SPR: Single-Scan Radar Place Recognition

Daniel Casado Herraez Le Chang Matthias Zeller Louis Wiesmann
Jens Behley Michael Heidingsfeld Cyrill Stachniss

***Abstract*—Localization is a crucial component for the navigation of autonomous vehicles. It encompasses global localization and place recognition, allowing a system to identify locations that have been mapped or visited before. Place recognition is commonly approached using cameras or LiDARs. However, these sensors are affected by bad weather or low lighting conditions. In this paper, we exploit automotive radars to address the problem of localizing a vehicle within a map using single radar scans. The effectiveness of radars is not dependent on environmental conditions, and they provide additional information not present in LiDARs such as Doppler velocity and radar cross section. However, the sparse and noisy radar measurement makes place recognition a challenge. Recent research in automotive radars addresses the sensor's limitations by aggregating multiple radar scans and using high-dimensional scene representations. We, in contrast, propose a novel neural network architecture that focuses on each point of single radar scans, without relying on an additional odometry input for scan aggregation. We extract pointwise local and global features, resulting in a compact scene descriptor vector. Our model improves local feature extraction by estimating the importance of each point for place recognition and enhances the global descriptor by leveraging the radar cross section information provided by the sensor. We evaluate our model using nuScenes and the 4DRadarDataset, which involve 2D and 3D automotive radar sensors. Our findings illustrate that our approach achieves state-of-the-art results for single-scan place recognition using automotive radars.**

***Index Terms*—Localization, SLAM, Autonomous Vehicle Navigation**

I. Introduction

GLOBAL localization is a central pillar in autonomous mobility navigation stacks. After a map is created, the ability to recognize previously visited locations enables precise pose estimation and accurate loop closure for SLAM in GNSS-denied environments, such as parking garages and indoor scenes. Until now, the main sensors employed for place recognition have been cameras and LiDARs. Cameras are compact to fit within an end-user vehicle. However, their performance degrades depending on the lighting conditions. LiDARs are much less affected by lighting conditions, but their laser ranging capabilities underperform in bad weather scenarios like fog, snow, or heavy rain.

In this work, we explore the challenge of place recognition using only automotive radars, eliminating the reliance on GNSS or other supplementary sensor data during operation. We focus on the problem of obtaining a compressed yet informative scene descriptor, as in Fig. 1. We achieve this employing automotive radars, that are already integrated into consumer vehicles today. They are popular sensors as they are compact, affordable, and resilient to adverse weather conditions. They also provide supplementary signals that can support scene understanding [9] [57] [58]. These include the Doppler velocity of the radar targets, and the radar cross section (RCS). The Doppler velocity provides an estimate of the relative radial velocity of the measurement, proving a useful cue for dynamic object detection [9]. RCS provides information about the detectability of an object and depends on its materials and point of view, making it a valuable feature for our network. Nonetheless, the noise from clutter and multi-path propagations, along with sparse output scans, makes radar place recognition a complex undertaking.

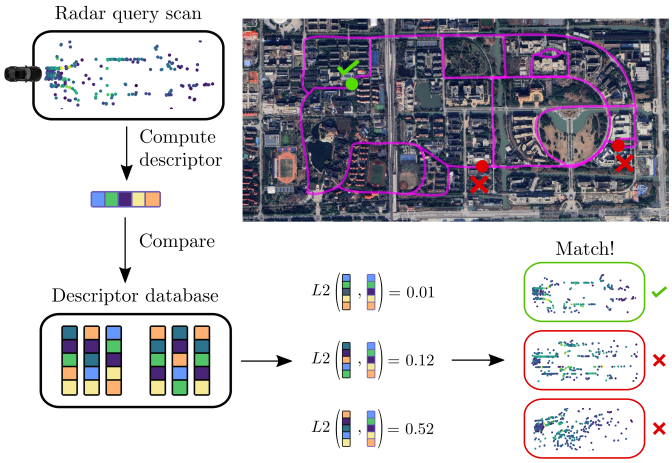

Fig. 1: Radar place recognition. The query scan is encoded by our network and compared to the map database using the L2 distance metric. Place recognition is performed by identifying the closest matching descriptor from the database.

Due to the high amount of noise, it can be challenging to discriminate between direct measurements of objects and those coming from clutter or reflections. The difficulty becomes even larger due to the low number of points per scan, as it can be hard to identify objects and structural elements in the environment. Some solutions rely on machine learning [31] or geometric models [22] [30] to discern between direct measurements and surface reflections. Other methods aim to approximate radar scans to LiDAR point clouds using occupancy grids [32] [50]. However, the sparse and noisy nature of radar scans makes it difficult to directly apply LiDAR solutions for radar place recognition [8]. To mitigate the challenges of sparsity in radar point clouds, some place recognition methods

Manuscript received: Mar. 19, 2024; Revised: May 29, 2024; Accepted: Jun. 21, 2024. This paper was recommended for publication by Javier Civera upon evaluation of the Associate Editor and Reviewers' comments.

D. Casado Herraez and M. Zeller are with CARIAD SE and with the University of Bonn, Germany. L. Chang is with CARIAD SE and with the University of Stuttgart, Germany. L. Wiesmann and J. Behley are with the Center for Robotics, University of Bonn, Germany. M. Heidingsfeld is with CARIAD SE, Germany. C. Stachniss is with the Center for Robotics, University of Bonn and with the Lamarr Institute for Machine Learning and Artificial Intelligence, Germany.

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

use radar scan aggregation to create denser sub-maps of the scene [8]. This relies on having an additional odometry input such as an IMU, wheel encoder, or another means of pose estimation [9]. Other current methods also provide localization solutions of radar over a map collected using a different type of sensors [34] [42] [49] [55]. Most approaches, however, come primarily from scanning radars [1] [16] [56], which are larger and harder to mount in end-user vehicles.

The main contribution of this paper is a novel solution for place recognition using single scans of automotive radars, without relying on an additional odometry input. Our focus encompasses a deep learning model that handles sparsity by processing the scan in a pointwise manner. It leverages the point coordinates and the RCS information provided by the sensor to collectively enhance the capabilities of radar-based place recognition. Our model is based on a point encoder that extracts features from the original point clouds, a network that encodes the RCS values, a point scoring module that estimates the importance of a point for place recognition, and a global descriptor extractor for spatial clustering of the global descriptor vectors. Our method achieves state-of-the-art results for 2D and 3D automotive radar place recognition using a comparably smaller scene descriptor.

