2025 IEEE International Conference on Robotics and Automation (ICRA)
May 19-23, 2025. Atlanta, USA

RLCNet: A Novel Deep Feature-Matching-Based Method for Online Target-Free Radar-LiDAR Calibration

Kai Luan Chenghao Shi* Xieyuanli Chen Rui Fan Zhiqiang Zheng Huimin Lu*

***Abstract*— While millimeter-wave radars are widely used in robotics and autonomous driving, extrinsic calibration with other sensors remains challenging due to the sparsity and uncertainty of radar point clouds. In this paper, we propose a novel deep feature-matching-based online extrinsic calibration approach for a 4D millimeter-wave radar and 3D LiDAR system. We formulate the calibration problem as a cross-modal point cloud registration task, initiating with keypoint-level matching followed by dense matching refinement. Efficient yet powerful neural networks are employed to extract prior keypoint matches, which are then expanded to surrounding regions, establishing dense point correspondences. Our approach effectively leverages the majority of the information from millimeter-wave radar, mitigating the impact of radar point cloud sparsity. We evaluate our approach on two datasets, and experimental results demonstrate that it outperforms state-of-the-art baseline methods and achieves an average improvement of 66.96% in calibration success rate, while reducing translational error and rotational error by 23.84% and 30.31%, respectively. Our implementation will be made open-source at https://github.com/nubot-nudt/RLCNet.**

I. Introduction

Millimeter-wave (mmWave) radars and LiDAR are two commonly employed sensors for online perception of mobile robots or autonomous vehicles. mmWave radar demonstrates superior stability in adverse conditions such as dust, rain, fog, and snow, offering robust all-weather perception capabilities. However, millimeter-wave radar has limited data resolution leading to very sparse point clouds [1], [2]. In contrast, LiDAR stands out for its exceptional three-dimensional environmental reconstruction capabilities, providing high-resolution spatial information for detailed scene representation. However, its performance can be affected by extreme weather conditions. To overcome their individual limitations and enhance overall perception accuracy and reliability, fusing data from both sensors has become a prevalent technique for autonomous mobile systems.

K. Luan, C. Shi, X. Chen, Z. Zheng, and H. Lu are with the College of Intelligence Science and Technology, National University of Defense Technology, China. R. Fan is with the College of Electronics & Information Engineering, Shanghai Institute of Intelligent Science and Technology, Shanghai Research Institute for Intelligent Autonomous Systems, the State Key Laboratory of Intelligent Autonomous Systems, and Frontiers Science Center for Intelligent Autonomous Systems, Tongji University, Shanghai 201804, P. R. China (e-mail: rui.fan@ieee.org).

∗ indicates the corresponding authors: C. Shi and H. Lu

This work was partly supported by the National Science Foundation of China under Grant 62403478, 62473288 and 62233013, the Fundamental Research Funds for the Central Universities, NIO University Programme (NIO UP), and the Xiaomi Young Talents Program. Young Elite Scientists Sponsorship Program by CAST (No. 2023QNRC001), and Major Project of Natural Science Foundation of Hunan Province under Grant 2021JC0004.

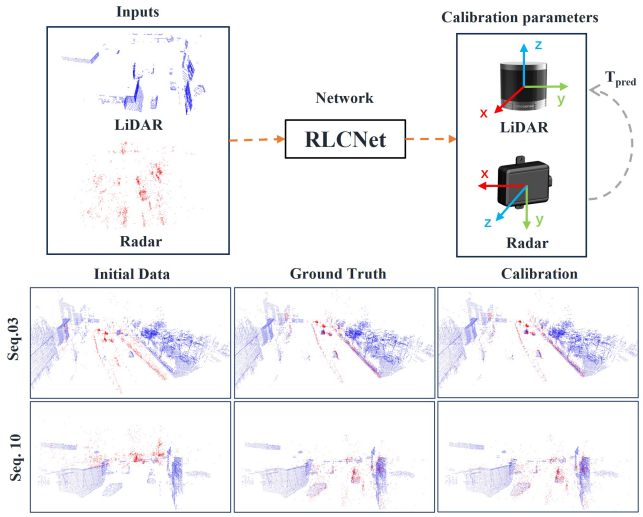


Fig. 1: Our RLCNet establishes correspondences between point clouds generated by a 4D millimeter-wave radar and a LiDAR scanner online, enabling the estimation of a 6-DOF extrinsic calibration between the two modalities.

In this work, we study the radar-LiDAR extrinsic calibration problem so that the radar and LiDAR data can be registered under a common reference frame for subsequent fusion. Current calibration approaches for radar primarily rely on corner reflectors [3], [4] or specialized calibration boards [5]. Heng proposes a target-free calibration method [6] which builds a point cloud map of the environment and subsequently calibrates the pose of the mmWave radar relative to this map. While these approaches have demonstrated promising outcomes, they are constrained to offline implementation. However, during online operation, the sensor pose drift is inevitable, thereby potentially undermining the accuracy of the initial calibration results. Furthermore, in sensor systems lacking hardware time synchronization, the asynchronous nature of data acquisition across sensors can render static extrinsic calibration parameters unreliable. Therefore, there is a compelling need for an online calibration technique.

To the best of our knowledge, no online calibration method has been specifically developed for 4D radar and LiDAR systems. The main challenge in calibrating millimeter-wave radar with other sensors lies in the extreme sparsity of the radar point cloud, which poses significant difficulty for conventional online extrinsic calibration methods. Existing ego-motion-based methods [7], [8], [9] often require accurate odometry estimation which is not always available based on sparse radar data. Similarly, keypoints matching

2025 IEEE International Conference on Robotics and Automation (ICRA) | 979-8-3315-4139-2/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICRA55743.2025.11127342

methods [10], [11], [12] for LiDAR point clouds calibration also fails, since extracting keypoints from highly sparse radar point cloud not only complicates the matching process but also significantly constrains the calibration accuracy. To tackle this challenge, we propose a deep dense feature-matching-based calibration framework that directly establishes correspondences on the raw point clouds to enhance the accuracy and robustness of calibration. The network consists of two main components: first, a keypoint detection and matching module is employed to extract keypoints from radar and LiDAR point clouds and perform matching at the keypoint level. Subsequently, leveraging neighborhood consistency, we extend the matching relationships of the keypoints to their neighboring regions, ultimately constructing dense point matches at the raw point cloud level ensuring calibration robustness and accuracy.

