AutoPlace: Robust Place Recognition with Single-chip Automotive Radar

AutoPlace: Robust Place Recognition with Single-chip Automotive Radar

Kaiwen Cai[†], Bing Wang[§], Chris Xiaoxuan Lu[‡*]
[†]University of Liverpool, [§]University of Oxford, [‡]University of Edinburgh

***Abstract*— This paper presents a novel place recognition approach to autonomous vehicles by using low-cost, single-chip automotive radar. Aimed at improving recognition robustness and fully exploiting the rich information provided by this emerging automotive radar, our approach follows a principled pipeline that comprises (1) dynamic points removal from instant Doppler measurement, (2) spatial-temporal feature embedding on radar point clouds, and (3) retrieved candidates refinement from Radar Cross Section measurement. Extensive experimental results on the public nuScenes dataset demonstrate that existing visual/LiDAR/spinning radar place recognition approaches are less suitable for single-chip automotive radar. In contrast, our purpose-built approach for automotive radar consistently outperforms a variety of baseline methods via a comprehensive set of metrics, providing insights into the efficacy when used in a realistic system.**

I. Introduction

By recognizing revisited places when traveling, place recognition is a key enabler to mobile autonomy and plays essential roles in a wide range of downstream tasks such as scene understanding, loop closure detection, localization, etc.

Visual place recognition develops with the prevalence of commercial RGB cameras, which involves handcrafted features [1], [2] and deep learning-based ones [3]–[5]. On the other hand, a LiDAR sensor is usually adopted as an alternative optical sensor for place recognition due to its better robustness in dim and dark environments. While place recognition approaches based on these optical sensors have made considerable progress in the past decade [4]–[8], they still fall short under visual degradation common on the roads (e.g., rain, snow, dust, fog, and direct sunlight). An example is shown in Fig. 1, where the state-of-the-art RGB camera-based place recognition failed to retrieve the correct candidate due to raindrops blocking the camera.

Unlike the above optical sensors operating in the visible spectrum, radar operates in a millimeter-wave frequency band, lending itself a modality robust to scene illumination and airborne obstacles [9]. By taking input as the dense radar spectra/images, recent works [10], [11] have shown the feasibility of place recognition based on *mechanically spinning radar* (e.g., CTS-350X). Despite the impressive performance achieved, spinning radar is known to be bulky and costly [12] and require to be mounted on the roof of the vehicle, nor able to provide the Doppler information. In contrast, *automotive radar* (aka. single-chip millimeter-wave radar) emerge as a low-cost and lightweight alternative that is pervasively embraced by major vehicle manufacturers (e.g., Audi and Ford [13], [14]). As it adopts wave beamforming

*Corresponding author: Chris Xiaoxuan Lu (xiaoxuan.lu@ed.ac.uk)

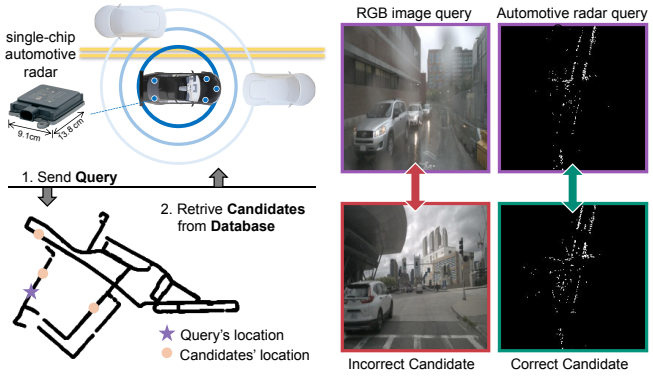

Fig. 1. Place recognition using the single-chip automotive radar: given a query (marked in purple) acquired from the same place on a rainy day [20], the state-of-the-art RGB camera-based place recognition [4] failed to retrieve the correct candidate due to raindrops blocking the camera, while the proposed `AutoPlace` successfully retrived the correct one.

rather than mechanical spinning to scan the environment [15], an automotive radar can measure the point's radial velocity through the Doppler effect as well as fine-grained Radar Cross Section (RCS).

These unique characteristics make automotive radar an attractive sensor for autonomous driving. While prior works use them to estimate the vehicle's (ego-)motion [16], [17] and localization [18], [19], realizing robust place recognition by automotive radar has never been explored and features different challenges. Compared with spinning radars [10], automotive radars' point clouds are significantly noisier, sparser and in much lower resolution [9]. These low-quality point clouds results in unreliable global descriptors that often 'mislead' a place recognition system.

In this work, we exploit the unique characteristics of automotive radar and propose a robust Automotive radar Place recognition approach dubbed `AutoPlace` to address the above challenges. Specifically, our contributions are:

- This paper is the first work that validates the capability of single-chip automotive radar for place recognition.
- We propose a novel place recognition method by fully utilizing the radial velocity and RCS measurement and effectively modeling the spatial and temporal radar points with a compact deep neural network.
- The proposed `AutoPlace` consistently outperforms a variety of competing approaches on the public nuScenes dataset [20], with code avaliable at: https://github.com/ramdrop/AutoPlace.

