Radar Tracker: Moving Instance Tracking in Sparse and Noisy Radar Point Clouds

Matthias Zeller Daniel Casado Herraez Jens Behley Michael Heidingsfeld Cyrill Stachniss

***Abstract*— Robots and autonomous vehicles should be aware of what happens in their surroundings. The segmentation and tracking of moving objects are essential for reliable path planning, including collision avoidance. We investigate this estimation task for vehicles using radar sensing. We address moving instance tracking in sparse radar point clouds to enhance scene interpretation. We propose a learning-based radar tracker incorporating temporal offset predictions to enable direct center-based association and enhance segmentation performance by including additional motion cues. We implement attention-based tracking for sparse radar scans to include appearance features and enhance performance. The final association combines geometric and appearance features to overcome the limitations of center-based tracking to associate instances reliably. Our approach shows an improved performance on the moving instance tracking benchmark of the RadarScenes dataset compared to the current state of the art.**

I. Introduction

Motion planning and reliable collision avoidance of autonomous vehicles in real-world environments depend on the precise tracking of moving instances. The information on how many agents are present and the prediction of movement is crucial for path planning and pose estimation. Cameras, LiDARs, and radars provide valuable information about the surroundings, and a versatile setup of autonomous vehicles often aims to reduce critical malfunctions. Radar sensor measurements are often noisy due to multi-path propagation, sensor noise, and ego motion. However, radar sensors work under adverse weather, overcoming the limitations of cameras and LiDARs, and thus are essential for reliable perception systems. Additionally, radar sensors directly measure the Doppler velocity of the so-called radar detections and determine the radar cross section, which depends on the material, the geometry, and the surface of the object, which could help to identify and track moving agents precisely.

In this paper, we elaborate on moving instance segmentation and tracking in noisy and sparse radar point clouds, as illustrated in Fig. 1. This requires differentiating between moving and static parts of the surroundings and consistently distinguishing instances of individual agents in the environment over time. The 4D moving object segmentation task falls within 4D panoptic segmentation [2]. However, all moving objects belong to the moving object class without further differentiation into a more detailed separation.

Matthias Zeller and Daniel Casado Herraez are with CARIAD SE and with the Center for Robotics, University of Bonn, Germany. Jens Behley is with the Center for Robotics, University of Bonn, Germany. Michael Heidingsfeld is with CARIAD SE, Germany. Cyrill Stachniss is with the Center for Robotics, University of Bonnand with the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

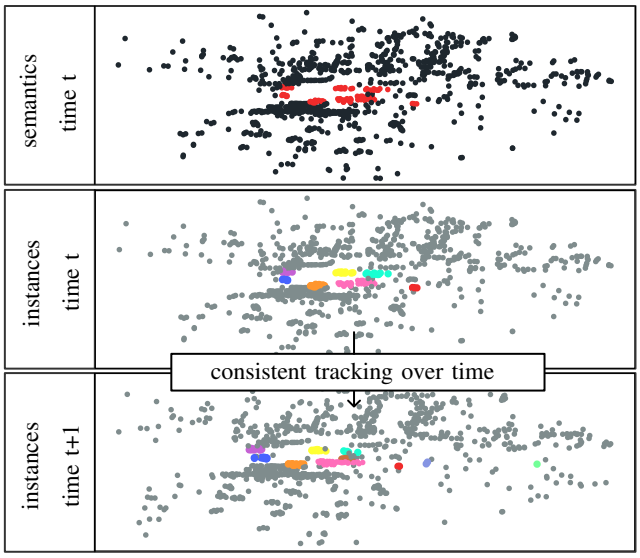

Fig. 1: Our method combines moving object segmentation (top), instance segmentation (middle), and tracking (bottom) to solve the 4D panoptic task of moving instance tracking from sparse radar point clouds. The corresponding colors in the middle and bottom images represent the respective tracked instances (static is grey).

Current state-of-the-art methods [5], [26] often address moving instance tracking within aggregated scans and associate instances and existing tracks based on the intersection over union (IoU) score. However, the aggregation of scans induces latency and is disadvantageous for tasks requiring immediate feedback, such as collision avoidance. Additionally, instances within sparse radar point clouds often comprise single points for which an IoU-based association is inappropriate. Other methods rely on dedicated trackers based on Kalman filters [17], which often neglect valuable appearance features [6], [46]. To extract appearance features, other approaches [5], [26] voxelize the point clouds, which is particularly harmful to sparse radar data processing [50].

The main contribution of this paper is a novel point-based approach that enables moving instance tracking by incorporating geometric and appearance features to accurately associate moving instances in sparse and noisy radar point clouds over time. Our approach, called Radar Tracker, utilizes neural networks and extends the prediction of a moving instance segmentation approach to derive temporal consistent tracking IDs. We efficiently incorporate temporal information for each point by a temporal offset prediction to enhance the segmentation and enable direct center-based tracking of moving instances. We propose an attention-based instance feature extraction network to reduce information loss and

keep the appearance features of the individual instances. Furthermore, we derive attention-based association scores to extend tracking by attention. The final geometric and appearance features are combined within our data association to improve the performance of the overall estimation task.

In sum, we make three claims: First, our approach shows state-of-the-art performance for moving instance tracking in sparse and noisy radar point clouds. Second, our temporal offset prediction incorporates valuable information for segmentation and enhances tracking performance. Third, our attention-based track association overcomes the shortcomings of center-based tracking by incorporating appearance feature information.

