distribution of ground points is unstable and discontinuous.
- The spatial distribution of the ground points is related to the mounting location of the radar on mobile robots.
- There may be dynamic objects within the location of the ground points.

To robustly estimate motion based on the ground points, we propose a joint ground points detection and ego-velocity estimation algorithm, as illustrated in Section III-B. The ground points are detected based on both Doppler and geometric characteristics of the ground. During this process, the accurate radar velocity is estimated jointly. Then, we fuse the estimated velocity with the angular velocity from IMU to obtain the odometry, as described in Section III-C. Furthermore, the estimated ego-velocity is used to remove the clutter points in radar point clouds, which improves the quality of point clouds for subsequent perception tasks, as shown in Section III-C. Thus, the outputs of the system are the estimated poses and the improved radar point clouds.

B. Joint Ground Points Detection and Velocity Estimation

For Lidar sensors, the ground is usually segmented based on the feature that the ground is a horizontal plane [23]. However, unlike Lidar point clouds, there is a significant error in the height of ground point clouds observed by the single-chip mmWave radar. Thus, we cannot directly extract ground points based on geometric features. Nevertheless, compared to Lidar point clouds, radar point clouds contain extra Doppler velocity of targets. By taking advantage of both Doppler and geometric characteristics, ground points can be accurately detected from radar point clouds. To be specific, as a static target, the Doppler velocity of the ground is the radial projection of radar velocity. Thus, for a ground point at (x, y, z), the Doppler velocity v_d is equal to

$$v_d = -\frac{v_x \cdot x + v_y \cdot y}{R}, \tag{1}$$

where $R = \sqrt{x^2 + y^2 + z^2}$ is the range of the ground point; v_x and v_y are radar velocities along the x-axis and y-axis respectively. Here we assume a planar motion for the radar.

According to the principles of radar signal processing [24], the radar directly measures the range R, the Doppler velocity v_d, the phase difference ω_x between horizontal arrays and the phase difference ω_z between vertical arrays. Then the coordinates (x, y, z) can be calculated using $x = R \cdot \omega_x/\pi$, $z = R \cdot \omega_z/\pi$, $y = \sqrt{R^2 - x^2 - z^2}$.

Due to limited elevation resolution, the z coordinates directly measured by the single-chip mmWave radar tend to be inaccurate [25]. According to (1), if the radar velocities are known, the z coordinate of the ground point can be recalculated using the following equations,

$$y_r = \frac{-v_d \cdot R - v_x \cdot x}{v_y},$$
$$z_r = -\sqrt{R^2 - x^2 - y_r^2}, \tag{2}$$

where y_r and z_r are reestimated y and z coordinates. The original height and the reestimated height of a frame of ground points are shown in Fig. 3, where z_r is calculated using the ground truth of radar velocities. As can be seen, the reestimated height of points is closer to the manually measured ground height than the original height. Moreover, the reestimated ground points are more consistent with the characteristic of being distributed on a horizontal plane. This can be attributed to the fact that single-chip mmWave radars usually possess higher Doppler resolution but lower angular resolution. To some extent, through (2), the high Doppler resolution compensates for the lack of angular resolution.

From another perspective, given the ground points, we can take advantage of the feature that the ground is a horizontal plane to estimate the radar velocity. To be specific, radar velocities can be estimated by minimizing the variance of the reestimated height of given ground points,

$$\{\hat{v}_x, \hat{v}_y\} = \underset{v_x, v_y}{\arg\min} \frac{1}{N} \sum_{i=1}^{N} \left(z_{r,i} - \mu_r\right)^2, \tag{3}$$

where $z_{r,i}$ is the height of the ith ground point reestimated using v_x and v_y, as detailed in (2); μ_r denotes the mean value of the reestimated height of all N ground points.

With this analysis, we propose a joint algorithm that can simultaneously detect ground points and estimate accurate radar velocity, inspired by the well-known Random Sample Consensus (RANSAC) [26] algorithm. The RANSAC algorithm can estimate model parameters from a set of data that contain outliers. It uses repeated random sampling to find hypothetical inliers and estimate model parameters based on these inliers. The fitted model is then tested on the entire dataset to find other inliers, called the consensus set. The algorithm repeats the above steps iteratively until the consensus set contains enough inliers.

Inspired by the basic idea of the RANSAC algorithm, we propose the joint ground points detection and velocity estimation algorithm, which takes account of the characteristics of the odometry problem and radar point clouds. The algorithm can be divided into the following three modules.

