mmPlace: Robust Place Recognition With Intermediate Frequency Signal of Low-Cost Single-Chip Millimeter Wave Radar

Chengzhen Meng, Yifan Duan, *Graduate Student Member, IEEE*, Chenming He, Dequan Wang, Xiaoran Fan, and Yanyong Zhang, *Fellow, IEEE*

***Abstract*—Place recognition is crucial for tasks like loop-closure detection and re-localization. Single-chip millimeter wave radar (single-chip radar in short) emerges as a low-cost sensor option for place recognition, with the advantage of insensitivity to degraded visual environments. However, it encounters two challenges. Firstly, sparse point cloud from single-chip radar leads to poor performance when using current place recognition methods, which assume much denser data. Secondly, its performance significantly declines in scenarios involving rotational and lateral variations, due to limited overlap in its field of view (FOV). We propose mmPlace, a robust place recognition system to address these challenges. Specifically, mmPlace transforms intermediate frequency (IF) signal into range azimuth heatmap and employs a spatial encoder to extract features. Additionally, to improve the performance in scenarios involving rotational and lateral variations, mmPlace employs a rotating platform and concatenates heatmaps in a rotation cycle, effectively expanding the system's FOV. We evaluate mmPlace 's performance on the milliSonic dataset, which is collected on the University of Science and Technology of China (USTC) campus, the city roads surrounding the campus, and an underground parking garage. The results demonstrate that mmPlace outperforms point cloud-based methods and achieves 87.37% recall@1 in scenarios involving rotational and lateral variations.**

***Index Terms*—Localization, recognition.**

I. Introduction

PLACE recognition plays a vital role in various fields [1], such as robotics, autonomous vehicles, augmented reality, and more. The main goal of place recognition is to identify previously visited locations based on sensor data and match them with a pre-built map database. For example, in Simultaneous Localization and Mapping (SLAM), place recognition plays an essential role in loop-closure detection, which helps correct the accumulated error in the robot's estimated trajectory. Moreover, in long-term navigation, place recognition assists with re-localization, enabling the robot to determine its position within the map after an extended period or after being temporarily lost.

TABLE I
Comparison of Different Radar Place Recognition Systems

System	Oxford [2]	nuScenes [3]	Ours w/o RP[1]	**Ours w/ RP[2]**
Type	mechanical	single-chip	single-chip	**single-chip**
Product	CTS350-X	ARS408 × 5	AWR1642	**AWR1642 + RP**
Cost($)	40 000	520 × 5	299	**299 + 90**
Point	1000	250 × 5	200	**200**
FOV(°)	360	360	120	**300**

[1] w/o RP: without the rotating platform.
[2] w/ RP: with the rotating platform.
The bold entities illustrate the low-cost of our system.

Camera and LiDAR are currently the dominant sensors for place recognition [4]. However, due to they are both optical sensors, their performance degrades severely in degraded visual environments, such as fog, rain, and snow [5], [6], [7]. On the other hand, the radar exhibits insensitivity to degraded visual environments due to its longer (than vision) wavelength ($\lambda \approx 4$ mm). Specifically, the radar utilizes the millimeter-wave signal with a wavelength larger than the small particles in fog, rain, and snow, enabling easy pass through raindrops and snowflakes [1], [5], [8].

Currently, radars used in place recognition systems [9] can be categorized into two types: mechanical and single-chip. As shown in Table I, although single-chip radar has fewer point clouds and a smaller FOV than mechanical radar, it offers the advantage of being much more affordable. However, leveraging a single-chip radar for place recognition presents two following challenges. Firstly, the point cloud data of a single-chip radar is sparse. Consequently, it encounters performance degradation when incorporated into current place recognition methods [10], [11], [12], [13], [14] that rely on the dense point cloud. Secondly, the restricted FOV of the single-chip radar exhibits limited overlap between the current query data and the candidate data stored in the pre-built map database, particularly in scenarios with rotational and lateral variations. Consequently, the performance of the single-chip radar place recognition noticeably degrades in scenarios involving rotational and lateral variations.

Manuscript received 4 November 2023; accepted 5 March 2024. Date of publication 14 March 2024; date of current version 17 April 2024. This letter was recommended for publication by Associate Editor D. Schulz and Editor S. Behnke upon evaluation of the reviewers' comments. This work was supported by the National Natural Science Foundation of China under Grant 62332016. *(Corresponding author: Yanyong Zhang.)*

Chengzhen Meng, Yifan Duan, Chenming He, and Dequan Wang are with the School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China (e-mail: czmeng@mail.ustc.edu.cn; dyf0202@mail.ustc.edu.cn; hechenming@mail.ustc.edu.cn; wdq15588@mail.ustc.edu.cn).

Xiaoran Fan is with the Technology Directions Office (TDO) of Google, Mountain View, CA 94043 USA (e-mail: ox5bc@winlab.rutgers.edu).

Yanyong Zhang is with the School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China, and also with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230026, China (e-mail: yanyongz@ustc.edu.cn).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2024.3377562, provided by the authors.

Digital Object Identifier 10.1109/LRA.2024.3377562

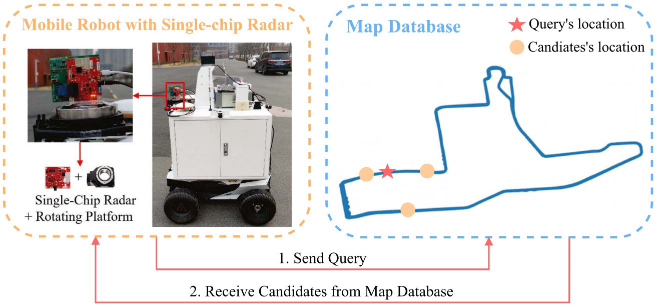


Fig. 1. Our single-chip radar place recognition system. It finds the same location in a pre-built map database based on the given query data.

