ISPRS Journal of Photogrammetry and Remote Sensing 183 (2022) 1–18

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

A heterogeneous 3D map-based place recognition solution using virtual LiDAR and a polar grid height coding image descriptor

Dong Xu [a], Jingbin Liu [a,b,*], Juha Hyyppä [b], Yifan Liang [a], Wuyong Tao [c]

[a] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China
[b] Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, Masala 02430, Finland
[c] School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China

ARTICLE INFO

Keywords:
Place recognition
Heterogeneous 3D map
Point cloud
Global feature descriptor
Polar grid height coding image

ABSTRACT

Place recognition is widely used for global localization technology. However, the existing place recognition solutions are limited by the requirement for the same type of sensors to be used in both the localization process and the mapping process. Therefore, the existing heterogeneous 3D map cannot be used for place recognition directly with the existing methods, leading to underutilization of information. In addition, most of the existing global feature descriptors used in place recognition solutions are still not highly descriptive and perform poorly under changed viewpoint scenes. To resolve these challenges, this paper presents a place recognition solution using virtual light detection and ranging (LiDAR) and polar grid height coding image (PGHCI) descriptors in the existing heterogeneous 3D map. First, virtual LiDAR is proposed to generate a series of virtual scans that are similar to the real scan of the localization sensor from the existing map, overcoming the limitation of the existing place recognition methods. Next, a novel PGHCI descriptor for place recognition is generated, and a method that overcomes the recognition difficulty of changed viewpoints in the same scene is presented. Two weighted distances for similarity estimation are analyzed, and the performance of the PGHCI descriptor with different parameters is evaluated. Finally, the performance of the PGHCI descriptor and the solution proposed in the paper is evaluated on several popular benchmark datasets and our own dataset. Comprehensive experiments demonstrated that the PGHCI descriptor has higher descriptiveness and is robust with respect to the changed viewpoint scene, as shown by the comparison of precision-recall (PR) curves using datasets with multiple scenes. The proposed place recognition solution has 100% success rates in the evaluation exclude under occluded conditions, showing that it is feasible to achieve robust place recognition using a heterogeneous 3D map.

1. Introduction

Place recognition is a classic problem in robotics applications and refers to a procedure of discerning places that have been visited. It is widely used to provide robust initial estimates for global localization technology by matching data of localization sensors with an existing map. Based on localization sensor data, current place recognition solutions can be divided mainly into two categories: image-based and 3D point cloud-based. Image-based methods suffer from a limited view field of photo and illumination changes (Wu et al., 2018; Wang et al., 2019). 3D point cloud-based solutions have recently attracted more attention because point cloud data cover a larger field and are insensitive to illumination.

Existing 3D point cloud-based place recognition solutions usually include the following steps. First, feature extraction is carried out for each scan in a 3D map (features here include but are not limited to global features and feature sets composed of local shape features). Then, the distances between the features of the query scan from the localization sensor and the features of each scan in the 3D map are obtained by similarity estimation. Finally, the minimum distance between two features that is smaller than a threshold value indicates that place recognition is successful. Although many insightful 3D point cloud-based solutions have been proposed, some problems still remain to be solved.

Existing 3D point cloud-based place recognition solutions are based on similarity estimation between query scans and scans in 3D maps. These solutions request scans in a 3D map to be from the same type of sensor as the localization sensor (He et al., 2016; Kim and Kim, 2018; Wang et al., 2019; Guo et al., 2019; Cop et al., 2018). Therefore, the

* Corresponding author.
E-mail address: jingbin.liu@whu.edu.cn (J. Liu).

https://doi.org/10.1016/j.isprsjprs.2021.10.020
Received 2 July 2021; Received in revised form 18 October 2021; Accepted 27 October 2021
Available online 10 November 2021

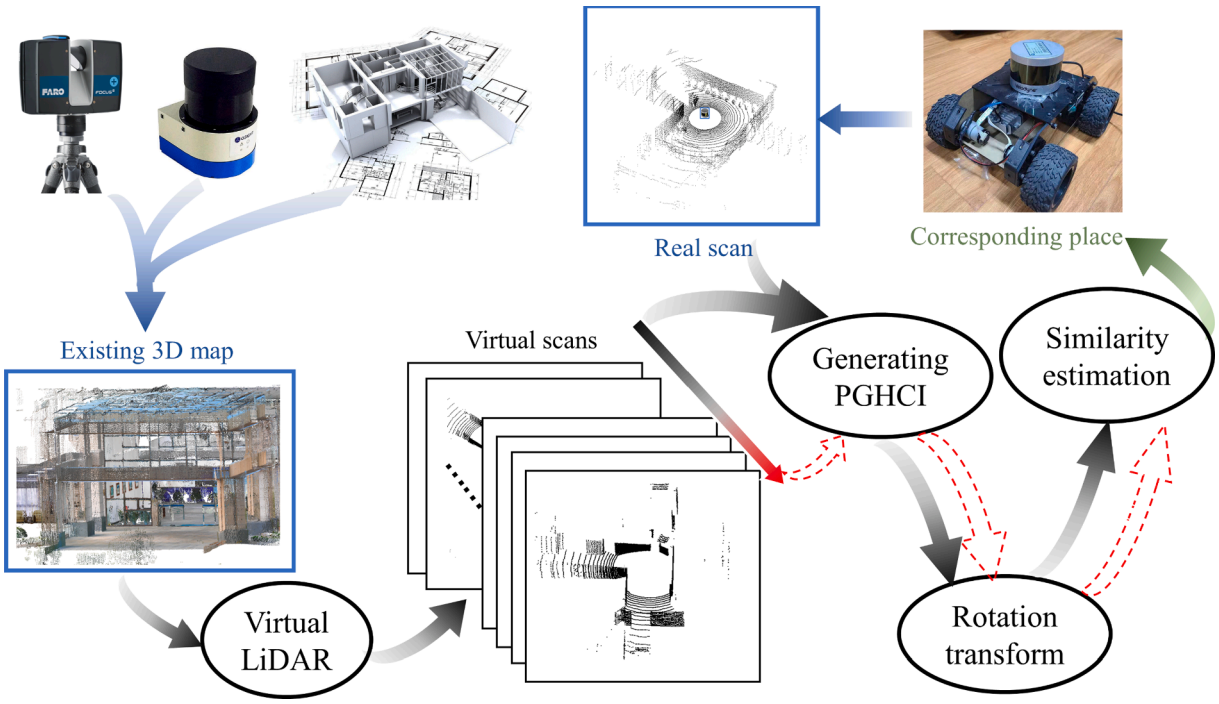

Fig. 1. The solution for place recognition in existing heterogeneous 3D map. The inputs of the solution are a real scan and an existing 3D map. The output is the place information of the sensor used for real scan collection. Virtual LiDAR is proposed to generate virtual scans. We can match real scan from the sensors with virtual scans. The match steps include calculating PGHCI descriptors, rotation transformation and estimating similarity.

existing heterogeneous 3D map from a different type of sensor cannot be used for place recognition directly with the existing solutions. In the localization process, rotating multibeam LiDAR is usually used (Wang et al., 2019; Guo et al., 2019). Heterogeneous 3D maps derived from terrestrial laser scanners (TLSs) (Liu et al., 2017), building information models (BIMs) (Liu et al., 2021); and 2D laser range finders (Zhang and Singh, 2017) cannot be used for place recognition with existing methods. However, there are many existing heterogeneous 3D maps. The use of the existing heterogeneous 3D maps for place recognition will avoid redundant data collection efforts. In addition, the descriptiveness of existing global feature descriptors used in the existing place recognition solutions can be improved. Poor recognition is obtained in some cases, particularly under the condition that LiDAR is set on the same place with changed viewpoints (Kim and Kim, 2018).

To resolve the above issues, we proposed a novel place recognition solution that can be used in the existing heterogeneous 3D map, as shown in Fig. 1. In our paper, one scan is equivalent to one frame. The contributions of this work beyond state-of-the-art methods are as follows:

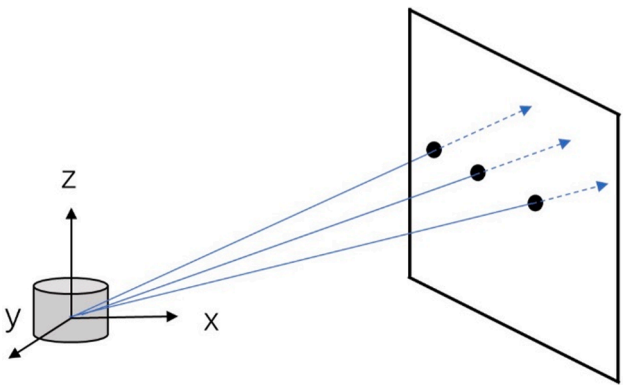

Fig. 2. The laser ray from a vritual LiDAR hits a planar model.

(1). We developed a heterogeneous 3D map-based place recognition solution using virtual LiDAR and a polar grid height coding image (PGHCI) descriptor. Using the concept of virtual LiDAR, the proposed solution overcomes the limitation of existing place recognition methods that require the use of same types of sensors in the mapping process and localization process.
(2). The proposed PGHCI descriptor adopts a polar grid and encodes a 3D point cloud into a 2D image. Therefore, the PGHCI descriptor is highly descriptive and features rotational invariance.
(3). Jensen–Shannon (JS) divergence was adopted to estimate the rotation transform between two PGHCI descriptors under changed viewpoints. We considered the distribution of each column within the PGHCI descriptor. The adopted method is more accurate than the forcible method and overcomes the difficulty of recognition of changed viewpoints in the same scene.
(4). Two weighted distances were applied to estimate the similarity between two PGHCI descriptors, and we analyzed the applicability of two weighted distances for different types of sensors.

The rest of the paper is organized as follows. Section 2 describes the related work; Section 3 introduces solution of place recognition that utilizes virtual LiDAR and the PGHCI descriptor; Section 4 presents the experimental results. Section 5 summarizes and concludes this work.

2. Related work

In this section, we first present a brief introduction of LiDAR and some applications using virtual LiDAR. Then, we review 3D point cloud-based place recognition solutions.

