RaDiVe: Robust 4D Radar Odometry with Distance-Bounded NDT and Velocity-Discrepancy Point Uncertainty

Sangwoo Jung[1], Dongjae Lee[1], Chiyun Noh[1], and Ayoung Kim[1]*

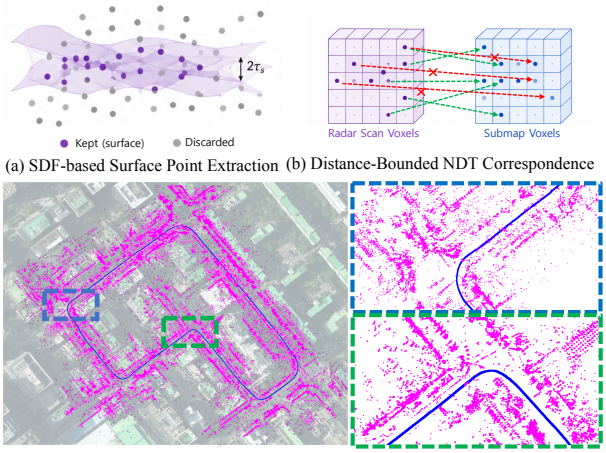
***Abstract*—Recent advances in 4D radar enable robust perception in adverse weather; however, the inherent sparsity, noise, and limited positional precision of radar point clouds pose significant challenges for registration-based odometry. In this letter, we propose RaDiVe, a 4D radar odometry framework that improves the accuracy and robustness of radar point cloud registration. We introduce a distance-bounded Normal Distributions Transform, which improves optimization stability and computational efficiency by restricting the correspondence search to near-distance voxel pairs. To mitigate measurement ambiguity, we propose a velocity-discrepancy point uncertainty model that weights each input 4D radar point according to the discrepancy between its measured Doppler radial velocity and the radial velocity predicted from the estimated ego-velocity. We also incorporate Signed Distance Function-based surface point extraction via implicit neural mapping to construct a geometrically consistent, noise-filtered local submap. Evaluations across multiple public datasets demonstrate that RaDiVe outperforms existing 4D radar odometry baselines by 44.4% in translational and 21.3% in rotational Absolute Trajectory Error on average, while maintaining real-time performance.**

***Index Terms*—Odometry, SLAM, Radar, Range Sensing**

I. Introduction

RADAR serves as a key sensor for robust odometry in challenging environments due to its all-weather perception capabilities [1–7]. As a range-sensing modality similar to Light Detection and Ranging (LiDAR), 4D radar-motivated approaches adopt LiDAR-style scan-to-submap registration to incorporate geometric information into odometry. Nevertheless, the inherent sparsity, limited precision, and high noise characteristic of 4D radar point clouds degrade odometry accuracy, even with recent efforts to increase point throughput.

Despite these data quality challenges, Iterative Closest Point (ICP) is widely explored for existing 4D radar odometry approaches [8–13]. However, since ICP inherently relies on point-level correspondences and assumes dense, high-precision measurements, its mismatch with the sparse, imprecise, and noisy nature of 4D radar point clouds often results in incorrect point correspondences. As an alternative to ICP, Normal Distributions Transform (NDT) registration has attracted attention for robustness on noisy data due to its voxel-based representation and smooth non-linear likelihood surfaces [14–16]. Previous NDT-based approaches have proven effective on Bird's-Eye-View (BEV) radar images; however, extending them to 4D radar point clouds remains challenging because fewer points per voxel yield unreliable voxel statistics and incorrect voxel correspondences.



(a) SDF-based Surface Point Extraction (b) Distance-Bounded NDT Correspondence

(c) Result Point Cloud Map and Trajectory overlapped on a Satellite Map

Fig. 1. (a) Neural points closer than threshold τ_s to the surface are kept, while others are discarded. (b) NDT correspondences are restricted to nearby pairs to enhance optimization stability and efficiency. (c) Qualitative results of RaDiVe on the `Street Day 1` sequence of `HeRCULES`. (Left) Neural point cloud map (purple) projected into a Google Maps satellite view. (Right) Zoomed view of the point cloud map on each region with the same borderline colors as in the left. The blue line represents the RaDiVe odometry estimation.

Data unreliability challenge extends beyond the registration strategy to the selection of reliable points from two key components of scan-to-submap registration: the incoming 4D radar scan and the local submap. For incoming scans, 4D radar provides precise radial velocity measurements, but previous studies primarily use them as optimization residuals [17, 18]. With limited use of velocity information, point uncertainty modeling largely relies on positional covariance [13]. For the local submap, naively accumulating historical 4D radar scans introduces another challenge [5, 10, 19]; noise and outliers accumulate in the local submap, thereby degrading the geometric consistency of the submap. Together, uncertainty in the input 4D radar scan and noise in the local submap compromise odometry accuracy.

Manuscript received: July 30, 2026; Revised: October 21, 2026; Accepted: November 12, 2026. This paper was recommended for publication by Editor Teresa A. Vidal-Calleja upon evaluation of the Associate Editor and Reviewers' comments. *(Corresponding author: Ayoung Kim)*

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) in 2024 (RS-2024-00461409, Development of Harmonic-space Operation using Seamless Information Sharing among Large-scale Heterogeneous Unmanned Vehicles), and in part by the Robotics and AI (RAI) Institute.

[1]S. Jung, D. Lee, C. Noh, and A. Kim are with the Department of Mechanical Engineering, SNU, Seoul, S. Korea `[dan0130, pur22, gch06208, ayoungk]@snu.ac.kr`

The source code is available at https://github.com/SangwooJung98/RaDiVe.

Digital Object Identifier (DOI): see top of this page.

To address these challenges in 4D radar odometry, we introduce **RaDiVe**, a 6-DoF 4D radar odometry framework designed to enable robust NDT registration for sparse 4D radar point clouds. RaDiVe mitigates the instability of NDT on sparse data and prioritizes reliable voxel matches by restricting correspondences to nearby voxel pairs between the input scan and the local submap. This restriction improves both optimization stability and computational efficiency. To further improve accuracy, we introduce two mechanisms that focus on reliable points from both the input 4D radar scan and the local submap. Specifically, we propose a velocity-discrepancy point uncertainty model to weight input scan points, while employing a Signed Distance Function (SDF)-based surface point extractor to prune outliers and preserve geometric structure in the local submap. Extensive evaluations across multiple public datasets demonstrate that RaDiVe reduces the average translational Absolute Trajectory Error (ATE) by 44.4% and rotational ATE by 21.3% relative to the best-performing baselines, while maintaining real-time performance. Fig. 1 shows a representative qualitative odometry and mapping result of RaDiVe and briefly explains the core modules.