II. Related Work

Moving instance tracking tasks can be solved as 4D panoptic segmentation [2], [20], [59] combining moving instance segmentation and tracking. Additionally, the task benefits from multi-object tracking [46], [53], single object tracking [15], [48], and panoptic segmentation [12], [51].

Panoptic Segmentation unifies instance and semantic segmentation. There exists extensive literature, including projection-based [4], [14], [19], [35], [42], [55], voxel-based [7], [23], [27], [41], [44], [45], [60], point-based [9], [33], [34], [37], and transformer-based [25], [31], [36], [40], [47], [50], [54], [57] approaches to solving individual tasks. However, projection-based and voxel-based approaches inherently introduce discretization artifacts and information loss, which is harmful to the targeted processing of sparse radar point clouds.

The point-based approaches directly process point clouds, keeping the spatial information intact to overcome the lossy encoding. Schumann et al. [37] adopted this approach by aggregating multiple radar point clouds as input for PointNet++ [34] and improved the method by adding a temporal module and additional features [39]. The aggregation of scans induces latency which is disadvantageous for safety-critical tasks requiring immediate feedback.

Recently, transformer-based networks have overcome this limitation by exploiting the self-attention mechanism [43] in point cloud understanding [11], [31], [36], [40], [47], [54], [57], which is inherently suitable to capture strong local and global dependencies and thus enable single-scan radar processing [50], [52]. In our prior work [51], we efficiently include the temporal information within a single scan and proposed an attention-based class-agnostic instance assignment to reliably segment moving instances. However, the trajectory and tracking information about moving agents is missing, which is required for safe autonomous mobility.

4D Panoptic Segmentation unifies instance segmentation and tracking [18] and thus incorporates spatial and temporal information about the environment. Only very recently, LiDAR-based 4D panoptic segmentation [2] was formally introduced, and it shares similarities to our targeted task of moving instance segmentation. Thus, we shortly summarize related work targeting this task but emphasize that moving instance tracking is a special case of 4D panoptic segmentation. State-of-the-art approaches [1], [2], [13], [21], [59] aggregate multiple point clouds and perform instance segmentation within a fused scan. Agarwalla et al. [1] follow CenterPoint [49] and directly predict the velocities of the objects in concatenated point clouds and perform a greedy nearest-neighbor association of the instances. In contrast, Zhu et al. [59] extend 4D-Stop [21] and propose to learn the offsets as equivariant vector fields. Despite the tremendous progress, aggregated scan processing comes with a significant computational burden and induces latencies.

Tracking-by-detection algorithms are the most common approaches [5], [8], [30] to work on a sequential scan basis. These algorithms first obtain object detections in the current frame and associate them across time which can be formulated as bipartite graph matching. The data association is often based on a cost matrix which can be solved by the Hungarian method [22] or greedy-matching algorithms [49]. The cost matrix is a similarity matrix comparing the existing tracks with the newly identified objects based on appearance or geometric features. To include motion information, filtering algorithms [3], [6], [46] such as the Kalman filter [17] utilize real-world physical models to estimate the state transition of instances. Based on the predictions, AB3DMOT [46] calculates the 3D IoU as the cost for the association. However, IoU-based association is inappropriate for radar signal processing because instances comprise single points. Chiu et al. [6] enhance performance by utilizing the Mahalanobis distance, and CenterPoint [49] performs a center-based greedy matching by predicting object velocities. Marcuzzi et al. [26] combine appearance and motion cues to associate instances, including class-dependent contrastive learning. CXTrack [48] and MotionTrack [53] utilize attention-based similarity features to track single and multi-objects, respectively. However, MotionTrack struggles to associate objects based on attention due to the sparsity of the point clouds, which is more severe for noisy radar data. Additionally, the proposed voxel-based backbones [26], [53] induce discretization artifacts, harming accuracy.

In this paper, we follow recent advancements and propose a novel moving instance tracking method that combines geometric and appearance features for sparse and noisy radar data. Our Radar Tracker includes a temporal offset prediction module to capture important motion information and enhance center-based tracking. Furthermore, our proposed transformer-based network encodes the instance information to derive attention-based association scores incorporating valuable appearance features. Our combined data association overcomes the shortcomings of center-based association and enhances state-of-the-art performance for moving instance tracking in sparse radar point clouds.

III. Our Approach to Track Moving Instances

Our approach aims to achieve reliable moving instance tracking in sparse radar point clouds. We follow the tracking-by-detection paradigm and extend the Radar Instance Transformer [51] with dedicated tracking modules, as illustrated in Fig. 2. We directly predict the temporal offset for each