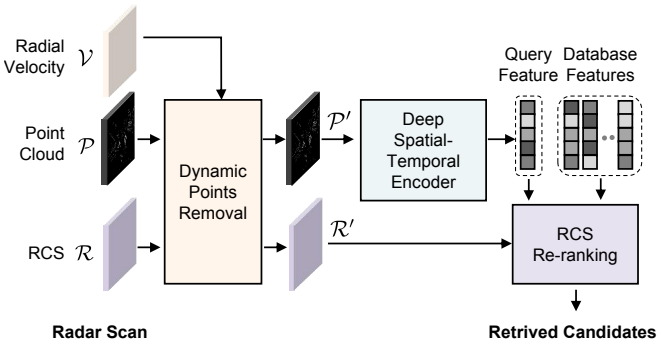

Fig. 2. Overview of AutoPlace: radial velocity, point cloud and RCS measurement of a radar scan are utilized by the proposed *Dynamic Points Removal* method, *Deep Spatial-Temporal Encoder* and *RCS Histogram Re-ranking* method, respectively.

II. Related Work

Place recognition with cameras is an established topic due to the sensor ubiquity, rich information and cost-effectiveness. Early works use handcrafted features and heuristic sequence matching methods [1], [2], while Convolution Neural Networks (CNNs) and attention mechanisms [4], [5] are recently explored. Unlike cameras, LiDAR sensors measure objects with explicit scales. LiDAR place recognition approaches include spatial segments methods [6], [21], point-wise neural networks [7] and 3D convolution networks [8].

Once as a common sensor used on ships [22], mechanically spinning radar has recently been utilized for vehicle place recognition. UnderTheRadar [23] uses intermediate features as global decsriptor for place recognition. RadarSLAM [24] uses M2DP [6] to generate compact descriptors for radar point clouds. KidnappedRadar [10] exploits a variant of NetVLAD [4] as the feature extractor with sophisticated modification to improve rotational invariance.

The closest work to ours is LookAroundYou [25], which improves KidnappedRadar [10] by using the off-the-shelf sequence matching mechanism from SeqSLAM [2] . However, LookAroundYou [25] inevitably inherits the limitations of SeqSLAM [2] in the following aspects [26]: (1) It requires at least two complete trajectories - query trajectory and database trajectory, and they should be aligned with the same number of samples, while our method can work with the queries and databases of any size, and has no dependency on fine-grained sequence alignments. (2) It requires the whole trajectory to be spatially continuous for performing the local contrast enhancement [2], while our method only requires a locally continuous sequence. (3) It heuristically matches sequences, while our method end-to-end trains a sequential matching network. More importantly, rather than uses the bulky spinning radar, our work goes for lightweight automotive radar, by which richer measurements are provided and used for more robust place recognition.

III. Methods

Fig. 2 illustrates the pipeline of the proposed AutoPlace. An automotive radar scan provides a set of measurements, including the *point cloud* $\mathcal{P} = \{(x_i, y_i)|i = 1, 2, .., N\}$, *radial velocity* $\mathcal{V} = \{v_i|i = 1, 2, .., N\}$ and *RCS* $\mathcal{R} = \{r_i|i = 1, 2, .., N\}$, where N is the number of observed points at a scanning instant. When receiving this input, our proposed Dynamic Points Removal (DPR) module first exploits $\mathcal{V}$ to identify and remove dynamic points in $\mathcal{P}$ and $\mathcal{R}$, and produces the refined $\mathcal{P}'$ and $\mathcal{R}'$. Then a Deep Spatial-Temporal Encoder extracts a discriminative feature vector from $\mathcal{P}'$. Finally, during the inference phase, our proposed RCS Histogram Re-ranking (RCSHR) module re-ranks the list of best matching candidates. We now detail the design of each module in what follows.

A. Dynamic Points Removal

It is common that dynamic and stationary objects coexist on a road, such as vehicles, pedestrians, traffic cones and building walls. While static objects are temporally consistent, dynamic ones are not: a moving vehicle may disappear when the vehicle revisits the same place. Consequently, dynamic objects could mislead the place recognition due to landmark inconsistency. To address this challenge, we propose a novel *Dynamic Point Removal* method to generate a dynamic points mask $\mathcal{M}$ considering the two motion status of the ego-vehicle (i.e., the status of the radar sensor):

1) Moving Ego-vehicle: Intuitively, when an ego-vehicle moves, the stationary objects move towards the opposite direction in ego-vehicle's field of view. In this case, the identification of these moving objects should be straightforward by using automotive radar because objects' velocities can be directly measured and returned from such sensors. Nevertheless, automotive radar can only provide the radial velocity for an observed object rather than the full velocity. The measured velocity from the radar can thus be "zero" when an object moves tangentially to the radar. To address this ambiguity of velocity measurement, we propose to distinguish the dynamic points based on radial velocities of all points in a scan rather than that of a single point.