Our main contributions are as follows:

- **Distance-Bounded NDT Correspondence:** We restrict NDT correspondences to nearby voxel pairs between the input scan and the submap, based on the high local overlap between consecutive radar scans. This improves both optimization stability and computational efficiency.
- **Velocity-Discrepancy Point Uncertainty:** We propose a point uncertainty model for 4D radar scans based on the discrepancy between measured and propagated radial velocities. The resulting uncertainty provides a reliability measure for each point in the input radar scan.
- **SDF-based Surface Point Extractor:** We introduce SDF-based surface point extraction for the local submap. This mitigates accumulated noise while preserving geometric structure, thereby improving registration accuracy.
- **Evaluation on Public Datasets:** RaDiVe is evaluated on multiple public datasets and outperforms previous state-of-the-art (SOTA) baselines on most evaluated sequences. We will release our algorithm to the robotics community.

II. Related Work

A. Point Cloud Registration on Radar Odometry

1) ICP-registration in 4D Radar Odometry: With recent advances in 4D radar point density, subsequent studies incorporate geometric information into odometry estimation by adopting LiDAR-inspired point cloud registration, particularly ICP. 4DRadarSLAM [8] employ Generalized Iterative Closest Point (GICP) with an adaptive probability distribution. Radar4motion [10] propose polar-grid-based local Radar Cross Section (RCS) feature extraction for scan-to-submap ICP. Herraez et al. [9] introduce Doppler-aware ICP for scan-to-scan registration and later incorporated ground constraints to radar-inertial simultaneous localization and mapping (SLAM) framework RaI-SLAM [11]. Kim et al. [20] integrate Doppler-velocity-based correspondences into ICP. ClarRO [21] fuses Doppler and RCS to down-weight the radar's weak elevation axis during ICP registration. Nevertheless, the reliance of ICP on accurate point-level correspondences remains poorly suited to the sparse and low-precision nature of 4D radar.

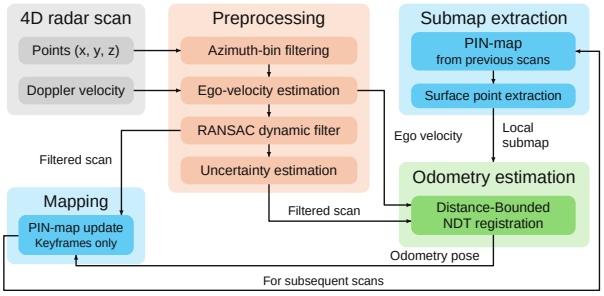


Fig. 2. Overall pipeline of RaDiVe.

2) NDT-registration in Radar Odometry: To address the limitations of ICP, NDT emerges as a robust alternative. Previous studies demonstrate its effectiveness on 2D BEV representations generated by scanning and 4D radars [14–16]. However, extending NDT to radar point clouds remains challenging because the voxel count increases while points per voxel decrease, leading to unreliable voxel statistics and unstable optimization. Li et al. [19] demonstrate the potential of distribution-based registration for 4D radar, but they limit registration to keyframes because of point cloud sparsity and computational cost. Thus, achieving stable, real-time NDT odometry at the sensor rate remains an open problem. RaDiVe introduces Distance-Bounded NDT, which restricts optimization to a reliable set of voxel correspondences, improving robustness while reducing computational cost.

B. Point Reliability in Radar Point Cloud

Beyond the registration algorithm, selecting reliable points is crucial for registration accuracy. Several methods apply heuristic input filtering, such as an RCS-bounded filter [22], whereas others estimate point-wise uncertainty based solely on point position [13] or jointly consolidate point and pose uncertainty [23]. Although the importance of radial velocity is recognized, it is generally incorporated as an additional residual term in optimization [17, 18], rather than a point-reliability term. Consequently, incorporating radial velocity measurements to derive uncertainty weight for each point remains insufficiently explored. RaDiVe addresses this by introducing a radial-velocity-informed point uncertainty model.

III. Method

We first preprocess to remove dynamic points and duplicate returns caused by radio-wave penetration. After extracting reliable points from the input radar scan, we estimate the ego-velocity and calculate point uncertainty using radial-velocity measurements. Surface point extraction module constructs a local submap from the global map. The current scan is then aligned with this submap using Distance-Bounded NDT registration. After pose estimation, selected keyframes are used to update the map. Fig. 2 presents the pipeline overview.

A. Radar Pointcloud Preprocessing

The raw radar scan is preprocessed to remove noise, dynamic returns, and duplicated points before registration.

For input 4D radar pointcloud $\mathcal{P} = \{(x_k, y_k, z_k, v_k)$, where $(x, y, z) \in \mathbb{R}^3$, $v =$ Radial Velocity$\}_{k=1}^{N}$, we apply the following preprocessing steps to retain reliable static points.

1) Azimuth-bin filtering: One notable difference between LiDAR and 4D radar is that 4D radar scans contain duplicated returns along the same direction due to radio-wave penetration. Points generated from scattering and reflection when penetrating objects may be inaccurate; therefore, we integrate azimuth-bin filtering. We start by computing the range, azimuth, and elevation $(r_k, \alpha_k, \varepsilon_k)$ of each point as follows:

$$r_k = \sqrt{x_k^2 + y_k^2 + z_k^2}, \tag{1}$$

$$\alpha_k = \arctan(y_k, x_k), \quad \alpha_k \in [-\pi, \pi], \tag{2}$$

$$\varepsilon_k = \arcsin\big(z_k / r_k\big), \quad \varepsilon_k \in [-\tfrac{\pi}{2}, \tfrac{\pi}{2}]. \tag{3}$$

Since many real-world environments contain meaningful structural information in the BEV, which primarily represents the azimuthal structure [24], we quantize only the azimuth direction while preserving the original elevation values. We quantize the azimuth into discrete bins of width $\Delta\alpha$:

$$b_k = \left\lfloor \frac{\alpha_k + \pi}{\Delta\alpha} \right\rfloor, \quad b_k \in \{0, 1, \ldots, B-1\}, \tag{4}$$

$$e_k = \varepsilon_k. \tag{5}$$

We group points by azimuth bin and elevation. Within each group (b, e), we retain the closest point as follows:

$$k^*_{b,e} = \underset{k:\,(b_k, e_k)=(b,e)}{\arg\min} \; r_k. \tag{6}$$

Azimuth-bin filtering serves as an auxiliary preprocessing step intended to suppress potential duplicated returns.

2) Dynamic point filtering via RANSAC and ego-velocity: To filter out dynamic points and calculate robust ego-velocity, we employ random sample consensus (RANSAC)-based filtering and Least Square (LSQ)-based ego-velocity calculation similar to existing methods [1, 3, 6, 25]. For a return generated by a static object, its radial velocity should equal the projection of the negative sensor ego-velocity onto the point direction. By extending sinusoidal fitting from 2D to 3D, the 3D ego-velocity $\mathbf{v}$ of the robot and static points are extracted.

B. Velocity-Discrepancy Point Uncertainty

Existing approaches model radar positional uncertainty in polar coordinates [13]. Doppler measurements provide complementary information for radar tracking [26]. Our velocity-discrepancy point uncertainty represents a Doppler-consistency-based reliability cue for registration, rather than a complete model of 3D positional uncertainty.