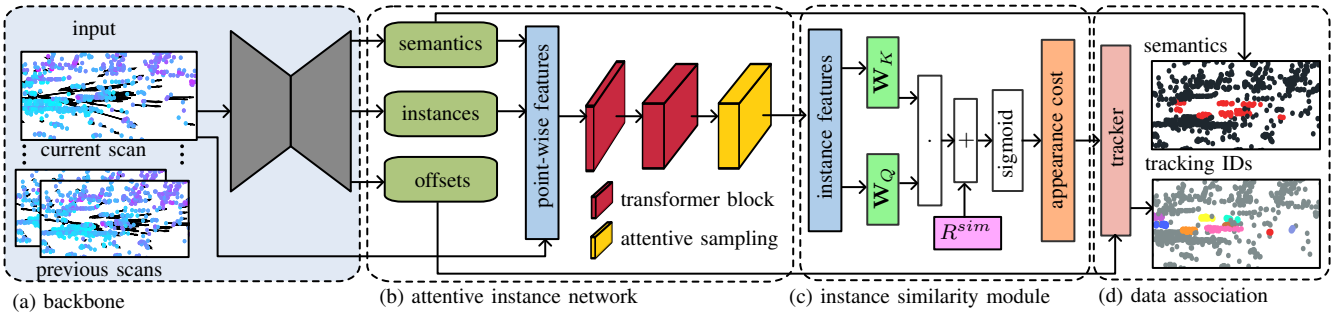

Fig. 2: The detailed design of the individual modules of our Radar Tracker. (a) The backbone is extended with the offset predictions and provides the semantic classes and instance predictions. (b) The attentive instance network extracts features to represent instances. (c) Our instance similarity module determines the appearance-based association matrix to enhance instance tracking. (d) The data association utilizes the appearance and geometric features to predict the tracking IDs $\mathcal{T}^{\text{track}}$ of moving instances in sparse radar point clouds.

detection within single radar scans to enhance segmentation and enable direct center-based association. We utilize the self-attention mechanism [43] to regress an additional cost function and include appearance features to improve tracking. The final association combines geometric and appearance features to enhance scene understanding.

A. Moving Instance Segmentation Backbone

The performance of the instance segmentation backbone limits the tracking of objects. Therefore, we utilize the state-of-the-art Radar Instance Transformer [51] as the backbone to extract moving instances reliably. The current radar scan $\mathcal{P}^t$ at time t, which comprises the point coordinates and the radar features such as the Doppler velocity and radar cross section, is efficiently enriched with temporal information from N^p previous scans $\mathcal{P}^{t-N^p}$ by the sequential attentive feature encoding module. The processed single scan, including temporally enriched point-wise features, is then passed through the network. The outputs of the backbone are the moving object segmentation (MOS) labels $\mathcal{S}^{\text{MOS}}$, the instance IDs $\mathcal{I} = \{I_1, \dots, I_N\}$ with $I_i \in \mathbb{N}$, and the point-wise features $\mathbf{X}^b$. We utilize the predictions as input to our Radar Tracker. Since we do not rely on additional information, we can potentially substitute the backbone for other moving instance segmentation networks.

B. Offset Prediction Module

Geometric features of instances are essential to track moving objects reliably. However, in sparse radar point clouds, the appearance of objects changes, making tracking based on bounding boxes difficult [46]. Especially the case of single-point instances is not covered adequately. Therefore, our Radar Tracker focuses on center-based associations to exploit important geometric properties.

Furthermore, future state prediction is crucial to associate tracks and objects over time. In contrast to other approaches [5], [49], which predict velocity vectors for bounding boxes or voxels, we directly process the point cloud on a per-point basis to include per-point motion cues. Our resulting approach first utilizes the commonly used offset prediction head [13] to regress offsets $\mathbf{O} \in \mathbb{R}^{N\times 2}$ to the instance center $C^t \in \mathbb{R}^{N\times 2}$ of the current scan. Secondly, we predict the temporal offset $\mathbf{O}^{\text{temp}} \in \mathbb{R}^{N\times 2}$ for each point, which is a vector that points to the center of the instance $C^{t+1} \in \mathbb{R}^{N\times 2}$ in the next scan. We calculate the center as the average of the coordinates of the points belonging to the instance.

The input to our offset prediction module is the concatenation of the features of the backbone $\mathbf{X}^b$, and the point coordinates $\mathbf{P}^t$ of the current scan to include fine-grained position information. For the individual offset prediction heads, we combine two fully connected layers, batch normalization [16] and a rectified linear unit (ReLU) [28]. The resulting offsets directly incorporate the information for center-based moving instance tracking and add motion cues about moving instances within each scan. Additionally, the temporal offset includes a regression target for single-point moving instances, which does not account for the standard offset prediction and enhances segmentation.

C. Attentive Instance Network

Center-based data association works remarkably well. However, the geometric association often neglects appearance features, which are essential if the geometric features are inaccurate or multiple agents interact, which makes a purely geometrically based approach challenging to solve. In contrast to other methods [5], [26], we propose to extract discriminative instance features by a transformer-based network to reduce information loss in sparse radar data.

Our attentive instance network comprises two transformer blocks and an attentive aggregation module. The input consists of the point coordinates $\mathbf{P}^{\text{in}} = [\mathbf{p}_1, \dots, \mathbf{p}_{N^{\text{mov}}}]^\top \in \mathbb{R}^{N^{\text{mov}}\times 2}$ and the features $\mathbf{X}^{\text{in}} = [\mathbf{x}_1, \dots, \mathbf{x}_{N^{\text{mov}}}]^\top \in \mathbb{R}^{N^{\text{mov}}\times D}$, where $\mathbf{p}_i \in \mathbb{R}^2$ and $\mathbf{x}_i \in \mathbb{R}^D$ for N^{mov} moving (mov) points. Hence, $\mathbf{X}^{\text{in}}$ only includes a subset of the features of the backbone $\mathbf{X}^b$, which includes moving and static points. During training, we select the instances based on the ground truth labels and for the inference based on the semantic and instance predictions of the backbone. We follow the backbone design proposed in Zeller et al. [51] to extract point-wise information. The transformer block is a residual block, including a feature dimension expansion that embeds a transformer layer. We first process the input features $\mathbf{X}^{\text{in}} \in \mathbb{R}^{N^{\text{mov}}\times D}$ by a linear layer with weight