In this letter, we propose mmPlace, a robust place recognition system based on a low-cost single-chip radar. As shown in Fig. 1, mmPlace identifies previously visited locations from a pre-built map database based on the single-chip radar. To begin with, mmPlace sets out from the IF signal, which is the raw data of single-chip radar for generating the point cloud. Specifically, mmPlace employs heatmap generation and feature extraction on the IF signal. Heatmap generation estimates range and angle to create the range azimuth heatmap, while feature extraction generates the place descriptor by applying a spatial encoder on the heatmap. Subsequently, mmPlace compares the similarity between the generated place descriptor and the place descriptors in the pre-built map database to recognize the current location. Moreover, mmPlace proposes to employ a rotating platform and concatenates heatmaps in a rotation cycle. This method not only effectively compensates for the antenna gains but also significantly enhances the FOV of the single-chip radar place recognition system, enhancing the system's performance in scenarios with rotational and lateral variations.

Although there are some datasets available for the IF signal [15], [16] of single-chip radar *or* for the place recognition [2], [3], we are unable to find a dataset specifically focusing on the IF signal of single-chip radar for place recognition. As a result, we establish a dataset named milliSonic for single-chip radar place recognition, incorporating data from the USTC campus, the city roads surrounding the campus, and an underground parking garage. Subsequently, we conduct a performance evaluation of mmPlace on this self-collected milliSonic dataset.

In summary, our contributions are as follows:

- We propose mmPlace, a robust place recognition system based on a low-cost single-chip radar. This system transforms the IF signal into the range azimuth heatmap and employs a spatial encoder to extract heatmap features for place recognition, outperforming both point cloud-based and point cloud image-based methods.
- Our mmPlace proposes to employ a rotating platform and concatenate heatmaps based on the relative pose in a rotation cycle, achieving 87.37% recall@1 in scenarios encompassing rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters.
- We collect a dataset, called milliSonic,[1] for the single-chip radar place recognition on the USTC campus, the city roads surrounding the campus, and an underground parking garage.

II. Related Work

In this section, we review the related work on place recognition, which can be classified into three categories based on the sensors: camera, LiDAR, and radar.

A. Visual Place Recognition

Visual place recognition is the most investigated technique in the place recognition area because of the camera's ubiquity, rich information, and cost-effectiveness. Initially, Cummins et al. [17] propose a non-learning method called FAB-MAP that uses SIFT features to construct a Bag-of-visual-words (Bow) architecture for place recognition. However, learning-based place recognition methods have become more prominent with the advent of learning-based feature extraction [18], [19]. For example, Arandjelovic et al. [20] introduce NetVLAD, a generalized VLAD layer that enhances the generalization ability of visual place recognition. Building upon NetVLAD, Hausler et al. [21] develop Patch-NetVLAD, which combines the benefits of local and global descriptor techniques by deriving patch-level features from the VLAD layer that are highly invariant to translation and rotation changes.

B. LiDAR-Based Place Recognition

Unlike cameras, LiDAR can capture the 3D geometric structure of the surrounding environment using laser beams. Kim et al. [10] propose ScanContext, a rotation-invariant 3D place descriptor that directly records the 3D structure of visible space. As an extension of ScanContext, Kim et al. [11] develop ScanContext++, a generic descriptor that is robust to both rotation and translation. With the emergence of learning-based 3D feature extraction, Vidanapathirana et al. [12] introduce LoGG3D-Net, which employs a local consistency loss to guide the network to learn the consistent local features.

C. Radar-Based Place Recognition

Unlike cameras and LiDAR, radar operates at much lower frequencies (GHz), making them insensitive to degraded visual environments such as rain, dust, fog, and direct sunlight. Hong et al. [13] develop RadarSLAM by directly employing the LiDAR place recognition method M2DP [22]. Barnes et al. [14] introduce a self-supervised framework for odometry estimation and use an intermediate feature as a global descriptor for place recognition. Cautoplai et al. [6] propose AutoPlace, which extracts spatial-temporal features from the radar point cloud for place recognition, utilizing five single-chip radars. These methods, which utilize the point cloud, involve two types of input: one directly using the point cloud and the other projecting it onto a fixed-size bird's-eye-view image (called point cloud image). These methods are effective with mechanical radar such as CTS350-X, or when using five single-chip radars like ARS408. However, their performance significantly degrades when applied to a low-cost single-chip radar, such as AWR1642, which is the single-chip radar utilized in our mmPlace and is generally more budget-friendly (as shown in Table I).

III. Method

The overview of mmPlace is illustrated in Fig. 2. The single-chip radar provides the raw data IF signal as input to mmPlace. The heatmap generation module performs range estimation and

[1] The dataset and its related code are released here: https://github.com/TC-MCZ/mmPlace

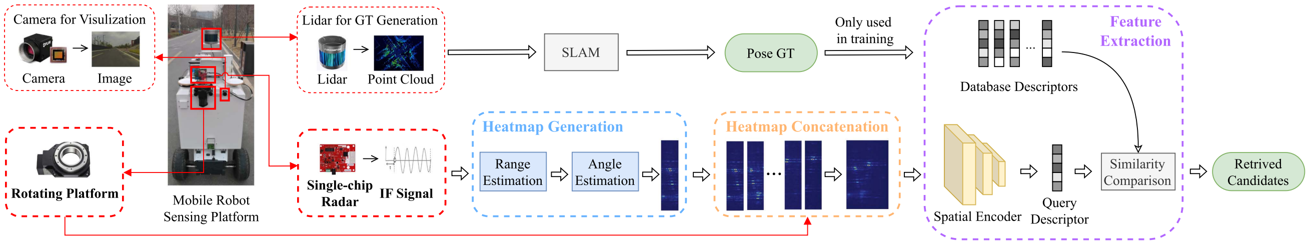


Fig. 2. mmPlace system consists of three main components: (1) heatmap generation, (2) feature extraction, and (3) heatmap concatenation. The heatmap generation module performs range estimation and angle estimation on the IF signal to generate the range azimuth heatmap. The feature extraction module employs a spatial encoder on the heatmap to produce the place descriptor. The heatmap concatenation module employs a rotating platform and concatenates the heatmaps over a full rotation cycle.

angle estimation on the IF signal to generate the range azimuth heatmap, which is discussed in detail in Section III-A. Next, the feature extraction module employs a spatial encoder on the heatmap to produce the place descriptor, which is explained in Section III-B. Subsequently, mmPlace compares the similarity between the place descriptor and the descriptors in the pre-built map database to recognize the current place. Additionally, mmPlace deploys a rotating platform and concatenates the heatmaps over a full rotation cycle, as described in Section III-C.