In sum, we make three key claims: Our system (i) achieves state-of-the-art performance on automotive radar single-scan place recognition while keeping a compact scene representation, (ii) provides a novel procedure to utilize RCS information to describe the scene improving accuracy, and (iii) enhances feature extraction by estimating the importance of points within the scan for place recognition.

II. Related Work

We present an overview of state-of-the-art approaches for place recognition. We divide it into two main categories, general place recognition and radar-based place recognition.

General place recognition estimates whether the current location has been visited before. The goal is to find the closest matching query in a database, as illustrated in Fig. 1. However, storing and comparing images or scans directly would be inefficient and computationally expensive. Camera-based solutions encode the images into handcrafted features like ORB [36], learned features such as NetVLAD [2], or enhance their system exploiting image sequence information [35] [46] [47]. LiDAR approaches commonly compress the scan into a more compact representation based on geometric or learned features. LiDAR geometric approaches usually transform the point cloud into a polar representation, as seen in ScanContext [25] and its multiple variations [13] [21] [24] [48], or in density grid maps, as shown by Gupta et al. [19]. The main concern with such methods is that they rely on structural assumptions about the environments like height, intensity or contours. Advances in LiDAR place recognition show that enhancing those classical approaches with learning-based solutions yields higher recognition recall, like DiSCO [53] and the method by Kim et al. [26]. Another group of approaches discretize the space into a voxel grid [10] [29] [41] [52], which may lead to a loss of information considering the low number of points in a radar scan. Other methods like PointNetVLAD [45] and KPPR [51] consider each individual point by using a pointwise feature encoder, which is more appropriate for our task of sparse radar scan place recognition. They aggregate their local point features into a global VLAD descriptor representative of the scene. Finally, some methods incorporate LiDAR intensity information [12]. A more extensive review of LiDAR methods for place recognition can be found in the recent survey by Yin et al. [54]. In terms of autonomous driving, bad weather conditions and difficult packaging are the main concerns when incorporating these sensors in end-user vehicles. Moreover, as shown by Cai et al. [8], adapting LiDAR methods [25] [28] directly for automotive radars can lead to a decrease in performance due to the sparsity and high amount of noise present in their output point cloud.

Radar-based place recognition estimates whether the current location has been visited before based on radar scans. Several methods employ spinning radars for this task using handcrafted features [1] [11] [20], learned features [5] [15] [40] [44], or contrastive learning [16] [56]. They use the intensity image provided by the spinning radar, encoding it into a descriptor that represents each place. Alternatively, solutions exist where the robot carrying a radar localizes over maps collected using different sensor types, such as LiDARs [34] [37] [55], satellite images [42], and binary maps [49]. This is particularly useful in locations where data can be previously collected with high-accuracy sensing techniques, but these maps are not always available. New existing approaches also combine multi-modal sensor fusion using camera and radars for place recognition [14] [18]. Those methods that exclusively rely on automotive radar, generally leverage a variant of intensity ScanContext as an additional component to their SLAM pipeline [33] [59]. However, their primary focus is on achieving a complete radar SLAM system, with minimal emphasis on the place recognition component. Notably, TransLoc4D [39] and Autoplace [8] aim for high-accuracy place recognition using automotive radars. Autoplace, however, imposes limitations on geometric understanding between the points and restricts the features to a planar space. Additionally, it relies on point cloud aggregation, which requires an additional odometer and the availability of radar sub-maps for place recognition. Autoplace's resulting descriptor vector is also high-dimensional, making it less compact for storage in a database.

In contrast to previous methodologies, this paper introduces a new approach for odometer-free single-scan place recognition using only automotive radars. It proposes a way of leveraging the radar point coordinates to capture the geometric features of the environment into a compressed, yet informative, descriptor. Our model exploits the additional RCS information provided by the sensor and handles the radar noise and sparsity by focusing on points that are important for place recognition. Our approach is capable of operating with 2D and 3D radar sensors in real-world scenarios, all substantiated by experimental evidence.

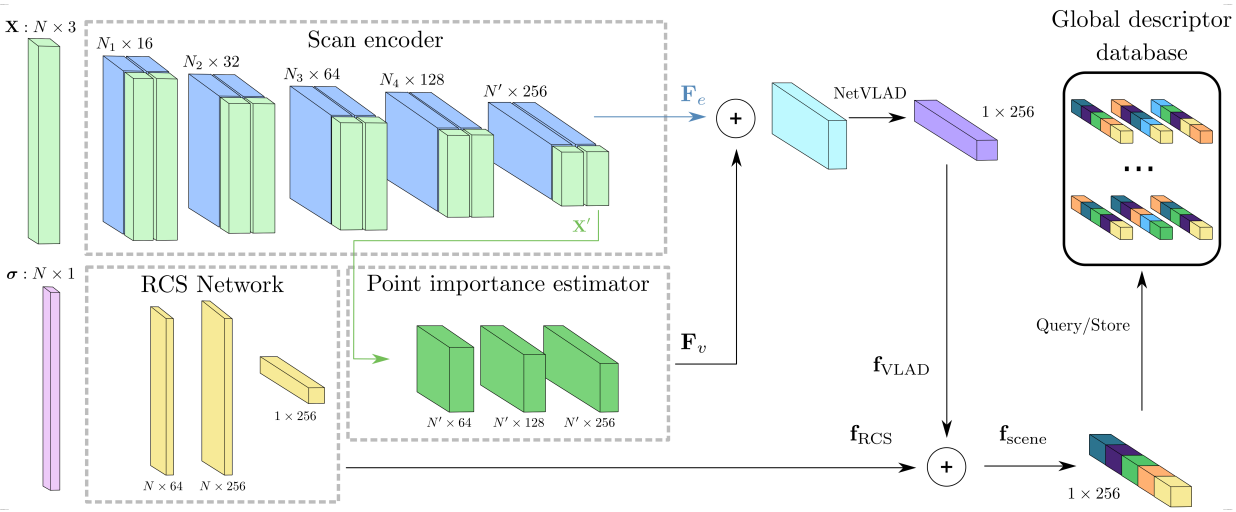

Fig. 2: System description diagram of our network architecture. The 3D radar scan point information is encoded resulting in $\mathbf{F}_e \in \mathbb{R}^{N' \times D}$. The contribution of each downsampled point for place recognition $\mathbf{F}_v \in \mathbb{R}^{N' \times D}$ is computed with our point importance estimator module. The feature vector is then transformed using NetVLAD to a global descriptor, and combined with the output of our RCS network module. The resulting descriptor can then be stored in the database during the first pass, or queried from the database for place recognition.