The main contribution of this paper is a novel deep feature-matching-based online calibration framework for radar-LiDAR systems. We tackle the challenges of sparse millimeter-wave radar point clouds by proposing a network that progressively constructs dense matches from keypoint to raw point cloud levels, enabling accurate calibration. Extensive experiments demonstrate that our method outperforms state-of-the-art baseline methods.

II. Related Work

A. Target-Based Extrinsic Calibration

Given the inherent high noise and sparsity of mmWave radar data, most calibration methods employ specialized trihedral reflectors to ensure stable signals within radar data [3], [4], [5], [13], [14]. Peršić et al. [13] achieves 6-degree-of-freedom (6DoF) extrinsic calibration between radar and other sensors by incorporating specially designed chessboard targets and leveraging two-step optimization procedure. Domhof et al. [5] utilize a combination of four-circular-holes foam boards and a trihedral reflector, to ensure simultaneous observation by diverse sensors. Wise et al. [15] propose to use Apriltags and trihedral reflectors achieving spatiotemporal calibration between cameras and radar.

B. Target-Free Extrinsic Calibration

The ego-motion-based approaches [7], [8], [9] exploiting the constraints between the individual sensor motions can be naturally implemented for the calibration of radar-LiDAR system. Persic et al. [16] executed an unsupervised online pairwise extrinsic calibration for 2D radars, cameras, and LiDAR sensors by synchronizing the trajectories of moving objects. This approach relies on priori knowledge of translational parameters, concentrating the estimation process on the yaw angles between the radar-camera and radar-LiDAR sensor pairs. Heng [6] introduced the first feature-based calibration approach. It constructed a point cloud map of the environment using LiDAR and subsequently calibrates the radar relative to this map, thereby determining the extrinsic parameters. An advantage of this method is that it does not require overlapping field-of-view (FOV) between LiDAR and radar; however, this approach can be only performed offline. To date, no feature-macthing-based online calibration method for radar-LiDAR system has been published.

Extensive research has been conducted on utilizing deep learning to extract features across different sensor modalities for online calibration. Existing works [10], [11], [12] adopt supervised or unsupervised methods for extrinsic calibration between LiDAR and cameras. These methods leverage neural networks to learn the shared features and inter-modality correlations, enabling data alignment across modalities. Scholler et al. [17] employed end-to-end deep learning to estimate the rotation matrix of extrinsic parameters, thereby aligning vehicle detections from radar measurements with those in camera images. While deep learning methods have demonstrated their effectiveness in online calibration of sensors such as cameras and LiDAR, their potential in mmWave radar calibration remains largely unexplored.

Due to the inherent similarities between radar and LiDAR point cloud data, point cloud registration methods [18], [19], [20], [21], [22] have the potential to be utilized for the calibration of radar-LiDAR system. However, the sparsity of radar point clouds, combined with the presence of substantial noise and ghost points, presents significant challenges.

III. Our Approach

In this work, we delve into the problem of target-free extrinsic calibration for a 4D mmWave radar and 3D LiDAR system. The goal is to compute the 6-Degrees-of-Freedom (6-DoF) extrinsic parameters between the two sensors. To this end, we introduce a novel framework that model the calibration task as a point matching task, and solve it in two steps, as illustrated in Fig. 2. Firstly, we extract keypoints from both the radar and LiDAR point clouds and establish correspondences on keypoint level. Secondly, we leverage neighborhood consistency and keypoint matches to construct dense point correspondences directly within the raw point clouds, thereby facilitating more precise calibration.

A. Data Preprocessing

The FOV of a typical 4D mmWave radar is less than 120°, meaning that at least two-thirds of the LiDAR data is redundant. Aligning the radar data with such an excess of information presents a significant challenge. Fortunately, within the context of calibration tasks, there is minimal fluctuations. Therefore, we assume that a preliminary estimation of the yaw angle is already known (derived from the preceding calibration iteration; for initial calibrations, this can be readily approximated through manual assessment). Utilizing the estimated yaw angle, we preserve only the LiDAR point cloud that lies within the 180° frontal arc aligned with the radar's orientation. Furthermore, considering the point cloud's diminished density at extended ranges, we restrict only those points from both sets of point clouds that are within a distance of no more than 60 m. Finally, we perform a downsampling of the two point clouds, employing a voxel size of 0.3 m, to standardize the data as input for our neural network.