A. Range Azimuth Heatmap Generation

The IF signal is the raw data of the single-chip radar for generating the radar point cloud [23]. However, it is important to note that the IF signal lacks spatial information and is unsuitable for subsequent feature extraction. Additionally, the sparsity of the single-chip radar point cloud also hinders its utility in feature extraction. Therefore, mmPlace performs sequential range and angle estimations on the IF signal to generate the range azimuth heatmap [24], which contains richer information than the point cloud and encompasses spatial details not present in the IF signal.

Range Estimation: The distance d between the object and the single-chip radar can be obtained through the range estimation [25]. The formula for calculating the distance d is given by:

$$f_{IF} = S\tau = \frac{S2d}{c} \Rightarrow d = \frac{f_{IF}c}{2S}, \tag{1}$$

where f_{IF} is the frequency of the IF signal, S is the slope of the millimeter wave chirp frequency change, τ is the round trip time, and c is the speed of light.

Angle Estimation: The angle of arrival (AoA) θ between the object and the single-chip radar can be determined through the angle estimation [25]. The formula for calculating the AoA θ is given by:

$$\omega = \frac{2\pi \triangle d}{\lambda} = \frac{2\pi l \sin(\theta)}{\lambda} \Rightarrow \theta = \sin^{-1}\left(\frac{\lambda\omega}{2\pi l}\right), \tag{2}$$

where ω is the phase difference, l is the distance between the two receiving antennas, and λ is the wavelength.

Our mmPlace sequentially performs range estimation and angle estimation on the sampled IF signal (I). The formula for calculating the range azimuth heatmap (H) is as follows:

$$H = \left| \sum_{j=1}^{N_C} \left(\underset{k=1}{\overset{N_R}{F}} \left(\underset{i=1}{\overset{N_S}{F}} \left(I(i,j,k) \right) \right) \right) \right|,$$
$$I \in C^{N_S \times N_C \times N_R}, H \in R^{N_S \times N_R}, \tag{3}$$

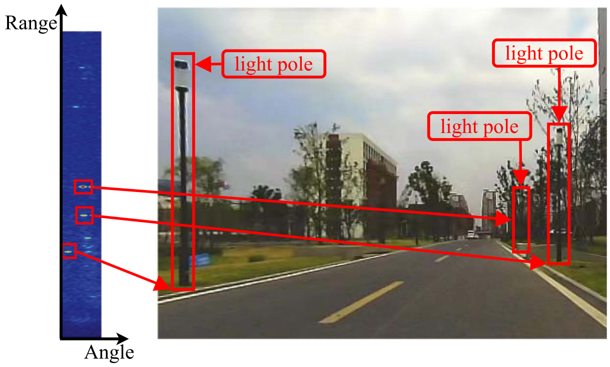


Fig. 3. Visualization of the range azimuth heatmap. The heatmap shows the location of objects and indicates the power of the signal reflected back from them. As an example, the bright spots in the heatmap correspond to the light poles seen in the photo.

where $F(\cdot)$ represents the Fast Fourier Transform (FFT), N_S is the analog-to-digital (ADC) sampling rate, N_C is the number of chirps and N_R is the number of antennas. Additionally, the range azimuth heatmap (H) can be resized by cropping the N_S dimension and interpolating the zeros in the N_R dimension of the IF signal (I).

The range azimuth heatmap encompasses important spatial data. As illustrated in Fig. 3, each cell within the heatmap represents the strength of the reflected signal from an object at a particular range and angle. Much like the visual image, the range azimuth heatmap possesses the capability to detect objects such as light poles.

B. Spatial Encoder-Based Feature Extraction

Taking into account the similarity between the range azimuth heatmap and the Bird's Eye View image, mmPlace employs a spatial encoder to perform feature extraction from the heatmap, resulting in the generation of the place descriptor. The spatial encoder is shown in Fig. 4. Initially, the heatmap consists of two dimensions: range and angle, and the convolution kernel is well-suited for extracting information from this two-dimensional space. Therefore, mmPlace uses four layers of the convolution kernels with a (3×3) kernel size and a (1×1) stride to extract information from the heatmap. Subsequently, for the extraction and synthesis of relevant information, mmPlace utilizes a max-pooling kernel after the convolution kernel layer. Given the distinct range and angle resolutions in the range azimuth heatmap, max-pooling kernels with different sizes are applied: 4 in the range dimension and 2 in the angle dimension.

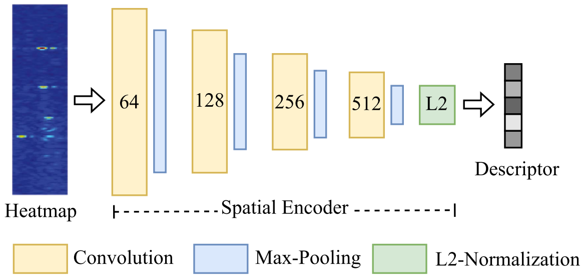


Fig. 4. Overview of the spatial encoder. After obtaining the range azimuth heatmap, the spatial encoder performs feature extraction on the heatmap to obtain the place descriptor.

After performing feature extraction on the range azimuth heatmap and obtaining the place descriptor, mmPlace compares the similarity between the generated place descriptor and the descriptors stored in the pre-built map database to recognize the current place. For enhanced retrieval speed within the map database, we utilize the Faiss [26] library.

In addition, we perform supervised training on the spatial encoder, utilizing ground truth data from the state-of-the-art LiDAR SLAM [27]. In this training process, we utilize the triplet margin loss [28] to train the spatial encoder. This ensures that locations closer in the real world are also closer in the feature space, while locations further away in the real world are correspondingly distant in the feature space. The calculation of the loss function is as follows:

$$L = \sum_{j} l\left(\min_{i}\left(d_E\left(f(q), f(p^i)\right)\right) - d_E\left(f(q), f(n^j)\right) + \alpha\right), \tag{4}$$

where $l(\cdot)$ is the hinge loss($l(x) = \max(x, 0)$), $d_E(\cdot)$ is the Euclidean distance, p^i represents the positive samples, and n^j represents the negative samples.