An overview of the proposed velocity-discrepancy point uncertainty is shown in Fig. 3. Given the robust ego-velocity $\mathbf{v}$ from III-A2, we project it to each static point. For a static point p_k, its unit direction vector is $\mathbf{d}_k = \frac{(x_k, y_k, z_k)^\top}{\|(x_k, y_k, z_k)\|}$. The propagated radial velocity v_k^{prop} is calculated as $v_k^{\text{prop}} = -\mathbf{d}_k^\top v$.

We calculate the discrepancy between the measured and propagated radial velocities as $\Delta v_k = \big| v_k^{\text{prop}} - v_k \big|$. Because $\Delta v_k \in [0, \infty)$, we bound the reliability weight $w_k \in (0, 1]$ using an exponential function as follows:

$$w_k = \exp\big(-\Delta v_k\big) = \exp\big(-|v_k^{\text{prop}} - v_k|\big). \tag{7}$$

Radial velocities are expressed in $\mathrm{m/s}$, while (7) uses their numerical values, corresponding to a velocity scale of $1\,\mathrm{m/s}$. For fixed measured Doppler velocity and ego-velocity, the weight is insensitive to displacement along the viewing ray. Ego-velocity estimation errors can also affect its reliability.

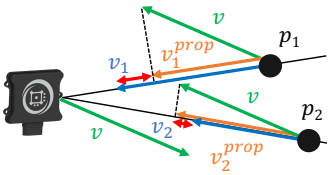


Fig. 3. Visualization of velocity-discrepancy point uncertainty. Blue and orange arrows represent Doppler measured and ego velocity-propagated radial velocity, respectively. The larger discrepancy at p_1 indicates higher uncertainty and a lower reliability weight.

C. Surface Point Extraction for Submap

To construct a low-noise local map for scan-to-submap registration in III-D, we extract points lying on stable surfaces. We leverage the Point-based Implicit Neural Representation (PIN)-mapping [27] for its fast, robust mapping based on an implicit neural representation that learns an SDF from input point clouds. PIN-map represents local geometry using neural points, each defined by a position $x_k \in \mathbb{R}^3$, orientation $q_k \in \mathrm{SO}(3)$, and latent geometric feature $f_k^g \in \mathbb{R}^d$. Following PIN-SLAM [27], the implicit SDF decoder D_θ is trained using samples along measurement rays, including measured end-points, Gaussian samples around them, and uniformly sampled points in front of and behind them. For a measured range ρ and a sample distance ℓ along the same ray, the supervision target is the signed projective distance $\rho - \ell$, which is positive before the return and negative behind it. Training uses sigmoid-based binary cross-entropy with Eikonal regularization, while we retain this sampling and supervision formulation.

Registration is performed for every scan, while the map is updated at selected keyframes after pose estimation. A keyframe is selected when the translation or yaw change relative to the previous keyframe exceeds δ_t or δ_r, respectively. Representative settings are $\delta_t = 2\mathrm{m}$ and $\delta_r = 3^\circ$, while the specific thresholds may vary across sequences. The local neural-point subset is refreshed for each scan based on distance to the sensor and the travel distance since each point was created, while excluded points remain in the global PIN-map.

For submap generation, rather than generating candidate points, we directly use each neural point position in the trained PIN-map as a query to evaluate its learned SDF. SDF of each neural point x_j is computed using its neighboring neural points x_k as $s_{jk} = D_\theta\big(f_k^g,\ d_{jk}\big)$, where $d_{jk} = q_k(x_j - x_k)q_k^{-1}$. These predictions are aggregated through inverse-distance weighting to emphasize nearby neural points:

$$S(x_i) = \frac{\sum_{j \in \mathcal{N}(x_i)} \nu_{ij} s_{ij}}{\sum_{j \in \mathcal{N}(x_i)} \nu_{ij}}, \quad \text{with} \quad \nu_{ij} = \|x_i - x_j\|^{-2}, \tag{8}$$

where $\mathcal{N}(x_i)$ is K-nearest neighbors found via voxel hashing.

The resulting SDF values allow us to extract reliable surface points by retaining points near the implicit surface. We empirically set the surface threshold to $\tau_s = 0.01\mathrm{m}$ to balance surface selectivity and point availability for registration.

$$\mathcal{P}_{\text{surf}} = \big\{\, x_i \in \mathcal{M} \;\big|\; |S(x_i)| \le \tau_s \big\}, \tag{9}$$

where $\mathcal{M}$ denotes the set of neural point positions in the local PIN-map. $\mathcal{P}_{\text{surf}}$ provides a noise-filtered and geometrically consistent surface representation for NDT-based registration.

D. Distance-Bounded NDT for Radar Odometry

We denote the filtered current radar scan as the source point cloud and the extracted surface points $\mathcal{P}_{\text{surf}}$ as the target

submap. We perform NDT registration using the reliability weights derived from the velocity-discrepancy point uncertainty. For voxel-based Gaussian construction, each voxel V_i containing n_i points is characterized by mean μ_i, covariance Σ_i, and average weight w_i:

$$\Sigma_i = \frac{1}{n_i}\sum_{j=1}^{n_i}(p_j-\mu_i)(p_j-\mu_i)^\top + \epsilon I, \tag{10}$$

$$\mu_i = \frac{1}{n_i}\sum_{j=1}^{n_i} p_j, \qquad w_i = \frac{1}{n_i}\sum_{j=1}^{n_i} w_{p_j}, \tag{11}$$

where p_j denotes a point within voxel V_i, w_{p_j} is velocity-discrepancy point uncertainty weight from III-B, and $\epsilon = 0.01\,\mathrm{m}^2$ is the covariance regularization constant. We use anisotropic voxels of size $2\mathrm{m}\times 2\mathrm{m}\times 1\mathrm{m}$ to provide sufficient spatial support. Source-point weights are averaged to scale the NDT matching cost, while voxel means and covariances remain unweighted. This reduces the influence of Doppler-inconsistent source Gaussians without changing their shape.

a) Correspondence Selection: We construct source Gaussians (μ_A, Σ_A, w) in the current radar frame and target Gaussians (μ_B, Σ_B) after expressing the submap in the previous radar frame. Before applying the initial relative transformation, we retain all pairs satisfying $\|\mu_A - \mu_B\| < d_{\max}$, with a representative threshold of $d_{\max} = 10\mathrm{m}$. These correspondences remain fixed throughout LM optimization. Restricting the pair set excludes distant matches and reduces the number of pairs evaluated during optimization.