matrix $\mathbf{W}_l \in \mathbb{R}^{D \times D_1}$ to increase the feature dimension. The resulting features $\mathbf{X}_l^{\text{in}}$ and corresponding point coordinates are fed into a transformer layer. The output features are processed by another linear layer and added to the skip connection features. For the transformer layer, we follow the design of the Point Transformer [57]. We encode the features of the moving instances $\mathbf{X}_l^{\text{in}}$ as queries $\mathbf{Q}$, keys $\mathbf{K}$, and values $\mathbf{V}$ as follows:

$$\mathbf{Q} = \mathbf{X}_l^{\text{in}}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{X}_l^{\text{in}}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{X}_l^{\text{in}}\mathbf{W}_V, \quad (1)$$

where $\mathbf{W}_Q$, $\mathbf{W}_K$, and $\mathbf{W}_V \in \mathbb{R}^{D_1 \times D_1}$ are learned linear projections. To reduce the computational burden, especially for large instances, we restrict the attention mechanism to local areas. We calculate the k-nearest neighbors with $k = N^l$ for the points in the current instance. We apply the sample and grouping algorithm [34] to extract the related queries, keys, and values, resulting in $\mathbf{Q}^{\text{sg}}, \mathbf{K}^{\text{sg}}$, and $\mathbf{V}^{\text{sg}} \in \mathbb{R}^{N^{\text{mov}} \times N^l \times D_1}$. Additionally, we calculate the relative positions $\mathbf{r}_{i,j} = \mathbf{p}_i - \mathbf{p}_j$ within the local areas of the instances where $\mathbf{p}_i$ and $\mathbf{p}_j \in \mathbf{P}^{\text{in}}$. We process the relative positions $\mathbf{r}_{i,j}$ by a multi-layer perceptron (MLP), including two linear layers with weight matrix $\mathbf{W}_1 \in \mathbb{R}^{2 \times 2}$ and $\mathbf{W}_2 \in \mathbb{R}^{2 \times D_1}$, batch normalization [16], and ReLU activation function [28] to derive the relative positional encoding [57] $\mathbf{R} \in \mathbb{R}^{N^{\text{mov}} \times N^l \times D_1}$.

We adopt vector attention [56] and subtract the encoded keys from the encoded queries to calculate the attention weights $\mathbf{A} \in \mathbb{R}^{N^{\text{mov}} \times N^l \times D_1}$ for the individual points i. Additionally, we add the relative positional encoding $\mathbf{R}_i$ before we determine the individual weighting of the features by the softmax function as follows:

$$\mathbf{A}_i = \text{softmax}(\mathbf{Q}_i^{\text{sg}} - \mathbf{K}_i^{\text{sg}} + \mathbf{R}_i). \quad (2)$$

To calculate the output features $\mathbf{X}_1^{\text{out}} \in \mathbb{R}^{N^{\text{mov}} \times D_1}$ of the transformer layer, we calculate the sum of the element-wise multiplication, indicated by $\odot$, and add the relative positional encoding as follows:

$$\mathbf{X}_{1,i}^{\text{out}} = \sum_{j=1}^{N^l} \mathbf{A}_{i,j} \odot (\mathbf{V}_{i,j}^{\text{sg}} + \mathbf{R}_{i,j}). \quad (3)$$

The attentive aggregation module is inspired by the attention mechanism and follows the attentive sampling operation [50]. We utilize the attentive aggregation module to keep and combine the information of instances within sparse radar point clouds. We process the output features $\mathbf{X}_2^{\text{out}} \in \mathbb{R}^{N^{\text{mov}} \times D_2}$ of the second transformer blocks by a linear layer with weight matrix $\mathbf{W}^{\text{agg}} \in \mathbb{R}^{D_2 \times D_2}$ and softmax activation function. The resulting outputs are our aggregation weights $\mathbf{A}^{\text{agg}}$, which we utilize to weight the N^I points within the individual instance. The final instance feature is derived by the summation of the weighted point features, resulting in:

$$\mathbf{X}_i^{\text{inst}} = \sum_{j=1}^{N^I} \mathbf{A}_{i,j}^{\text{agg}} \odot \mathbf{X}_{i,j}^{\text{out}}. \quad (4)$$

The instance-wise feature vectors comprise the information for the data association. Additionally, we extract the coordinates $\mathbf{P}^{\text{inst}}$ of each instance to include position information in the association step.

D. Instance Similarity Module

The essential part of improving the data association of tracks and newly detected objects is the cost function or similarity measure for the tracking. We utilize the features and center coordinates of our attentive instance network to determine the similarities and incorporate essential appearance features.

We encode the features $\mathbf{X}^{\text{inst}}$ as queries $\mathbf{Q}^{\text{sim}}$ and keys $\mathbf{K}^{\text{sim}}$ following Eq. (1) and perform dot product attention to derive an attention-based similarity value. Within the attention matrix, we include the self- and cross-attention of the instances. Additionally, we calculate the relative center positions $\mathbf{r}_{\text{center}}$ where $\mathbf{p}_i$ and $\mathbf{p}_j \in \mathbf{P}_{\text{center}}$ and encode the relative positions as $\mathbf{R}^{\text{sim}}$ by an MLP. We reduce the dimensionality of the encoding to one.