2.1. Brief introduction of LiDAR and virtual LiDAR

LiDAR (light detection and ranging) has become a common technology for the acquisition of 3D point clouds. A LiDAR sensor can obtain the distance of an object by emitting a pulsed laser and receiving the reflected signals from the object. The fundamental principle of major

LiDAR is time of flight (TOF). Direct TOF obtains the time between sending and receiving the pulsed laser by a high-precision timer. Indirect TOF obtains the time by measuring the phase shift. First, the time of flight is multiplied by the speed of light to obtain the distance. Then, combining the distance of the object with the angles at which the laser was emitted, the point of the object can be obtained.

Some model-based methods focus on producing virtual LiDAR data. Ray casting is used to simulate each laser ray emitted by the virtual LiDAR (Gusmão et al., 2020). Different from the LiDAR, the point of the model is obtained by calculating the intersection of the laser rays emitted from the virtual LiDAR and the surface of the model, as shown in Fig. 2. The laser ray is emitted at a preset yaw and pitch angle that are parameters of the virtual LiDAR. The first point that the laser ray hits is the point of model.

Based on the existing model, some studies focused on applications using virtual LiDAR. In the construction domain, virtual LiDAR plays an important role in construction progress tracking (Bosché et al., 2014; Bosché et al., 2015), building component quality control (Bosché and Guenet, 2014) and rescue after the disaster (Ma et al., 2016). In the above applications, virtual LiDAR is proposed to generate an as-plan point cloud from the BIM model. In the autonomous driving domain (Yue et al., 2018; Hanke, 2018; Zhao et al., 2021), virtual LiDAR is proposed to create large point cloud datasets with point-level labels from a dynamic virtual environment. Many models exist for the dynamic virtual environment. Synthetic datasets greatly develop supervised deep learning algorithms which are data driven, bringing the overall safety validation effort for automated driving functions to an economically feasible level (Linnhoff et al., 2020).

Major existing heterogeneous 3D maps are not models. Therefore, virtual LiDAR data cannot be obtained using the above methods. The virtual LiDAR proposed in this paper converts the original point cloud into data that are similar to the data of the localization sensor. More details can be found in Section 3.1.

2.2. 3D point cloud-based place recognition solution

3D point cloud-based place recognition solutions are divided mainly into three categories. The first category is the bag of words (BOW)-based methods. These methods detect the key points in the frame and then calculate the 3D local feature descriptor of each key point. The 3D local feature descriptor is a vector that is converted from the surface shape information contained in the detected feature point neighborhood. Detailed studies of local feature descriptors are reported in the literature (Guo et al., 2014; Yang et al., 2016; Han et al., 1802; Yang et al., 2020). The 3D local feature descriptors are converted into words of a dictionary that have been subsequently constructed offline. Then; place recognition is carried out based on a histogram of the words. Steder et al. (Steder et al., 2011) used normal aligned radial feature (NARF) to construct a bag-of-words model and achieved good results. Other key point detectors can also be applied in the construction of word bag models, such as intrinsic shape signatures (ISS) (Yu, 2009), Harris3D (Sipiran and Bustos, 2011) and KPQ-SI (Mian et al., 2010). Binary shape context (BSC) (Dong et al., 2017), Signature of Histograms of Orientations (SHOT) (Tombari et al., 2010), Quintuple Local Coefficient Images (QLCI) (Tao et al., 2020) and other local feature descriptors can also be used for local feature extraction. However, it is still a huge challenge to realize 3D key point detection with a high repetition rate (Boroson and Ayanian, 2019). Under the condition of the same scene with different viewpoints, the result of place recognition will be greatly affected if the repetitive detection result of key points cannot be achieved.

Extraction of global feature descriptors can solve the above problems. Methods based on global feature descriptors are widely used for place recognition. Compared with the BOW-based methods, these methods reduce the computational complexity and improve the robustness of the algorithm by extracting the features of the acquired whole point cloud. Muhammad et al. presented Z-projection (Muhammad and Lacroix, 2011) that calculates the normal vectors and saves the angles between the normal corresponding to each point and the Z axis to construct the feature descriptor. He et al. proposed multi-view 2D projection (M2DP) (He et al., 2016) that projects the 3D point cloud onto several two-dimensional planes and obtains the point cloud distribution matrix. The feature descriptor is generated through singular value decomposition (SVD) of the matrix. Under the condition of the same scene with changed viewpoints, the above methods cannot achieve place recognition results, because both descriptors are not rotationally invariant. Scan Context (SC) (Kim and Kim, 2018) and LiDAR IRIS (Wang et al., 2019) construct arc-shaped grids for rotating LiDAR and extracting features inside the grids to construct feature descriptors. The Scan Context selects the highest value within the raster to acquire feature descriptors, and the LiDAR IRIS conducts LoG-Gabor filtering and thresholding operations on the LiDAR IRIS image to obtain feature descriptors. To achieve rotation invariance, Scan Context uses the forcible solution to match feature descriptors, while LiDAR IRIS uses the Fourier transform. Although Scan Context and LiDAR-IRIS have rotational invariance characteristics, they also show poor recognition, particularly under the condition that LiDAR is set on the same place with a changed viewpoint. This is because the methods obtaining rotational invariance are not sufficiently accurate in addition to the low descriptiveness of the above descriptors. All of the above methods encode spatial information into feature descriptors. DEscriptor of LiDAR Intensities as a Group of HisTograms (DELIGHT) (Cop et al., 2018), Intensity Signature of Histograms of OrienTations (ISHOT) (Guo et al., 2019) and Intensity Scan Context (ISC) (Wang et al., 2020) make use of echo intensity information to construct feature descriptors. Intensity information needs to be calibrated prior to using different LiDARs. The calibration result of intensity information affects place recognition with these descriptors.

The proposed PGHCI descriptor is created based on a polar grid, endowing the descriptor with rotational invariance. The value of each grid is assigned by the height distribution of the points in the grid, making the descriptor more descriptive. In addition, a more accurate method that obtains rotational invariance is adopted in the place recognition procedure. The combination of all of the above characteristics makes the proposed descriptor more robust for place recognition in challenging scenes.

To date, two methods using deep learning have been employed to carry out place recognition. The first approach is to carry out place recognition by extracting features. 3D feature learning is also a research hotspot. Elbaz et al. (Elbaz et al., 2017) extended 2D feature extraction to 3D point clouds by projecting 3D point clouds onto 2D planes. In (Gojcic et al., 2019; Zeng et al., 2017); the authors constructed the voxel point cloud and extracted the features of each voxel space. Finally, the features were obtained through the connecting layer. After the publication of PointNet (Qi et al., 2017) and PointNet++ (Qi et al., 2017), the subsequent papers (Deng et al., 2019; Wang and Solomon, 2019; Lu et al., 2019; Wang et al., 2019) used this approach for reference in feature extraction. The approach takes the point as input directly and the overall feature of the point cloud as output. Based on 3D feature learning, Liu et al. (Liu, 2019) proposed the SEQLPD network to extract the features of the point cloud and carry out place recognition. In this paper, a complete workflow for place recognition using deep learning was built. Some other similar works have also been reported (Chen et al., 2020; Yin et al., 2020; Sun et al., 2020). The second approach is to extract semantic information for place recognition. Dube et al. (Dube et al., 2017) proposed the Seg-Match network that performs location recognition after semantic segmentation of point clouds. The result of semantic segmentation affects place recognition using this kind of method. Although deep learning has far surpassed traditional methods in the field of image target recognition, deep learning-based methods require a large amount of data for parameter training. There is still a lack of point cloud training data covering various scene categories.

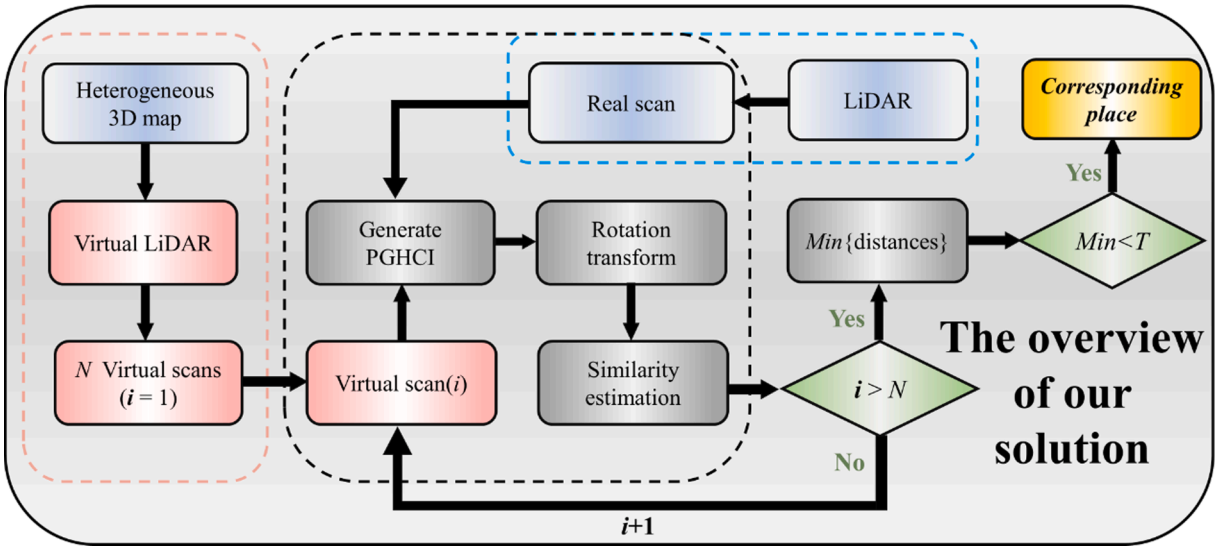

Fig. 3. The workflow of place recognition solution based existing heterogeneous 3D map.