b) Cost Function: The transformation between consecutive radar frames is parameterized as a 6-degree of freedom (DoF) vector $\xi = [r, p, y, t_x, t_y, t_z]^\top$, where (r, p, y) denote roll, pitch, yaw, and (t_x, t_y, t_z) represent translations. The optimal transformation ξ^* minimizes the following weighted NDT cost function as:

$$Cost(\xi) = \sum_i \left[1 - \frac{\exp\left(-\frac{1}{2}\Delta_i(\xi)^\top \Sigma_i(\xi)^{-1}\Delta_i(\xi)\right)}{\sqrt{(2\pi)^3|\Sigma_i(\xi)|}}\right] w_i, \tag{12}$$

where the difference and covariance are defined as

$$\Delta_i(\xi) = R(\xi)\mu_{A,i} + t(\xi) - \mu_{B,i}, \tag{13}$$

$$\Sigma_i(\xi) = R(\xi)\Sigma_{A,i}R(\xi)^\top + \Sigma_{B,i}. \tag{14}$$

Here, $R(\xi)$ and $t(\xi)$ are the rotation matrix and translation vector derived from Euler angles and translations, respectively. We use this local Euler angle parameterization for the small inter-scan rotations expected in the ground vehicle setting.

c) Ego-Velocity Translation Prior: To improve robustness and convergence for low-speed motion, we incorporate a translation prior using the ego-velocity estimate in III-A2:

$$Cost_{\text{prior}}(\xi) = \frac{\lambda_{\text{trans}}}{2}\|\mathbf{t}(\xi) - \mathbf{t}_{\text{init}}\|^2, \tag{15}$$

where $\mathbf{t}_{\text{init}}$ denotes the initial translation from ego-motion, activated when the ego-velocity is lower than $1.5\mathrm{m}/\,\mathrm{sec}$.

d) Levenberg-Marquardt (LM) Optimization: We employ the LM algorithm to iteratively update the transformation parameters ξ by solving $(\mathbf{H} + \lambda\,\mathrm{diag}(\mathbf{H}))\,\delta\xi = -\mathbf{g}$. Upon convergence, the optimal transformation is given by:

$$T = \begin{bmatrix} R(\xi^*) & t(\xi^*) \\ 0 & 1 \end{bmatrix}. \tag{16}$$

LM is initialized using ego-velocity and the previous inter-scan rotation. We use at most five iterations, terminating when an update decreases the cost by less than 10^{-6}.

IV. Experiments

A. Dataset and Evaluation

Our approach is validated on three public 4D radar benchmarks: `Ntu4dradlm` [28], `HeRCULES` [29], and `SNAIL` [30]. `Ntu4dradlm` provides Oculii Eagle scans on both a car and a handcart platform; `HeRCULES` uses a car-mounted Continental ARS548 over multiple loops or a traffic-jammed scenario; `SNAIL` contains sequences from both sensors attached to a car-based system. Oculii Eagle yields ∼7500 points/frame at 12 Hz, whereas Continental ARS548 returns ∼150 points/frame at 20 Hz, differing heavily in sensor frequency, point throughput, and even point precision. Azimuth-bin filtering is enabled for Oculii Eagle with $\Delta\alpha = 0.175°$, while disabled for Continental ARS548 to avoid further reducing the already sparse input. To accommodate the extreme sparsity of Continental ARS548, each source point is represented by a Gaussian centered at that point with fixed isotropic covariance $\Sigma_i = (0.25\mathrm{m})^2 I$. Target Gaussians remain voxel-based, with covariances computed using the formulation in III-D, while the registration formulation remains unchanged.

Odometry accuracy is measured with the Evo evaluator [31] via root mean square error (RMSE) of translational [m] and rotational ATE [°] to evaluate the accuracy and stability of methods. Estimated and ground-truth trajectories are associated by nearest timestamps, rigidly aligned in SE(3) without scale correction, and then projected onto the XY plane. We report the RMSE of translational error as ATE_t [m] and rotational error as ATE_r [°]. We additionally report translational [%] and rotational [°/m] Relative Trajectory Error (RTE), averaged over all evaluated sub-trajectories spanning 10–50% of each sequence's ground-truth path length [32]. We benchmark against open-source ICP based 4D radar-only odometry methods: 4DRadarSLAM [8], Radar4Motion [10]. Furthermore, to compare with recent ICP algorithms, KISS-ICP [33] and Doppler-ICP [18] are chosen for their generality and explicit design for 4D point clouds. We report the quantitative evaluation of RaDiVe to horizontal translation and yaw. For a fair odometry comparison, loop closure is disabled in all baseline methods. In the result tables, best results are in **Bold**, the second best results are Underlined, and '–' indicates convergence failure during optimization.

B. Evaluation on the `Ntu4dradlm` Dataset

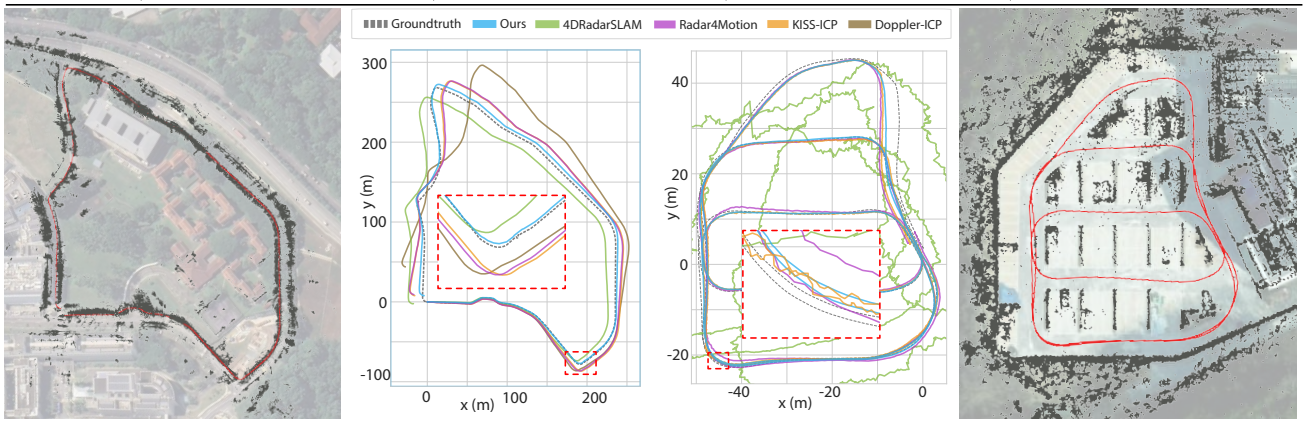
As Table I, RaDiVe achieves the lowest ATE_t and ATE_r across most sequences in `Ntu4dradlm`, outperforming point correspondence–based methods by a wide margin. In particular, for shaky handcart sequences (`cp`, `nyl`), RaDiVe demonstrates strong robustness from the combination of NDT-based registration and a translation prior, with the ATE_t in `cp` being about $5\times$ lower than any baseline. Even on smoother car-platform sequences (`loop2`, `loop3`), RaDiVe maintains superior accuracy, highlighting its ability to preserve global consistency. 4DRadarSLAM achieves the second-best result in most cases and even outperforms RaDiVe on `loop2` with ATE_t for its compatibility with `Ntu4dradlm`. RaDiVe also achieves the lowest translational and rotational RTE on all sequences. Notably, on `loop2`, RaDiVe yields lower relative