To calculate the resulting attention-based instance similarity scores, we replace the softmax function with an element-wise sigmoid function and add the positional center encoding as follows:

$$\mathbf{A}^{\text{sim}} = \text{sigmoid}\left(\mathbf{Q}^{\text{sim}}\mathbf{K}^{\text{sim}\top} + \mathbf{R}^{\text{sim}}\right). \quad (5)$$

The similarity scores incorporate the feature and position information and directly indicate how likely two instances belong together. To utilize the scores as an additional cost function, we calculate the similarity cost as:

$$\mathbf{C}^{\text{sim}} = \frac{1}{(\mathbf{A}^{\text{sim}} + \epsilon)}, \quad (6)$$

where ϵ is an arbitrarily small constant for numerical stability. The similarity cost depends on the appearance features and thus incorporates essential information for moving instance tracking.

E. Data Association

The central part of our data association is the offset predictions $\mathbf{O}^{\text{temp}}$ and $\mathbf{O}$ for the coordinates of the instance $\mathbf{P}^{\text{inst}}$ because the center-based association within a small distance is reliable for tracking moving instances. However, to improve data association when the geometric association is imprecise, we utilize the attention-based instance similarity scores $\mathbf{A}^{\text{sim}}$ to enhance tracking performance.

We first calculate the Euclidean distance d based on the center predictions of our method. For the existing tracks, the center is defined by the temporal offset predictions $\mathbf{O}^{\text{temp}}$, whereas for the newly identified instances, the offset $\mathbf{O}$ is utilized. Since a global mapping includes multiple misleading connections, which would be considered in the optimization step, we directly restrict the optimization to local areas. Therefore, we cluster the instances based on the distances d into local areas with DBSCAN [10]. After the clustering, we utilize the local cost matrices, i.e. only instances in each cluster, and perform Hungarian matching [22].

Additionally, we process the input features $\mathbf{X}^{\text{in}}$ by our model to extract instance features $\mathbf{X}^{\text{inst}}$ and determine the similarity scores $\mathbf{A}^{\text{sim}}$ of the tracks and the objects. Since the geometric information is precious within the short-range displacement, we determine a threshold t_{d1} where the data association is purely based on the geometric assignment. Above that threshold, we include the similarity cost function $\mathbf{C}^{sim}$ to perform the association. Therefore, we determine a similarity cost threshold t_c, which resolves whether the object is assigned to the corresponding track. The similarity cost threshold also handles occlusion and initializes new tracks. Furthermore, we define a second distance-based threshold t_{d2} where the appearance-based association is difficult, and the association is omitted. We update the existing tracks with the information of the assigned objects. Occluded tracks are propagated according to the temporal offset predictions $\mathbf{O}^{\text{temp}}$, and we initialize new tracks with the corresponding information of the object. The data association incorporates geometric and appearance features to enhance tracking performance.

F. Implementation Details

We implemented our approach in PyTorch [32] and trained the instance segmentation backbone and our Radar Tracker with one Nvidia A100 GPU. We adopt the training parameter of the Radar Instance Transformer [51]. To learn the standard and temporal offsets of the radar detections, we add for both offsets the following loss function:

$$L^{\text{offset}} = \frac{1}{N}\sum_{i=1}^{N}\|\mathbf{o}_i - (\mathbf{c}_i - \mathbf{p}_i)\|_1, \tag{7}$$

where N is the number of points in the point cloud, and $\mathbf{c}_i$ is the respective center of the instance that $\mathbf{p}_i$ belongs to.

We utilize the AdamW [24] optimizer with an initial learning rate of 0.001 to train our Radar Tracker. We process the original features with dimension $D = 4$, comprising the point coordinates (x_i^C, y_i^C), the radar cross section σ_i, and the ego-motion compensated Doppler velocity v_i, by the transformer blocks where $D_1 = 64$ and $D_2 = 256$. We define the local areas for sample and grouping by $N_l = 6$ points. The attentive aggregation module keeps the feature dimension and combines the information within one feature vector. The batch of our method includes 64 scan pairs where only the first scan is considered during the loss calculation. Hence, the network is able to predict the data association and if the objects within the first scan are present in the second scan. We supervise the attentive similarity output by a binary cross entropy loss.

We set the bandwidth $b = 10$ for the clustering using DBSCAN to determine local areas for the association. We set the distance based thresholds $t_{d1} = 5\,\text{m}$ and $t_{d2} = 10\,\text{m}$. We keep the tracks for 12 consecutive scans. The cost threshold for the attentive similarity is set to $t_c = 1.5$. We add points belonging to the static class as additional instances for data augmentation to include the differentiation between static and moving points in the attentive similarity.

Approach	LSTQ	S_{assoc}	S_{cls}
MOT [46]	42.4	19.4	**92.7**
Center tracking [49] + Hungarian [22]	59.3	38.0	**92.7**
CA-Net [26]	34.8	13.0	**92.7**
Ours	**66.8**	**48.2**	**92.7**

TABLE I: Moving instance tracking results on the RadarScenes test set in terms of LSTQ, S_{cls}, and S_{assoc} scores.

IV. Experimental Evaluation

The main focus of this work is to enable reliable moving instance tracking in sparse and noisy radar point clouds. We present our experiments to show the capabilities of our method and to support our key claims, which include that our method outperforms existing state-of-the-art methods in moving instance tracking. Secondly, our temporal offset prediction enhances the classification and tracking score by adding additional motion cues. Thirdly, our attention-based association scores incorporate valuable appearance features enhancing performance.