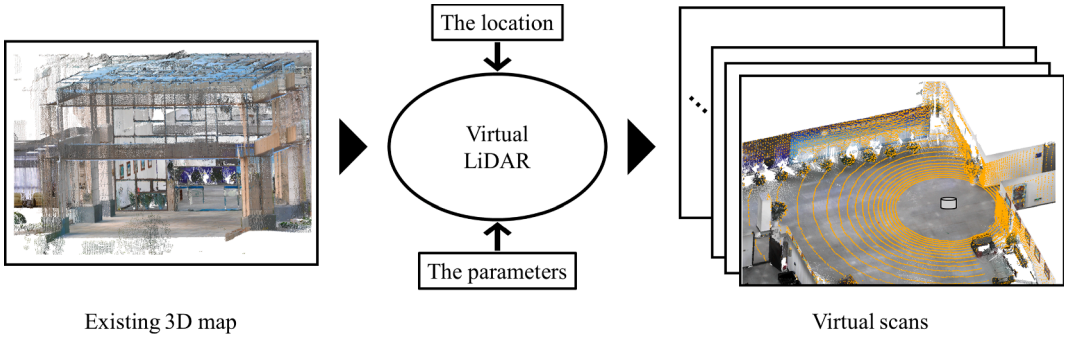

Fig. 4. The workflow for generating virtual scan for place recognition in existing 3D map.

3. Proposed place recognition solution

In this section, we present a solution using virtual LiDAR and polar grid height coding image descriptors for place recognition in existing heterogeneous 3D maps. The workflow of our solution is shown in Fig. 3. The heterogeneous 3D map is converted into a series of virtual scans that are similar to real scans through virtual LiDAR. The real scan from the localization sensor is matched with virtual scans. The match steps include the calculation of PGHCI descriptors, rotation transformation and similarity estimation. If the minimum distance between the PGHCI descriptors of the real scan and virtual scan is smaller than the threshold value, the corresponding place can be obtained. In this section, virtual LiDAR is presented first. Then, the PGHCI descriptor is described in detail.

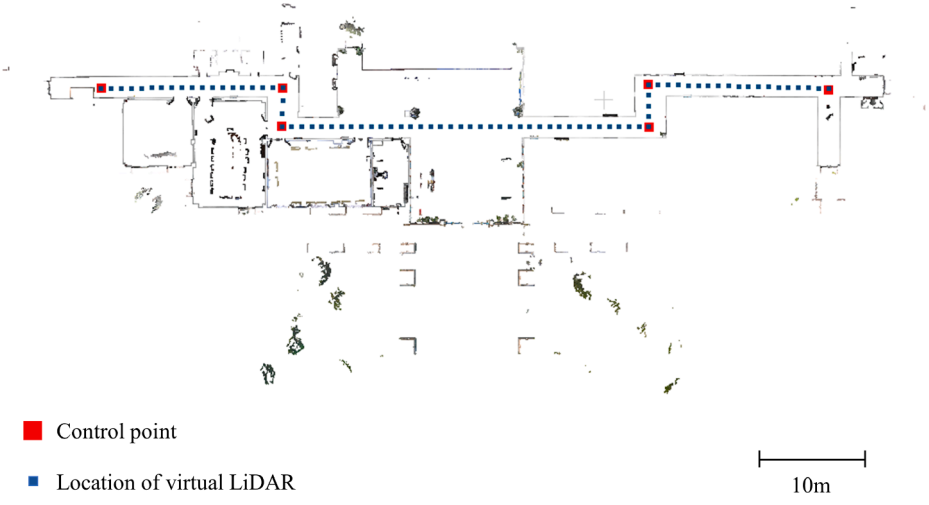

Fig. 5. The location of virtual LiDAR in an existing heterogeneous 3D map.

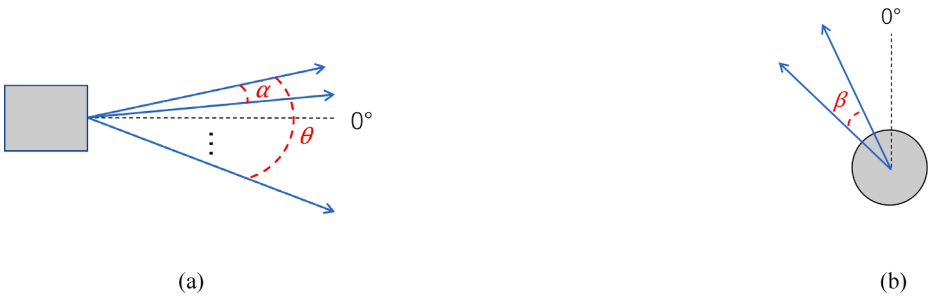

Fig. 6. Several configurable parameters of virtual LiDAR. (a) is a side view of virtual LiDAR: θ is the vertical FOV, α is the resolution of vertical angle; (b) is a top view of virtual LiDAR: β is the resolution of horizontal angle. The horizontal FOV of virtual LiDAR is 360°.

3.1. Virtual LiDAR

In this section, virtual LiDAR is proposed as the basis of our place recognition solution in an existing heterogeneous 3D map. First, a semiautomatic method that generates the location of virtual LiDAR in an existing heterogeneous 3D map is introduced. The requisite parameters of virtual LiDAR are subsequently described. Finally, the principle of the proposed virtual LiDAR is demonstrated. According to the location and parameters of virtual LiDAR, out method can convert heterogeneous 3D maps into a series of virtual scans that are similar to real scans of localization sensors. The workflow for generating virtual scans is shown in Fig. 4.

3.1.1. Location of virtual LiDAR

To convert the existing heterogeneous 3D map into a series of virtual scans, we should obtain the position and posture of each scan in the map. First, we pick the requisite control points manually. The requisite control points are usually in the inflection corner. The number of requisite control points depends on the complexity of the passable area in the existing heterogeneous 3D map. For example, six control points need to be picked in a map, as shown in Fig. 5. After picking the control points, the location of virtual LiDAR $vL_i\left(vL_i^x, vL_i^y, vL_i^z\right), i \in [vL_n]$ will be generated by interpolation. The symbol $[vL_n]$ denotes $\{vL_1, vL_2, \ldots, vL_{n-1}, vL_n\}$. Because the location of the control points is known, the location of virtual LiDAR can be acquired. The posture of virtual LiDAR includes yaw, roll and pitch angle. Considering the rotation invariance of our PGHCI descriptor, we set the yaw of each scan to a constant value. The localization sensor is usually placed horizontally. Therefore, the yaw, roll and pitch angles of virtual LiDAR are set to zero values.

3.1.2. Requisite parameters of virtual LiDAR

Rotating multibeam LiDAR is used for place recognition as a major sensor. Its parameters (as shown in Fig. 6) mainly include the measurement range, vertical field of view (FOV) and resolution of the angle (horizontal and vertical). To generate a series of virtual scans that are similar to the real scan of the rotating multibeam LiDAR, the vertical FOV and resolution of the angle (horizontal and vertical) of the virtual LiDAR need to be set.

After obtaining the range and resolution of the vertical and horizontal angles, the vertical and horizontal angles of the laser ray $L_{i,j}(i \in [\theta/\alpha], j \in [360/\beta])$ can be calculated by interpolation. In addition, $[A_v]$ and $[A_h]$can be obtained after interpolation. $[A_v]$ is a set of vertical angles of all laser rays. $[A_h]$is a set of horizontal angles of all laser rays. The symbol *maxDis* refers to the maximum range of virtual LiDAR.

3.1.3. The principle of proposed virtual LiDAR.

After obtaining the location, posture and parameters of virtual LiDAR, we can generate virtual scans based on existing heterogeneous 3D maps. Different from the scanning process of LiDAR, the generation of virtual scans of virtual LiDAR is a filtering process. The laser ray $L_{i,j}$ hits at most one point. Therefore, there are most $(\theta/\alpha)*(360/\beta)$ points in a virtual scan. The filtering process filters all of the points of the existing heterogeneous 3D map to obtain the virtual scan points. We denote all points of the existing heterogeneous 3D map as $P = \{P_1, P_2, P_3, \cdots, P_m\}$, where m is the number of points. The generation of a virtual scan at location $vL_k$is described in detail as follows:

For each point $P_i\left(P_i^x \quad P_i^y \quad P_i^z\right)$ in P, it is first evaluated whether its distanceD_i is smaller than *maxDis*:

$$D_i = \|vL_k - P_i\| \tag{1}$$

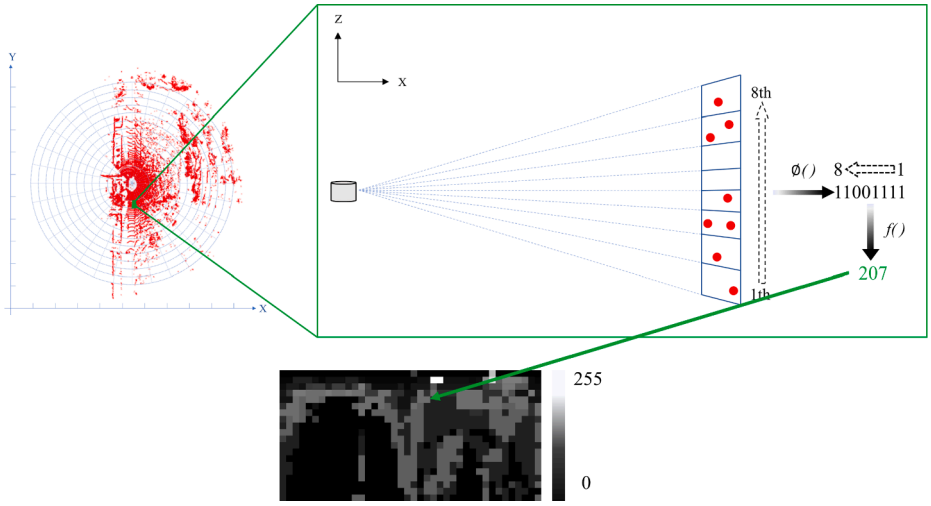

Fig. 7. An illustration of the generation of a PGHCI descriptor.