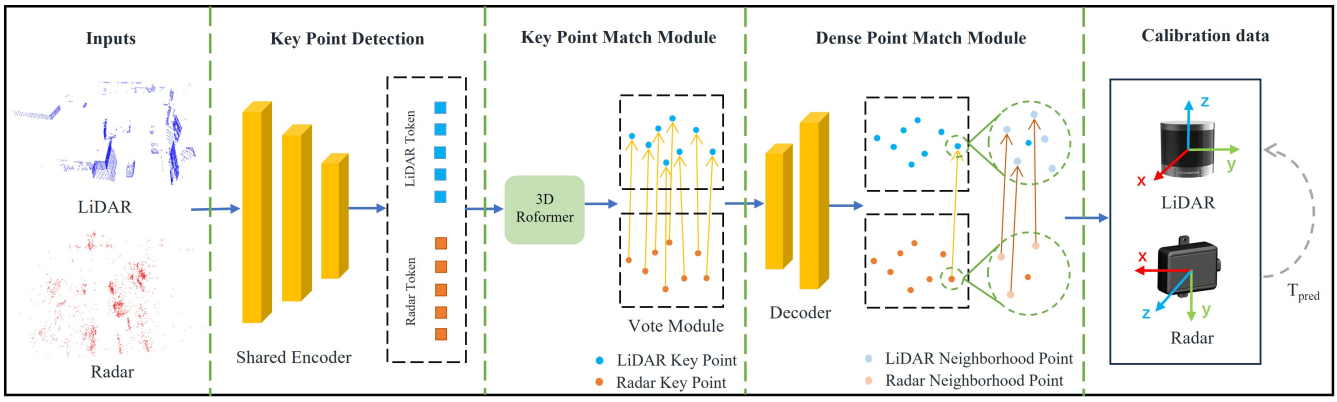


Fig. 2: Our proposed network architecture is used to estimate extrinsic calibration parameters between LiDAR and millimeter wave radar. The inputs to the network are LiDAR point clouds and millimeter wave radar point clouds. The output of the network is the 6-DoF rigid body transformation Tpred between the two sensors. The entire network is divided into Key Point Detection, Key Point Match Module and Dense Point Matching Module.

B. Keypoint Detection

Our approach aims to first extract stable keypoints from two point clouds and establish correspondences as an initial guess for dense point matching. As a start point for stable keypoint extraction, we exploit the KPEncoder [23] as the backbone that hierarchically downsamples and encodes the point cloud into the uniformly distributed nodes with descriptors $[\hat{\mathcal{P}}|\hat{\mathbf{F}}]$. To enable contextual information aggregation and exchange between the two point clouds, we follow 3D-RoFormer [24] and map a point $\boldsymbol{p}_i^Q$ with its feature $\boldsymbol{h}_i^Q$ in the query point cloud Q and all the points in the source point cloud S with a linear projection, given by:

$$\begin{aligned} \boldsymbol{q}_i &= \boldsymbol{W}_1 \boldsymbol{h}_i^Q + \boldsymbol{b}_1, \\ \boldsymbol{k}_j &= \boldsymbol{W}_2 \boldsymbol{h}_j^S + \boldsymbol{b}_2, \\ \boldsymbol{v}_j &= \boldsymbol{W}_3 \boldsymbol{h}_j^S + \boldsymbol{b}_3. \end{aligned} \tag{1}$$

If Q, S are the same point cloud, Eq. (1) generates the feature maps for self-attention operation, otherwise cross-attention. It leverages an MLP and maps the position $\boldsymbol{p}_i \in \mathbb{R}^3$ into the rotary embedding $\boldsymbol{\Theta}_i = [\theta_1, \theta_2, \cdots, \theta_{d/2}] \in \mathbb{R}^{\frac{d}{2}}$, and then transform it to a rotation matrix as:

$$\boldsymbol{R}_{\Theta_i} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & \cdots & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \cos\theta_{\frac{d}{2}} & -\sin\theta_{\frac{d}{2}} \\ 0 & 0 & \cdots & \sin\theta_{\frac{d}{2}} & \cos\theta_{\frac{d}{2}} \end{bmatrix}. \tag{2}$$

We apply $\boldsymbol{R}_{\Theta_i}$ and $\boldsymbol{R}_{\Theta_j}$ to query $\boldsymbol{q}_i$ and key $\boldsymbol{k}_j$ respectively in self-attention and obtain the rotary self-attention as:

$$\alpha''_{ij} = \mathrm{softmax}_j((\boldsymbol{R}_{\Theta_i}\boldsymbol{q}_i)^\mathsf{T} \boldsymbol{R}_{\Theta_j}\boldsymbol{k}_j), \tag{3}$$

$$\tilde{\boldsymbol{h}}_i = \sum_{j=1}^{|\mathcal{S}|} \alpha''_{ij}\boldsymbol{v}_j. \tag{4}$$

Consequently, we can further derive Eq. (3) as:

$$\begin{aligned} \alpha''_{ij} &= \mathrm{softmax}_j(\boldsymbol{q}_i^\mathsf{T} \boldsymbol{R}_{\Theta_i}^\mathsf{T} \boldsymbol{R}_{\Theta_j}\boldsymbol{k}_j), \\ &= \mathrm{softmax}_j(\boldsymbol{q}_i^\mathsf{T} \boldsymbol{R}_{\Theta_j - \Theta_i}\boldsymbol{k}_j), \end{aligned} \tag{5}$$

where the relative information $\boldsymbol{\Theta}_j - \boldsymbol{\Theta}_i$ is naturally incorporated into the calculation of output $\tilde{\boldsymbol{h}}_i$. We denote the enhanced feature by 3D-RoFormer as $\tilde{\mathbf{H}} \in \mathbb{R}^{|\hat{\mathcal{P}}| \times \tilde{d}}$.

After 3D-RoFormer, we extract enhanced features that fuses contextual information and geometric structure. Subsequently, we deploy a suite of Multi-Layer Perceptrons (MLPs) to estimate the geometric offset. This process maps the uniformly sampled nodes to the proposal keypoints, i.e., $[\Delta\mathbf{P}, \Delta\mathbf{F}] = \mathrm{Vote}(\tilde{\mathbf{F}})$, $\mathcal{S} = \hat{\mathcal{P}} + \Delta\mathbf{P}$, and $\mathbf{H} = \tilde{\mathbf{F}} + \Delta\mathbf{F}$. By adding offsets, multiple proposals emerge clustered within salient areas of the point cloud. Finally, we randomly select a single proposal from each cluster as the final keypoints.

C. Dense Point Matching

Keypoint matching. Given that the keypoints have just been shifted and filtered by random selecting, we input these keypoints to another 3D-RoFormer to update the features as $\hat{\mathbf{H}}$. A Gaussian correlation matrix $\mathbf{C} \in \mathbb{R}^{|\hat{\mathcal{S}}^A| \times |\hat{\mathcal{S}}^B|}$ is then calculated to model the similarity between normalized $\hat{\mathbf{H}}^A$ and $\hat{\mathbf{H}}^B$ with $c_{i,j} = \exp(-\|\hat{\mathbf{h}}_i^A - \hat{\mathbf{h}}_j^B\|^2)$.