1) Sampling Based on Velocity and Ground Priors: For odometry, the velocity between adjacent frames is assumed to be continuous, with no abrupt change. Besides, although the distribution of ground points is unstable, significant changes to the overall range of distribution are not expected to occur. Thus, we can improve the efficiency of sampling by taking advantage of the velocity and ground position priors.

Assume that the estimated radar velocity of the last frame is $(\hat{v}_{x,t-1}, \hat{v}_{y,t-1})$ and the detected ground point set of the last frame is $\boldsymbol{G}_{t-1}$. The point set of the current frame is denoted as $\boldsymbol{P}_t$. Then the sampling method based on velocity and ground priors can be divided into the following steps.

- *Step 1:* Refine the ground point set of the last frame $\boldsymbol{G}_{t-1}$ using the velocity of the frame $(\hat{v}_{x,t-1}, \hat{v}_{y,t-1})$ based on (2). Then perform a farthest point sampling (FPS) [27] algorithm on the refined point set $\boldsymbol{G}'_{t-1}$ and obtain $\boldsymbol{G}''_{t-1}$, which contains N_{sp} samples.
- *Step 2:* Refine the point set of the current frame $\boldsymbol{P}_t$ using the velocity prior $(\hat{v}_{x,t-1}, \hat{v}_{y,t-1})$. The refined point set is denoted as $\hat{\boldsymbol{P}}'_t$.
- *Step 3:* For the nth iteration, if n is not larger than N_{sp}, the sampling index $i_n = \arg\min_l d(\boldsymbol{p}''_n, \hat{\boldsymbol{q}}'_l)$, where $\boldsymbol{p}''_n$ is the nth point in $\boldsymbol{G}''_{t-1}$, $\hat{\boldsymbol{q}}'_l$ is the lth point in $\hat{\boldsymbol{P}}'_t$ and $d(\cdot)$ is the Euclidean distance. If n is larger than N_{sp}, then select a point randomly from the points that have not been sampled.
- *Step 4:* Given the sampling index i_n, the i_nth point $\boldsymbol{p}_{i_n}$ in the original point set $\boldsymbol{P}_t$ is taken as the centroid of all samples. Then all the points in the neighborhood of $\boldsymbol{p}_{i_n}$ are considered as hypothetical inliers, i.e. $\boldsymbol{H}_n = \{\boldsymbol{p} \in \boldsymbol{P}_t, \text{ s.t. } d(\boldsymbol{p}, \boldsymbol{p}_{i_n}) \leq \delta_r\}$, where δ_r is the radius of the neighborhood and $\boldsymbol{H}_n$ is the output hypothetical inliers of the nth iteration.

Due to the inaccuracy of the height of ground points, in the first three steps, we do not use the original point clouds to find a sampling index. Instead, we first refine the point sets of consecutive frames using the velocity prior. Then the point closest to the detected ground points of the last frame is sampled. In the last step, we assume that the points very close to the ground point are probably ground points as well. Thus, all the points in the neighborhood of the point that we just sampled are considered hypothetical inliers.

2) Model Fitting and Consensus set Searching Based on Ground Height Refinement: With the hypothetical inliers $\boldsymbol{H}_n$ that are sampled from the original point clouds, we can estimate the radar velocity by optimizing the height variance of the points, as expressed in (3). After that, we test the estimated model on the entire point clouds and find the consensus set. Specifically, we use the estimated velocity $\{\hat{v}_{x,n}, \hat{v}_{y,n}\}$ to refine the coordinates of all points using (2). The refined point set is denoted as $\boldsymbol{P}'_{t,n}$. Then all the points whose height meets the following condition are considered to belong to the consensus set,

$$\boldsymbol{I}_{c,n} = \left\{i \,\middle|\, \boldsymbol{p}_i \in \boldsymbol{P}'_{t,n},\ \left|h(\boldsymbol{p}_i) - \mu'_{r,n}\right| \leq \delta_h\right\}, \tag{4}$$

where $\boldsymbol{p}_i$ is the ith point in $\boldsymbol{P}'_{t,n}$; $h(\cdot)$ denotes the height of the point; $\mu'_{r,n}$ represents the mean height of $\boldsymbol{P}'_{t,n}$; δ_h is the threshold of the height error of ground points and $\boldsymbol{I}_{c,n}$ denotes the indices of the consensus set in the nth iteration. Then the

consensus set that includes all possible inliers can be represented by $\boldsymbol{C}_n = \{\boldsymbol{p}_i \in \boldsymbol{P}_t \mid i \in \boldsymbol{I}_{c,n}\}$, where $\boldsymbol{p}_i$ is the ith point in the original point clouds $\boldsymbol{P}_t$.