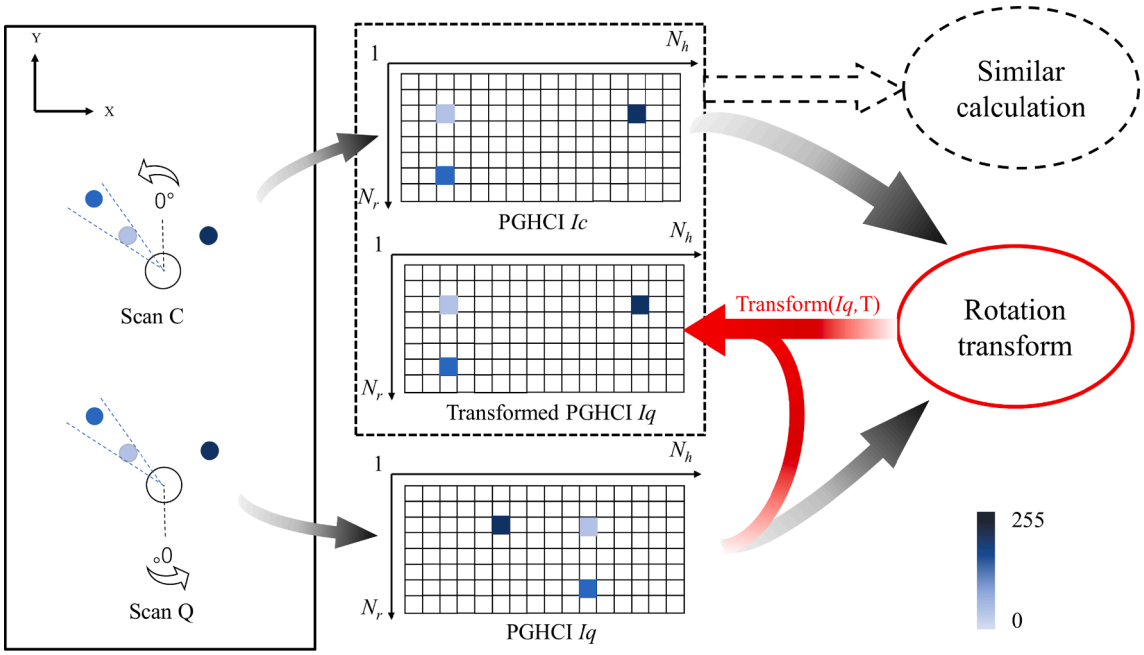

Fig. 8. An illustration of changed viewpoint affection on PGHCI descriptors.

If $D_i > maxDis$, point $P_i$will be removed. Else, the procedure will advance to the next step.

It is judged whether P_i belongs to a laser ray according to the vertical angle V_i and horizontal angleH_i.

$$V_i = \arctan\left(\frac{P_i^z - vL_k^z}{\sqrt{\left(P_i^x - vL_k^x\right)^2 + \left(P_i^y - vL_k^y\right)^2}}\right)$$
$$H_i = \arctan\left(\frac{P_i^y - vL^y}{P_i^x - vL^x}\right) \tag{2}$$

If $V_i \in [A_v] \& H_i \in [A_h]$, P_i will be considered to belong to a laser ray. Then, P_i will be compared with the previous point that belongs to the same laser ray. The point closer to the virtual LiDAR will be retained.

After the process of filtering P, the point cloud of a virtual scan in a location will be obtained.

When the virtual scans at each location of virtual LiDAR are calculated, we can obtain all virtual scans that represent heterogeneous 3D maps for place recognition. Compared to the original heterogeneous 3D map, the data size of virtual scans is smaller.

3.2. Polar grid height coding image descriptor for place recognition

In this section, we present a method to generate PGHCI descriptors and describe how to overcome the recognition difficulty of changed viewpoints in the same scene by JS divergence. Next, two weighted distances for similarity estimation are analyzed, and the performance of the PGHCI descriptor with different parameters is evaluated.

3.2.1. Generation of PGHCI descriptor

We take the measurement theory of rotating multibeam LiDAR used for place recognition into consideration and refer to some approaches that use the same type of sensors for road detection (Sun Peng-peng et al., 2018). After comprehensive consideration, we propose PGHCI descriptors, as shown in Fig. 7. After we obtain a scan from rotating multibeam LiDAR or virtual LiDAR, we place all of the points of a scan into a polar grid first. Then, we assign the value of each grid by the height distribution of the points in the grid.

The origin of the polar coordinate is the center of a 3D scan. The polar grid is acquired after we divide a 3D scan into horizontal azimuthal and radial bins in polar coordinates. Then, the whole point cloud P of a 3D scan is separated into N_h*N_r mutually exclusively point cloud P_{ij} ($i \in [N_h], j \in [N_r]$). N_h is the number of partitions in the 360° range, and N_r is the number of partitions in a limited range (it refers to the effective observation range of a laser scanner for a specific application). N_h and N_r are set to 40 and 20, respectively, in the paper. Please see Section 3.2.4 for details about these two parameters.

After we partition all of the points of a 3D scan into bins, a value is assigned to each bin by the height distribution of the points in the bin. Each bin is divided into N_v partitions according to the vertical angle between the point and the center. The symbol P_{ijk} ($i \in [N_h], j \in [N_r], k \in [N_v]$) refers to the kth partition of point cloud P_{ij}. The value of Nv is set according to the analysis of frequently used rotating multibeam LiDAR, rather than analysis of the experiments for the parameters. The vertical angular range of rotating multibeam LiDAR is limited. For example, the vertical angle field values of Velodyne HDL-64E (64 beam LiDAR), Velodyne HDL-32E (32 beam LiDAR) and Velodyne VLP-16C (16 beam LiDAR) are 26.8°, 40° and 30°, respectively. Considering that the proposed descriptor should be general to the frequently used rotating multibeam LiDAR, we set $Nv = 8$. This ensures that there are at least two laser beams in each partition within the bin, helping to resist the effects of noise. If the value of Nv is larger than 8, there are not enough two laser beams in each partition within the bin when using Velodyne VLP-16C. If the value of Nv is smaller than 8, an overlarge partition within the bin will hide many details. In conclusion, we set $Nv = 8$ in the paper. The bin encoding function is:

$$\varnothing\left(P_{ijk}\right) = \begin{cases} 1, if\left(P_{ijk} \neq \varnothing\right) \\ 0, othrewise \end{cases}$$
$$f\left(P_{ij}\right) = \sum_{k=1}^{N_v} \varnothing\left(P_{ijk}\right) * 2^k \tag{3}$$
$$V_{ij} = f\left(P_{ij}\right)$$

where $\varnothing()$ is the function that determines whether the point set P_{ijk} is empty and $f()$ is the function that calculates the value of each bin. If the values of all bins are obtained, we can obtain a PGHCI descriptor I.

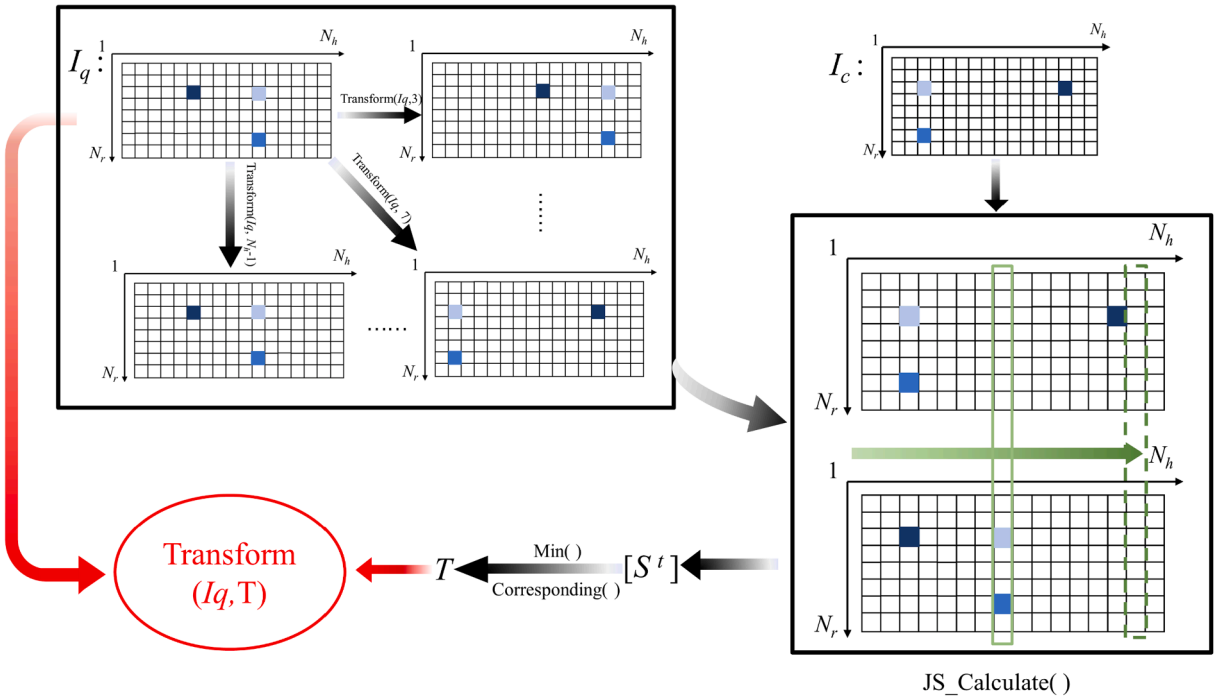

Fig. 9. The workflow of obtaining rotation invariance.

$$I = \bigcup_{i=1,j=1}^{N_h,N_r} V_{ij} \tag{4}$$

Because a polar grid is adopted in the generation of the PGHCI descriptor, it compensates for the insufficient information caused by the sparsity of far points and brings rotational invariance to the descriptor. In addition, the height distribution of the points in the grid is considered. The coding method retains the 3D information, making the PGHCI descriptor more descriptive.

3.2.2. JS divergence for rotation-invariance

Prior to estimating the similarity of two PGHCI descriptors, we need to transform the descriptors to obtain rotational invariance. The zero axis of polar coordinates (which has a constant angle between the front of the vehicle) affects the PGHCI descriptors. This phenomenon makes the two PGHCI descriptors different under the changed viewpoints in the same place, as shown in Fig. 8. Notably, the changed viewpoint causes the fluctuating zero axis of polar coordinates, and it affects the PGHCI descriptor involuntarily. However, we find that the difference between two PGHCI descriptors on the changed viewpoint scene is the column index. If we find the corresponding column index within both PGHCI descriptors, rotational invariance will be acquired.