Formally, the points' radial velocity for stationary objects has a sinusoidal function as follows:

$$v_{r,i} = -v_s \cos(\alpha - \theta_i), i = 1, 2, ..., N \quad (1)$$

where v_s is the radar velocity, α heading direction, N the number of points, $v_{r,i}$ and θ_i the radial velocity and azimuth angle of the i^{th} point. Recall that point's radial velocity for moving object does not depend on the ego-vehicle motion and can be arbitrary, which means it does not necessarily fit the Eq. (1) and thus is a outlier. Following [27], we use the Least Square approach and the Random Sample Consensus (RANSAC) [28] algorithm to solve v_s and α and identify outliers, by which the moving points are found.

2) Static Ego-vehicle: One limitation of the above dynamic points removal approach is that it works on the assumption that the ego-vehicle is moving such that the Eq.(1) can be utilized to identify outliers. However, such an assumption does not hold when the ego-vehicle is static. To address this problem, we propose a simple yet effective

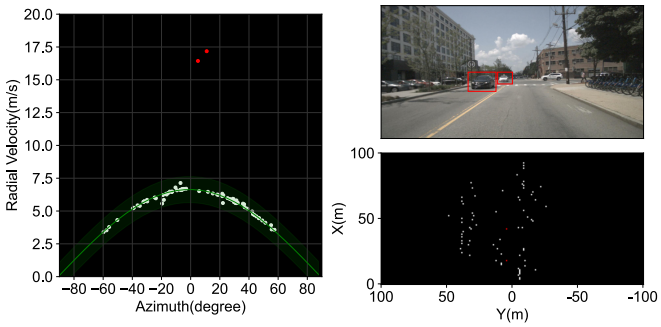

Fig. 3. Remove dynamic points based on radial velocity: the left figure shows the points' radial velocity distribution measured by the front radar in a single radar scan, and the right is the front-view images from the front camera, the bird view radar point cloud from the front radar, respectively. The two dynamic points (the moving cars) are successfully identified by the proposed DPR method and marked in red.

method to identify dynamic points when the ego-vehicle is static. The key intuition behind our method is: at a scanning instant, if most points have zero radial velocities, then the ego-vehicle is very likely to be static. Furthermore, as a nonzero radial velocity can only result from a nonzero velocity, it is reasonable to regard points with nonzero radial velocity as moving points. The implementation flow of our method is described in Algorithm 1.

Fig. 3 shows an example of points' radial velocity distribution, the corresponding bird view radar point cloud, and the identified dynamic points using the proposed DPR method[1].

Input : radial velocity $\mathcal{V} = \{v_{r,i}, \theta_i | i = 1, 2.., N)\}$
veolcity fitting threshold τ
static velocity threshold v_r^τ
percentage of static points threshold p^τ
Output: dynamic points mask
$\mathcal{M} = \{d_i | d_i \in \{0, 1\}, i = 1, 2, .., N\}$ (value 0 means dynamic points)

Initialization: $p \leftarrow 0$, **foreach** d_i **do** $d_i \leftarrow 1$;

for i =*1* **to** N **do**
  if $v_{r,i} < v_r^\tau$ **then** $p \leftarrow p + \frac{1}{N}$;
  else $d_i \leftarrow 0$;
end
`/* static ego-vehicle */`
if $p > p^\tau$ **then** output $\mathcal{M}$;
`/* moving ego-vehicle */`
else
  foreach d_i **do** $d_i \leftarrow 1$;
  execute RANSAC and Least Square algorithms to find outliers indexed as $j_1, j_2, .., j_n$;
  foreach j in $\{j_1, j_2, .., j_n\}$ **do** $d_j \leftarrow 0$;
  output $\mathcal{M}$;
end

Algorithm 1. Dynamic Points Removal.

B. Deep Spatial-Temporal Encoder

Once the dynamic points mask $\mathcal{M}$ is derived using the above DPR, refined $\mathcal{P}'$ and $\mathcal{R}'$ are then obtained by removing dynamic points in $\mathcal{P}$ and $\mathcal{R}$. Next, we introduce the *Deep Spatial-Temporal Encoder* to encode radar point clouds spatial and temporal information for robust place recognition.

[1]More illustrative examples can be found on our website: https://github.com/ramdrop/autoplace.

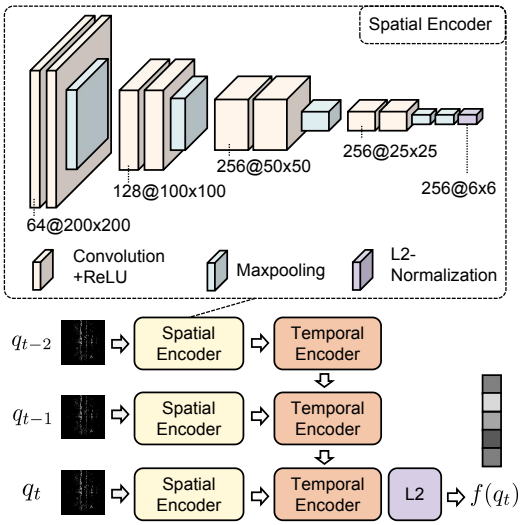

Fig. 4. Overview of the proposed deep spatial-temporal encoder: in the Spatial Encoder diagram, the block color reflects layer type while the block size indicates the layer's output size, which is annotated as channel@width×height.