3) Determining Whether to Iterate Based on a Geometric Characteristic and Coarse Velocity Estimation: Unlike the RANSAC algorithm, we cannot determine whether to repeat the above two modules based on the number of inliers in the consensus set, as the number of ground points may not necessarily exceed that of other targets. To address this problem, we design a stopping criterion based on a geometric characteristic and coarse velocity estimation.

First, if the data in the consensus set are real ground points, the points should be distributed on a horizontal plane after refinement. We reestimate the velocity using all the points in the consensus set based on (3). Then the consensus set is refined with the velocity based on (2). The height variance of the refined consensus set $\boldsymbol{C}'_n$ should be lower than a given threshold when the iteration is to terminate.

Besides, considering that there are many noise points in radar point clouds due to multipath effects, we incorporate an additional check of the estimated velocity, to improve the robustness of the algorithm. Although the original height of ground points is inaccurate, they can be used to estimate a coarse velocity. Specifically, we use the original coordinates of the consensus set to fit a model based on (1) with the least squares algorithm. The fitted coarse velocity is denoted as $\{\hat{v}^c_{x,n}, \hat{v}^c_{y,n}\}$. Then the velocity $\{\hat{v}_{x,n}, \hat{v}_{y,n}\}$ estimated by optimization needs to meet the condition $\|\hat{\boldsymbol{v}}_n - \hat{\boldsymbol{v}}^c_n\|_2 \leq \delta_v$ when the iteration terminates, where $\hat{\boldsymbol{v}}_n = (\hat{v}_{x,n}, \hat{v}_{y,n})$, $\hat{\boldsymbol{v}}^c_n = (\hat{v}^c_{x,n}, \hat{v}^c_{y,n})$ and δ_v represents the threshold of velocity error.

If the consensus set obtained in module *2)* satisfies both of the conditions, the iteration will terminate. The consensus set is the detected ground point set and the reestimated velocity $\hat{\boldsymbol{v}}_n$ on the consensus set is the output radar velocity.

Notably, for the initial frame, as there are no prior ground points, a random sampling approach is adopted. To ensure the efficiency and accuracy of initialization, it is required that there are no other targets at close range in front of the radar during the initialization process.

The overall flow of the joint ground points detection and velocity estimation algorithm is illustrated in Algorithm 1.

Algorithm 1: Joint Ground Points Detection and Velocity Estimation Algorithm.

Input: radar point clouds of the current frame $\boldsymbol{P}_t$, ground points of the last frame $\boldsymbol{G}_{t-1}$, radar velocity of the last frame $\hat{\boldsymbol{v}}_{t-1}$, sampling radius δ_r, height error threshold δ_h, height variance threshold δ_{hvar}, velocity error threshold δ_v, max iterations n_{max}

Output: ground points $\boldsymbol{G}_t$, radar velocity $\hat{\boldsymbol{v}}_t$