TABLE I
EVALUATION RESULTS IN NTU4DRADLM WITH OCULII

	loop3				loop2				cp				nyl			
	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r
4DRadarSLAM	12.802	2.074	2.166	0.00208	**54.733**	5.954	5.349	0.00412	1.852	1.189	3.644	0.01672	6.253	1.477	3.279	0.00752
Radar4Motion	91.364	7.685	4.553	0.00485	255.422	17.837	11.504	0.01367	2.710	2.051	4.823	0.02309	5.397	3.082	4.555	0.00742
KISS-ICP	17.297	1.594	1.807	0.00181	85.657	6.937	4.536	0.00474	2.980	2.493	5.086	0.02176	6.952	2.653	4.719	0.00690
Doppler-ICP	29.897	2.287	2.567	0.00250	133.028	10.658	7.085	0.00738	2.304	2.393	3.602	0.02355	14.552	6.507	5.780	0.02087
Ours	**9.914**	**1.147**	**1.396**	**0.00151**	70.825	**5.928**	**3.451**	**0.00396**	**0.387**	**0.768**	**1.179**	**0.01012**	**2.970**	**1.472**	**1.888**	**0.00364**

TABLE II
EVALUATION RESULTS IN HERCULES WITH ARS548

	Library				Sports Complex				Parking Lot				Street			
	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r
4DRadarSLAM	120.454	21.888	28.690	0.07228	108.178	59.607	29.304	0.15423	17.173	13.974	16.308	0.08448	151.787	61.270	91.328	0.44790
Radar4Motion	8.311	4.149	1.808	0.00894	3.899	1.752	1.760	0.00447	1.712	2.642	2.744	0.01415	5.463	4.582	4.585	0.01475
KISS-ICP	5.347	3.365	1.281	0.00731	2.843	2.004	1.715	0.00582	1.329	2.959	2.683	**0.01045**	4.154	5.119	**3.730**	0.00937
Doppler-ICP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Ours	**2.179**	**2.901**	**0.927**	**0.00333**	**1.691**	**1.484**	**1.555**	**0.00366**	**1.034**	**1.026**	**2.484**	0.01728	**2.600**	**2.571**	3.743	**0.00466**



(a) nyl map and odometry **(b)** Estimated trajectory in nyl **(c)** Estimated trajectory in P. Lot **(d)** P. Lot map and odometry

Fig. 4. Qualitative analysis of nyl and Parking Lot Night 3. Red lines in Fig. 4a and Fig. 4d are estimated odometry.

errors than 4DRadarSLAM despite its higher translational ATE. Qualitative results are visualized in Fig. 4a and Fig. 4b for the nyl sequence, where other methods suffer from global drift while RaDiVe robustly estimates odometry.

C. Evaluation on the HeRCULES Dataset

Table II shows that RaDiVe substantially outperforms other radar-only baselines on both ATE_t and ATE_r across every sequence. This is primarily due to the distance-bounded NDT and velocity-discrepancy point uncertainty, which are particularly effective in the highly sparse but precise radial velocity of Continental ARS548, leading to accurate odometry estimation. Street Day 1 introduces additional challenges: traffic congestion and stop-and-go motion in the urban environment, which degrade data quality compared to open-area sequences. While 4DRadarSLAM and Doppler-ICP fail under these conditions, RaDiVe maintains reliable odometry due to its translation prior scheme based on ego-velocity. For relative errors, RaDiVe achieves the lowest translational and rotational RTE on three of the four sequences for each metric.

Fig. 4c and Fig. 4d include odometry and mapping results of RaDiVe in Parking Lot Night 3, confirming its robustness and stability. 4DRadarSLAM and Doppler-ICP diverge due to registration failure. KISS-ICP exhibits high local jittering, as evident in the zoomed view, resulting from the registration instability of ICP on the extremely sparse 4D radar point cloud of Continental ARS548. Although Radar4Motion presents the only smooth trajectory among the comparison baselines, it suffers from global drift, resulting in a different trajectory on every loop. In comparison, RaDiVe successfully estimated a smooth and accurate trajectory due to its distance-bounded NDT and velocity-discrepancy uncertainty.

D. Evaluation on the SNAIL Dataset

Table III and Table IV summarize results on the SNAIL dataset for Continental ARS548 and Oculii Eagle, respectively. On Continental ARS548, RaDiVe consistently outperforms radar-only baselines in ATE_t and maintains competitive ATE_r. Similar to HeRCULES, Doppler-ICP fails to converge due to the extreme sparsity of the Continental ARS548, while 4DRadarSLAM and Radar4Motion also present reduced accuracy across both ATE_t and ATE_r. KISS-ICP is the only baseline that competes with RaDiVe on ATE_r, but due to its inherent reliance on point-to-point correspondence design, its ATE_t highly diverges compared with the result from RaDiVe. Fig. 5a and Fig. 5b present qualitative results on 20240113_1, presenting others suffer from drift or registration failure, whereas RaDiVe robustly estimates full odometry.

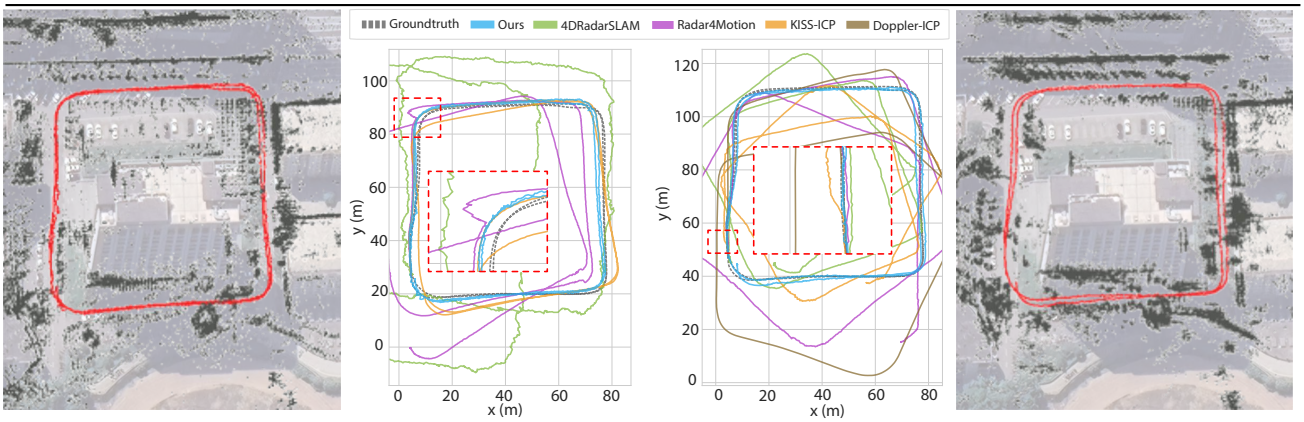
With the Oculii Eagle, RaDiVe achieves the best ATE_t and ATE_r in every sequence except ATE_t in 20240113_3, where Doppler-ICP performs better. However, Doppler-ICP's instability on other sequences and its high computational cost limit its practical use. In contrast, RaDiVe provides robust and stable odometry on all sequences, maintaining real-time performance. Fig. 5c and Fig. 5d present qualitative results