As the automotive radar in this study provides 2D point clouds, we convert a radar point cloud to a *radar image* by projecting all 2D points to the image panel with the occupied pixel assigned value 1, and the pixels that are not occupied by any points are assigned value 0.

1) Spatial Encoder: To encode the spatial information of a radar image, we propose a convolution-based *spatial encoder* as shown in Fig. 4: all convolutional layers have the same kernel size (3×3) and stride (1×1), followed by a ReLU layer. Max-pooling layers are used to downsample feature map, and all have a kernel size (2×2) and stride (2×2), except the last one has a larger kernel size (3×3) and a stride (3×3). An L2-normalization layer is used at the end of the spatial encoder. The feature map of the last layer has a size of $C \times H \times W$, which can be regarded as C-dimensional features in $(H \times W)$ spatial locations.

2) Temporal Encoder: As point clouds given by automotive radar are noisy and sparse [9], a measured object may come and go at random across consecutive radar images. Such inconsistency, however, makes it hard for the network to learn a consistent feature map even for the same place. To address this, we propose to utilize the temporal information in a series of radar images. The underlying idea here is that the object's inconsistency between consecutive scans can be mitigated by sequential smoothing. Inspired by the recent success of recurrent neural networks in smoothing sequential data [9], [29], we here adopt the single-layer LSTM [30] as the *temporal encoder* subsequent to the aforementioned spatial encoder. Fig. 4 illustrates the overall architecture of the proposed deep spatial-temporal encoder.

3) Loss Function: Similar to NetVLAD [4], the triplet margin loss is used to train the model, which is given by

$$L = \sum_k \max\{d_E(f(q), f(p)) - d_E(f(q), f(n^k)) + m, 0\}$$

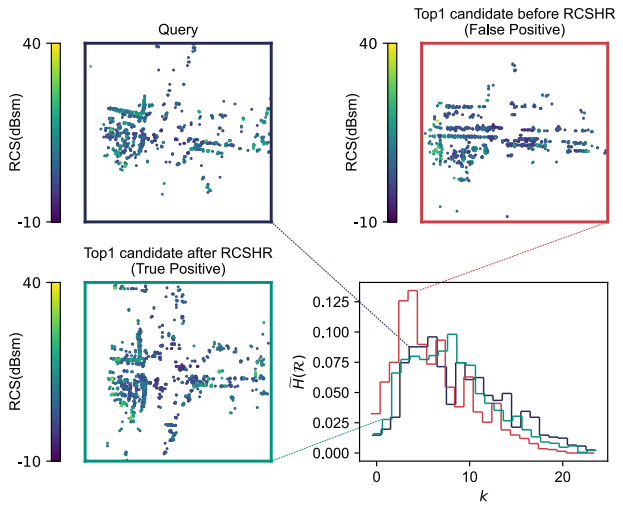

Fig. 5. An example of RCSHR: The lower right figure is the RCS histograms of a query and two candidates, while others are their bird view radar images. Given a query, the retrieved top-1 candidate before RCSHR is a false positive; after performing RCSHR based on their RCS histograms, we identified the true positive candidate. (we use point size larger than 1 pixel for better visualization)

where $f(\cdot)$ denotes the network mapping a radar image to a feature vector, $d_E(\cdot)$ Euclidean distance, q the query sample, p the best positive matching sample, n^k the true negative samples, $m = 0.1$ the predefined margin, and $k = 10$ the number of negative samples. See NetVLAD [4] for more details about the triplet loss function.

C. RCS Histogram Re-ranking

RCS values, as a type of radar output, are finally used in `AutoPlace` to refine the recognition accuracy. RCS is a property of the object's reflectivity, mainly determined by the object's material, size and reflected angle. This measurement provides an additional feature for an object and has proven effective in assisting odometry task [31], [32]. However, the question remains whether and how the RCS can contribute to a place recognition task. To answer this question, we propose a novel method, dubbed *RCS Histogram Re-Ranking* to further improve place recognition accuracy. The intuition is that geometrically-close places should share similar RCS histograms regardless of weather and illumination variances. Therefore, when measuring the similarity of two radar point clouds, we consider not only their feature distance but also RCS histogram distance. Concretely, the steps of deriving the RCS histogram distance are as follows:

For the RCS measurement $\mathcal{R}'$, we first normalize it to range $(0, 1)$. To filter RCS of trivial objects, we only retain RCS within $(b_m, 1)$, where b_m is the lower RCS bound and the bin width b_w is empirically determined from a validation set. Finally, by calculating counts in each bin, the RCS histogram is obtained. For consistency across different scans, the histogram is normalized to have a unit sum over all bins.