C. Heatmap Concatenation Within a Rotation Cycle

Accurate place recognition is a challenging task for single-chip radar in scenarios with rotational and lateral variations. This challenge arises from the variability of antenna gains in different directions and the limited FOV of the single-chip radar. In particular, as the orientation of the single-chip radar changes, there are fluctuations in the signal strength reflected back from an object. In addition, there is only a limited FOV overlap of the single-chip radar between the current query data and the candidate data stored in the pre-built map database in scenarios with rotational and lateral variations. As a result, the performance of the single-chip radar place recognition experiences a noticeable degradation under such conditions, as shown in Table V.

To address this challenge, as shown in Fig. 5, we propose to employ a rotating platform to capture single-chip radar data from varying angles and concatenate the heatmaps from a complete rotation cycle. For instance, if the rotational velocity is 150 degrees per second, and the frame rate of the single-chip radar is 10 Hz, we have 12 heatmaps within a rotation cycle. Therefore, the heatmap concatenation not only can effectively compensate for antenna gains but also significantly enhances the FOV of the single-chip radar place recognition system, expanding it to 300 degrees. Thus, mmPlace also has a larger FOV overlap in scenarios involving rotational and lateral variations.

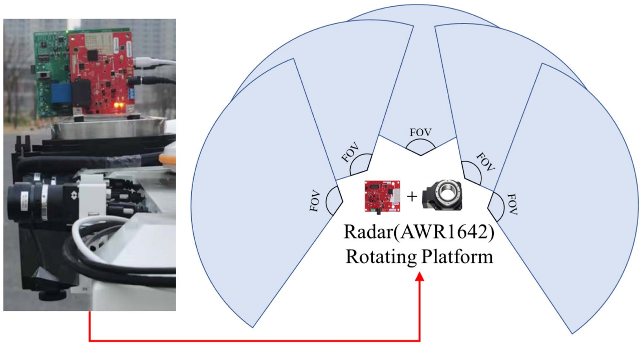


Fig. 5. Radar is deployed on a rotating platform. The rotating platform rotates horizontally over 180 degrees at a speed of 150 degrees per second, which effectively captures data from multiple angles.

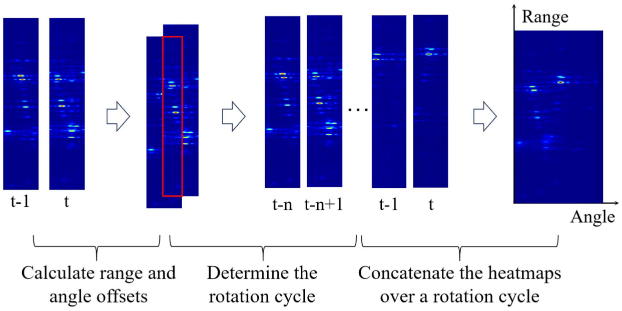

Range
Angle
t-1
t
t-n
t-n+1
...
t-1
t
Calculate range and angle offsets
Determine the rotation cycle
Concatenate the heatmaps over a rotation cycle

Fig. 6. Process of range azimuth heatmap concatenation. Our mmPlace concatenates heatmaps across a rotation cycle, considering both range and angle offsets. This results in a concatenated heatmap with an expanded FOV of up to 300 degrees.

Next, we discuss how to concatenate the range azimuth heatmaps within a cycle, which requires precise pixel-level alignment. The rotating platform doesn't consistently maintain a constant speed because it undergoes periodic starts and stops during the rotation. As a result, directly concatenating heatmaps at fixed step intervals can lead to a significant accumulation of concatenation errors. Therefore, mmPlace concatenates the heatmaps based on the relative pose between neighboring frames, effectively eliminating the cumulative concatenation errors. This method takes into consideration both the range offset (r_{offset}) and the angle offset (a_{offset}) for improved accuracy. The process of heatmap concatenation is shown in Fig. 6. To begin with, we calculate the range and angle offsets between neighboring heatmap frames by minimizing the cosine similarity in their overlapping regions. The formula is expressed as follows:

$$r_{offset_{[t]}}, a_{offset_{[t]}} = \arg\min_{r,a} s\left(o\left(H_{t-1}, H_t(r, a)\right)\right), \tag{5}$$

where H is the range azimuth heatmap, $H(r, a)$ means H translates r and a pixels along the horizontal axis and the vertical axis respectively, $o(\cdot)$ represents the overlap area of H_{t-1} and $H_t(r, a)$, $s(\cdot)$ computes the cosine similarity of the overlap area of H_{t-1} and $H_t(r, a)$, and $r_{offset_{[t]}}$ and $a_{offset_{[t]}}$ are the estimated results.

Secondly, the angle offset (a_{offset}) possesses a positive value when the rotating platform undergoes clockwise rotation and a negative value during counterclockwise rotation. Therefore, by assessing the sign of the angular offset, we can ascertain the direction of rotation in the current frame. This enables us to identify a complete rotation cycle, where the angular offset's sign remains constant throughout that cycle. The formula is presented as follows:

$$a_{offset_{[i-n]}} \oplus a_{offset_{[i-n+1]}} \oplus \cdots \oplus a_{offset_{[i]}} = 0, \qquad (6)$$

where n represents the number of heatmaps for a full rotation cycle, and $\oplus$ is used to determine whether the signs of the operands on both sides of $\oplus$ are equal. If they are equal, the result is 0; if they differ, the result is 1. When conducting chained calculations, determine whether the signs of all operands involved in the computation are equal.