We follow Qin et al. [25] to perform a dual-normalization to suppress ambiguous matches, given by:

$$\hat{c}_{i,j} = \frac{c_{i,j}}{\sum_{k=1}^{|\hat{\mathcal{S}}^A|} c_{i,k}} \cdot \frac{c_{i,j}}{\sum_{k=1}^{|\hat{\mathcal{S}}^B|} c_{k,j}}. \tag{6}$$

The largest N_c entries are chosen as the keypoint correspondences.

Dense point matching. A matched keypoint pair indicates similarity in their respective neighborhoods, suggesting the potential to find additional point matches within these regions. Therefore, we first determine the region belonging to each keypoint. For each keypoint $\hat{\boldsymbol{s}}_i$, we use a point-to-node strategy [26] to assign each point to its nearest keypoint by:

$$\mathcal{G}_i = \{\boldsymbol{p} \in \mathcal{P} | i = \underset{j}{\mathrm{argmin}}(\|\boldsymbol{p} - \hat{\boldsymbol{s}}_j\|_2), \hat{\boldsymbol{s}}_j \in \hat{\mathcal{S}}\}, \tag{7}$$

where $\mathcal{G}_i$ is a patch of neighbor points of $\hat{\boldsymbol{s}}_i$.

We leverage the KPDecoder [23] to recover point-level descriptors $\mathbf{F}$. For each keypoint correspondence $(\hat{\boldsymbol{s}}_{x_i}^A, \hat{\boldsymbol{s}}_{y_i}^B)$, we have its corresponding point patch match $(\mathcal{G}_{x_i}^A, \mathcal{G}_{y_i}^B)$. Subsequently, we then compute a match score matrix $\mathbf{O}_i =$

$\mathbf{F}^{\mathcal{A}}_{x_i}(\mathbf{F}^{\mathcal{B}}_{y_i})^{\mathsf{T}}/\sqrt{\bar{d}}$ using point-level descriptors. A "dustbin" row and column filled with a learnable parameter $\alpha \in \mathbb{R}$ are further appended for $\mathbf{O}_i$ to handle non-matching points. The Sinkhorn algorithm is then used to solve the soft assignment matrix $\mathbf{Z}^i \in \mathbb{R}^{(M_i+1)\times(N_i+1)}$. Given the inherent disparity in data volume between LiDAR and millimeter-wave radar point clouds, we adopt a row-wise maximization strategy to allocate each radar point to a corresponding LiDAR point or to a "dustbin as our final assignment $\mathcal{C}^i$. A point is assigned to a "dustbin", it is classified as a non-match.

D. Loss Function

The loss function is composed of three parts: the keypoint detection loss L_{s}, the keypoint match loss L_{c}, the dense point matching loss L_{f}. The final loss is the sum of these three components: $L = L_{\mathrm{s}} + L_{\mathrm{c}} + L_{\mathrm{f}}$.

Keypoint detection loss. An ideal keypoint should exhibit stability across both the radar and LiDAR point clouds,we, therefore, employ a Chamfer loss to minimize the Euclidean distance between the matched keypoints, calculated as:

$$L_{\mathrm{s}} = \sum_{i=1}^{|\mathcal{S}^{\mathcal{A}}|} \min_{\boldsymbol{s}^{\mathcal{B}}_j \in \mathcal{S}^{\mathcal{B}}} \|\boldsymbol{s}^{\mathcal{A}}_i - \boldsymbol{s}^{\mathcal{B}}_j\|_2^2 + \sum_{i=1}^{|\mathcal{S}^{\mathcal{B}}|} \min_{\boldsymbol{s}^{\mathcal{A}}_j \in \mathcal{S}^{\mathcal{A}}} \|\boldsymbol{s}^{\mathcal{B}}_i - \boldsymbol{s}^{\mathcal{A}}_j\|_2^2. \quad (8)$$

Keypoint matching loss. We follow [25] and use overlap-aware circle loss to guide network to match keypoints with relatively high overlap, as given by:

$$L^{\mathcal{A}}_{\mathrm{c}} = \frac{1}{|\mathcal{A}|} \sum_{\mathcal{G}^{\mathcal{A}}_i \in \mathcal{A}} \log[1 + \sum_{\mathcal{G}^{\mathcal{B}}_j \in \varepsilon^+_i} e^{\lambda^j_i \beta^+_{i,j}(d^j_i - \Delta^+)} \cdot \sum_{\mathcal{G}^{\mathcal{B}}_k \in \varepsilon^-_i} e^{\beta^-_{i,k}(\Delta^- - d^k_i)}], \quad (9)$$

where ε^+_i is the positive patch sharing at least 10% overlap with $\mathcal{G}^{\mathcal{A}}_i$, and ε^-_i is the negative patch that do not overlap with $\mathcal{G}^{\mathcal{A}}_i$, $d^j_i = \|\hat{\mathbf{h}}^{\mathcal{A}}_i - \hat{\mathbf{h}}^{\mathcal{B}}_j\|_2$, λ^j_i refers to the overlap ratio between $\mathcal{G}^{\mathcal{A}}_i$ and $\mathcal{G}^{\mathcal{B}}_j$, and $\beta^+_{i,j} = \gamma(d^j_i - \Delta^+)$ and $\beta^-_{i,k} = \gamma(d^k_i - \Delta^-)$ represent the positive and negative weights. The hyper-parameters setting is followed by convention: $\Delta^+ = 0.1$ and $\Delta^- = 1.4$. The overall coarse match loss is the average of overlap-aware circle loss on $\mathcal{A}$ and $\mathcal{B}$, i.e., $L_{\mathrm{c}} = (L^{\mathcal{A}}_{\mathrm{c}} + L^{\mathcal{B}}_{\mathrm{c}})/2$