Now the problem arises how to measure the similarity of two histograms. As a histogram is an empirical estimate of a probability distribution [33], we can use a wide range of similarity functions for comparing two distributions to measure the similarity of two histograms. In this work we

Input : RCS measurements
$\mathcal{R}'_1 = \{r_{1,i} | i = 1, 2, ..., N_1\}$
$\mathcal{R}'_2 = \{r_{2,i} | i = 1, 2, ..., N_2\}$, b_m, b_w
Output: RCS histogram distance d_R

bin the range $(b_m, 1)$ with equal width b_w;
foreach $\mathcal{R}'$ in $\{\mathcal{R}'_1, \mathcal{R}'_2\}$ **do**
  foreach r in $\mathcal{R}'$ **do**
    $r \leftarrow \frac{r - \min \mathcal{R}'}{\max \mathcal{R}' - \min \mathcal{R}'}$
  end
  calculate counts in each bin and we get
  $H(\mathcal{R}') = \{c_k | k = 1, 2, ..[b_m/b_w]\}$;
  $\widetilde{H}(\mathcal{R}') = \{\frac{c_k}{\sum_k c_k} | k = 1, 2, ..[b_m/b_w]\}$;
end

$d_R = \sum_k \widetilde{H}(\mathcal{R}'_1; k) \log(\frac{\widetilde{H}(\mathcal{R}'_1; k)}{\widetilde{H}(\mathcal{R}'_2; k)})$

Algorithm 2. Calculate RCS Histogram Distance

use KL divergence as the similarity function since it is proven to be the an effective metric among a variety of similarity functions [33]. By denoting RCS histogram distance of two RCS histograms $\widetilde{H}(\mathcal{R}'_1)$ and $\widetilde{H}(\mathcal{R}'_2)$ as d_R, we have:

$$d_R(\widetilde{H}(\mathcal{R}'_1; k), \widetilde{H}(\mathcal{R}'_2; k)) = \sum_k \widetilde{H}(\mathcal{R}'_1; k) \log(\frac{\widetilde{H}(\mathcal{R}'_1; k)}{\widetilde{H}(\mathcal{R}'_2; k)})$$

where k is the bin index. Note that a more similar RCS histogram pair leads to a smaller RCS histogram distance d_R. Algorithm 2 summarises the above procedures. With the derived RCS histograms and top-M candidates retrieved based on their feature distance, a re-rank is performed by holistically considering the combined total distance d_{total}:

$$d_{total}(q, c) = \alpha \cdot d_R(\widetilde{H}(q), \widetilde{H}(c)) + (1 - \alpha) \cdot d_E(f(q), f(c))$$

where α is an adjustable parameter used to balance the two distances, q and c denotes the query sample and the candidate sample. And we empirically set M=100 in this study. An example of RCSHR is shown in Fig. 5 where we can see that the retrieved top-1 candidate without using the RCSHR is only plausibly correct but not really close to the query. After the RCSHR re-rank is applied, the real match pops up as the new top-1 candidate due to their more similar RCS histograms, and consequently smaller d_R over the others.

IV. Experimental Setup

A. nuScenes Dataset

As automotive radar is relatively new, the public dataset of this novel sensor is limited. The newly released nuScenes dataset [20] is the first and only one for large-scale environments with multi-modal sensors, including automotive radar. Since no previous work has been done for place recognition on this dataset, we provide necessary information about the dataset and our data pre-processing pipelines.

There are five radar sensors installed at the front, left, right and back parts of the vehicle, covering a 360° FOV. Each radar works at 77 GHz with 250 m measurement range and 13 Hz capture frequency. The final data is comprised of 1000

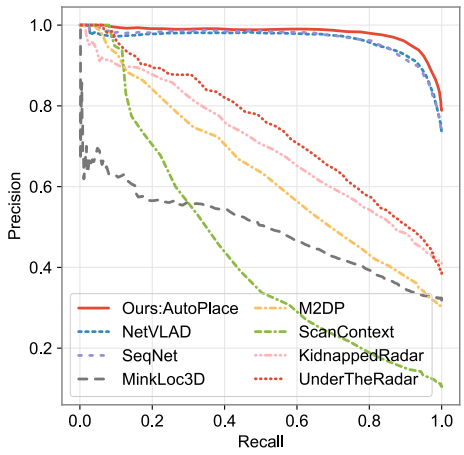

Fig. 6. Precision-recall curve of SOTA methods on the nuScenes dataset.

scenes of 20 s duration from four locations, which is 15 h of driving data (242 km traveled at an average of 16 km h^{-1}).

In order to train and evaluate place recognition methods, a large number of loop closures and diverse road situations are desired. Therefore, we choose to train and evaluate `AutoPlace` on the largest split, *Boston* split, among all the four splits. For the 550 scenes collected in Boston over 120 days, we take the data from the last 15 days as the *validation query set* and the *test query set*, while the rest as the *training set*. The *training set* is further divided into the *database set* and the *training query set*. Specifically,

- The *database set* is created by taking as many places as possible from the *training set*, until a newly added place is no 1 m farther than any existing places in the *database set*.
- The *training query set* is created by substracting the *database set* from the *training set*.
- The *validation query set* : the *test query set*=1 : 4, and they are refined by removing places that have no ground truth true positives in the *database set*.