TABLE III
EVALUATION RESULTS IN SNAIL WITH ARS548

	20240113_1				20240113_3				20240123_3			
	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r
4DRadarSLAM	21.602	3.617	15.950	0.03530	134.225	8.739	13.897	0.00826	264.611	25.883	26.134	0.04492
Radar4Motion	10.065	10.735	5.800	0.05072	72.672	10.342	7.189	0.00738	22.974	9.351	4.443	0.01107
KISS-ICP	3.139	3.393	2.752	0.01580	25.282	**3.064**	**2.537**	0.00196	5.166	**3.030**	2.884	0.00510
Doppler-ICP	-	-	-	-	-	-	-	-	-	-	-	-
Ours	**0.885**	**0.963**	**2.037**	**0.00780**	**3.521**	4.782	2.870	**0.00118**	**2.480**	5.554	**2.403**	**0.00191**

TABLE IV
EVALUATION RESULTS IN SNAIL WITH OCULII EAGLE

	20240113_1				20240113_3				20240123_3			
	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r	ATE_t	ATE_r	RTE_t	RTE_r
4DRadarSLAM	10.750	60.474	8.821	0.07664	116.532	88.049	7.381	0.01241	64.069	18.662	11.436	0.02108
Radar4Motion	25.690	33.468	12.660	0.17292	88.049	45.855	8.579	0.02507	79.615	17.896	11.312	0.03661
KISS-ICP	19.460	24.796	13.808	0.12205	57.273	34.600	5.226	**0.00974**	87.179	19.811	15.122	0.04265
Doppler-ICP	40.944	54.793	19.596	0.26977	**34.600**	19.370	5.307	0.01204	295.243	44.363	26.529	0.07998
Ours	**1.497**	**1.385**	**1.195**	**0.00923**	41.886	**5.459**	**4.861**	0.01050	**18.912**	**17.668**	**2.671**	**0.00748**



(a) ARS548 map and odometry **(b)** Estimated trajectory on ARS548 **(c)** Estimated trajectory on Oculii **(d)** Oculii map and odometry

Fig. 5. Qualitative result of 20240113_1 on Continental ARS548 and Oculii Eagle. Red lines in Fig. 5a and Fig. 5d are estimated odometry.

on 20240113_1, where the other baselines suffer from accumulating orientational drift and fail to revisit the loop.

When comparing two 4D radars between Table III and Table IV, 4DRadarSLAM benefits from the denser Oculii Eagle as it was primarily developed for, while RaDiVe outperforms others consistently across both sensors. The comparison also indicates that higher point density does not necessarily yield lower odometry error. The internal point cloud enhancement of Oculii Eagle increases output density but may also affect geometric fidelity, which could be one factor contributing to the observed performance difference between the two sensors.

RaDiVe achieves the lowest RTE in most comparisons across both sensors. It ranks second to KISS-ICP, with modest differences in translational RTE for ARS548 20240113_3 and rotational RTE for Oculii Eagle 20240113_3, demonstrating competitive relative error performance.

E. Effect of Distance-Bounded NDT

We assessed the effectiveness of our distance-bounded NDT registration by comparing it to four alternatives on Ntu4dradlm with all other processing held constant: (1) **Full NDT** (using all possible voxel correspondences without spatial constraints), (2) **GICP**, (3) **P2P ICP** (point-to-plane ICP), and (4) **ICP** (point-to-point ICP), implemented using the Small_gicp [34] library. As shown in Table V, our approach outperforms all baselines on loop2 and nyl for both ATE_t and ATE_r due to its stable, distance-based correspondence

TABLE V
DISTANCE-BOUNDED NDT IN NTU4DRADLM

		Ours	Full NDT	GICP	P2P ICP	ICP
loop2	ATE_t	**70.825**	74.039	209.090	444.414	222.925
	ATE_r	**5.928**	5.995	13.145	27.275	14.420
nyl	ATE_t	**2.970**	5.945	8.804	7.696	7.628
	ATE_r	**1.472**	1.961	3.192	3.021	2.961

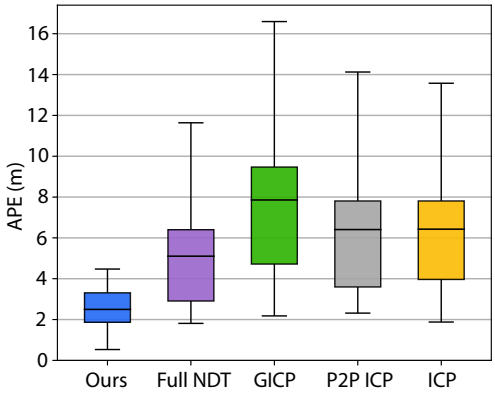


Fig. 6. An evaluation about different registration methods on nyl sequence in the Ntu4dradlm dataset. The median values of ATE_t and standard deviations are (**2.345**, **1.102**), (4.296, 2.903), (6.087, 5.255), (4.903, 4.174), and (5.000, 4.061). Distance-bounded NDT presents the lowest median and standard deviation on ATE_t

TABLE VI
VELOCITY UNCERTAINTY AND SURFACE POINT EXTRACTION

V. Unc.	S. P. E.	PIN-map	Ntu. nyl ATE$_t$	Ntu. nyl ATE$_r$	HeRC. P. Lot ATE$_t$	HeRC. P. Lot ATE$_r$
✗	✗	✗	97.395	65.926	3.928	6.690
✓	✗	✗	81.029	56.116	2.102	4.861
✗	✗	✓	78.051	74.801	30.544	68.228
✓	✗	✓	19.053	12.581	3.267	5.735
✗	✓	✓	5.872	2.649	1.206	3.052
✓	✓	✓	**2.970**	**1.472**	**1.034**	**1.026**

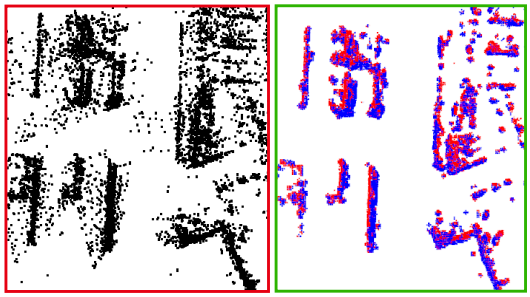


Fig. 7. SDF-based Surface Point Extractor on `Parking Lot`. Noisy points in the left red-bordered figure are deleted in the right green-bordered figure based on the SDF-based surface point extraction. In the right green-bordered figure, the color of the points depicts SDF values; blue indicates positive, and red indicates negative SDF values.

selection. Though Full NDT presents a competitive result with our distance-bounded NDT, this comes at a cost: Full NDT requires $2 \sim 3\times$ longer runtime per frame, making it unsuitable for real-time usage. These results highlight the balance between accuracy and practical efficiency in RaDiVe.