Dense point matching loss. We use a gap loss [20] on the soft assignment matrix $\mathbf{Z}^i$, calculated as:

$$\begin{aligned} L^i_{\mathrm{f}} = &\frac{1}{M_i} \sum_{m=1}^{M_i} \log(\sum_{n=1}^{N_i+1} [(-r^i_m + \mathbf{Z}^i_{m,n} + \eta)_+ + 1]) \\ &+ \frac{1}{N_i} \sum_{n=1}^{N_i} \log(\sum_{m=1}^{M_i+1} [(-c^i_n + \mathbf{Z}^i_{m,n} + \eta)_+ + 1]), \end{aligned} \quad (10)$$

where $(\cdot)_+ = max(\cdot, 0)$, $r^i_m = \sum_{n=1}^{N_i+1} \mathbf{Z}^i_{m,n}\mathbf{M}^i_{m,n}$ refers to the soft assignment value for the true match of m-th point in $\mathcal{G}^{\mathcal{A}}_{x_i}$, and $c^i_n = \sum_{m=1}^{M_i+1} \mathbf{Z}^i_{m,n}\mathbf{M}^i_{m,n}$ refers to the soft assignment value for the true match of n-th point in $\mathcal{G}^{\mathcal{B}}_{x_i}$, and $\mathbf{M}^i \in \{0,1\}^{(M_i+1)\times(N_i+1)}$ refers to the ground truth correspondences with a match threshold τ. The final dense point matching loss is the average over all the matched patch pairs: $L_{\mathrm{f}} = \frac{1}{2|\mathcal{M}|} \sum_{i=1}^{|\mathcal{M}|} L^i_{\mathrm{f}}$.

IV. Experimental Evaluation

A. Experimental Setup

Ground truth calibration precision and sensor data quality are crucial for data-driven methods. However, there are few open-access datasets that encompass high-quality 4D millimeter wave radar point cloud data. Among the available resources, the VoD dataset [27] and the MSC RAD4R dataset [28] are two datasets that meet our requirements.

The VoD dataset has multi-sensor data from an HDL-64 LiDAR and FRGen21 radar in urban areas. The MSC RAD4R dataset includes data from urban and rural environments, collected by a 128-channel LiDAR and Eagle radar, making it a good choice for evaluating online calibration methods. In experiments, we split the VoD dataset into 6000 frames for training, 100 for validation, and 2582 for testing. For the MSC dataset, we select 16,982 frames from six urban and four rural sequences for training and 2941 frames from two urban and three rural sequences for testing.

We adopt data augmentation as in [11], [29], applying random translations within 5 meters in the x, y, and z directions, as well as rotations within 10 ° in roll, pitch, and yaw to all LiDAR point cloud data. This range is set based on estimating and analyzing installation and operational sensor errors. Data augmentation within this range helps the model adapt to real world sensor distributions. This approach introduces varying degrees of random extrinsic parameters to both the training and testing sets, facilitating comprehensive training and evaluation. Our model is trained on a single NVIDIA RTX 3060 GPU with a batch size of 1. The total training time is about 30 hours. The runtime for inference is 172 ms per scan, enabling online keyframe calibration.

B. Metrics

We use these metrics for calibration performance evaluation: i) Relative Translational Error (RTE): Measuring Euclidean distance between the estimated and true translation vectors, and errors in each of the x, y, and z axes, denoted as X, Y, and Z. ii) Relative Rotational Error (RRE): Calculating geodesic distance between the estimated and ground truth rotation matrices, and angular discrepancies in Euler angles, denoted as Rx, Ry, and Rz. iii) Registration Recall (RR): Calculating the fraction of estimation whose RRE and RTE are below certain thresholds, e.g., 5 °and 2 m.

C. Cross-Modal Calibration Performance

To the best of our knowledge, no online calibration method specifically designed for radar-LiDAR systems has been reported. Existing motion-based calibration [7], [8], [9] requires odometry information from both sensors. However, existing radar odometry methods fail to produce satisfactory results in our experiments. Consequently, motion-based techniques have not been included in our comparisons. We compare our approach with baselines from the most closely related task in the domain: point cloud registration. Specifically, we benchmark our results against three representative methods: i) the recent SOTA registration method, LCR-Net [22], and ii) the most widely employed traditional