The final *database set*, *training query set*, *validation query set* and *test query set* contains 6312, 7075, 924 and 3696 radar images, respectively.

B. Implementation Details

Since a single radar point cloud is too sparse to extract useful information due to the low sensing quality, we follow typical data pre-processing of [17], [32] and concatenate the nearest seven radar point clouds to form a denser point cloud. Ground truth ego-motion is used for concatenation, but this could be relaxed by using a simple local pose estimator, e.g., IMU. Besides, as the far-range measurement is less accurate, we crop the radar measurements to retain the points within 100 m from the sensor. When projected to the image plane, each point occupies one pixel. Thus, the converted 2D bird view radar image has a size of 200×200.

After searching hyper-parameters on validation set, we set $\tau = 0.15, v_r^\tau = 1, p^\tau = 0.5$ for DPR, and $b_m = 0.02, b_w = 0.04, \alpha = 0.41$ for RCSHR. For the network training, we use a batch size of 8 and SGD with an initial learning rate of 0.01, momentum 0.9 and weight decay 0.001. We decay the learning rate by 0.5 every 5 epochs. Following the scale of KidnappedRadar [10] and PointNetVLAD [7], we regard places in database that are within the radius=9 m area to the query as true positives, while those are outside the radius=18 m area as true negatives.

TABLE I

PERFORMANCE OF SOTA METHODS ON THE NUSCENES DATASET

Method	Recall@1/5/10	max F_1	AP
MinkLoc3D [8]	31.8/53.6/61.1	0.53	0.49
M2DP [6]	30.3/43.4/48.4	0.58	0.65
ScanContext [21]	10.4/15.3/17.2	0.42	0.45
UnderTheRadar [11]	38.5/53.8/59.1	0.67	0.75
KidnappedRadar [10]	41.0/56.6/61.5	0.65	0.71
NetVLAD [4]	73.1/80.5/82.4	0.92	0.96
NetVLAD+`TE`*[1]	70.8/79.5/81.2	0.90	0.95
SeqNet [36]*	73.3/80.0/82.1	0.92	0.97
Ours: `SE`	73.3/80.0/81.9	0.89	0.96
Ours: `SE+TE`*	76.7/81.7/83.4	0.93	0.97
Ours: `AutoPlace` *[2]	**78.9/83.1/84.3**	**0.94**	**0.98**

[1] * denotes using sequential frames for place recognition.
[2] i.e., `SE+TE+DPR+RCSHR`.

TABLE II

ABLATION STUDY OF AUTOPLACE

SE	TE	DPR	RCSHR	Recall@1/5/10	max F_1	AP
√				73.3/80.0/81.9	0.89	0.96
√	√			76.7/81.7/83.4	0.93	0.97
√		√		75.3/81.4/83.0	0.91	0.96
√			√	73.4/80.3/82.2	0.89	0.96
√		√	√	75.8/82.1/83.8	0.91	0.96
√	√		√	77.7/82.1/83.4	0.93	0.98
√	√	√		77.8/82.3/83.7	0.94	0.98
√	√	√	√	**78.9/83.1/84.3**	**0.94**	**0.98**

C. Evaluation Metrics

We follow four standard metrics in place recognition tasks: recall@N [4], [5], precision-recall curve [34], [35], max F_1 [10] and average precision (AP) [34]. See [34] for details of generating precision-recall curve and AP.

V. RESULTS

A. Comparison with State-Of-The-Art Methods

In what follows, we denote our proposed Spatial Encoder and Temporal Encoder as **`SE`** and **`TE`** for brevity. We compare our approach with the SOTA methods, including:

- Visual place recognition: **NetVLAD** [4], **SeqNet** [36]. We adapt the implementation of the above works to the settings of the nuScenes dataset. To fairly evaluate the effectiveness of our spatial encoder, we also investigated the performance of **NetVLAD+`TE`** by adding our `TE` to the original NetVLAD network.
- LiDAR place recognition: **M2DP** [6], **ScanContext** [21] and **MinkLoc3D** [8]. We feed them the pseudo-3D point clouds by adding a pseudo axis $z = 0$ and then normalize point clouds to range $(-1, 1)$.
- Spinning radar place recognition: **UnderTheRadar** [11] and **KidnappedRadar** [10]. For KidnappedRadar, we convert the radar images from Cartesian coordinates to polar coordinates.

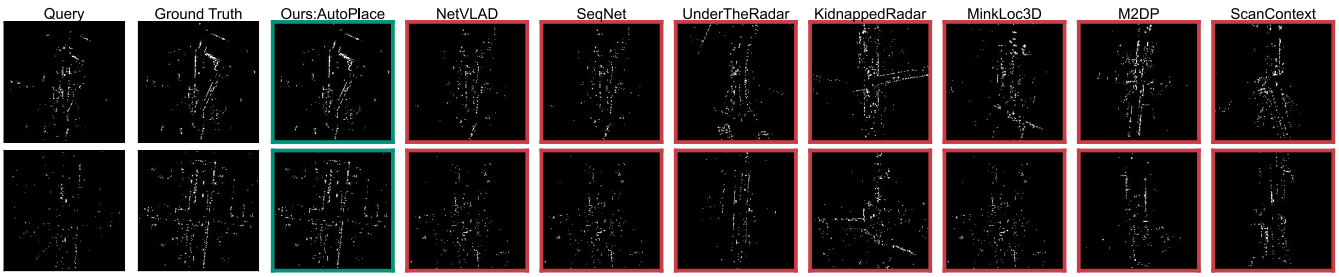

Fig. 7. Qualitative analysis of SOTA methods. The first and second columns show the query radar images and ground truth radar images, and the other columns are the retrieved top 1 candidate via different methods. Green means the retrived candidate is a true positive, while red denotes false positive.