We can think of each column within the PGHCI descriptor as a discrete distribution. JS divergence can be used to measure the similarity of two discrete distributions. JS divergence is a variant of Kullback–Leibler (KL) divergence (Peng et al., 2008), and KL divergence also refers to relative entropy. For the discrete probability distributions P(x) and Q(x), defined on the same probability space, KL divergence is given by:

$$\mathrm{KL}(\mathrm{P}||\mathrm{Q}) = \sum P(x) log \frac{P(x)}{Q(x)} \tag{5}$$

Both the KL and JS divergences can be used to analyze the similarity between distributions. However, the KL divergence is asymmetric, and its range is not fixed. By contrast, JS divergence is symmetric, and its range is (Wu et al., 2018). Therefore, JS divergence is adopted for similarity analysis. The zero value of the JS divergence indicates that two distributions are the same, and the value of 1 indicates that two distributions are the opposite. In other words, a smaller JS divergence indicates that the two distributions are more similar. Defining the quantity M = (P + Q)*(0.5), we can express the JS divergence as:

$$\mathrm{JS}(\mathrm{P}||\mathrm{Q}) = \frac{1}{2} KL(P||M) + \frac{1}{2} KL(Q||M) \tag{6}$$

JS divergence is adopted to estimate the similarity of each column within PGHCI descriptors. The different indices of the corresponding columns are calculated by Algorithm 1, and the overview of this algorithm is shown in Fig. 9.

Algorithm 1.

Input: Given two PGHCI descriptors I_q and I_c.
Output: corresponding column index difference T; the max similar index S_min.
For t: $0 \rightarrow N_h$
  I_q^t = Transform(I_q,t);
  S^t = JS_Calculate(I_q^t, I_c);
end
S_min = Min([S^t]);
T = Corresponding(S_min)

where Transform() is the function that changes the index of each column in the PGHCI descriptor. JS_Calculate() is the function that calculates the mean JS divergence of all corresponding columns. Corresponding () is the function that returns the index difference of the corresponding columns.

3.2.3. Similarity estimation using weighted distances

After obtaining two transformed PGHCI descriptors, we need to develop the distance measure to estimate the similarity between two descriptors for place recognition. In this paper, we obtain two distances by comparing each pixel value of the PGHCI descriptor. Because each pixel value of the PGHCI descriptor is encoded with an 8-bit binary method, there are two measures to calculate similarity. The first approach is to adopt pixel values to calculate directly. The second approach is to convert each pixel value into binary values for calculation. In the first approach, the pixel value must be normalized first. We

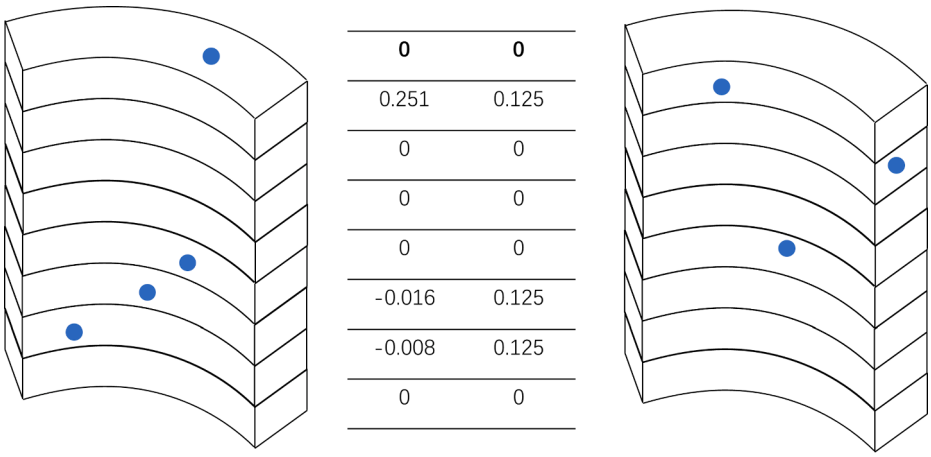

Fig. 10. Two similar values between two pixels using two weighting methods.

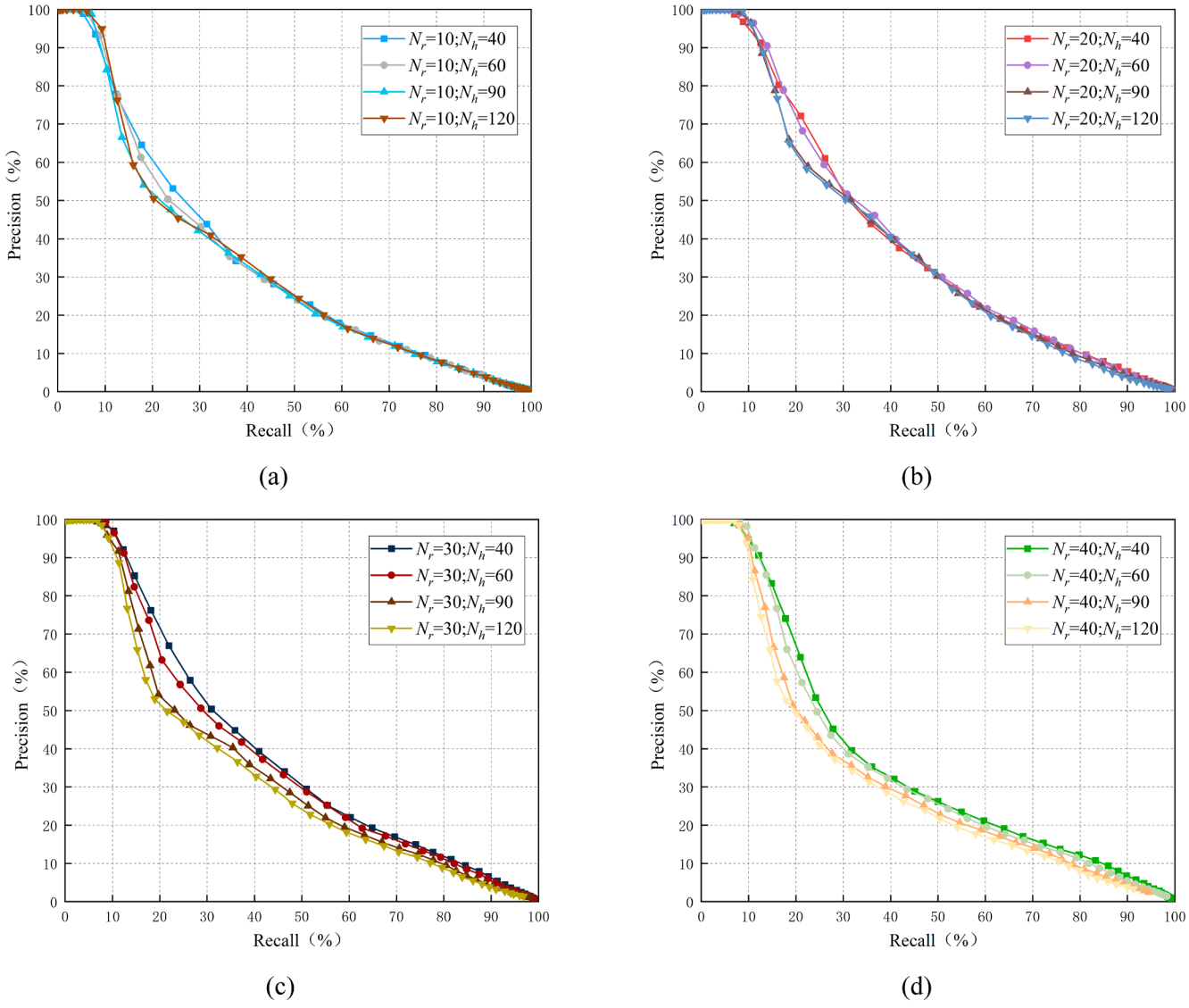

Fig. 11. The PR curves of PGHCI with different parameters.

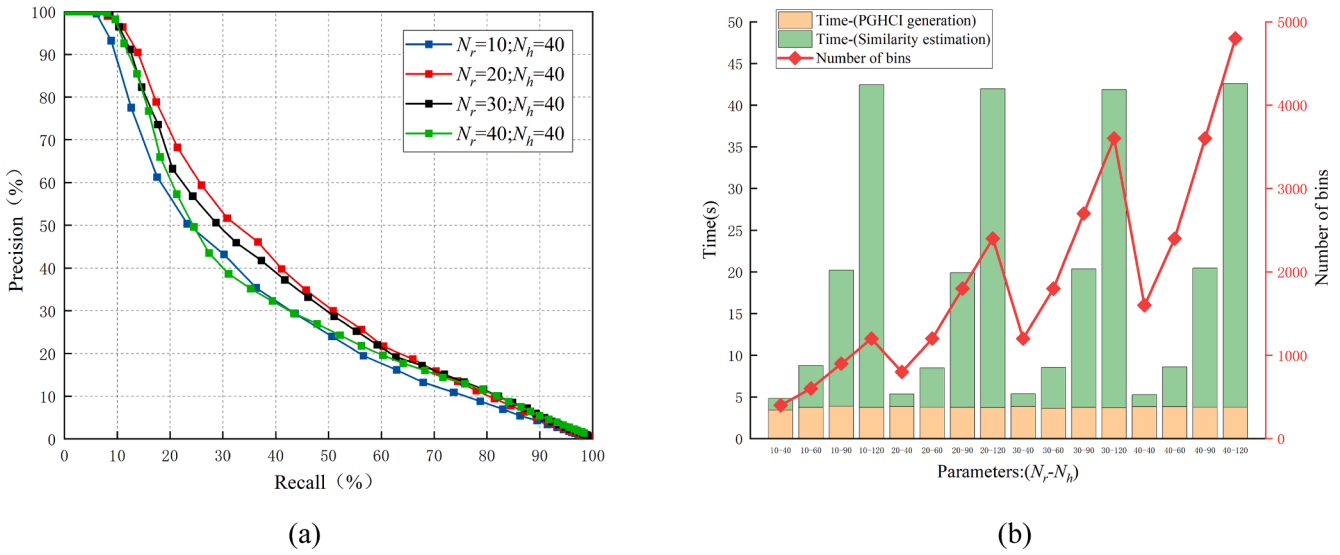

Fig. 12. (a) The PR curves of PGHCI with different parameters. (b) The time performances of PGHCI with different parameters.

can write the distance as:

$$\left.\begin{aligned} PN1 &= \frac{P}{255} \\ c_j^q \in I^q, c_j^c &\in I^c \end{aligned}\right) \qquad (7)$$
$$M1 = \frac{1}{N_h}\sum_{j=1}^{N_h}\left(1 - \frac{c_j^q \bullet c_j^c}{\|c_j^q\|\|c_j^c\|}\right)$$

where P is the pixel value, $PN1$ is the normalized pixel value in the first way, I^q is the normalized image of the query scan, I^c is the normalized image of the candidate scan,c_j^q is the *jth* clowns of I^q, and$M1$ is the distance calculated using the first approach. The second method is given by:

$$PS2 = 1 - \frac{cnt(toBinary(P1)^t oBinary(P2))}{8} \qquad (8)$$
$$M2 = \frac{1}{\mathrm{N}}\sum_{i=1}^{N} PS2_i$$

where $PS2$is the normalized similar value in the second way, *toBinary()* is the function that converts pixel value into binary, and *cnt()* is the function that counts the number of 1.$M2$ is the distance calculated using the second approach.