III. Our Approach

Our approach aims to achieve odometer-free single-scan radar place recognition. The process involves comparing scans stored in a database with current scans during navigation, see Fig. 1. Initially, radar sensors capture the environment during the robot's first pass at a location. We convert each scan into a place descriptor using our neural network (Fig. 2), which combines local and global point information from the measured point cloud. Then, we store the descriptor in a map database. During navigation, we query the database with current measurements converted into descriptors using the same encoder-descriptor network from Fig. 2. This allows us to identify locations with high similarity according to our scoring function. Our approach utilizes the Doppler velocity provided by the radar in order to filter dynamic point outliers. We leverage point convolutions to capture local information together with a scoring method to estimate the contribution of each point for place recognition. Additionally, we incorporate global point information within the scan through RCS data and a NetVLAD [45] global descriptor.

Dynamic point pre-filtering. A key advantage of automotive radars is the measurement of the Doppler velocity of the point targets. It represents the relative radial velocity of the measured point with respect to the vehicle. This velocity information cannot be used as an additional channel for the place recognition network, as a vehicle can revisit the same place at different speeds. Nevertheless, it allows to differentiate between static and dynamic points [57]. We follow Cai et al. [8] to focus solely on the static points of the radar scan for place recognition. The main idea is that static points of the environment should match the ego-vehicle's velocity. Points with a different velocity are likely to correspond to moving objects, and thus, are considered outliers. We preprocess all the scans resulting in filtered point clouds containing only points of the scene likely to be static.

Scan encoder. The main goal of the scan encoder is to obtain a feature space representation $\mathbf{F}_e \in \mathbb{R}^{N' \times D}$ of the filtered static radar point cloud $\mathbf{X} \in \mathbb{R}^{N \times 3}$. In the case of 2D radars, we assume the z dimension to be zero. The encoding should contain spatial data descriptive enough for place recognition, which involves capturing contextual information of points at different scales. Autoplace [8] achieves this by projecting its radar point cloud into a 2D image and encoding it with a convolutional neural network. As a consequence, most vertical and geometric information is lost. Similar to KPPR [51], we focus on capturing 3D contextual information of individual points directly from the radar scan using rigid kernel point convolutions (KPConv) [43]. For a point $\mathbf{x}_i^l$ from point cloud $\mathbf{X}^l = [\mathbf{x}_1^l, \mathbf{x}_2^l, ..., \mathbf{x}_n^l]^\top$ at layer l, the convolution of features $\mathbf{F}^{l-1} \in \mathbb{R}^{N_{l-1} \times D_{l-1}}$ with kernel g^l is given as:

$$(\mathbf{F}^{l-1} * g^l)(\mathbf{x}_i^l) = \sum_{\mathbf{x}_k^l \in \mathcal{N}^l(\mathbf{x}^l)} g(\mathbf{x}_k^l - \mathbf{x}_i^l)^l \boldsymbol{f}_k^{l-1}, \qquad (1)$$

where $\mathcal{N}^l(\mathbf{x}_i^l) = \{\mathbf{x}_k^l \mid \|\mathbf{x}_k^l - \mathbf{x}_i^l\| < r^l\}$ denotes the neighbor points of $\mathbf{x}_i^l$ in a radius $r^l \in \mathbb{R}$.

Our encoder is composed of a sequence of five convolutional and downsampling layers that capture local features $\mathbf{F}^l$ at different levels with different radii. We use grid downsampling at different scales, and contrary to KPConv [43], we find that downsampling before computing convolution helps process contextual information of the points. We also increase the radius of $\mathcal{N}^l(\mathbf{x}_i^l)$ in every new convolution-downsampling block for an extended receptive field. We also test adding an additional channel to the input point cloud accounting for the RCS information of each point.

Point importance estimator. Due to the sparsity and high amount of outliers coming from noise and multi-path reflections in radar point clouds, it is crucial that the place recognition network is able to identify useful anchor points within the scan. Knowing that noise can randomly appear and disappear between single scans, we propose a point importance

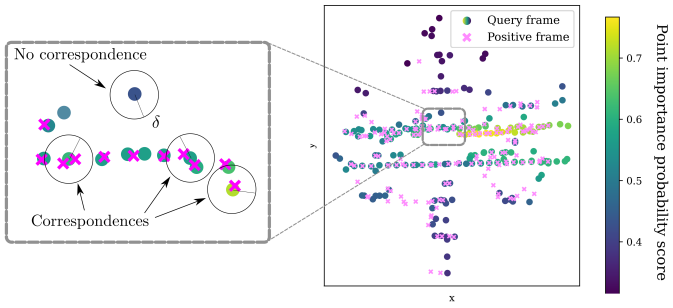

Fig. 3: Point importance estimation module. (left) During training, those query points that have a correspondence in the positive frame within a radius δ, represented as a circle, are considered as valuable for place recognition. Those that do not have a correspondence can be highly inconsistent between scans, thus their predicted importance should be reduced. Although all points are checked, we only display the radius on a small subset for clear visualization. (right) Resulting probabilities of our point importance estimation module. Those areas with higher densities and geometric line patterns are considered as more important than random points located far away from the sensor.

estimation module to guide the training of the network, focusing on points that are relevant for place recognition. As a result of finding valuable measurements, this module also helps focusing on those points that we are more certain that exist, such as high-density locations, patterns in the environment, and the point distribution within the scans, as shown in Fig. 3. We achieve this by adding an additional feature to our network architecture that outputs the probability of a point being important for place recognition.