A. Experimental Setup

We train and evaluate our model on the RadarScenes [38] dataset since it is the only large-scale high-resolution radar dataset [58] that includes per-point annotations for moving instance tracking under versatile scenarios. We follow Zeller et al. [52] and split the 158 sequences into 130 sequences for training, 6 for validation, and 22 for testing. We perform the ablation studies on the validation set. We merge the data of the four individual radar sensors to obtain information about the surroundings of the vehicle [52].

We utilize the LiDAR segmentation and tracking quality (LSTQ) score [2] to evaluate the moving instance tracking performance. The LSTQ is designed for evaluating point-based segmentation and tracking methods and does not depend on LiDAR-specific properties. Additionally, the adaptation of the metric enables comparability for follow-up research. The LSTQ combines the classification score S_{cls} and association score S_{assoc}, resulting in $\text{LSTQ} = \sqrt{S_{\text{cls}} \times S_{\text{assoc}}}$.

B. Moving Instance Tracking

The first experiment evaluates the performance of our approach and its outcomes support the claim that our method achieves state-of-the-art performance in moving instance tracking in sparse and noisy radar scans. We compare our Radar Tracker to the high-performing networks with strong performance in point-based tracking benchmarks. However, we do not consider the best performing Eq-4D-StOP [59] since it incorporates large rotations of the input point clouds, which is detrimental to radar data [29]. Therefore, we utilize CA-Net [26], MOT [46], and the center tracking approach proposed by Yin et al. [49] as baselines. We extend the center tracking with Hungarian matching [22] and directly use the measured Doppler velocities to perform the tracking instead of predicting the velocities of the individual bounding boxes. For the MOT [46] approach, we utilize the IoU as the cost to illustrate the limitations. We adopt the Radar Instance Transformer [51] as the backbone for all methods

Approach	standard offset	temporal offset	$\mathrm{IoU}^{\mathrm{mov}}$
RIT [51]			84.4
Ours	✓		85.2
Ours		✓	85.3
Ours	✓	✓	**85.4**

TABLE II: Influence of the temporal and standard offset predictions in terms of $\mathrm{IoU}^{\mathrm{mov}}$ on the RadarScenes validation set.

since it is the best-performing approach for moving instance segmentation and thus enables a fair comparison.

Our Radar Tracker outperforms the existing methods, especially in terms of LSTQ and $\mathrm{S}_{\mathrm{assoc}}$, as displayed in Tab. I. As mentioned, the MOT [46] approach struggles to associate small instances due to the IoU-based association. The center-based tracking overcomes these limitations and enhances performance. Nevertheless, both methods neglect the appearance features of the instances and thus can not compensate for the shortcomings of geometric tracking. CA-Net combines both features within one cost function. However, the method struggles to capture instance information and to associate the instances based on appearance features. We argue that extracting appropriate features is challenging in sparse radar data, and the design of the network and association function is crucial to enhance accuracy. Our method exceeds these limits and reliably tracks moving instances by combining geometric and appearance features.

C. Ablation Study on Offset Predictions

The second experiment evaluates our offset predictions, especially the temporal offset, and illustrates that our approach is capable of including valuable motion cues to enhance segmentation and tracking quality. To evaluate the segmentation performance, we utilize the $\mathrm{IoU}^{\mathrm{mov}}$ since segmentation of moving detection is essential for tracking. We extend the backbone, the Radar Instance Transformer, with the standard offset prediction and the temporal offset prediction as additional regression targets, as depicted in Tab. II. The standard offset, which points to the center of the instance within the current scan, already improves the $\mathrm{IoU}^{\mathrm{mov}}$ by 0.8 absolute percentage points. Despite that improvement, the temporal offset prediction enhances the performance by an additional 0.1 absolute percentage point. We presume that the temporal offset prediction includes stronger motion cues, especially for instances comprising single detection that do not have a regression target for the standard offset. We combine both offset predictions to enable direct center-based tracking and achieve the best $\mathrm{IoU}^{\mathrm{mov}}$ of 85.4%.

To verify that the offset predictions improve the tracking performance, we evaluate a simple center-based association with and without the center predictions. We remove the appearance features to strictly assess the performance of the geometric approach. The offset prediction improves the $\mathrm{S}_{\mathrm{assoc}}$ from 49.5% to 50.2%, which underlines the advantage of direct temporal offset predictions.

D. Ablation Study on Attentive Association

Finally, we analyze our method concerning the ability to extract reliable attentive similarity scores to associate instances. Therefore, we evaluate the different components of our method as detailed in Tab. III. First, we remove the positional encoding within our attentive instance association, which results in a decrease of 0.3 absolute percentage points. We argue that positional encoding is important to differentiate between similar instances within the scan.

Approach	$\mathrm{S}_{\mathrm{assoc}}$
Ours w/o positional encoding	54.0
Ours with no object class	54.1
Ours w/o sigmoid	54.0
Geometric association $t_{d2} = 10\,\mathrm{m}$	52.2
Ours	**54.3**

TABLE III: Influence of the design decision for the attentive association on the RadarScenes validation set.

In the second step, we add an additional no-object regression target [53] to the attention score to address the occlusion within the appearance features. However, distant instances are often detected in one scan but not covered in the next one, leading to several no-object assignments as ground truth. We assume that this forces the network to assign more instances to the no object class, and the information to track the instances is not covered adequately, which results in a 0.2 absolute percentage points decrease of $\mathrm{S}_{\mathrm{assoc}}$. Additionally, we tried to remove the sigmoid function [53] to directly learn the attention scores. However, this also results in a decrease in performance. To verify that the association based on the attention scores enhances accuracy, we evaluate our method, including only geometric information for the threshold $t_{d2} = 10\,\mathrm{m}$. The geometric association performs worse compared to our combined approach. Hence, the appearance features are essential to track the instances and resolve ambiguities within larger distances. In summary, our evaluation suggests that our method provides competitive moving instance tracking results in sparse and noisy radar point clouds by incorporating geometric and appearance features. Thus, we supported all our claims with this experimental evaluation.