$\boldsymbol{G}'_{t-1} \leftarrow$ refine $\boldsymbol{G}_{t-1}$ with $\hat{\boldsymbol{v}}_{t-1}$ using Eq. (2);
$\boldsymbol{G}''_{t-1} \leftarrow$ perform farthest point sampling on $\boldsymbol{G}'_{t-1}$;
$\hat{\boldsymbol{P}}'_t \leftarrow$ refine $\boldsymbol{P}_t$ with $\hat{\boldsymbol{v}}_{t-1}$ using Eq. (2);
$N_{sp} \leftarrow$ the number of points in $\boldsymbol{G}''_{t-1}$; $n \leftarrow 0$;
Do
  $n \leftarrow n + 1$;
  if $n \leq N_{sp}$ **then**
    $i_n = \arg\min_l d(\boldsymbol{p}''_n, \hat{\boldsymbol{q}}'_l)$, $\boldsymbol{p}''_n \in \boldsymbol{G}''_{t-1}$, $\hat{\boldsymbol{q}}'_l \in \hat{\boldsymbol{P}}'_t$;
  else
    $i_n \leftarrow$ index from random sampling
  end
  $\boldsymbol{H}_n \leftarrow \{\boldsymbol{p} \in \boldsymbol{P}_t, \text{ s.t. } d(\boldsymbol{p}, \boldsymbol{p}_{i_n}) \leq \delta_r\}$, $\boldsymbol{p}_{i_n} \in \boldsymbol{P}_t$
  $\hat{\boldsymbol{v}}_n \leftarrow$ estimate velocity with $\boldsymbol{H}_n$ using Eq. (3);
  $\boldsymbol{P}'_{t,n} \leftarrow$ refine $\boldsymbol{P}_t$ with $\hat{\boldsymbol{v}}_n$ using Eq. (2);
  $\mu'_{r,n} \leftarrow$ the mean height of $\boldsymbol{P}'_{t,n}$
  $\boldsymbol{I}_{c,n} \leftarrow \left\{i \mid \boldsymbol{p}_i \in \boldsymbol{P}'_{t,n}, \left|h(\boldsymbol{p}_i) - \mu'_{r,n}\right| \leq \delta_h\right\}$;
  $\boldsymbol{C}_n \leftarrow \{\boldsymbol{p}_i \in \boldsymbol{P}_t \mid i \in \boldsymbol{I}_{c,n}\}$;
  $\hat{\boldsymbol{v}}_n \leftarrow$ estimate velocity with $\boldsymbol{C}_n$ using Eq. (3);
  $\boldsymbol{C}'_n \leftarrow$ refine $\boldsymbol{C}_n$ with $\hat{\boldsymbol{v}}_n$ using Eq. (2);
  $hvar_n \leftarrow$ the variance of the height of $\boldsymbol{C}'_n$;
  $\hat{\boldsymbol{v}}^c_n \leftarrow$ fitted to $\boldsymbol{C}_n$ based on Eq. (1);
  $\Delta v_n \leftarrow \|\hat{\boldsymbol{v}}_n - \hat{\boldsymbol{v}}^c_n\|_2$;
while $(hvar_n > \sigma_{hvar}$ *or* $\Delta v > \sigma_v)$ *and* $n < n_{max}$;
$\boldsymbol{G}_t \leftarrow \boldsymbol{C}_n$; $\hat{\boldsymbol{v}}_t \leftarrow \hat{\boldsymbol{v}}_n$;

C. Radar Inertial Odometry

With the ego-velocity estimated using radar point clouds in Section III-B and the angular velocity measured by the IMU sensor, we can obtain the odometry,

$$\begin{aligned} x_{t+1} &= x_t + \left(\hat{v}_{x,t}\cos\left(\psi_t - \frac{\pi}{2}\right) + \hat{v}_{y,t}\cos\left(\psi_t\right)\right)\cdot\Delta t, \\ y_{t+1} &= y_t + \left(\hat{v}_{x,t}\sin\left(\psi_t - \frac{\pi}{2}\right) + \hat{v}_{y,t}\sin\left(\psi_t\right)\right)\cdot\Delta t, \\ \psi_{t+1} &= \psi_t + \omega_t\cdot\Delta t. \end{aligned} \tag{5}$$

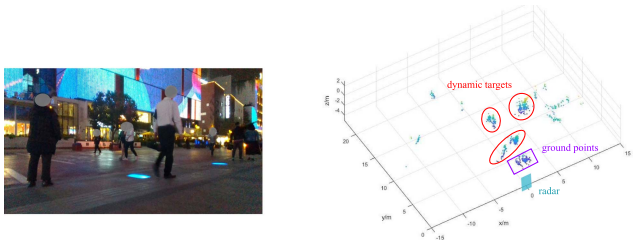
where (x_t, y_t, ψ_t) is the position and yaw angle at time t; $\hat{v}_{x,t}$ and $\hat{v}_{y,t}$ are the estimated radar velocities at time t; ω_t denotes the angular velocity at time t measured by IMU and Δt is the interval between time t and $t+1$. The radar is oriented towards the x-positive axis of the global coordinate system when the yaw angle is zero.

In addition to robust odometry, our method can improve the quality of radar point clouds for subsequent perception tasks by removing clutter points, including the points of dynamic targets and noise points. Specifically, the Doppler velocity of static targets is equal to the radial projection of the radar velocity, as expressed in (1). Based on this, we can distinguish between static targets and dynamic targets. With the estimated radar velocity in the odometry, the targets whose Doppler velocities satisfy (1) are considered static targets and the others are considered to be clutter points.