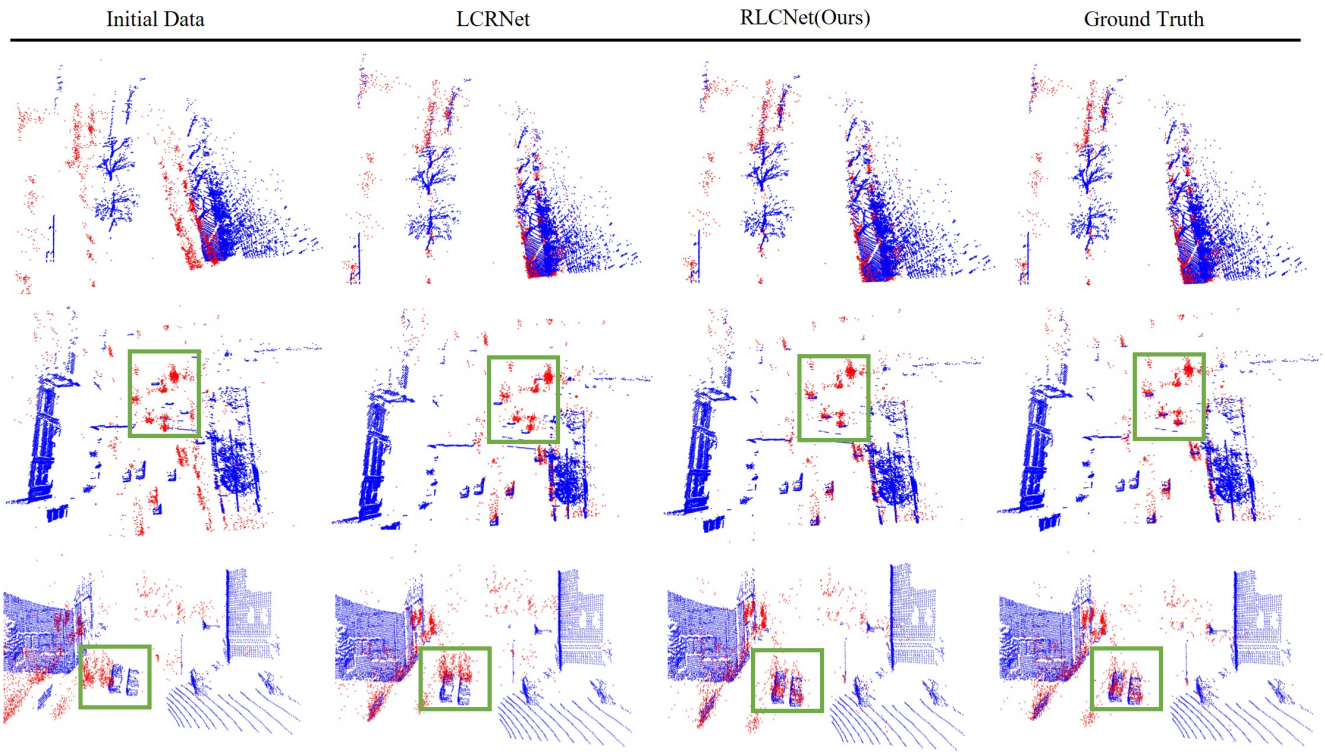


Fig. 3: Qualitative comparison of single-shot calibration using our method and LCR-Net. While LCR-Net exhibiting recognizable error, our method precisely aligning the two point clouds.

TABLE I: Calibration Performance on the MSC Dataset. The best results are highlighted in bold.

	Seq. 1				Seq. 3				Seq. 10				Seq. 12			
	ICP	TEASER	LCR-Net	RLCNet	ICP	TEASER	LCR-Net	RLCNet	ICP	TEASER	LCR-Net	RLCNet	ICP	TEASER	LCR-Net	RLCNet
RR(%)↑	4.18	0.09	61.54	**87.56**	3.67	0.380	62.59	**89.47**	8.81	1.02	83.39	**96.27**	5.26	0.380	65.04	**99.25**
RTE(m)↓	0.940	0.661	0.347	**0.298**	1.119	1.496	0.364	**0.307**	0.603	1.508	0.210	**0.179**	0.915	1.100	0.283	**0.252**
RRE(°)↓	1.705	2.954	1.818	**1.492**	1.570	3.020	1.677	**1.253**	1.524	2.227	1.089	**0.895**	2.696	3.986	1.592	**1.008**
X(m)↓	0.188	0.345	0.173	**0.147**	0.283	0.545	0.168	**0.149**	**0.088**	0.828	0.093	0.095	0.364	0.286	0.147	**0.138**
Y(m)↓	0.251	0.277	0.159	**0.149**	0.303	0.768	0.165	**0.129**	0.128	0.554	0.077	**0.066**	0.288	0.152	0.114	**0.079**
Z(m)↓	0.829	0.491	0.180	**0.147**	0.939	0.894	0.204	**0.178**	0.561	0.746	0.135	**0.105**	0.562	1.051	0.152	**0.149**
Rx(°)↓	0.565	0.826	0.558	**0.502**	0.547	2.033	0.631	**0.480**	0.596	1.902	0.452	**0.375**	1.320	0.539	0.650	**0.431**
Ry(°)↓	0.868	**0.673**	0.937	0.752	1.082	0.943	0.857	**0.725**	0.770	0.780	0.578	**0.490**	1.028	3.916	0.930	**0.681**
Rz(°)↓	**0.980**	2.771	1.159	0.994	**0.615**	1.586	1.048	0.705	0.871	0.617	0.584	**0.474**	1.517	0.549	0.850	**0.533**

TABLE II: Calibration Performance on the VoD Dataset. The best results are highlighted in bold.

	RR(%)↑	RTE(m)↓	RRE(°)↓	X(m)↓	Y(m)↓	Z(m)↓	Rx(°)↓	Ry(°)↓	Rz(°)↓
ICP	1.92	1.047	2.606	0.704	0.509	0.289	1.376	1.213	1.241
TEASER	1.74	0.934	3.407	0.531	0.448	0.354	1.980	1.743	1.067
LCR-Net	37.95	0.417	1.899	0.174	0.175	0.257	1.242	0.912	0.699
RLCNet (ours)	**72.36**	**0.295**	**1.266**	**0.140**	**0.120**	**0.171**	**0.867**	**0.541**	**0.469**

method, ICP [18], and iii) the recent SOTA traditional method, TEASER [21]. We retrain LCR-Net on the same dataset for a fair comparison. For TEASER, to balance performance and runtime, we employ the top 10 nearest neighbors as an initial estimation of correspondences.

The experimental results on the MSC-RAD4D dataset [28] are shown in Tab. I, our approach achieved an RR exceeding 85% across all sequences. The total RR of the dataset reached 90.45%, with average translational errors of 0.279 m (x: 0.156 m, y: 0.120 m, z: 0.138 m) and average angular errors of 1.260° (Roll: 0.462°, Pitch: 0.690°, Yaw: 0.765°). The experimental results on the VoD dataset are presented in Tab. II, our approach continues to demonstrate a significant advantage over the benchmark methods. Tradition methods, i.e., ICP and TEASER, struggle to produce satisfactory results due to the extremely sparse radar data and presence of noise points. The underwhelming performance of TEASER attributes to the absence of good initial correspondences.