Finally, considering both the range and angle offsets, we concatenate the heatmap for the entire rotation cycle in a clockwise direction. The formula is presented as follows:

$$H' = \bigcup_{i=1}^{n} H_i \left(\sum_{0}^{i} r_{offset_{[t]}}, \sum_{0}^{i} a_{offset_{[t]}} \right), \qquad (7)$$

The resulting heatmap (H'), derived from the concatenation of multiple angles, not only compensates for antenna gains but also extends the system's FOV, thereby significantly enhancing the system's performance in scenarios involving rotational and lateral variations.

It is worth noting that the rotating platform is quite affordable at $90. As shown in Table I, the rotating platform is significantly cheaper than a mechanical millimeter-wave radar and costs less than one-third of AWR1642, the single-chip radar used in mmPlace.

IV. Experiments

In this section, we present the performance evaluation results of mmPlace. Firstly, due to the absence of publicly available place recognition datasets that encompass the single-chip radar IF signal, we collect a dataset on the USTC campus, the city roads surrounding the campus, and an underground parking garage, called milliSonic, as presented Section IV-A. Next, we provide the implementation details in Section IV-B. Following that, we compare mmPlace 's performance with the point cloud-based and point cloud image-based methods in Section IV-C. Then we present the performance of mmPlace utilizing heatmap concatenation in rotational and lateral variation scenarios in Section IV-D.

A. Our Millisonic Dataset

The existing publicly available radar datasets are not suitable for our mmPlace. Indeed, Oxford [2] and nuScenes [3] don't include the IF signal, and RADIal [15] and RAMP-CNN [16] are designed specifically for object detection. Consequently, we collect the milliSonic dataset on the USTC campus, the city roads surrounding the campus, and an underground parking garage for the evaluation of mmPlace.

For data collection, we employ a mobile robot sensing platform. As shown in Fig. 2, this platform is equipped with a rotating platform, a TI AWR1642 single-chip radar sensor,[2] a LUSTER FLRA camera sensor,[3] and a RoboSense RS-32 LiDAR sensor.[4] Thus, our milliSonic dataset includes the single-chip radar IF signal, the LiDAR point cloud, and the camera image. The LiDAR data offers precise ground truth position information via LiDAR SLAM [27], while the camera data enables observation of the robot's surroundings. Furthermore, this dataset has the potential to facilitate future research endeavors, such as multi-sensor place recognition and single-chip radar SLAM, among others.

TABLE II
Statistics of Millisonic Dataset

Seq	Frame	Len(m)	Rot($^\circ$)	Lat(m)	Area	Other
0	23 201	3200.34	0–10	0–1	campus	w/o RP[1]
1	20 528	2706.96	0–10	0–1	campus	w/o RP[1]
2	31 417	4259.45	0–10	0–1	campus	w/o RP[1]
3	21 048	2756.76	0–10	0–1	city	w/o RP[1]
4	8253	809.37	0–10	0–1	parking	w/o RP[1]
5	42 351	3976.59	0–40	0–3	campus	w/o RP[1]
6	5055	539.97	0–40	0–3	campus	w/ RP[2]
7	12 684	1292.33	0–40	0–3	campus	w/ RP[2]
8	11 911	764.24	0–40	0–3	parking	w/ RP[2]
9	12 018	425.30	0–40	0–3	parking	w/ RP[2]

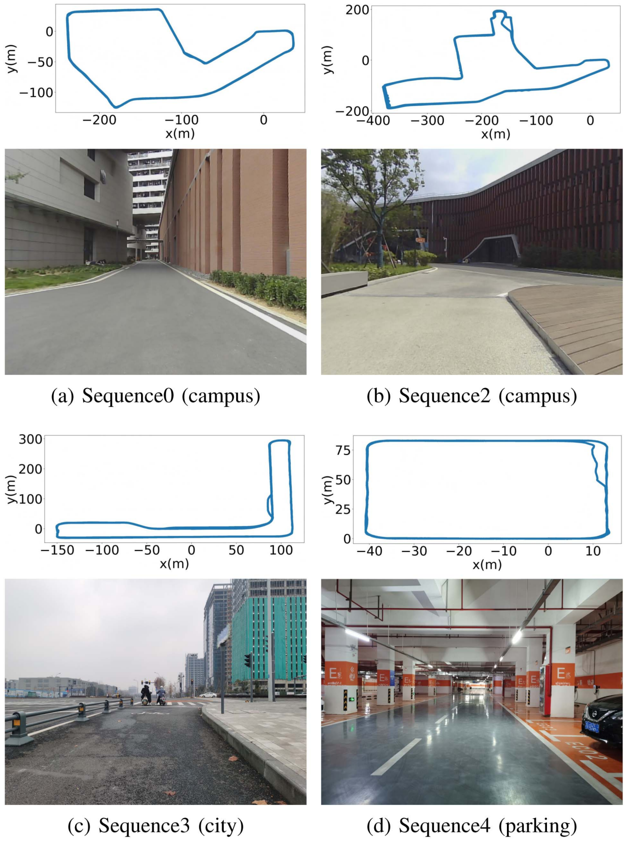
[1] w/o RP: without the rotating platform.
[2] w/ RP: with the rotating platform.

As shown in Table II, the milliSonic dataset comprises ten sequences, encompassing a total travel distance of 20,731 meters and comprising 188,466 frames of data. The dataset is gathered across the USTC campus, the city roads surrounding the campus, and an underground parking garage. Furthermore, as demonstrated in Fig. 7, we present data collection routes and scenarios for the four sequences in the dataset. In the subsequent experiments, we employ sequence 0 for training, while sequence 1 to 9 are utilized for testing. Notably, data collection routes for sequence 1 and 2 exhibit a 30% overlap with sequence 0, while routes for sequence 3 to 9 have no overlap with sequence 0. Additionally, sequence 0 to 4 encompass rotational variations from 0 to 10 degrees and lateral translation variations from 0 to 1 m, while sequence 5 to 9 involve rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters. Sequence 0 to 5 collect without the rotating platform, whereas sequence 6 to 9 utilize the rotating platform. It's worth noting that sequence 7 includes data collected both with the rotating platform (the first 6700 frames) and without the rotating platform (the last 5984 frames).