Using a subsampled scan $\mathbf{X}' = [\mathbf{x}'_1, \mathbf{x}'_2, ..., \mathbf{x}'_n]^\top \in \mathbb{R}^{N' \times 3}$ to estimate the importance of their corresponding encoded local features, we encode the probability of a point $\mathbf{x}'_j \in \mathbb{R}^3$ being valuable into a feature vector $\boldsymbol{f}_{v_j} \in \mathbb{R}^D$ using an MLP with three learnable layers, ReLU activations and layer normalization such that

$$\boldsymbol{f}_{v_j} = \text{MLP}_v(\mathbf{x}'_j). \tag{2}$$

The importance probability of a point is within the range $[0, 1]$. It indicates how valuable the point is for place recognition. It is estimated as

$$P(\mathbf{x}'_j) = \text{sigmoid}\left(\phi\left(\boldsymbol{f}_{v_j}\right)\right), \tag{3}$$

where ϕ indicates a linear projection of the feature vector from $\mathbb{R}^D$ to $\mathbb{R}$. We use this scoring later in Eq. (9) to compute the loss during training between points that only exist in one of the scans that are being compared.

The encoded probabilities $\boldsymbol{f}_v$ of all downsampled points are stacked into $\mathbf{F}_v \in \mathbb{R}^{N' \times D}$, and added to the output of the embedding, resulting in:

$$\mathbf{F}_{e+v} = \mathbf{F}_e + \mathbf{F}_v, \tag{4}$$

where $\mathbf{F}_{e+v} \in \mathbb{R}^{N' \times D}$. Rather than multiplying the probabilities as weights, we add them to the normalized features from the encoder in order to enhance the feature embeddings corresponding to those points that are more valuable for place recognition.

Radar cross section network. To maximize the potential of radar sensors, we propose an additional module that uses radar cross section information to enhance the final feature descriptor. RCS measures how detectable an object is by the radar based on the object properties themselves and the measurement angle. It provides additional information for each point in $\mathbf{X}$ about the properties of a specific location, making it a valuable attribute for place recognition. However, unlike research in object detection [38], our experiments show that adding the RCS as an additional channel minimally enhances place recognition performance.

Instead, we propose a RCS network module that learns the RCS feature representation of the entire radar scan $\mathbf{f}_{\text{RCS}} \in \mathbb{R}^D$ and is permutation invariant. This enhances the global descriptor vector making it more informative than a descriptor only containing point information.

To capture the RCS information $\boldsymbol{\sigma} \in \mathbb{R}^N$ from the scan, we propose to use a two layer MLP encoding the RCS value of each point σ_i into a feature encoding $\boldsymbol{f}_{\sigma_i} \in \mathbb{R}^D$ such that

$$\boldsymbol{f}_{\sigma_i} = \text{MLP}_r(\sigma_i). \tag{5}$$

We then aggregate and normalize the features from all points over the feature dimension making the result permutation invariant following

$$\mathbf{f}_{\text{RCS}} = \frac{\sum_{i=1}^{N} \boldsymbol{f}_{\sigma_i}}{\|\sum_{i=1}^{N} \boldsymbol{f}_{\sigma_i}\|_2}. \tag{6}$$

The resulting $\mathbf{f}_{\text{RCS}}$ acts as a global descriptor containing the distribution of RCS values for that particular scan. While Autoplace [8] uses an additional stage that performs histogram re-ranking after their network prediction, we integrate our RCS network module inside the model avoiding the additional postprocessing step.

Global descriptor database. The descriptor extraction layer aggregates local features into a single global descriptor vector. We exploit a NetVLAD layer [45] to aggregate the local features $\mathbf{F}_{e+v}$ from Eq. (4) into K learnable cluster centers, resulting in $\mathbf{f}_{\text{VLAD}} \in \mathbb{R}^D$. These learnable centers represent points calculated from groups of similar local descriptors. We combine the VLAD descriptor with the RCS descriptor leading to the final global descriptor vector $\mathbf{f}_{\text{scene}}$ following

$$\mathbf{f}_{\text{scene}} = \frac{\mathbf{f}_{\text{VLAD}} + \mathbf{f}_{\text{RCS}}}{\|\mathbf{f}_{\text{VLAD}} + \mathbf{f}_{\text{RCS}}\|_2}. \tag{7}$$

The resulting descriptor vector represents the location measured by a radar scan. It contains the local information obtained from the point encoder and the point importance estimator, as well as the global features extracted using the RCS network and NetVLAD.

During the map recording process, every scan is stored as a different descriptor vector $\mathbf{f}^{\text{m}}_{\text{scene}} \in \mathbb{R}^D$ in a KDTree [6] map database $\mathcal{M} = \{\mathbf{f}^{\text{m}}_{\text{scene}_1}, \mathbf{f}^{\text{m}}_{\text{scene}_2}, ..., \mathbf{f}^{\text{m}}_{\text{scene}_M}\}$. The query scan is also encoded as the feature descriptor vector $\mathbf{f}^{\text{q}}_{\text{scene}}$, and compared to those stored in $\mathcal{M}$ using the L2 distance metric. We use superscript "m" to indicate that the feature vector belongs to the map database.

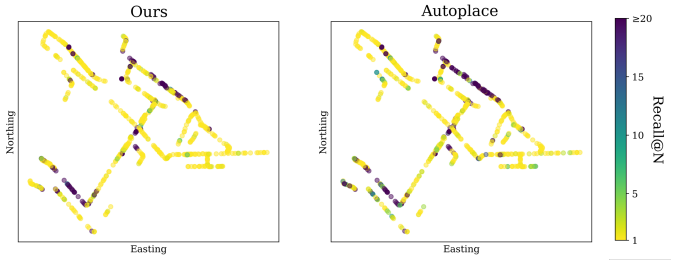

Fig. 4: Comparison of the place recognition recalls at each location of the nuScenes for single radar scans for (left) our network and (right) Autoplace without temporal encoding.