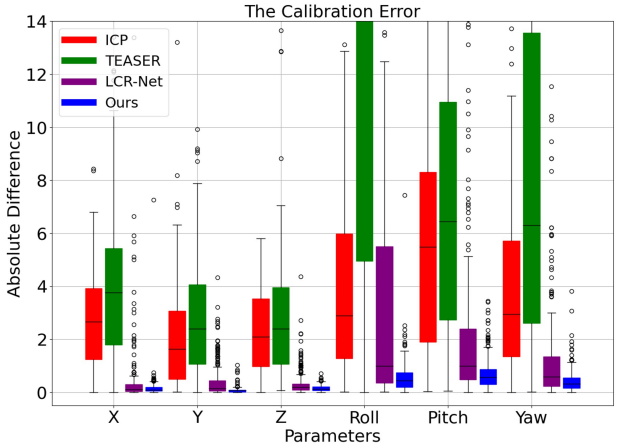


Fig. 4: Comparison of calibration error distribution on Seq. 12 of the MSC-RAD4R dataset.

TABLE III: Ablation study.

	RR(%)↑	RTE(m)↓	RRE(°)↓
RLCNet1encoder41(ours)	**90.45**	**0.279**	1.260
RLCNet1encoder40	61.82	0.306	1.446
RLCNet2encoder41	83.31	0.296	**1.250**

We evaluate the distribution of calibration errors across all frames. As shown in Fig. 4, our method is highly tolerant of translation errors, achieving excellent results. We visualize the calibration results as in Fig. 3. The initial parameters of the two point clouds exhibit substantial discrepancies in the translation and rotation matrices, and the radar point cloud is notably sparser. However, our method accurately predicts the extrinsic calibration parameters despite these challenges.

D. Ablation Studies

We perform ablation studies in Tab. III to show the effectiveness of the design. Initially, we study the keypoint extraction. As shown, using a single shared encoder, denoted as RLCNet1encoder41, performs better than using two distinct encoders for two point clouds, denoted as RLCNet2encoder41. This is because mmWave radar and LiDAR point clouds are fundamentally geometric representations of the environment; So, a unified encoder can be more effective to capture common geometric and semantic features in both types of point clouds.

Subsequently, we investigate the dense point matching. Benefiting from the flexibility of our network architecture, we can match at the top layer, denoted as RLCNet1encoder40 or the second layer, denoted as RLCNet1encoder41, i.e., at the raw point cloud or at the downsampled point cloud. The results show that downsampled matching performs better. This may be due to the fact that, at the top layer, the excessive number of LiDAR point clouds introduces redundant information, which complicates the matching process.

E. Potential Downstream Applications

As shown in Fig. 5, we visualize the projected radar point cloud onto the LiDAR map with our estimated extrinsic parameters. The precise alignment between the two point clouds achieved by our calibration demonstrates its potential for downstream cross-modal localization. Additionally, for a test sequence, we map radar point cloud. We accumulate radar point cloud using the LiDAR pose and estimated extrinsic parameters, as illustrated in Fig. 6. The radar map stitched using the extrinsic parameters estimated by our network and corresponding LiDAR odometry poses is the closest to the ground-truth-based map. This highlights the consistent accuracy of our online calibration and its potential for multi-sensor fused odometry.

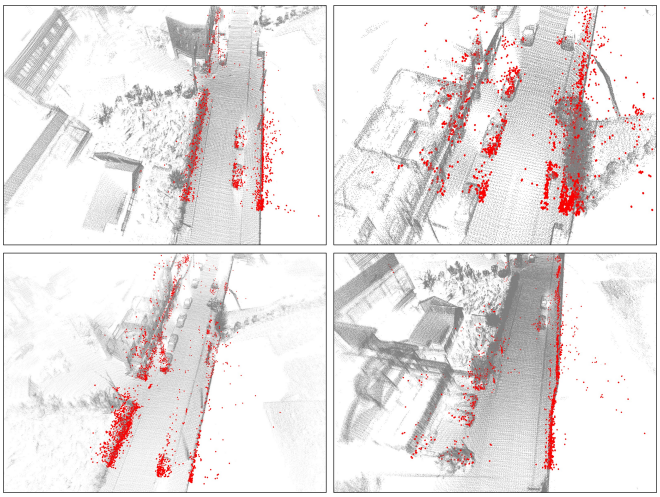
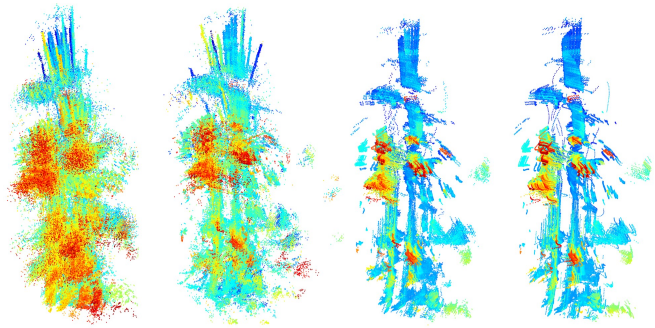


Fig. 5: Radar points (red) projected on LiDAR point map (gray) using our estimated extrinsic parameters.



(a) ICP (b) LCR-Net (c) Ours (d) Groundtruth

Fig. 6: Accumulated radar point map using LiDAR pose and our estimated extrinsic parameters.

V. Conclusion

In this work, we introduce RLCNet, a novel approach for feature-matching-based online extrinsic calibration of radar-LiDAR systems. Our approach first extracts stable keypoints for the purpose of matching priors. Given the sparsity inherent in mmWave radar data, utilizing sparser keypoint matches poses a significant constraint on the reliability and precision of calibration. To overcome this limitation, we leverage neighborhood consistency to seek denser point matches from the vicinity of keypoint matches, thereby facilitating a more robust and accurate calibration. Experimental results demonstrate that our method outperforms the baselines in terms of calibration recall and precision across two datasets.