B. Implementation Details

The spatial encoder of mmPlace is implemented using the PyTorch framework and trained on an Nvidia RTX 3060 GPU. For the network training, we use a batch size of 16 and employ SGD with an initial learning rate of 0.01, momentum of 0.9, and weight decay of 0.001. The learning rate is decayed by 0.5 every 5 epochs, and training continues until convergence, up to a maximum of 50 epochs. Furthermore, following the scale of AutoPlace [6] and KidnappedRadar [7], we consider places in the database within a radius of 3 meters from the query as true positives, while those outside a radius of 18 meters are considered true negatives. Additionally, aligning with the

[2] https://www.ti.com/tool/AWR1642BOOST
[3] https://lusterinc.com/product/series1552.html
[4] https://www.robosense.cn/rslidar/RS-LiDAR-32



(a) Sequence0 (campus) (b) Sequence2 (campus)

(c) Sequence3 (city) (d) Sequence4 (parking)

Fig. 7. Data collection routes and scenarios for the four sequences in the dataset. We collect the dataset on the USTC campus, the city roads surrounding the campus, and an underground parking garage.

AutoPlace [6], we utilize 1 positive sample and 10 negative samples for (4).

C. Evaluation in Normal Scenarios With IF Signal Data

In this subsection, we evaluate mmPlace in normal scenarios (sequence 1 to 4) involving rotational variations from 0 to 10 degrees and lateral translation variations from 0 to 1 m. We begin by comparing mmPlace with point cloud-based and point cloud image-based methods. Subsequently, we evaluate mmPlace 's performance under different range azimuth heatmap sizes.

The existing radar place recognition methods [6], [7], [10], [12], [22] utilizing point cloud employ two types of input, one directly using the point cloud and the other using the point cloud image, as discussed in Section II-C. To compare with these methods that rely on point cloud or point cloud image, we also process the IF signal to generate these two types of data. Firstly, we convert the IF signal collected from the single-chip radar into 2D point clouds with the method proposed in [29]. Next, we generate the pseudo-3D point clouds by adding a pseudo-axis $z = 0$. Finally, we obtain the point cloud images by projecting the 2D points onto the image panel. Occupied pixels are assigned a value of 1, while unoccupied pixels are assigned a value of 0. Moreover, considering the sparse nature of the point cloud from the single-chip radar, we follow the data processing approach in [30] by concatenating 5 frames of the point cloud to generate a denser point cloud.

IF Signal vs Point Cloud: We compare mmPlace with current point cloud-based [10], [12], [22] and point cloud image-based [6], [7] methods in sequence 1 to 4. These sequences encompass rotational variations from 0 to 10 degrees and lateral translation variations from 0 to 1 m. The evaluation of these methods employs the standard metrics for place recognition, including recall@N [20], maxF_1 [7].

As shown in Table III, our IF signal-based mmPlace outperforms both point cloud-based and point cloud image-based methods significantly. Specifically, mmPlace achieves a 39.35% and 30.93% higher recall@1 than ScanContext (the best point cloud-based method) and AutoPlace (the best point cloud image-based method), respectively. Our mmPlace achieves up to 88.39% recall@1, while ScanContext and AutoPlace only achieve 49.04% and 57.46% recall@1, respectively. While point cloud-based and point cloud image-based methods prove effective with mechanical radars like CTS350-X or when employing five single-chip radars such as ARS408, their performance diminishes when applied to a low-cost single-chip radar. The poor performance of these methods is attributed to the sparse point cloud data of a low-cost single-chip radar, which makes it difficult to extract valid scenario information for place recognition. Whereas, mmPlace makes full use of the information-rich IF signal data for place recognition, thus outperforming other point could-based and point cloud image-based methods significantly, as discussed in Section III-A and Section III-B. This also indicates that point cloud-based and point cloud image-based methods are not suitable for low-cost single-chip radar, such as AWR1642, which is the radar utilized in our mmPlace and is generally more budget-friendly (as shown in Table I).

Different Heatmap Sizes: We evaluate the effects of different range azimuth heatmap sizes in sequence 1 to 4. We modify the generated heatmap size by cropping and zero-padding on the IF signal data, as discussed in Section III-A. As shown in Table IV, we observe that the larger heatmap sizes result in improved performance but at the expense of increased system latency. This trade-off between performance and efficiency arises because the larger heatmap contains more information, but it also demands a higher time overhead. Notably, when enlarging the heatmap size to 64 × 1024, the improvements in performance become marginal compared to the 64 × 768 configuration, while incurring a significantly higher computational overhead. Hence, We opt for the 64 × 768 heatmap size in our mmPlace.

D. Evaluation for Heatmap Concatenation

In this subsection, we evaluate mmPlace in difficult scenarios (sequence 5 to 9) involving rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters. First, we evaluate mmPlace's performance without heatmap concatenation under different rotational and lateral variations. Afterward, we compare different heatmap concatenation methods in these difficult scenarios.

Different Rotational and Lateral Variations: We evaluate the influence of different rotations and lateral translation in sequence 5, without utilizing the heatmap concatenation. This sequence encompasses rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters. As shown in Table V, the system performance declines as the rotation angle and lateral distance increase. This issue mainly arises from the limited overlap between the current query data and the candidate data stored in the pre-built map database in scenarios with

TABLE III
PERFORMANCE OF **MMPLACE** AND **POINT CLOUD-BASED METHODS** AND **POINT CLOUD IMAGE-BASED METHODS**

Input Data	Method	Sequence 1		Sequence 2		Sequence 3		Sequence 4		Mean
		Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	Recall@1
Pointcloud	M2DP [22]	52.60/62.77/67.14	0.714	40.16/49.03/53.59	0.596	30.39/41.78/47.38	0.471	45.40/57.68/63.74	0.677	42.13
	LoGG3D [12]	56.59/70.14/74.80	0.772	53.36/69.55/75.38	0.723	38.71/54.89/62.29	0.568	46.49/67.16/71.85	0.681	48.78
	ScanContext [10]	51.46/59.80/64.12	0.679	47.98/56.85/61.07	0.653	43.61/54.00/59.37	0.609	53.13/57.91/59.86	0.737	49.04
Pointcloud Image	Kidnapped [7]	43.78/51.37/55.35	0.696	39.98/49.44/53.75	0.614	35.24/46.74/52.30	0.552	50.03/62.09/66.51	0.720	42.25
	AutoPlace [6]	58.40/63.59/65.72	0.781	59.01/65.47/68.65	0.753	53.69/62.71/67.47	0.705	58.74/64.63/67.51	0.764	57.46
IF signal	**Ours**	**88.47/89.15/89.27**	**0.941**	**88.60/90.40/90.98**	**0.943**	**88.16/89.42/89.90**	**0.940**	**88.34/89.38/89.48**	**0.941**	**88.39**

Sequence 1 to 4 encompass rotational variations from 0 to 10 degrees and lateral translation variations from 0 to 1 meter.
The bold entities illustrates the comparison methods of the experiment and the best performance results.