Metric learning for place recognition. The goal of training is that the network learns a useful and compressed representation of the environment. For a query descriptor $\mathbf{f}^{\text{q}}_{\text{scene}}$, the feature descriptor vector must be similar for places that are alike, named positive samples $\mathbf{f}^{\text{pos}}_{\text{scene}}$, and dissimilar for those that are very different, denoted as negative samples $\mathbf{f}^{\text{neg}}_{\text{scene}}$. Positive and negative samples are defined during training based on GNSS distance information. However, during inference, GNSS is no longer required for the operation of our place recognition network.

Positive samples are defined as those within a radius distance R_1 to the query measured with the GNSS location. Furthermore, due to occlusions or different viewing perspectives, the positive samples taken from a close geo-location may look very different from the queries themselves. Therefore during training, the selected positive samples are those within R_1 that have the lowest L2 distance between descriptors. We also observe an increase in performance when training on multiple positives for the same location, as the network learns that the same location can be measured in multiple ways and from different view points.

Negative samples are those that are farther away from a bigger radius R_2 such that $\text{R}_2 > \text{R}_1$. However, as the dataset may contain very different scans from different locations, randomly selecting negative samples may lead to low discrimination and generalization capabilities. As proposed by Uy et al. [45], we use a hard negative mining strategy to find the most similar negative sample to the query descriptor $\mathbf{f}^{\text{q}}_{\text{scene}}$ in the feature space. This helps the network learn to discriminate challenging scenes where a wrong database match closely resembles the query scan.

We leverage the triplet margin loss by Balntas et al. [3] minimizing the distance of the query to a positive sample $d_{\text{pos}} = L2(\mathbf{f}^{\text{q}}_{\text{scene}}, \mathbf{f}^{\text{pos}}_{\text{scene}})$, and maximizing it with respect to the H hardest negative samples $d_{\text{neg}_h} = L2(\mathbf{f}^{\text{q}}_{\text{scene}}, \mathbf{f}^{\text{neg}_h}_{\text{scene}})$. The triplet margin loss for a constant margin α is given as:

$$\mathcal{L}_t = \sum_{h=1}^{H} \max(d_{\text{pos}} - d_{\text{neg}_h} + \alpha, 0). \tag{8}$$

Additionally, to account for the estimated value of the points from our point importance module (Sec. III), we propose to measure the nearest point correspondences between the query and its associated positive scan. First, we align both scans by transforming them to GNSS global coordinates. Then, as shown in Fig. 3, we introduce a radius distance hyperparameter δ that determines whether a given point in the query scan has a correspondence in the positive scan. We consider valuable points for place recognition only those that demonstrate consistency, thus existing in both scans. Note that the GNSS signal is only required for computing the loss during training but is dispensable during operation.

Interpreting this as a binary classification problem, we assign those points in the query scan that have a correspondence in the positive scan a label $\hat{P}(\mathbf{x}'_j) = 1$, and $\hat{P}(\mathbf{x}'_j) = 0$ to those without a correspondence. Denoting $P_j = P(\mathbf{x}'_j)$ for brevity, the final binary cross-entropy loss is:

$$\mathcal{L}_v = -\sum_{j=1}^{N'} P_j \log(\hat{P}_j) + (1 - P_j) \log(1 - \hat{P}_j). \tag{9}$$

The final loss is a weighted sum between the triplet loss in Eq. (8) and the point loss in Eq. (9)

$$\mathcal{L} = \mathcal{L}_t + \gamma \mathcal{L}_v, \tag{10}$$

where γ is a tuneable parameter.

IV. Experimental Evaluation

This work aims to achieve odometer-free single-scan radar place recognition using automotive radars. We present our experiments to show the capabilities of our system. The results of our evaluation support our key claims that our work (i) achieves state-of-the-art performance on automotive radar single-scan place recognition while keeping a compact scene representation, (ii) provides a novel procedure to utilize RCS information to describe the scene improving accuracy, and (iii) enhances feature extraction by estimating the importance of points within the scan for place recognition.

A. Experimental Setup

The goal of our approach is to reliably retrieve the position in a given map based on a single radar scan. On the evaluation of our method, we run experiments on real-world driving scenarios, using 2D and 3D radar datasets, nuScenes [7], and the 4DRadarDataset [33], respectively. These datasets contain the point cloud output provided by the radar sensors, hence the result of our point-based method being independent from the key point extraction algorithm, required for point-based approaches in NavTech radar datasets [4] [23]. We first evaluate our work on single-scan radar place recognition and compare it to state-of-the-art solutions. We provide quantitative and qualitative results for the comparison. Then, we carry out an ablation study of our system to analyze the contribution of each module to the final result, as well as how the new hyperparameters have an effect on place recognition.

B. Implementation Details

The architecture of our network is described in Fig. 2. We train the model for 80 epochs using a batch size of 4. Following KPPR [51], we set the descriptor size $D = 256$. We use the Adam [27] optimizer with learning rate 5×10^{-6} and a learning rate decay of 0.95 every 20 steps. We set the triplet

TABLE I: Comparison with state of the art in nuScenes.

	R@1% [%]	R@1/5/10 [%]	Output dim.
KPPR	88.0	66.1 / 78.1 / 81.6	**256**
RadVLAD	33.5	1.74 / 6.09 / 10.5	32768
RadVLAD RCS	17.2	0.97 / 3.19 / 4.84	32768
FFT-RadVLAD	16.3	0.29 / 1.74 / 3.29	32768
FFT-RadVLAD RCS	11.2	0.19 / 0.58 / 1.64	32768
ScanContext	24.7	15.3 / 20.8 / 22.2	1200
ScanContext RCS	6.38	3.58 / 4.84 / 5.22	1200
Auto+TE 7 sweep	85.9	78.9 / 83.1 / 84.3	4096
Ours 7 sweep	87.5	78.7 / 82.7 / 83.3	256
Auto 1 sweep	83.0	60.8 / 72.8 / 76.5	9216
Auto+TE 1 sweep	86.4	73.4 / 81.2 / 83.0	4096
Ours 1 sweep	**88.2**	**76.2 / 82.3 / 83.9**	**256**