Interestingly, GICP failed to provide robust odometry with its distribution-based representation. This is because GICP generates a single distribution per point using the K-Nearest Neighbor (KNN) approach, which results in an overly dense distribution in regions where dense, noisy points exist, leading to poor correspondence. Comparably, the NDT approach is robust in this situation as a single NDT is generated per voxel. To evaluate our distance-bounded NDT more thoroughly, we drew interquartile range graphs with Absolute Position Error (APE) among the different registration metrics in Table V as Fig. 6. Both median and standard deviation of APE are lowest on our distance-bounded NDT, which supports its robustness and accuracy on a sparse 4D radar point cloud.

F. Effect of Uncertainty and Surface Point Extraction

We evaluate velocity uncertainty (V. Unc.) and surface point extraction (S.P.E.) on `nyl` and `Parking Lot Night 3`, and additionally compare PIN-map with simple accumulated global map. Disabling S.P.E. retains PIN-map construction but omits SDF-based filtering. The accumulation baseline uses the same preprocessing, keyframe criteria, local map selection rules, and registration settings. As shown in Table VI, V. Unc. always reduces ATE on both sequences. On `Parking Lot Night 3`, accumulation outperforms PIN-map without S.P.E. This may reflect the influence of off-surface neural points on registration under sparse ARS548. With S.P.E., PIN-map selects points near the SDF zero-level set and achieves lower errors than accumulation. These results support using PIN-map together with S.P.E. for local submap construction.

TABLE VII
AVERAGE TIME CONSUMPTION IN MULTIPLE PUBLIC DATASETS

Oculii (83 ms)	ms/frame	**ARS548 (50 ms)**	ms/frame
`loop2`	72.47 ms	`Library`	49.30 ms
`nyl`	74.38 ms	`Street`	42.92 ms
`20240123_3`	65.28 ms	`20240123_3`	49.03 ms

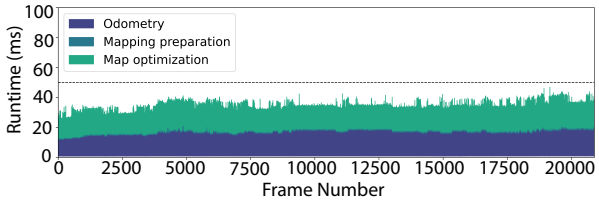


Fig. 8. Time consumption on `Street Day 1`. The horizontal dotted line represents the 50 ms real-time criterion.

G. Computational Cost Evaluation

We analyzed the whole runtime of our Python-based pipeline, including LM optimization and NDT generation. Critical functions are compiled using `@torch.jit.script` decorator for optimized computational efficiency. To evaluate the real-time performance of RaDiVe in general, we selected at least one sequence from every public dataset. As shown in Table VII, the average time consumption per frame of RaDiVe is lower than real-time thresholds of 83ms (Oculii Eagle) and 50ms (Continental ARS548) on every chosen sequence. Furthermore, a qualitative result on time consumption with `Street Day 1`, the most extended sequences among `HeRCULES`, is included in Fig. 8. The map optimization time consumption is plotted based on the average time consumption, as the original time consumption graph is jagged due to the keyframe-based mapping. All benchmarks were conducted on a system including an Intel i7-11700 CPU, 64GB RAM, and an NVIDIA RTX 3080 GPU, verifying that RaDiVe meets real-time constraints on consumer-grade hardware.

H. Cross-Modal Evaluation on FMCW LiDAR

The importance of a cross-modal framework between 4D radar and Frequency Modulated Continuous Wave (FMCW) LiDAR, which both present 4D point clouds but differ in sensor characteristics, has recently emerged, as presented in Doppler-SLAM [35]. Though RaDiVe is not primarily designed for cross-modal purposes, it can process both types of input data. Therefore, we additionally evaluated RaDiVe on dense, high-precision 4D point clouds from the Aeva Aeries II FMCW LiDAR in `HeRCULES`. We compare RaDiVe with FAST-LIO2 [36], a LiDAR-inertial odometry method, and 4DRadarSLAM [8], a radar-based SLAM method, using the same FMCW LiDAR data. For FMCW LiDAR, we reuse the voxel-based NDT settings employed for NTU Oculii.

As shown in Table VIII, radar-based RaDiVe achieves lower translational ATE on all four evaluated sequences and

TABLE VIII
COMPARISON WITH LIDAR-BASED SLAM IN HERCULES WITH AEVA AERIES II SOLID STATE FMCW LIDAR

		Library ATE$_t$	Library ATE$_r$	Sports Complex ATE$_t$	Sports Complex ATE$_r$	Parking Lot ATE$_t$	Parking Lot ATE$_r$	Street ATE$_t$	Street ATE$_r$
LiDAR	**Fast-LIO2**	3.982	2.088	4.635	2.514	1.982	5.382	3.139	5.023
	4DRadarSLAM	26.145	14.591	22.091	8.813	22.373	53.377	23.867	15.727
	Ours (L)	3.425	**1.703**	3.738	2.513	2.033	6.427	2.654	4.885
	Ours (R)	**2.179**	2.901	**1.691**	**1.484**	**1.034**	**1.026**	**2.600**	**2.571**

lower rotational ATE on three of them than both LiDAR-based RaDiVe and FAST-LIO2. On `Library`, however, both LiDAR-based methods achieve lower rotational ATE. This comparison characterizes performance under the evaluated configurations, without establishing the best achievable accuracy of each sensor modality. LiDAR-based RaDiVe remains competitive with FAST-LIO2 and also achieves lower ATE than 4DRadarSLAM on all sequences, supporting its potential applicability to FMCW LiDAR.

V. Conclusion

In this letter, we presented RaDiVe, a robust 4D radar odometry framework that improves the accuracy of radar point cloud registration. Our approach integrates three key components: distance-bounded NDT correspondence, velocity-discrepancy point uncertainty, and SDF-based surface point extraction via PIN mapping. These innovations effectively mitigate the challenges of sparse, noisy, and low-precision radar data on odometry estimation. Extensive evaluations on public datasets demonstrate that RaDiVe outperforms the baselines on most sequences while maintaining real-time performance. As future work, we plan to integrate proprioceptive sensors, such as IMU or leg kinematics, to further improve odometry accuracy.

Acknowledgment

The authors used AI-assisted tools to refine Fig. 1a and Fig. 1b. Final verification was performed by the authors.

References

[1] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using doppler radar," in *Proc. IEEE Intell. Transport. Sys. Conf.*, 2013, pp. 869–874.

[2] C. Doer and G. F. Trommer, "An ekf based approach to radar inertial odometry," in *Proc. IEEE Intl. Conf. on Multisensor Fusion and Integration for Intell. Sys.*, 2020, pp. 152–159.

[3] Y. S. Park, Y.-S. Shin, J. Kim, and A. Kim, "3d ego-motion estimation using low-cost mmwave radars via radar velocity factor for pose-graph slam," *IEEE Robot. and Automat. Lett.*, vol. 6, no. 4, pp. 7691–7698, 2021.