TABLE IV
COMPARISON OF **DIFFERENT RANGE AZIMUTH HEATMAP SIZES**

Heatmap Size	Sequence 1		Sequence 2		Sequence 3		Sequence 4		Latency(ms)
	Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	Recall@1/5/10	max F_1	
32 × 512	86.63/88.40/88.88	0.932	85.84/88.95/89.85	0.924	85.19/88.15/89.00	0.922	86.56/87.26/87.44	0.932	**17.56**
32 × 768	88.01/88.79/89.17	0.939	87.00/89.92/90.61	0.939	87.23/89.03/89.67	0.939	86.73/88.25/89.15	0.934	19.12
32 × 1024	88.16/88.94/89.02	0.941	87.67/90.01/90.73	0.938	88.01/89.83/90.30	0.939	87.81/88.31/89.51	0.938	20.29
64 × 512	86.84/88.31/88.74	0.935	86.33/88.95/89.80	0.930	87.49/87.63/87.72	0.938	86.57/88.63/89.34	0.932	19.27
64 × 768*	88.47/**89.15**/89.27	0.941	**88.60**/90.40/90.98	**0.943**	88.16/89.42/89.90	0.940	**88.34/89.38**/89.48	**0.941**	20.14
64 × 1024	**88.49**/89.04/**89.30**	**0.943**	88.33/**90.64/91.51**	0.942	**88.18/89.62/90.03**	**0.941**	88.07/89.06/**89.51**	0.940	24.39

* denotes the range azimuth heatmap size used in mmPlace.
Sequence 1 to 4 encompass rotational variations from 0 to 10 degrees and lateral translation variations from 0 to 1 meter.
The bold entities illustrates the experimental parameter settings and the best performance results.

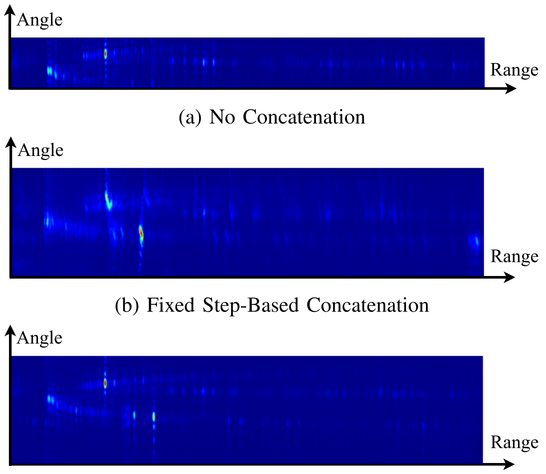
TABLE V
PERFORMANCE OF MMPLACE **WITHOUT HEATMAP CONCATENATION** UNDER DIFFERENT ROTATIONAL AND LATERAL VARIATIONS

$Rot(^\circ)$ / $Lat(m)$	0–5	5–10	10–20	20–40
0.0–0.5	93.77	86.60	65.56	39.23
0.5–1.0	84.09	84.05	63.27	36.40
1.0–2.0	73.06	70.56	60.95	35.38
2.0–3.0	60.83	51.08	44.67	27.78

rotational and lateral variations, as discussed in Section III-C. To address this issue, we propose to employ a rotating platform and concatenate the heatmaps in a rotation cycle to enhance the system's FOV.

Comparison of Different Concatenation Methods: We conduct a comparison of different range azimuth heatmap concatenation methods, including no concatenation, fixed step-based heatmap concatenation, and relative pose-based heatmap concatenation in sequence 6 to 9. These sequences encompass rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters.

As shown in Table VI, our proposed relative pose-based heatmap concatenation achieves 87.37% racall@1 in scenarios involving rotational and lateral variations. Specifically, it achieves 18.35% and 16.01% higher recall@1 than no concatenation in scenarios with 10 to 40 degrees of rotation and 1 to 3 meters of lateral translation, respectively. Also in these scenarios, our mmPlace achieves up to 83.48% and 84.47% racall@1, which outperforms fixed step-based concatenation by 4.25% and 6.78%, respectively. This is because the relative pose-based heatmap concatenation not only enables pixel-level alignment of the heatmaps in a rotation cycle but also enlarges the FOV overlapping area in rotational and lateral variation scenarios, as discussed in Section III-C. Again, as shown in Fig. 8, the heatmap concatenated based on relative pose has a larger FOV than the initial heatmap and is more accurate than the heatmap concatenated with fixed step size.



(a) No Concatenation

(b) Fixed Step-Based Concatenation

(c) Relative Pose-Based Concatenation

Fig. 8. Visualisation of different heatmap concatenation methods. (a) Illustrates the heatmap without concatenation, while (b) and (c) illustrate the heatmap concatenated with fixed step size and the heatmap concatenated based on relative pose, respectively.