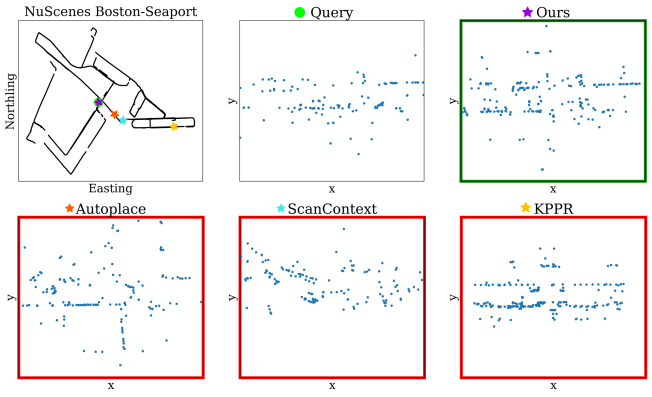

Fig. 5: Qualitative place recognition results in nuScenes Boston-Seaport. The first image displays the location where the query measurement was taken and the predictions. The other images present the corresponding radar scans to the predicted location of compared methods for the same query. A green frame indicates correct predictions while a red frame refers to incorrect predictions.

margin loss parameter $\alpha = 0.1$ We set the KPConv radius at each layer to $\mathrm{r} = [3, 6, 12, 24, 48]$ m and a grid downsampling of sizes $0.25, 0.5, 1, 2, 4$ m. Kernel size is set to 5 for nuScenes and 7 for the 4DRadarDataset. The number of cluster centers K for NetVLAD is set to 64.

NuScenes provides insight into how our approach behaves over long periods of time on 2D automotive radars. We follow a similar procedure to Cai et al. [8], training on the scans from the "Boston Sea-port" location obtained in the first 105 days. We split our evaluate : test sets with a $4:1$ ratio using the scans after the 105 day threshold. We consider a prediction correct if it is within $\mathrm{R}_1 = 9$ m of the actual location. During training, negative samples are taken from outside a $\mathrm{R}_2 = 18$ m radius. Moreover, we train on 5 positive samples for each location for better performance.

The 4DRadarDataset [33] shows the performance of our method in place recognition for loop closure detection using automotive 3D radars, as the recordings were taken within a short time span with minimal variations in the environment. The dataset is split into four sequences. We use "Campus 1" and "Campus 4" for training, and evaluate on "Campus 2" and "Campus 3". Additionally, we consider a prediction correct if it is within $\mathrm{R}_1 = 2$ m. During training, negative samples are taken from outside a $\mathrm{R}_2 = 4$ m radius.

TABLE II: Comparison with state of the art in 4DRadarDataset.

	R@1% [%]	R@1/5/10 [%]	Output dim.
KPPR	99.4	95.4 / 99.2 / 99.2	**256**
RadVLAD	99.3	91.9 / 96.3 / 97.3	32768
RadVLAD RCS	99.3	94.6 / 97.4 / 98.2	32768
FFT-RadVLAD	50.4	20.0 / 26.5 / 30.8	32768
FFT-RadVLAD RCS	57.3	21.5 / 28.8 / 34.2	32768
ScanContext	87.7	79.7 / 86.6 / 87.4	1200
ScanContext RCS	94.7	94.0 / 94.7 / 94.7	1200
Auto	99.9	94.4 / 99.4 / **99.8**	9216
Auto+TE*	99.5	**97.7** / 99.0 / 99.1	4096
Ours	**100.0**	97.1 / **99.6** / **99.8**	**256**

TE*: Multi-scan temporal encoding introduced in Auroplace [8].

C. Comparison with the State of the Art

The first experiment evaluates the accuracy and the resulting descriptor size of our single-scan radar place recognition system in comparison to other methods in 2D and 3D radar datasets. Similar to other works, we denote recall as "R" and measure the R@1/5/10 and R@1% metrics, which represent if the selected query by the system ranks among the top 1, 5, 10 or 1% candidates in the database. We also display the output descriptor dimensionality describing the size of the descriptor being stored in the database. A higher dimensional descriptor requires higher storage space, while more compressed descriptors require less space and can be stored more efficiently.

We show the comparison of our method with state of the art approaches in Tab. I and Tab. II, and qualitative results in Fig. 4 and Fig. 5. We evaluate it against the feature-based LiDAR methods of ScanContext [48] and RCS ScanContext [23], a learning-based LiDAR method KPPR [51], a frequency-based scanning radar approach FFT-RadVLAD [17] and a clustering-based approach RadVLAD [17] with and without RCS values, and a learning-based automotive radar solution Autoplace (Auto) [8]. We place special focus on Autoplace [8]. The original implementation is done with 7 aggregated radar point clouds, and their LSTM-based temporal encoding (TE) considers 3 consecutive aggregated scans. In addition to the results in their paper, we show for comparison in nuScenes their results for 1 sweep with and without temporal encoding. The top results for 1 and 7 sweeps are displayed **bolded** and underlined, respectively.

Our method achieves state-of-the-art results for both datasets achieving comparable performance to multi-scan approaches while using a more compact scene descriptor. In nuScenes, the low number of points per radar scan makes it difficult for non-learning-based methods to extract useful patterns from the environment. The discrete nature of projected point clouds also poses a challenge for spinning radar frequency-based approaches like FFT-RadVLAD. For the Radar4DDataset, our method on a single scan obtains a similar R@1 to Autoplace using TE and 3 scans. The experiment demonstrates how our compact descriptor remains highly informative for place recognition.

D. Ablation studies

The second set of experiments supports our claim that integrating RCS into the network and estimating the impor-

TABLE III: Ablation studies of the network modules on nuScenes.