IV. Experiments and Evaluation

A. Datasets

1) Our Datasets: Although there have been many radar datasets available that contain dynamic objects [28], [29], [30], the extent of dynamism of the scenes in these datasets does



(a) Dynamic environment (b) Corresponding radar points

Fig. 4. Example of the collection environments in our dataset.

not qualify as "high-dynamic". Therefore, we build a real-world dataset containing different high-dynamic environments. Our collection platform is equipped with multi-sensors that are commonly used in mobile robots. The mmWave radar we use is the low-cost and commercial Texas Instruments AWR1843 radar with an azimuth resolution of 14.3° and an elevation resolution of 57.2°. The range and Doppler resolution of the radar waveform we design are 0.045 m and 0.032 m/s respectively. For inertial sensing, we use an LPMS-IG1 IMU. Besides, our mobile platform is equipped with a 32-beam Ouster Lidar and a real-time kinematic (RTK) enabled Global Navigation Satellite System (GNSS) localization system.

Our dataset is collected in different dynamic scenarios including streets and parks, one of which is demonstrated in Fig. 4. As can be seen, many pedestrians are walking in the environment, which is a common dynamic scene for real-world odometry. The trajectories in the dataset are all hundreds of meters long with different shapes. The ground truth of poses is obtained by fusing the results of GNSS and the Lidar-inertial odometry LIO-SAM [7].

2) Public Datasets: To verify the generalization of our method, we evaluate it on the public ColoRadar [31] dataset. The radar waveform used has a range resolution of 0.125 m and a Doppler resolution of 0.04 m/s. Besides, the dataset includes the data from an IMU and a 3D Lidar, along with accurate ground truth of poses. The whole dataset is collected in diverse 3D scenes, which are mainly static environments.

B. Evaluation Metrics and Baselines

Following the widely-used odometry benchmark [2], we adopt the Absolute Trajectory Error (ATE) and Relative Pose Error (RPE) to evaluate the proposed method. The absolute trajectory investigates the global consistency of a trajectory and the relative pose error investigates the local accuracy of localization. Through the root mean square error (RMSE) of these two metrics, we can comprehensively evaluate the performance of the proposed odometry.

We compare our method with the radar-inertial odometry methods EKF-RIO [13] and iRIOM [18], the radar-velocity estimation method in [11], and the Lidar-inertial odometry method LIO-SAM [7]. EKF-RIO is the latest open-source RIO method based on a single-chip mmWave radar known to us and iRIOM[2] is the current state-of-the-art RIO method. Since our method focuses on odometry, we compare it with the version of iRIOM without loop closure (iRIO). Besides, to verify the advantage of exploiting the ground points, we compare the proposed method with the widely-used method in [11] that performs the RANSAC algorithm on all targets to estimate radar velocity. In addition to radar-based methods, we also compare our method with the Lidar-based method LIO-SAM, an open-source Lidar-inertial odometry method that can achieve high positioning accuracy.

[2]The work is not open-source, so we reproduce it.