V. Conclusion

In this paper, we presented a novel approach for moving instance tracking in sparse and noisy radar point clouds. Our method exploits temporal offset predictions to encode geometric information to enhance segmentation and tracking. We incorporate appearance features and introduce an attention-based association cost to improve the tracking quality. This allows us to successfully associate individual instances based on valuable geometric and appearance features over time. We furthermore evaluated our method on the radar moving instance tracking benchmark based on the RadarScenes dataset, providing comparisons to other methods and supporting all claims made in this paper. The experiments suggest that combining geometric and appearance features is essential to achieve good performance on moving instance tracking in sparse radar data. Overall, our approach outperforms the state-of-the-art methods, taking a step towards reliable moving instance tracking and sensor redundancy for autonomous vehicles.

References

[1] A. Agarwalla, X. Huang, J. Ziglar, F. Ferroni, L. Leal-Taixé, J. Hays, A. Osep, and D. Ramanan. Lidar Panoptic Segmentation and Tracking without Bells and Whistles. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[2] M. Aygün, A. Osep, M. Weber, M. Maximov, C. Stachniss, J. Behley, and L. Leal-Taixe. 4D Panoptic Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[3] N. Benbarka, J. Schröder, and A. Zell. Score refinement for confidence-based 3D multi-object tracking. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[4] X. Chen, S. Li, B. Mersch, L. Wiesmann, J. Gall, J. Behley, and C. Stachniss. Moving Object Segmentation in 3D LiDAR Data: A Learning-based Approach Exploiting Sequential Data. *IEEE Robotics and Automation Letters (RA-L)*, 6(4):6529–6536, 2021.

[5] Y. Chen, J. Liu, X. Zhang, X. Qi, and J. Jia. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[6] H.K. Chiu, A. Prioletti, J. Li, and J. Bohg. Probabilistic 3d multi-object tracking for autonomous driving. *arXiv preprint*, arXiv:2001.05673, 2020.

[7] C. Choy, J. Gwak, and S. Savarese. 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[8] D.B, Reid. An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, 24(6):843–854, 1979.

[9] A. Dubey, A. Santra, J. Fuchs, M. Lübke, R. Weigel, and F. Lurz. HARadNet: Anchor-free target detection for radar point clouds using hierarchical attention and multi-task learning. *Machine Learning with Applications (MLWA)*, 8:100275, 2022.

[10] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of the Conf. on Knowledge Discovery and Data Mining (KDD)*, 1996.

[11] M.H. Guo, J. Cai, Z.N. Liu, T.J. Mu, R.R. Martin, and S. Hu. PCT: Point Cloud Transformer. *Computational Visual Media*, 7(2):187–199, 2021.

[12] F. Hong, H. Zhou, X. Zhu, H. Li, and Z. Liu. Lidar-based panoptic segmentation via dynamic shifting network. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[13] F. Hong, H. Zhou, X. Zhu, H. Li, and Z. Liu. Lidar-based 4d panoptic segmentation via dynamic shifting network. *arXiv preprint*, arXiv:2203.07186, 2022.

[14] S. Huang, Z. Gojcic, J. Huang, A. Wieser, and K. Schindler. Dynamic 3D Scene Analysis by Point Cloud Accumulation. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[15] L. Hui, L. Wang, L. Tang, K. Lan, J. Xie, and J. Yang. 3D Siamese transformer network for single object tracking on point clouds. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[16] S. Ioffe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2015.

[17] R. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, 82:35–45, 1960.

[18] D. Kim, S. Woo, J. Lee, and I.S. Kweon. Video Panoptic Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] J. Kim, J. Woo, and Sunghoon. RVMOS: Range-View Moving Object Segmentation Leveraged by Semantic and Motion Features. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):8044–8051, 2022.

[20] L. Kreuzberg, I.E. Zulfikar, S. Mahadevan, F. Engelmann, and B. Leibe. 4d-stop: Panoptic segmentation of 4d lidar using spatio-temporal object proposal generation and aggregation. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[21] L. Kreuzberg, I.E. Zulfikar, S. Mahadevan, F. Engelmann, and B. Leibe. 4D-StOP: Panoptic Segmentation of 4D LiDAR using Spatio-temporal Object Proposal Generation and Aggregation. In *Proc. of the Europ. Conf. on Computer Vision Workshops*, 2022.

[22] H. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[23] J. Li, X. He, Y. Wen, Y. Gao, X. Cheng, and D. Zhang. Panoptic-phnet: Towards real-time and high-precision lidar panoptic segmentation via clustering pseudo heatmap. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[24] I. Loshchilov and F. Hutter. SGDR: Stochastic Gradient Descent with Warm Restarts. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2017.

[25] R. Marcuzzi, L. Nunes, L. Wiesmann, J. Behley, and C. Stachniss. Mask-Based Panoptic LiDAR Segmentation for Autonomous Driving. *IEEE Robotics and Automation Letters (RA-L)*, 8(2):1141–1148, 2023.