Encoder	RCSN	PIM	5 Pos.	Aggr.	R@1/5 [%]	Runtime [ms]
(x, y, z)	✓	$\oplus$	✓	✓	77.8 / 81.9	269
(x, y, z)					62.7 / 76.8	150
(x, y, z, RCS)					63.2 / 78.0	172
(x, y, z)	✓				70.7 / 80.8	159
(x, y, z)		$\oplus$			63.1 / 76.5	159
(x, y, z)	✓	$\otimes$			70.9 / 81.0	168
(x, y, z)	✓	$\oplus$			73.3 / **82.2**	167
(x, y, z, RCS)	✓	$\oplus$	✓		73.9 / 80.1	167
(x, y, z)	✓	$\oplus$	✓		**74.3** / 81.6	167

Scan encoder (Encoder) using only point coordinate inputs (x, y, z) and with RCS as an additional channel (x, y, z, RCS), our RCS network (RCSN), our point importance module (PIM), our training strategy with five positive samples per query (5 Pos), and aggregation of seven scans (Aggr.).

tance of each point leads to improved accuracy in place recognition. We carry out the experiments on nuScenes [7] test set from Sec. IV-C. In Tab. III, we evaluate how each component contributes to the final result and how it affects the runtime performance during inference.

The main modules are the scan encoder, which accepts as inputs only point coordinates (x, y, z) or point coordinates with an additional channel for the RCS (x, y, z, RCS), the proposed RCS network (RCSN), and the point importance module (PIM). We also experiment with multiplication $\otimes$, and addition $\oplus$ of the features in Eq. (4). Furthermore, we also test our training strategy where five positive samples are used for each query. We also show how scan aggregation affects the runtime performance.

We can observe how much each component contributes to the final result, with the biggest effect being caused by the RCS network module. We also observe how addition is preferred over multiplication of the PIM module, as it enhances the useful points for place recognition without affecting the remaining point cloud. Moreover, the runtime is minimally affected by the implementation of the RCS network and PIM modules (< 20 ms) but greatly increases with scan aggregation. Adding the RCS as an additional channel leads to a slight improvement in performance compared to the RCS network module. This highlights the importance of using the RCS information in a the context of an entire scan, rather than adding it as an additional feature to each point.

In Tab. IV, we test the influence of the distance parameter δ and loss function weight γ introduced by our point importance estimation module. High radii δ cause wrong correspondences, while low radii lead to not finding any correspondence, resulting in an equal weighting of all points. This demonstrates how the performance can be improved by focusing on those points that are important for place recognition and focusing less on the noise points. Additionally, varying γ shows the influence caused by the point importance estimator module, and how the multi-objective loss from Eq. (10) affects the final result.

TABLE IV: Ablation studies on nuScenes for the radius δ and loss weighting factor γ from our point importance module.

δ [m]	R@1/5/10 [%]	γ	R@1/5/10 [%]
1.0	71.4 / **81.1** / **83.6**	0.00	71.0 / 80.0 / 82.6
2.0	71.5 / 80.2 / 82.2	0.10	71.6 / 80.6 / 83.1
3.0	**71.7** / 80.6 / 83.0	0.50	**73.3** / **82.2** / **84.6**
4.0	70.8 / 80.7 / 82.5	1.00	71.4 / 81.1 / 83.6

V. Conclusion

In this paper, we achieved place recognition using single scans from standard automotive radar sensors, without relying on additional GNSS or odometers. We proposed a novel point-based neural network architecture that encodes local and global information of the scene into a single compressed descriptor. We achieve this by encoding the local information using point convolution and combining it with the scene's RCS data. Additionally, we integrate an additional point importance estimation module that helps the network learn those measurements that are useful for place recognition. We implemented and evaluated our approach on different datasets and provided comparisons to other existing techniques, supporting all claims made in this paper. The experiments suggest that our method achieves high performance for estimating the global location of a car within a map using single automotive radar scans, while keeping a compact scene representation.

References

[1] D. Adolfsson, M. Karlsson, V. Kubelka, M. Magnusson, and H. Andreasson. Tbv radar slam: Trust but verify loop candidates. *IEEE Robotics and Automation Letters (RA-L)*, 8(6):3613–3620, 2023.

[2] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] V. Balntas, E. Riba, D. Ponsa, and K. Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *Proc. of British Machine Vision Conference (BMVC)*, 2016.

[4] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner. The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[5] D. Barnes and I. Posner. Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[6] J. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[7] H. Caesar, V. Bankiti, A. Lang, S. Vora, V. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuScenes: A Multimodal Dataset for Autonomous Driving. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[8] K. Cai, B. Wang, and C.X. Lu. Autoplace: Robust place recognition with single-chip automotive radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[9] D. Casado Herraez, M. Zeller, L. Chang, I. Vizzo, M. Heidingsfeld, and C. Stachniss. Radar-only odometry and mapping for autonomous vehicles. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[10] M.Y. Chang, S. Yeon, S. Ryu, and D. Lee. Spoxelnet: spherical voxel-based deep place recognition for 3d point clouds of crowded indoor spaces. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[11] B. Choi, H. Kim, and Y. Cho. Referee: Radar-based efficient global descriptor using a feature and free space for place recognition. *arXiv preprint arXiv:2403.14176*, 2024.

[12] L. Di Giammarino, I. Aloise, C. Stachniss, and G. Grisetti. Visual Place Recognition using LiDAR Intensity Information. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[13] Y. Fan, X. Du, L. Luo, and J. Shen. Fresco: Frequency-domain scan context for lidar-based place recognition with translation and rotation invariance. In *Proc. of the Intl. Conf. on Control, Automation, Robotics and Vision (ICARCV)*, 2022.

[14] S. Fu, Y. Duan, Y. Li, C. Meng, Y. Wang, J. Ji, and Y. Zhang. Crplace: Camera-radar fusion with bev representation for place recognition. *arXiv preprint arXiv:2403.15183*, 2024.

[15] M. Gadd, D. De Martini, and P. Newman. Look around you: Sequence-based radar place recognition with learned rotational invariance. In *Proc. of the IEEE Symp. on Position, Location and Navigation*, 2020.

[16] M. Gadd, D. De Martini, and P. Newman. Contrastive learning for unsupervised radar place recognition. In *Proc. of the Int. Conf. on Advanced Robotics (ICAR)*, 2021.

[17] M. Gadd and P. Newman. Open-radvlad: Fast and robust radar place recognition. In *Proc. of the IEEE Radar Conference (RaConf)*, 2024.