REFERENCES

[1] L. Fan, J. Wang, Y. Chang, Y. Li, Y. Wang, and D. Cao, "4d mmwave radar for autonomous driving perception: a comprehensive survey," *IEEE Transactions on Intelligent Vehicles (TIV)*, 2024.

[2] K. Luan, C. Shi, N. Wang, Y. Cheng, H. Lu, and X. Chen, "Diffusion-based point cloud super-resolution for mmwave radar data," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[3] T. Wang, N. Zheng, J. Xin, and Z. Ma, "Integrating millimeter wave radar with a monocular vision sensor for on-road obstacle detection applications," *Sensors*, vol. 11, no. 9, pp. 8992–9008, 2011.

[4] J. Peršić, I. Marković, and I. Petrović, "Extrinsic 6dof calibration of 3d lidar and radar," in *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, pp. 1–6, IEEE, 2017.

[5] J. Domhof, J. F. Kooij, and D. M. Gavrila, "An extrinsic calibration tool for radar, camera and lidar," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pp. 8107–8113, IEEE, 2019.

[6] L. Heng, "Automatic targetless extrinsic calibration of multiple 3d lidars and radars," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 10669–10675, IEEE, 2020.

[7] J. Maye, H. Sommer, G. Agamennoni, R. Siegwart, and P. Furgale, "Online self-calibration for robotic systems," *Intl. Journal of Robotics Research (IJRR)*, vol. 35, pp. 357–380, 2016.

[8] N. Keivan and G. Sibley, "Online slam with any-time self-calibration and automatic change detection," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pp. 5775–5782, IEEE, 2015.

[9] T. Välimäki, B. Garigipati, and R. Ghabcheloo, "Motion-based extrinsic sensor-to-sensor calibration: Effect of reference frame selection for new and existing methods," *Sensors*, vol. 23, no. 7, p. 3740, 2023.

[10] P. Jiang, P. Osteen, and S. Saripalli, "Semcal: Semantic lidar-camera calibration using neural mutual information estimator," in *2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 1–7, IEEE, 2021.

[11] X. Lv, B. Wang, Z. Dou, D. Ye, and S. Wang, "Lccnet: Lidar and camera self-calibration using cost volume network," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 2894–2901, 2021.

[12] P. Jiang and S. Saripalli, "Contrastive learning of features between images and lidar," in *Proc. of the International Conf. on Automation Science and Engineering (CASE)*, pp. 411–417, IEEE, 2022.

[13] J. Peršić, I. Marković, and I. Petrović, "Extrinsic 6dof calibration of a radar–lidar–camera system enhanced by radar cross section estimates evaluation," *Journal on Robotics and Autonomous Systems (RAS)*, vol. 114, pp. 217–230, 2019.

[14] G. El Natour, O. A. Aider, R. Rouveure, F. Berry, and P. Faure, "Radar and vision sensors calibration for outdoor 3d reconstruction," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pp. 2084–2089, IEEE, 2015.

[15] E. Wise, Q. Cheng, and J. Kelly, "Spatiotemporal calibration of 3-d millimetre-wavelength radar-camera pairs," *IEEE Trans. on Robotics (TRO)*, 2023.

[16] J. Peršić, L. Petrović, I. Marković, and I. Petrović, "Online multi-sensor calibration based on moving object tracking," *Advanced Robotics*, vol. 35, no. 3-4, pp. 130–140, 2021.

[17] C. Schöller, M. Schnettler, A. Krämmer, G. Hinz, M. Bakovic, M. Güzet, and A. Knoll, "Targetless rotational auto-calibration of radar and camera for intelligent transportation systems," in *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, pp. 3934–3941, IEEE, 2019.

[18] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611, pp. 586–606, Spie, 1992.

[19] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[20] C. Shi, X. Chen, K. Huang, J. Xiao, H. Lu, and C. Stachniss, "Keypoint Matching for Point Cloud Registration using Multiplex Dynamic Graph Attention Networks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, pp. 8221–8228, 2021.

[21] H. Yang, J. Shi, and L. Carlone, "TEASER: Fast and Certifiable Point Cloud Registration," *IEEE Trans. on Robotics (TRO)*, vol. 37, no. 2, pp. 314–333, 2020.

[22] C. Shi, X. Chen, J. Xiao, B. Dai, and H. Lu, "Fast and accurate deep loop closing and relocalization for reliable lidar slam," *IEEE Trans. on Robotics (TRO)*, 2024.

[23] H. Thomas, C. Qi, J. Deschaud, B. Marcotegui, F. Goulette, and L. Guibas, "KPConv: Flexible and Deformable Convolution for Point Clouds," in *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[24] C. Shi, X. Chen, H. Lu, W. Deng, J. Xiao, and B. Dai, "Rdmnet: Reliable dense matching based point cloud registration for autonomous driving," *IEEE Trans. on Intelligent Transportation Systems (T-ITS)*, 2023.

[25] Z. Qin, H. Yu, C. Wang, Y. Guo, Y. Peng, and K. Xu, "Geometric transformer for fast and robust point cloud registration," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 11143–11152, 2022.

[26] R. Li, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "Pu-gan: a point cloud upsampling adversarial network," in *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[27] A. Palffy, E. Pool, S. Baratam, J. F. Kooij, and D. M. Gavrila, "Multi-class road user detection with 3+ 1d radar in the view-of-delft dataset," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, pp. 4961–4968, 2022.

[28] M. Choi, S. Yang, S. Han, Y. Lee, M. Lee, K. H. Choi, and K.-S. Kim, "Msc-rad4r: Ros-based automotive dataset with 4d radar," *IEEE Robotics and Automation Letters (RA-L)*, 2023.

[29] H. Zhao, Y. Zhang, Q. Chen, and R. Fan, "Dive deeper into rectifying homography for stereo camera online self-calibration," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, pp. 14479–14485, IEEE, 2024.