TABLE VI
COMPARISON OF **DIFFERENT HEATMAP CONCATENATION METHODS** IN SCENARIOS (SEQUENCE 6 TO 9) INVOLVING ROTATIONAL VARIATIONS FROM 0 TO 40 DEGREES AND LATERAL TRANSLATION VARIATIONS FROM 0 TO 3 METERS

Method	Data with 10–40 Rot(°)		Data with 1–3 Lat(m)		All Data	
	Recall@1/5/10	maxF_1	Recall@1/5/10	maxF_1	Recall@1/5/10	maxF_1
Ours w/o HC[1]	65.13/68.31/69.05	0.823	68.46/74.51/75.61	0.839	69.18/76.27/80.81	0.840
Ours w/ FS-HC[2]	79.23/80.90/82.05	0.906	77.69/82.55/83.73	0.881	82.43/84.61/85.73	0.915
Ours w/ RP-HC[3]	**83.48/86.42/89.88**	**0.915**	**84.47/86.84/88.84**	**0.926**	**87.37/89.07/90.92**	**0.939**

[1] w/o HC: without the heatmap concatenation.
[2] w/ FS-HC: with the fixed step-based heatmap concatenation.
[3] w/ RP-HC: with the relative pose-based heatmap concatenation.
The bold entities illustrates the comparison methods of the experiment and the best performance results.

V. CONCLUSION

In this letter, we propose mmPlace, a robust place recognition system based on a low-cost single-chip radar. Since point cloud-based and point cloud image-based methods perform poorly due to the sparse point cloud of the single-chip radar, mmPlace starts with the IF signal. Firstly, mmPlace generates range azimuth heatmap by performing range and angle estimation processing on the IF signal. Then, a spatial encoder is used for feature extraction on the heatmap. Additionally, mmPlace deploys a rotating platform and concatenates heatmaps in a rotation cycle to enhance the system's performance in scenarios involving rotational and lateral variations. We collect the milliSonic dataset on the USTC campus, the city roads surrounding the campus, and an underground parking garage. Our experiments on the milliSonic dataset demonstrate that mmPlace surpasses both point cloud-based and point cloud image-based methods. The heatmap concatenation enhances the system's performance in scenarios encompassing rotational variations from 0 to 40 degrees and lateral translation variations from 0 to 3 meters.

REFERENCES

[1] P. Yin, S. Zhao, I. Cisneros, A. Abuduweili, G. Huang, and M. Milford, "General place recognition survey: Towards the real-world autonomy age," 2022, *arXiv:2209.04497*.

[2] 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. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 6433–6438.

[3] H. Caesar et al., "nuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11618-11628.

[4] S. Lowry et al., "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[5] D.-H. Paek, S.-H. Kong, and K. T. Wijaya, "K-Radar: 4D Radar object detection for autonomous driving in various weather conditions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 3819–3829.

[6] K. Cai, B. Wang, and C. X. Lu, "Autoplace: Robust place recognition with low-cost single-chip automotive radar," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 3475–3481.

[7] Ş. Săftescu, M. Gadd, D. De Martini, D. Barnes, and P. Newman, "Kidnapped radar: Topological radar localisation using rotationally-invariant metric learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4358–4364.

[8] K. Qian, S. Zhu, X. Zhang, and L. E. Li, "Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 444–453.

[9] K. Harlow, H. Jang, T. D. Barfoot, A. Kim, and C. Heckman, "A new wave in robotics: Survey on recent mmWave radar applications in robotics," 2023, *arXiv:2305.01135*.

[10] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2018, pp. 4802–4809.

[11] G. Kim, S. Choi, and A. Kim, "Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1856–1874, Jun. 2022.

[12] K. Vidanapathirana, M. Ramezani, P. Moghadam, S. Sridharan, and C. Fookes, "LoGG3D-Net: Locally guided global descriptor learning for 3D place recognition," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 2215–2221.

[13] Z. Hong, Y. Petillot, and S. Wang, "RadarSLAM: Radar based large-scale SLAM in all weathers," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2020, pp. 5164–5170.

[14] D. Barnes and I. Posner, "Under the radar: Learning to predict robust keypoints for odometry estimation and metric localisation in radar," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 9484–9490.

[15] J. Rebut, A. Ouaknine, W. Malik, and P. Pérez, "Raw high-definition radar for multi-task learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17000–17009.

[16] X. Gao, G. Xing, S. Roy, and H. Liu, "RAMP-CNN: A novel neural network for enhanced automotive radar object recognition," *IEEE Sensors J.*, vol. 21, no. 4, pp. 5119–5132, Feb. 2021.

[17] M. J. Cummins and P. M. Newman, "FAB-MAP: Probabilistic localization and mapping in the space of appearance," *Int. J. Robot. Res.*, vol. 27, no. 6, pp. 647–665, 2008.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.

[19] I. Namatēvs, "Deep convolutional neural networks: Structure, feature extraction and training," *Inf. Technol. Manage. Sci.*, vol. 20, no. 1, pp. 40–47, 2017.

[20] R. Arandjelovic, P. Gronát, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1437–1451, Jun. 2018.

[21] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-NetVLAD: Multi-scale fusion of locally-global descriptors for place recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14136–14147.

[22] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2016, pp. 231–237.

[23] C. Iovescu and S. Rao, "The fundamentals of millimeter wave sensors," *Texas Instrum.*, pp. 1–8, 2017.

[24] J. Kim, Y. Kim, and D. Kum, "Low-level sensor fusion network for 3D vehicle detection using radar range-azimuth heatmap and monocular image," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 388–402.

[25] X. Li, X. Wang, Q. Yang, and S. Fu, "Signal processing for TDM MIMO FMCW millimeter-wave radar sensors," *IEEE Access*, vol. 9, pp. 167959–167971, 2021.

[26] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Trans. Big Data*, vol. 7, no. 3, pp. 535–547, Jul. 2021.

[27] Y. Duan, J. Peng, Y. Zhang, J. Ji, and Y. Zhang, "PFilter: Building persistent maps through feature filtering for fast and accurate LiDAR-based SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2022, pp. 11087–11093.

[28] J. Wang et al., "Learning fine-grained image similarity with deep ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*, 2014, pp. 1386–1393.

[29] E. Pan, J. Tang, D. Kosaka, R. Yao, and A. Gupta, "OpenRadar," 2019. [Online]. Available: https://github.com/presenseradar/openradar

[30] J.-T. Lin, D. Dai, and L. V. Gool, "Depth estimation from monocular images and sparse radar data," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2020, pp. 10233–10240.