For example, for P1: 142, its binary is 10001110, and for P2: 200, its binary is 11001000. In the first approach, the normalized similar value between P1 and P2 is 0.773. In the second approach, the normalized similar value is 0.625. We can see the interpretation in Fig. 10.

The range of distance is (Wu et al., 2018); and a smaller distance indicates that the two PGHCI descriptors are more similar. In fact, M1 and M2 are two different weighted approaches. In the second approach, the weights at different heights are the same. By contrast in the first approach, greater height has more weight. In a real scene, according to the data from different rotating multibeam LiDARs, different distances have their own advantages. More analysis details can be found in experiment Section 4.2.2.

3.2.4. Parameters analysis of the PGHCI descriptor

First, considering that the effective observation range of a rotating multibeam LiDAR is approximately 100 m and the sparsity of far points, we set the maximum sensing range of the LiDAR sensor to 80 m.

N_h and N_r are the important parameters of the PGHCI descriptor. These parameters affect the size of bins. Larger N_h and N_r will make the bin smaller, indicating that the PGHCI descriptor is more descriptive but is also more sensitive to noise. In addition, more time needs to be spent calculating the similarity of two PGHCI descriptors for place recognition. By contrast, smaller N_h and N_r with larger bins are more robust to noise and spend less time matching. Consequently, the value of N_h and N_r should be set to achieve the trade-off between the descriptor's descriptiveness and efficiency.

To determine the appropriate values of N_h and N_r, we examine the performance of the PGHCI descriptor on the KITTI data 00 (the dataset details can be found in Section 4.1.1) under different values of N_h and N_r. The PR curve (evaluation criteria can be found in Section 4.1.2) is used to assess the performance. Using four N_h and four N_r ($N_h \in \{40,60,90,120\}$, $N_r \in \{10,20,30,40\}$), a total of 16 experiments were performed.

When the value of N_r is 10 (Fig. 11(a)), the value of N_h is increased from 40 to 120. It is observed that the performance becomes increasingly worse with increasing values of N_h. A similar trend is observed in Fig. 11(b)–(d). When the values of N_r are constant, the performance improves with smaller N_h values. This indicates that the large horizontal angle range plays an important role in the descriptiveness.

Fig. 12(a) shows the PR curves for varying values of N_r when the value of N_h is 40. As shown in Fig. 12(a), the worst performance is obtained when the value of N_r is 10. Too large bins hide many details, affecting the descriptiveness of the PGHCI descriptors. For the N_r values of 20, 30 and 40, the PGHCI descriptors have comparable performance. However, when the value of N_r is 20, there are only 800 bins within the PGHCI descriptor. This is beneficial for saving the time in the calculation of similarity, as shown in Fig. 12(b). Therefore, we set $N_r = 20$, $N_h = 40$ in the paper. In subsequent experiments, we found that the performance of the PGHCI descriptor with selected parameters was more robust than those of the other methods on several datasets from different sensors (64-beam LiDAR and 16-beam LiDAR). This indicates that our selected parameters have high generalizability. If place recognition is needed in a new dataset with higher resolution, the PGHCI descriptor will have more descriptiveness with slightly larger N_h and N_r. Because the resolution is high enough, a too large bin will hide many details. By contrast, if place recognition is needed in a new dataset with lower resolution, slightly smaller N_h and N_r will be good choices.

4. Experiment

In this section, our global feature descriptor is evaluated over three datasets against two similar open source state-of-the-art algorithms:

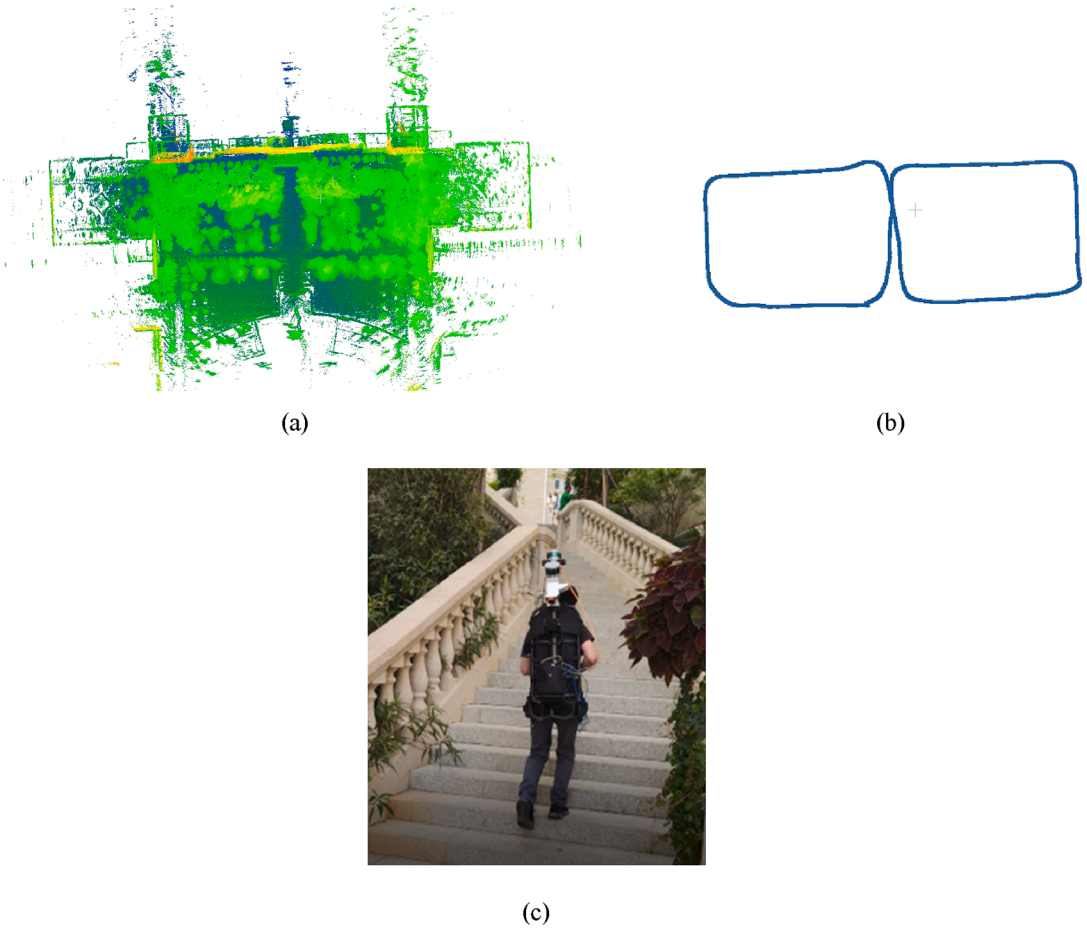

Fig. 13. Our own dataset collected at our campus. (a) The reconstructed 3D map. (b) Trajectory chart of the people walking with a backpack mobile laser system. (c) Our backpack mobile laser scanner.

Scan Context (Kim and Kim, 2018) and LiDAR IRIS (Wang et al., 2019). In addition, two existing heterogeneous 3D maps are used to evaluate our place recognition solution.

4.1. Experimental setup

In this section, the datasets and evaluation criteria are demonstrated.

4.1.1. Datasets

Experiments evaluating descriptor performance are carried out on three datasets: KITTI odometry sequences (Geiger et al., 2012), Queensland Centre for Advanced Technologies (QCAT) 3D map (Guo et al., 2019) and our own dataset. These datasets have a clear differences, such as the type of LiDAR sensors used and changed or unchanged viewpoint at the same place.

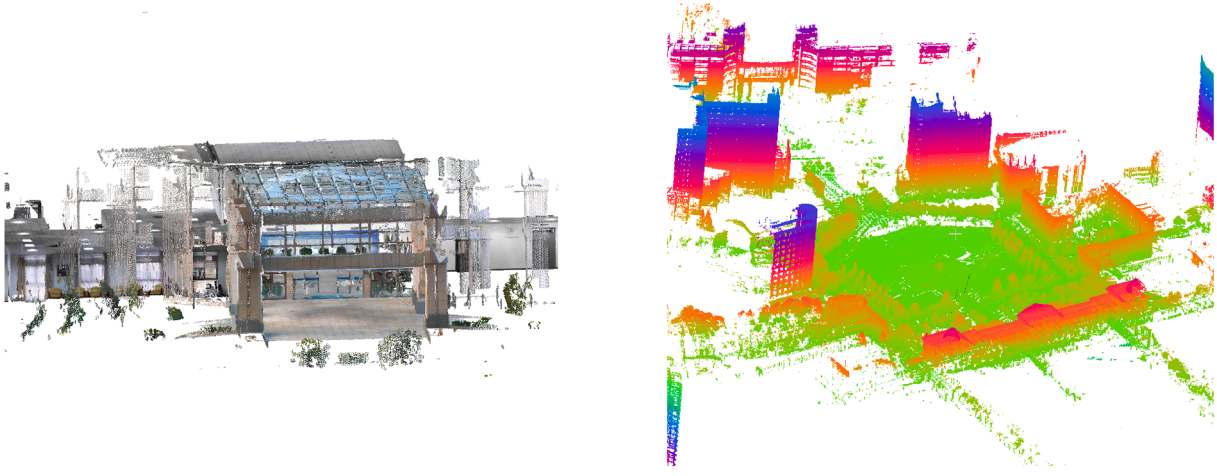

Fig. 14. (a) A 3D map of an indoor scene. (b) A 3D map of an outdoor scene.