[4] J. Michalczyk, R. Jung, and S. Weiss, "Tightly-coupled ekf-based radar-inertial odometry," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.*, 2022, pp. 12 336–12 343.

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

[6] S. Jung, W. Yang, and A. Kim, "Co-ral: Complementary radar-leg odometry with 4-dof optimization and rolling contact," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.*, 2024, pp. 13 289–13 296.

[7] C. Noh, S. Jung, H. Kim, Y. Hu, L. Herlant, and A. Kim, "Garlileo: Gravity-aligned radar-leg-inertial enhanced odometry," *Intl. J. of Robot. Research*, p. 02783649261457941, 2026.

[8] J. Zhang, H. Zhuge, Z. Wu, G. Peng, M. Wen, Y. Liu, and D. Wang, "4dradarslam: A 4d imaging radar slam system for large-scale environments based on pose graph optimization," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2023, pp. 8333–8340.

[9] D. C. Herraez, M. Zeller, L. Chang, I. Vizzo, M. Heidingsfeld, and C. Stachniss, "Radar-only odometry and mapping for autonomous vehicles," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2024, pp. 10 275–10 282.

[10] S. Kim, J. Seok, J. Lee, and K. Jo, "Radar4motion: Imu-free 4d radar odometry with robust dynamic filtering and rcs-weighted matching," *IEEE Trans. Intell. Vehicles*, 2024.

[11] D. C. Herraez, F. Kaschner, M. Zeller, D. Muhle, J. Behley, M. Heidingsfeld, D. Cremers, and C. Stachniss, "Ground-aware automotive radar odometry," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2025.

[12] W. Yang, H. Jang, and A. Kim, "Ground-optimized 4d radar-inertial odometry via continuous velocity integration using gaussian process," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2025.

[13] Y. Xu, Q. Huang, S. Shen, and H. Yin, "Incorporating point uncertainty in radar slam," *IEEE Robot. and Automat. Lett.*, 2025.

[14] P.-C. Kung, C.-C. Wang, and W.-C. Lin, "A normal distribution transform-based radar odometry designed for scanning and automotive radars," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2021, pp. 14 417–14 423.

[15] R. Zhang, Y. Zhang, D. Fu, and K. Liu, "Scan denoising and normal distribution transform for accurate radar odometry and positioning," *IEEE Robot. and Automat. Lett.*, vol. 8, no. 3, pp. 1199–1206, 2023.

[16] M. Hilger, N. Mandischer, and B. Corves, "Randt slam: Radar slam based on intensity-augmented normal distributions transform," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.*, 2024, pp. 7831–7838.

[17] C. Noh, W. Yang, M. Jung, S. Jung, and A. Kim, "Garlio: Gravity enhanced radar-lidar-inertial odometry," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* IEEE, 2025, pp. 9869–9875.

[18] B. Hexsel, H. Vhavle, and Y. Chen, "Dicp: Doppler iterative closest point algorithm," in *Proc. Robot.: Science & Sys. Conf.*, 2022.

[19] X. Li, H. Zhang, and W. Chen, "4d radar-based pose graph slam with ego-velocity pre-integration factor," *IEEE Robot. and Automat. Lett.*, vol. 8, no. 8, pp. 5124–5131, 2023.

[20] J. Kim, G. Bae, C. Kim, J. Lee, W. Shin, and H. Oh, "Doppler correspondence: Non-iterative scan matching with doppler velocity-based correspondence," *arXiv preprint arXiv:2502.11461*, 2025.

[21] E. Furuno, P. Megarajan, C. Kowalski, T. C. Stratmann, M. Pfingsthorn, and A. Hein, "Claro: A cluster-based method for radar odometry," *IEEE Robot. and Automat. Lett.*, 2026.

[22] Q. Huang, Y. Liang, Z. Qiao, S. Shen, and H. Yin, "Less is more: Physical-enhanced radar-inertial odometry," in *Proc. IEEE Intl. Conf. on Robot. and Automat.*, 2024, pp. 15 966–15 972.

[23] W. Yang, D. Lee, M. Jung, and A. Kim, "Geometrically-constrained radar-inertial odometry via continuous point-pose uncertainty modeling," *IEEE Robot. and Automat. Lett.*, vol. 11, no. 5, pp. 6098–6105, 2026.

[24] L. Luo, S. Zheng, Y. Li, Y. Fan, B. Yu, S.-Y. Cao, J. Li, and H.-L. Shen, "Bevplace: Learning lidar-based place recognition using bird's eye view images," in *Proc. IEEE Intl. Conf. on Comput. Vision*, 2023, pp. 8700–8709.

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

[26] L. Jacobs, P. Veelaert, H. Steendam, and W. Philips, "On the accuracy of automotive radar tracking," in *IEEE Vehicular Tech. Conf.*, 2023, pp. 1–6.

[27] Y. Pan, X. Zhong, L. Wiesmann, T. Posewsky, J. Behley, and C. Stachniss, "Pin-slam: Lidar slam using a point-based implicit neural representation for achieving global map consistency," *IEEE Trans. Robot.*, 2024.

[28] J. Zhang, H. Zhuge, Y. Liu, G. Peng, Z. Wu, H. Zhang, Q. Lyu, H. Li, C. Zhao, D. Kircali *et al.*, "Ntu4dradlm: 4d radar-centric multi-modal dataset for localization and mapping," in *Proc. IEEE Intell. Transport. Sys. Conf.*, 2023, pp. 4291–4296.

[29] H. Kim, M. Jung, C. Noh, S. Jung, H. Song, W. Yang, H. Jang, and A. Kim, "Hercules: Heterogeneous radar dataset in complex urban environment for multi-session radar slam," in *Proc. IEEE Intl. Conf. on Robot. and Automat.* IEEE, 2025, pp. 4649–4656.

[30] J. Huai, B. Wang, Y. Zhuang, Y. Chen, Q. Li, and Y. Han, "Snail radar: A large-scale diverse benchmark for evaluating 4d-radar-based slam," *Intl. J. of Robot. Research*, p. 02783649251329048, 2024.

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

[32] Z. Zhang and D. Scaramuzza, "A Tutorial on Quantitative Trajectory Evaluation for Visual(-Inertial) Odometry," in *Proc. IEEE/RSJ Intl. Conf. on Intell. Robots and Sys.*, 2018, pp. 7244–7251.

[33] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "Kiss-icp: In defense of point-to-point icp–simple, accurate, and robust registration if done the right way," *IEEE Robot. and Automat. Lett.*, vol. 8, no. 2, pp. 1029–1036, 2023.

[34] K. Koide, "small_gicp: Efficient and parallel algorithms for point cloud registration," *Journal of Open Source Software*, vol. 9, no. 100, p. 6948, Aug. 2024.

[35] D. Wang, H. Haag, D. C. Herraez, S. May, C. Stachniss, and A. Nüchter, "Doppler-slam: Doppler-aided radar-inertial and lidar-inertial simultaneous localization and mapping," *IEEE Robot. and Automat. Lett.*, 2025.

[36] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, 2022.