[18] A. Garcia-Hernandez, R. Giubilato, K.H. Strobl, J. Civera, and R. Triebel. Unifying local and global multimodal features for place recognition in aliased and low-texture environments. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[19] S. Gupta, T. Guadagnino, B. Mersch, I. Vizzo, and C. Stachniss. Effectively detecting loop closures using point cloud density maps. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[20] H. Jang, M. Jung, and A. Kim. Raplace: Place recognition for imaging radar using radon transform and mutable threshold. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[21] B. Jiang and S. Shen. Contour context: Abstract structural distribution for 3d lidar loop detection and metric pose estimation. *arXiv preprint arXiv:2302.06149*, 2023.

[22] A. Kamann, P. Held, F. Perras, P. Zaumseil, T. Brandmeier, and U.T. Schwarz. Automotive radar multipath propagation in uncertain environments. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2018.

[23] G. Kim, Y. Park, Y. Cho, J. Jeong, and A. Kim. Mulran: Multimodal range dataset for urban place recognition. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[24] G. Kim, S. Choi, and A. Kim. Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments. *IEEE Trans. on Robotics (TRO)*, 38(3):1856–1874, 2021.

[25] G. Kim and A. Kim. Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[26] G. Kim, B. Park, and A. Kim. 1-day learning, 1-year localization: Long-term lidar localization using scan context image. *IEEE Robotics and Automation Letters (RA-L)*, 4(2):1948–1955, 2019.

[27] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2015.

[28] J. Komorowski. Minkloc3d: Point cloud based large-scale place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[29] J. Komorowski. Improving point cloud based place recognition with ranking-based loss and large batch training. In *Proc. of the Intl. Conf. on Pattern Recognition (ICPR)*, 2022.

[30] J. Kopp, D. Kellner, A. Piroli, and K. Dietmayer. Fast rule-based clutter detection in automotive radar data. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2021.

[31] F. Kraus, N. Scheiner, W. Ritter, and K. Dietmayer. Using machine learning to detect ghost images in automotive radar. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2020.

[32] P.C. Kung, C.C. Wang, and W.C. Lin. Radar occupancy prediction with lidar supervision while preserving long-range sensing and penetrating capabilities. *IEEE Robotics and Automation Letters (RA-L)*, 7(2):2637–2643, 2022.

[33] X. Li, H. Zhang, and W. Chen. 4d radar-based pose graph slam with ego-velocity pre-integration factor. *IEEE Robotics and Automation Letters (RA-L)*, 8(8):5124–5131, 2023.

[34] Y. Ma, X. Zhao, H. Li, Y. Gu, X. Lang, and Y. Liu. Rolm: Radar on lidar map localization. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[35] M. Milford and G. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2012.

[36] R. Mur-Artal, J.M.M. Montiel, and J.D. Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE Trans. on Robotics (TRO)*, 31(5):1147–1163, 2015.

[37] A. Nayak, D. Cattaneo, , and A. Valada. Ralf: Flow-based global and metric radar localization in lidar maps. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[38] 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)*, 7(2):4961–4968, 2022.

[39] G. Peng, H. Li, Y. Zhao, J. Zhang, Z. Wu, P. Zheng, and D. Wang. Transloc4d: Transformer-based 4d radar place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[40] Ş. Săftescu, M. Gadd, D. De Martini, D. Barnes, and P. Newman. Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[41] S. Siva, Z. Nahman, and H. Zhang. Voxel-based representation learning for place recognition based on 3d point clouds. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[42] T.Y. Tang, D. De Martini, S. Wu, and P. Newman. Self-supervised learning for using overhead imagery as maps in outdoor range sensor localization. *The Intl. Journal of Robotics Research*, 40(12-14):1488–1509, 2021.

[43] 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.

[44] M. Usuelli, M. Frosi, P. Cudrano, S. Mentasti, and M. Matteucci. Radarlcd: Learnable radar-based loop closure detection pipeline. *arXiv preprint arXiv:2309.07094*, 2023.

[45] M.A. Uy and G.H. Lee. Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[46] O. Vysotska and C. Stachniss. Lazy Data Association For Image Sequences Matching Under Substantial Appearance Changes. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):213–220, 2016.

[47] O. Vysotska and C. Stachniss. Effective Visual Place Recognition Using Multi-Sequence Maps. *IEEE Robotics and Automation Letters (RA-L)*, 4:1730–1736, 2019.

[48] H. Wang, C. Wang, and L. Xie. Intensity scan context: Coding intensity and geometry relations for loop closure detection. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[49] X. Wei, I.A. Bârsan, S. Wang, J. Martinez, and R. Urtasun. Learning to localize through compressed binary maps. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[50] R. Weston, S. Cen, P. Newman, and I. Posner. Probably unknown: Deep inverse sensor modelling radar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[51] L. Wiesmann, L. Nunes, J. Behley, and C. Stachniss. Kppr: Exploiting momentum contrast for point cloud-based place recognition. *IEEE Robotics and Automation Letters (RA-L)*, 8(2):592–599, 2022.

[52] T.X. Xu, Y.C. Guo, Z. Li, G. Yu, Y.K. Lai, and S.H. Zhang. Transloc3d: Point cloud based large-scale place recognition using adaptive receptive fields. *Communications in Information and Systems*, 23(1):57–83, 2023.

[53] X. Xu, H. Yin, Z. Chen, Y. Li, Y. Wang, and R. Xiong. Disco: Differentiable scan context with orientation. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):2791–2798, 2021.

[54] H. Yin, X. Xu, S. Lu, X. Chen, R. Xiong, S. Shen, C. Stachniss, and Y. Wang. A Survey on Global LiDAR Localization: Challenges, Advances and Open Problems. *Intl. Journal of Computer Vision (IJCV)*, 2024.

[55] H. Yin, X. Xu, Y. Wang, and R. Xiong. Radar-to-lidar: Heterogeneous place recognition via joint learning. *Frontiers in robotics and AI*, 8:661199, 2021.

[56] J. Yuan, P. Newman, and M. Gadd. Off the radar: Uncertainty-aware radar place recognition with introspective querying and map maintenance. *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[57] 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, 2022.

[58] M. Zeller, D. Casado Herraez, J. Behley, M. Heidingsfeld, and C. Stachniss. Radar tracker: Moving instance tracking in sparse and noisy radar point clouds. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[59] 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. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.