TABLE I
EVALUATION RESULTS ON OUR DATASET

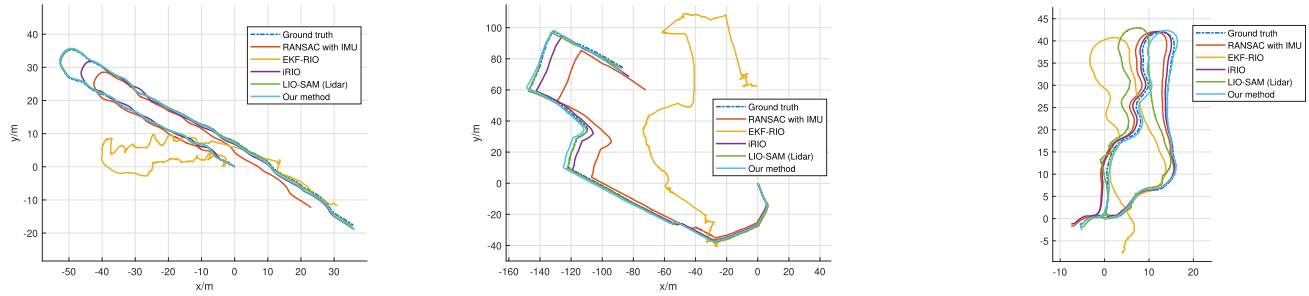
Data	Modality	Method	ATE RMSE		RPE RMSE	
			Trans (m)	Rot (°)	Trans	Rot (°/m)
Seq 1	Radar	EKF-RIO	29.014	7.636	0.901	1.762
		RANSAC[a]	32.698	0.871	3.172	0.903
		iRIO	3.846	3.089	0.147	1.197
		Ours	**0.435**	**0.871**	**0.056**	**0.752**
	Lidar	LIO-SAM	0.722	1.360	0.054	0.787
Seq 2	Radar	EKF-RIO	17.509	19.449	0.648	1.506
		RANSAC	9.460	1.222	0.847	0.858
		iRIO	4.009	5.486	0.156	1.116
		Ours	**0.647**	**1.222**	**0.054**	**0.784**
	Lidar	LIO-SAM	0.193	1.170	0.043	0.754
Seq 3	Radar	EKF-RIO	57.640	25.648	0.282	0.885
		RANSAC	13.779	0.896	0.114	0.798
		iRIO	10.418	6.792	0.095	0.820
		Ours	**1.530**	**0.896**	**0.062**	**0.783**
	Lidar	LIO-SAM	1.126	1.408	0.046	0.799
Seq 4	Radar	EKF-RIO	7.888	8.263	0.272	0.605
		RANSAC	5.370	0.632	0.105	0.498
		iRIO	1.739	2.859	0.096	0.516
		Ours	**0.545**	**0.632**	**0.084**	**0.481**
	Lidar	LIO-SAM	0.845	0.528	0.059	0.439
Seq 5	Radar	EKF-RIO	66.435	34.024	0.417	0.859
		RANSAC	15.747	2.260	0.536	0.652
		iRIO	3.986	4.697	0.128	0.734
		Ours	**2.011**	**2.260**	**0.084**	**0.635**
	Lidar	LIO-SAM	0.842	0.951	0.054	0.559

[a] The estimated velocity is fused with IMU data to obtain odometry.

C. Quantitative Evaluation

1) Results on Our Datasets: The evaluation results on five sequences of our dataset are shown in Table I. The estimated trajectories of two sequences are visualized in Fig. 5(a) and (b). The results in Table I indicate that the proposed method outperforms two RIO methods and the RANSAC method. This can be attributed to the fact that these methods all rely upon the assumption that the static objects in the environment are in the majority. In the scenes where the number of dynamic targets exceeds that of static targets, as observed in our dataset, the methods would produce wrong velocity estimates and thus degrade severely. Although in iRIO, a more robust graduated non-convexity is employed to deal with moving objects, it still cannot handle scenes that primarily consist of dynamic objects. In addition, for the EKF-RIO and iRIO methods, the inaccurate velocity estimates would further adversely impact the estimation of the IMU state.

For the Lidar-based method, due to the larger field of view, the high-dynamic scenes for mmWave radar may not meet the



(a) Seq 2 in our dataset (b) Seq 5 in our dataset (c) Seq 1 in public dataset

Fig. 5. Visualization of odometry results on part of sequences.

TABLE II
EVALUATION RESULTS ON COLORADAR DATASET

Data	Modality	Method	ATE RMSE		RPE RMSE	
			Trans (m)	Rot (°)	Trans	Rot (°/m)
Seq 1[a]	Radar	EKF-RIO	7.258	12.057	0.242	0.707
		RANSAC	1.269	1.752	0.082	1.279
		iRIO	**0.710**	5.395	**0.081**	**0.693**
		Ours	0.725	**1.752**	0.137	1.283
	Lidar	LIO-SAM	3.351	3.945	0.211	2.578
Seq 2[b]	Radar	EKF-RIO	8.659	18.857	0.198	0.763
		RANSAC	1.011	2.545	**0.125**	1.238
		iRIO	0.999	6.193	0.139	**0.672**
		Ours	**0.551**	**2.545**	0.136	1.244
	Lidar	LIO-SAM	0.484	1.974	0.173	2.474

[a] Seq1 denotes 2_28_2021_outdoors_run6 sequence.
[b] Seq2 denotes 2_28_2021_outdoors_run3 sequence.

criteria of being high-dynamic for Lidar. Thus, LIO-SAM does not experience significant degradation. By comparison with LIO-SAM, it can be seen that our odometry reaches positioning accuracy close to the Lidar level, which suggests that our method achieves both robustness and high accuracy in complex dynamic environments.