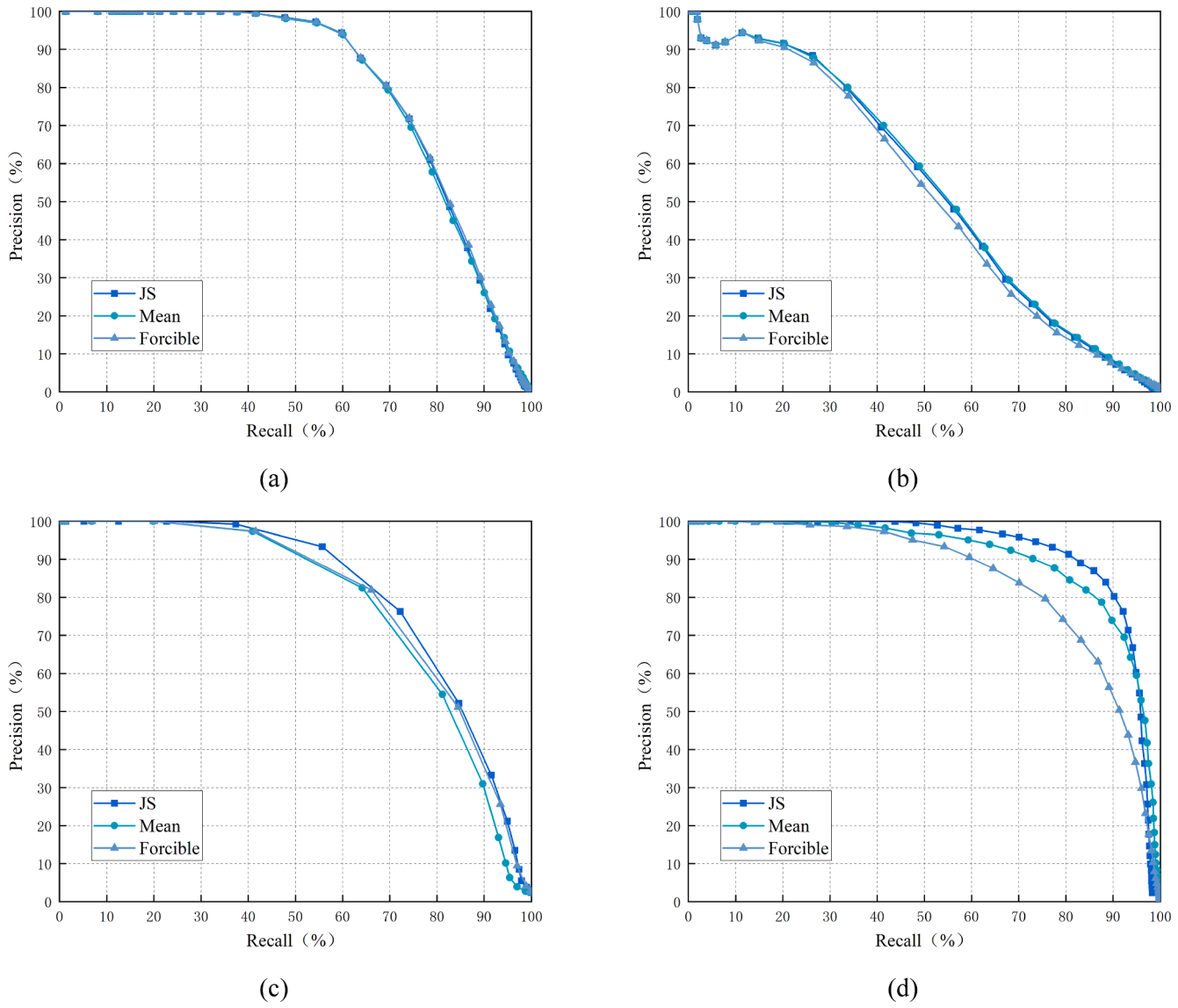

Fig. 15. The PR curves of PGHCI with different methods for overcoming changed viewpoint in the same scene. (a) KITTI sequence 05; (b) KITTI sequence 08; (C) our own dataset; (d) the QCAT dataset.

We selected three KITTI odometry sequences 00, 05 and 08 that contained 4541, 2761 and 4071 scans, respectively. 64-beam LiDAR (Velodyne HDL-64E) was used for data collection. In sequences 00 and 05, the viewpoint is unchanged when the vehicle revisits the place. In the sequence 08, the viewpoint is changed. A QCAT 3D map is generated by the continuous-time SLAM algorithm (Bosse and Zlot, 2009) on the "Gator" platform (Romero et al., 2016). It can be divided 2055 scans. The 16-beam LiDAR (Velodyne VLP-16C) is mounted above the vehicle and is rotated by a motor at a 45° angle. In this dataset, the robot moves in both the same and opposite directions at revisited places. Our 3D map is generated by Google's cartographer (Hess et al., 2016) on the campus; as shown in Fig. 13(a). It contains 2946 scans. The 16-line LiDAR (Velodyne VLP-16C) is mounted above the backpack mobile laser scanner, as shown in Fig. 13(c). The high-fidelity IMU and GPS are integrated into the backpack mobile laser system, ensuring a high precision for the location and posture of each scan. In this dataset, the people move in the same direction at revisited places. The resolution of our own dataset is lower than that of the QCAT dataset.

The existing heterogeneous 3D maps include indoor scenes and outdoor scenes, as shown in Fig. 14. The indoor 3D map was captured on the first floor of a building at Wuhan University and was derived from Navvis m3. Navvis m3 is a trolley mobile laser scanner composed of three 2D laser range finders and other sensors. Two 2D laser range finders are mounted vertically to capture the point cloud. The outdoor 3D map is the WHU-TLS campus dataset (Dong et al., 2020) that was captured at the Friendship Square at Wuhan University using the RIEGL VZ-400. RIEGL VZ-400 is a TLS. Indoor and outdoor heterogeneous 3D maps contain 21.83 million and 109.05 million points; respectively.

4.1.2. Evaluation criteria

The performance of our PGHCI descriptor is evaluated using a precision-recall (PR) curve. The PR curve is obtained by calculating the precision and recall under different thresholds.

$$\begin{aligned} \text{Precision} &= \frac{numberofcorrectmatches}{totalnumberofmatches} \\ \text{Recall} &= \frac{numberofcorrectmatches}{totalnumberofcorrespondingmatches} \end{aligned} \tag{9}$$

where the *number of correct matches* refers to the number of pairs whose Euclidean distance is smaller than 4 m and their descriptor similarity is less than the threshold. *The total number of matches* refers to the number

(a) (b)

Fig. 16. The PGHCI descriptor for scans of the different scenes. (a) The PGHCI descriptor of a scan of the narrow road scene. (b) The PGHCI descriptor of a scan of the spacious scene.

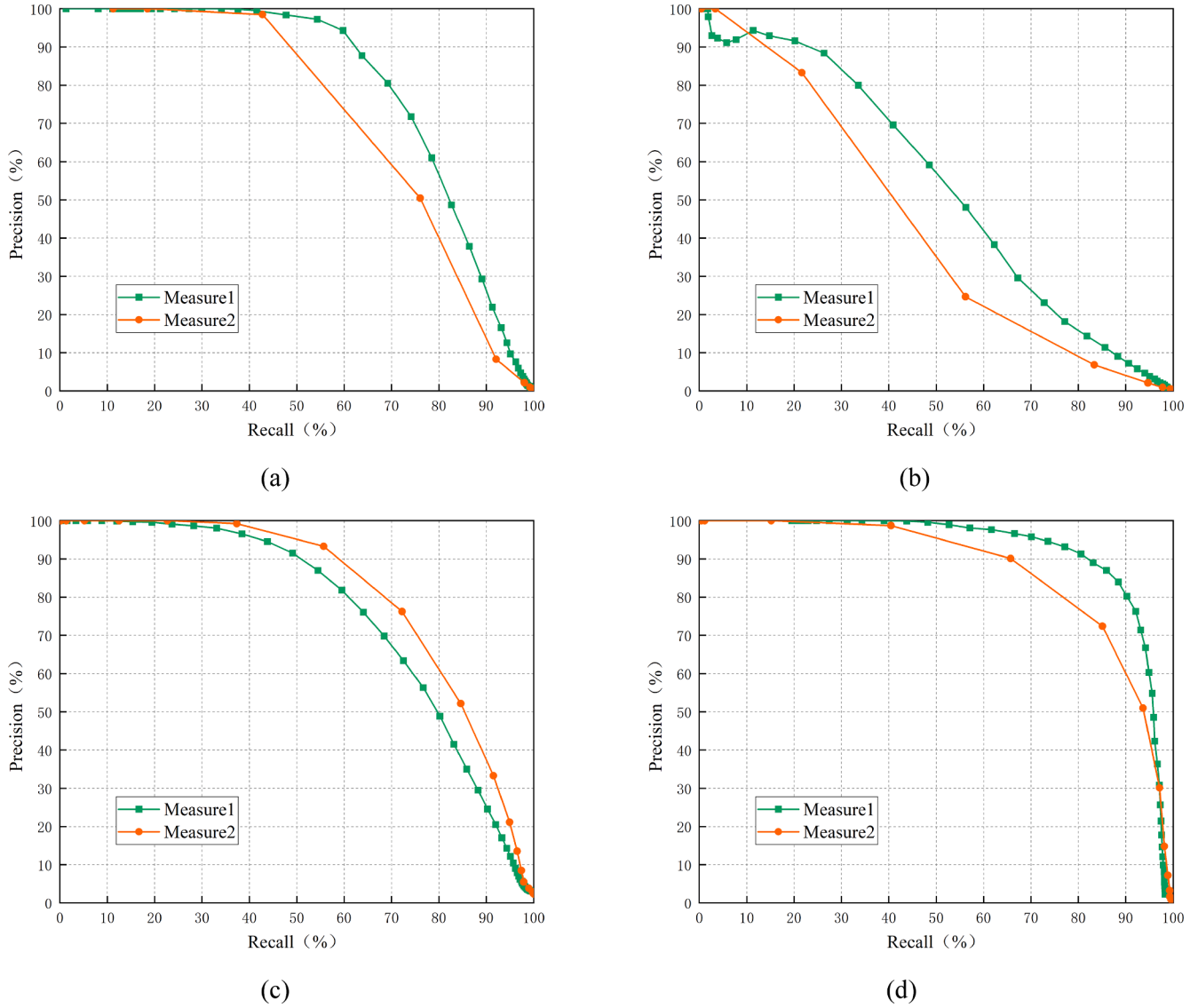

Fig. 17. The PR curves of PGHCI with different weight distances. (a) KITTI sequence 05; (b) KITTI sequence 08; (C) our own dataset; (d) QCAT dataset.