[26] R. Marcuzzi, L. Nunes, L. Wiesmann, I. Vizzo, J. Behley, and C. Stachniss. Contrastive Instance Association for 4D Panoptic Segmentation for Sequences of 3D LiDAR Scans. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[27] B. Mersch, X. Chen, I. Vizzo, L. Nunes, J. Behley, and C. Stachniss. Receding Moving Object Segmentation in 3D LiDAR Data Using Sparse 4D Convolutions. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7503–7510, 2022.

[28] V. Nair and G. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2010.

[29] 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 Robotics and Automation Letters (RA-L)*, 7(2):4961–4968, 2022.

[30] Z. Pang, Z. Li, and N. Wang. Simpletrack: Understanding and rethinking 3d multi-object tracking. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[31] C. Park, Y. Jeong, M. Cho, and J. Park. Fast Point Transformer. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[32] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2019.

[33] C.R. Qi, H. Su, K. Mo, and L.J. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[34] C. Qi, K. Yi, H. Su, and L.J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2017.

[35] H. Qiu, B. Yu, and D. Tao. GFNet: Geometric Flow Network for 3D Point Cloud Semantic Segmentation. *Trans. on Machine Learning Research (TMLR)*, 2022.

[36] J. Schult, F. Engelmann, A. Hermans, O. Litany, S. Tang, and B. Leibe. Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[37] O. Schumann, M. Hahn, J. Dickmann, and C. Wöhler. Semantic Segmentation on Radar Point Clouds. In *Proc. of the Intl. Conf. on Information Fusion*, 2018.

[38] O. Schumann, M. Hahn, N. Scheiner, F. Weishaupt, J.F. Tilly, J. Dickmann, and C. Wöhler. RadarScenes: A real-world radar point cloud data set for automotive applications. In *Proc. of the Intl. Conf. on Information Fusion*, 2021.

[39] O. Schumann, J. Lombacher, M. Hahn, C. Wöhler, and J. Dickmann. Scene Understanding With Automotive Radar. *IEEE Trans. on Intelligent Vehicles*, 5(2):188–203, 2019.

[40] Y. Shi and K. Ma. SAFIT: Segmentation-Aware Scene Flow with Improved Transformer. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[41] S. Su, J. Xu, H. Wang, Z. Miao, X. Zhan, D. Hao, and X. Li. PUPS: Point cloud unified panoptic segmentation. *arXiv preprint*, arXiv:2302.06185, 2023.

[42] J. Sun, Y. Dai, X. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen. Efficient Spatial-Temporal Information Fusion for LiDAR-Based 3D Moving Object Segmentation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2017.

[44] T. Vu, K. Kim, T.M. Luu, T. Nguyen, and C.D. Yoo. SoftGroup for

3D Instance Segmentation on Point Clouds. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[45] N. Wang, C. Shi, R. Guo, H. Lu, Z. Zheng, and X. Chen. InsMOS: Instance-Aware Moving Object Segmentation in LiDAR Data. *arXiv preprint*, arXiv:2303.03909, 2023.

[46] X. Weng, J. Wang, D. Held, and K. Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[47] L. Xin, L. Jianhui, J. Li, W. Liwei, Z. Hengshuang, L. Shu, Q. Xiaojuan, and J. Jiaya. Stratified Transformer for 3D Point Cloud Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[48] T.X. Xu, Y.C. Guo, Y.K. Lai, and S.H. Zhang. CXTrack: Improving 3D point cloud tracking with contextual information. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[49] T. Yin, X. Zhou, and P. Krähenbühl. Center-Based 3D Object Detection and Tracking. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[50] M. Zeller, J. Behley, M. Heidingsfeld, and C. Stachniss. Gaussian Radar Transformer for Semantic Segmentation in Noisy Radar Data. *IEEE Robotics and Automation Letters (RA-L)*, 8(1):344–351, 2023.

[51] M. Zeller, V.S. Sandhu, B. Mersch, J. Behley, M. Heidingsfeld, and C. Stachniss. Radar Instance Transformer: Reliable Moving Instance Segmentation in Sparse Radar Point Clouds. *IEEE Trans. on Robotics (TRO)*, pages 1–17, 2023.

[52] M. Zeller, V.S. Sandhu, B. Mersch, J. Behley, M. Heidingsfeld, and C. Stachniss. Radar Velocity Transformer: Single-scan Moving Object Segmentation in Noisy Radar Point Clouds. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[53] C. Zhang, C. Zhang, Y. Guo, L. Chen, and M. Happold. Motiontrack: End-to-end transformer-based multi-object tracking with lidar-camera fusion. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops*, 2023.

[54] C. Zhang, H. Wan, X. Shen, and Z. Wu. PVT: Point-voxel transformer for point cloud learning. *Intl. Journal of Intelligent Systems*, 37(12):11985–12008, 2022.

[55] Y. Zhang, Z. Zhou, P. David, X. Yue, Z. Xi, B. Gong, and H. Foroosh. Polarnet: An improved grid representation for online lidar point clouds semantic segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[56] H. Zhao, J. Jia, and V. Koltun. Exploring self-attention for image recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[57] H. Zhao, L. Jiang, J. Jia, P.H. Torr, and V. Koltun. Point Transformer. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[58] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue. Towards Deep Radar Perception for Autonomous Driving: Datasets, Methods, and Challenges. *Sensors*, 22, 2022.

[59] M. Zhu, S. Han, H. Cai, S. Borse, M.G. Jadidi, and F. Porikli. 4D Panoptic Segmentation as Invariant and Equivariant Field Prediction. *arXiv preprint*, arXiv:2303.15651, 2023.

[60] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.