2) Results on Public Datasets: The evaluation results on the ColoRadar dataset are presented in Table II and the estimated trajectories of one sequence are shown in Fig. 5(c). It can be seen that the proposed method achieves similar performance on the ColoRadar dataset as on our dataset. This indicates that our method has good generalization on the data using different radar waveforms and collected in different scenes with different radar mounting locations.

3) Running Time: We evaluate the running time of our method using an embedded platform NVIDIA Xavier NX. The results in Fig. 6 show that our method can run in real time with an average time of 41.6 ms per frame. The most costly part is using all ground points to fit a model at the end of each iteration. Compared to RANSAC, although our method requires slightly more time in regular environments, it remains stable in high-dynamic environments. By contrast, the time of RANSAC increases significantly in high-dynamic environments due to the difficulty of convergence. In terms of memory consumption, our method requires about 500 KB of memory to process one frame of data.

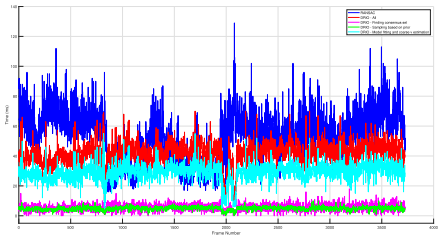
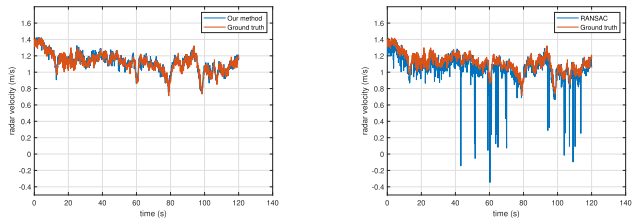


Fig. 6. Running time of our method and RANSAC method.



(a) Our method (b) RANSAC on manually extracted ground points

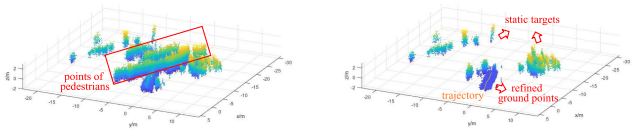
Fig. 7. Ablation study of ground points detection.

D. Ablation Study

To verify the necessity and effect of detecting ground points, we compare our method in Section III-B with the method that estimates velocity by performing the RANSAC algorithm on the manually extracted ground points. To be specific, according to the mounting location of the radar on our platform, we manually extract the points according to the distribution of ground points as shown in Fig. 3(a). The estimated radar velocities on Seq 1 are shown in Fig. 7. As can be seen, when there are many moving targets close to the radar, the velocities estimated with the manually extracted ground points are highly inaccurate, while our method can jointly detect ground points and estimate accurate velocity.

E. Point Cloud Improvement

To verify the effect of our method to improve the quality of radar point clouds, we demonstrate the mapping results that use original point clouds and improved point clouds respectively, which is shown in Fig. 8. To be specific, we transform the point clouds into the global coordinate system according to the ground



(a) Map with original point clouds (b) Map with improved point clouds

Fig. 8. Mapping results using original point clouds and improved point clouds respectively.

truth of poses. Then, the point clouds of all frames are merged to build the map. From Fig. 8, we can see that the map built with the original point clouds contains many clutter points, including the points of dynamic targets and noise points. By contrast, the map built with the improved point clouds has removed the clutter points and identified the refined ground, which is more efficient for subsequent tasks such as relocalization and path planning.

V. Conclusion

In this letter, we propose a robust radar-inertial odometry method in high-dynamic environments by taking advantage of the ground points. To tackle the discontinuous distribution of the ground points, we propose a joint ground detection and velocity estimation algorithm by exploiting the Doppler and geometric characteristics of the ground. Experimental results indicate that the proposed odometry achieves reliable performance in complex dynamic environments. Moreover, our method can improve the quality of radar point clouds effectively for subsequent tasks. In future research, we would like to extend the work to 6-DoF pose estimation and explore a tighter fusion between radar and IMU.

References

[1] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, Dec. 2021.

[2] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[3] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2320–2327.

[4] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.

[5] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, Apr. 2017.