Table. I presents the performance of SOTA methods and our methods. As expected , brute-force applying the place recognition approaches of LiDAR sensors to automotive radar results in inferior performance: ScanContext, M2DP and MinkLoc3D only achieve 10.4%, 30.3% and 31.8% recall@1, respectively. Their failures can be attributed to the reasons that (1) automotive radar point clouds are much sparser than LiDAR point clouds [17], and (2) pseudo 3D point clouds of the automotive radar lack valid information on z axis. These factors make 3D point cloud-based methods ill-suited to automotive radar.

We can also observe from this table that spinning radar place recognition approaches perform slightly better than LiDAR-based methods: KidnappedRadar and UnderTheRadar achieve 41.0% and 38.5% recall@1, respectively, which are still far from being satisfactory. KidnappedRadar takes mechanically spinning radar's spectra as input and performs max-pooling upon the last feature map along the azimuth axis to achieve rotational invariance. However, since automotive radar point clouds are higher-level but less informative than spectra, max-pooling operation on automotive radar's feature map only makes things worse, preventing the network from producing a discriminative radar image descriptor.

Visual place recognition methods, NetVLAD and SeqNet, achieve 73.1% and 73.3% recall@1, respectively. We suppose their relatively good performance results from the strong feature representative capability of VGG16 [37]. Our `SE` surpasses NetVLAD and achieves a comparable performance of the sequence-based method, SeqNet. Notice that NetVLAD+`TE` performs even worse than NetVLAD, this is because NetVLAD is designed to produce high dimensional features (larger than 30k-dimensions in [4]) while LSTM in `TE` works well when fed low dimensional data (less than 10k-dimensions in [9], [38]), such a incompatibility results in its poor performance. In contrast, our `SE` produces compact features and works better with `TE` than NetVLAD. By utilizing all information from automotive radar, `AutoPlace` extends the gap to the runner-up to 5.6%, 0.02 and 0.01 for recall@1, $\max F_1$ and AP. A similar trend can also be observed in Fig. 6 that LiDAR and spinning radar place recognition methods are outperformed by their visual counterparts, NetVLAD and SeqNet. Still, `AutoPlace` exceeds the others by a significant margin.

We also provide qualitative analysis in Fig. 7. As we can see, when the queried scene structure is incomplete (first row) or the point cloud in a query is extremely sparse (second row), competing approaches struggle while `AutoPlace` can still retrieve the correct match.

In summary, the experimental results suggest: (1) existing visual, LiDAR or spinning radar place recognition methods perform unsatisfactorily when being directly applied to automotive radar, and (2) by fully utilizing all the information provided by automotive radar, our `AutoPlace` is superior to all competing methods.

B. Ablation Study

We study each component of `AutoPlace` by evaluating different groups shown in Table. II. It can be observed that

- **SE** (c.f. Sec. III-B.1) alone achieves recall@1=73.3%, which is comparable with NetVLAD.
- **TE** (c.f. Sec. III-B.2) boosts recall@1 by 3.4% when added to `SE`, and decreases it by 3.1% when removed from `AutoPlace`.
- **DPR** (c.f. Sec. III-A) increases recall@1 by 2.0% when added to `SE`, and decreases it by 1.2% when removed from `AutoPlace`.
- **RCSHR** (c.f. Sec. III-C) improves recall@1 by 0.1%, 0.5%, 1.0% and 1.1% when added to `SE`, `SE+DPR`, `SE+TE` and `SE+TE+DPR`, respectively. Improvements vary because `RCSHR` works as a refinement module, bringing more remarkable improvement when the descriptors themselves are more discriminative (i.e., `RCSHR` hardly changes candidates' order when mismatches have small feature distances). This is in accordance with our observation that more discriminative descriptors lead to fewer mismatches between geometrically-close places, and thus, `RCSHR` helps more in distinguishing these places (more illustrative examples are available in our supplementary video).

The results indicate that each component is critical in improving the overall performance, of which the most significant benefit is from `TE`.

VI. Conclusion

Observing that existing place recognition methods ill-suits the emerging automotive radar, we propose `AutoPlace`, a novel place recognition framework by fully exploiting automotive radar's rich measurements. Experimental results show remarkable performance gain of `AutoPlace` on the public nuScenes dataset. Future work will further investigate `AutoPlace` for simultaneous localization and mapping.

REFERENCES

[1] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, June 2008.

[2] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation*. St Paul, MN, USA: IEEE, May 2012, pp. 1643–1649.