[6] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," in *Proc. Int. Conf. Robot.: Sci. Syst.*, 2014, pp. 1–9.

[7] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.

[8] Y. Cheng, J. Su, M. Jiang, and Y. Liu, "A novel radar point cloud generation method for robot environment perception," *IEEE Trans. Robot.*, vol. 38, no. 6, pp. 3754–3773, Dec. 2022.

[9] M. Barjenbruch, D. Kellner, J. Klappstein, J. Dickmann, and K. Dietmayer, "Joint spatial-and Doppler-based ego-motion estimation for automotive radars," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 839–844.

[10] C. X. Lu et al., "milliEgo: Single-chip mmWave radar aided egomotion estimation via deep sensor fusion," in *Proc. 18th Conf. Embedded Netw. Sensor Syst.*, 2020, pp. 109–122.

[11] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using Doppler radar," in *Proc. IEEE 16th Int. Conf. Intell. Transp. Syst.*, 2013, pp. 869–874.

[12] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using multiple Doppler radars," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 1592–1597.

[13] C. Doer and G. F. Trommer, "An EKF based approach to radar inertial odometry," in *Proc. IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, 2020, pp. 152–159.

[14] Y. Almalioglu, M. Turan, C. X. Lu, N. Trigoni, and A. Markham, "Milli-RIO: Ego-motion estimation with low-cost millimetre-wave radar," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3314–3323, Feb. 2021.

[15] C. Doer and G. F. Trommer, "Yaw aided radar inertial odometry using Manhattan world assumptions," in *Proc. IEEE 28th Saint Petersburg Int. Conf. Integr. Navigation Syst.*, 2021, pp. 1–9.

[16] A. Kramer, C. Stahoviak, A. Santamaria-Navarro, A.-A. Aghamohammadi, and C. Heckman, "Radar-inertial ego-velocity estimation for visually degraded environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 5739–5746.

[17] J. Michalczyk, R. Jung, and S. Weiss, "Tightly-coupled EKF-based radar-inertial odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 12336–12343.

[18] Y. Zhuang, B. Wang, J. Huai, and M. Li, "4D iRIOM: 4D imaging radar inertial odometry and mapping," *IEEE Robot. Automat. Lett.*, vol. 8, no. 6, pp. 3246–3253, Jun. 2023.

[19] Y. S. Park, Y.-S. Shin, and A. Kim, "PhaRaO: Direct radar odometry using phase correlation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 2617–2623.

[20] M. Gadd, D. De Martini, and P. Newman, "Look around you: Sequence-based radar place recognition with learned rotational invariance," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2020, pp. 270–276.

[21] A. Prabhakara et al., "High resolution point clouds from mmWave radar," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 4135–4142.

[22] Y. Cheng, C. Pang, M. Jiang, and Y. Liu, "Relocalization based on millimeter wave radar point cloud for visually degraded environments," *J. Field Robot.*, vol. 40, no. 4, pp. 901–918, 2023.

[23] B. Douillard et al., "On the segmentation of 3D LIDAR point clouds," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 2798–2805.

[24] C. Iovescu and S. Rao, "The fundamentals of millimeter wave sensors," Texas Instrum., pp. 1–8, 2017.

[25] Y. Cheng, H. Xu, and Y. Liu, "Robust small object detection on the water surface through fusion of camera and millimeter wave radar," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 15263–15272.

[26] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[27] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. 31st Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5105–5114.

[28] A. Palffy, E. Pool, S. Baratam, J. F. P. Kooij, and D. M. Gavrila, "Multi-class road user detection with 3+ 1D radar in the View-of-Delft dataset," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 4961–4968, Apr. 2022.

[29] D.-H. Paek, S.-H. Kong, and K. T. Wijaya, "K-Radar: 4D Radar object detection dataset and benchmark for autonomous driving in various weather conditions," in *36th Conf. Neural Inf. Process. Syst. Datasets Benchmarks Track*, 2022. [Online]. Available: https://openreview.net/forum?id=W_bsDmzwaZ7.

[30] L. Zhenget al., "TJ4DRadSet: A 4D radar dataset for autonomous driving," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst.*, 2022, pp. 493–498.

[31] A. Kramer, K. Harlow, C. Williams, and C. Heckman, "ColoRadar: The direct 3D millimeter wave radar dataset," *Int. J. Robot. Res.*, vol. 41, no. 4, pp. 351–360, 2022.