[3] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2015, pp. 4297–4304.

[4] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[5] Y. Zhu, J. Wang, L. Xie, and L. Zheng, "Attention-based Pyramid Aggregation Network for Visual Place Recognition," in *Proceedings of the 26th ACM International Conference on Multimedia*, ser. MM '18. New York, NY, USA: Association for Computing Machinery, Oct. 2018, pp. 99–107.

[6] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Daejeon, South Korea: IEEE, Oct. 2016, pp. 231–237.

[7] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep Point Cloud Based Retrieval for Large-Scale Place Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4470–4479.

[8] J. Komorowski, "Minkloc3d: Point cloud based large-scale place recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1790–1799.

[9] C. X. Lu, M. R. U. Saputra, P. Zhao, Y. Almalioglu, P. P. de Gusmao, C. Chen, K. Sun, N. Trigoni, and A. Markham, "milliego: single-chip mmwave radar aided egomotion estimation via deep sensor fusion," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 109–122.

[10] Ş. Săftescu, M. Gadd, D. De Martini, D. Barnes, and P. Newman, "Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4358–4364.

[11] D. Barnes and I. Posner, "Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9484–9490.

[12] 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 *2021 IEEE 24th International Conference on Information Fusion (FUSION)*. IEEE, 2021, pp. 1–8.

[13] adaptive cruise control with stop and go function. [Online]. Available: https://www.audi-technology-portal.de/en/electrics-electronics/driver-assistant-systems/adaptive-cruise-control-with-stop-go-function

[14] Adaptive cruise control. [Online]. Available: hhttps://www.ford.com/technology/driver-assist-technology/adaptive-cruise-control/

[15] C. X. Lu, S. Rosa, P. Zhao, B. Wang, C. Chen, J. A. Stankovic, N. Trigoni, and A. Markham, "See through smoke: robust indoor mapping with low-cost mmwave radar," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 14–27.

[16] P.-C. Kung, C.-C. Wang, and W.-C. Lin, "A Normal Distribution Transform-Based Radar Odometry Designed For Scanning and Automotive Radars," *arXiv:2103.07908 [cs]*, Mar. 2021.

[17] J.-T. Lin, D. Dai, and L. Van Gool, "Depth estimation from monocular images and sparse radar data," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 233–10 240.

[18] M. Rapp, T. Giese, M. Hahn, J. Dickmann, and K. Dietmayer, "A feature-based approach for group-wise grid map registration," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015, pp. 511–516.

[19] F. Schuster, C. G. Keller, M. Rapp, M. Haueis, and C. Curio, "Landmark based radar slam using graph optimization," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 2559–2564.

[20] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[21] G. Kim and A. Kim, "Scan Context: Egocentric Spatial Descriptor for Place Recognition Within 3D Point Cloud Map," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 4802–4809.

[22] 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 *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Paris, 2020.

[23] D. Barnes and I. Posner, "Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9484–9490.

[24] Z. Hong, Y. Petillot, and S. Wang, "Radarslam: Radar based large-scale slam in all weathers," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5164–5170.

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

[26] M. Chancán and M. Milford, "DeepSeqSLAM: A Trainable CNN+RNN for Joint Global Description and Sequence-based Place Recognition," *arXiv:2011.08518 [cs]*, Nov. 2020.

[27] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using Doppler radar," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, Oct. 2013, pp. 869–874.

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

[29] R. Wang, S. M. Pizer, and J.-M. Frahm, "Recurrent neural network for (un-) supervised learning of monocular video visual odometry and depth," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5555–5564.

[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[31] E. Jose and M. Adams, "An augmented state SLAM formulation for multiple line-of-sight features with millimetre wave RADAR," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Edmonton, Alta., Canada: IEEE, 2005, pp. 3087–3092.

[32] Y. Li, Y. Liu, Y. Wang, Y. Lin, and W. Shen, "The Millimeter-Wave Radar SLAM Assisted by the RCS Feature of the Target and IMU," *Sensors*, vol. 20, no. 18, p. 5421, Jan. 2020.

[33] K. Meshgi and S. Ishii, "Expanding histogram of colors with gridding to improve tracking accuracy," in *2015 14th IAPR International Conference on Machine Vision Applications (MVA)*. IEEE, 2015, pp. 475–479.

[34] Y. Hou, H. Zhang, and S. Zhou, "Evaluation of Object Proposals and ConvNet Features for Landmark-based Visual Place Recognition," *Journal of Intelligent & Robotic Systems*, vol. 92, no. 3, pp. 505–520, Dec. 2018.

[35] Z. Chen, L. Liu, I. Sa, Z. Ge, and M. Chli, "Learning Context Flexible Attention Model for Long-Term Visual Place Recognition," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4015–4022, Oct. 2018.

[36] S. Garg and M. Milford, "SeqNet: Learning Descriptors for Sequence-Based Hierarchical Place Recognition," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4305–4312, July 2021.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[38] S. Wang, R. Clark, H. Wen, and N. Trigoni, "End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 513–542, 2018.