of pairs in which their descriptor similarity is less than the threshold. *The total number of corresponding matches* refers to the number of pairs whose Euclidean distance is smaller than 4 m. We note that the distance of 4 m is set as the default according to (Kim and Kim, 2018; Wang et al., 2019).

The performance of our place recognition solution is evaluated using the success rate. The success rate is equal to the precision with a 100% value of recall.

4.2. Evaluation of PGHCI descriptor

In this section, the method for overcoming changed viewpoints in the same scene, two weighted distances that calculate the similarity of two descriptors and the performance of our PGHCI descriptor are evaluated. To conduct a more complete experiment, we calculate the similarity between the selected scan and all of the remaining scans. For example, the similarity estimation will be conducted 10 ($4 + 3 + 2 + 1 + 0$) times

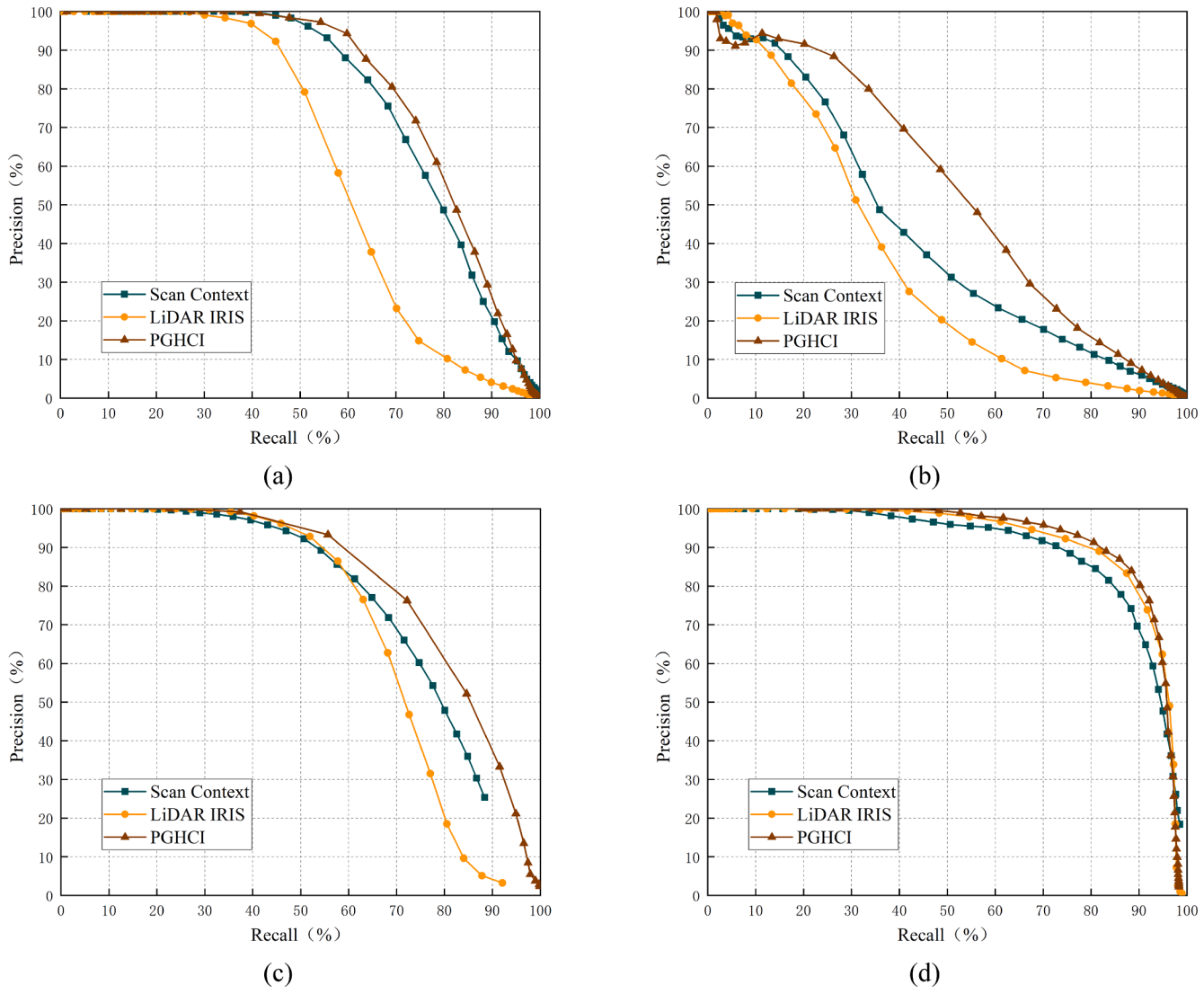

Fig. 18. The PR curves of different descriptors. (a) KITTI sequence 05; (b) KITTI sequence 08; (C) our own dataset; (d) QCAT dataset.

if there are 5 scans in the dataset. However, if there are 4500 scans in the dataset, the similarity estimation will be conducted 10,127,250 times. This is time-consuming. Therefore, we made a trade-off between time and number of experimental times. In the KITTI odometry sequences and our own dataset, we pick a frame every three frames. Sequence 00 is used to assess the parameters of our PGHCI descriptors, and the other datasets are used to evaluate the performance of the three descriptors. We use Scan Context implemented in MATLAB[1] and LiDAR IRIS implemented in C++[2].

4.2.1. *Performance of PGHCI with different rotation-invariance methods*

In this section, three methods that overcome the difficulty of recognition of changed viewpoints in the same scene are evaluated. The first method adopts the JS divergence proposed in this paper. The second method obtains rotational invariance using the mean of each column within the PGHCI descriptor. The mean value is used instead of the JS divergence to estimate the similarity between two corresponding columns. The third method transforms the PGHCI descriptors and calculates the similarity until the smallest similarity is obtained. The first, second and third methods are referred to as JS, Mean and Forcible, respectively. The PR curve of the PGHCI descriptor was used as an assessment index, as shown in Fig. 15. After acquiring rotational invariance, similarity estimation is necessary for the PR curve. To control the variable, the best distance for similarity estimation is adopted.

It is observed from Fig. 15 that the three methods perform equally well for KITTI sequence 05. For KITTI sequence 08, the forcible method shows slightly worse performance. For our own dataset and the QCAT dataset, the JS method shows the best performance, followed by the Mean method and the Forcible method shows the worst performance. For KITTI sequences 05 and 08, narrow roads cause many columns within PGHCI descriptors to lack content, as shown in Fig. 16(a). As a result, there is no obvious gap between the three methods. The major scans for our own dataset and QCAT dataset are collected in the spacious scene, making each column within PGHCI descriptors contain abundant information, as shown in Fig. 16(b). Therefore, the JS method shows the best performance.

4.2.2. *Performance of PGHCI with two weighted distance*

In this section, two distances that calculate the similarity of two descriptors for place recognition are evaluated. The first method is to adopt pixel values for direct calculation. The second method is to convert each pixel value into binary values for calculation. We called the first method measure1 and the second method measure2. The PR curve

[1] https://github.com/irapkaist/scancontext.
[2] https://github.com/JoestarK/LiDAR-Iris.

of the PGHCI descriptor as an evaluation index. To control the variable, the JS divergence is adopted for rotational invariance.

As shown in Fig. 17, measure1 shows better performance on the KITTI sequences 05, 08 and QCAT datasets. For our own dataset, the performance of the PR curve is better with measure2. The common feature of the KITTI sequences 05, 08 and QCAT is that they have a higher resolution of point clouds. By contrast the resolution of our own dataset is lower. High-resolution point clouds contain redundant information. For example, the redundant points of ground make little difference for the result of place recognition and the redundant points of moving vehicles have the opposite effect for the result. By contrast, the higher points of immobile buildings and trees play an important role in place recognition. Therefore, measure1 with a higher weight in higher positions shows better performance with the high resolution of the point cloud. If the resolution of the point cloud is low, information about each altitude is indispensable. Because the hither points are not abundant, if we reduce the weight of the lower points, it will make the result worse. Therefore, measure2 with equal weight performs better in the low resolution of the point cloud.

4.2.3. *Performance of PGHCI against with other descriptors.*

In this section, our global feature descriptor PGHCI is evaluated against two similar open source state-of-the-art algorithms, Scan Context and LiDAR IRIS.

As shown in Fig. 18, our PGHCI descriptor shows better performance than other descriptors in all datasets. This indicates that the PGHCI descriptor has greater descriptiveness. Particularly for sequence 08 (the viewpoint is changed), our PGHCI descriptor performs far better than the other descriptors. This indicates that the PGHCI descriptor is robust with respect to the changed viewpoint scene. The Scan Context assigns the value of each bin by the maximum height of points within the bin. It is clear that Scan Context encodes less information than the PGHCI descriptor. In addition, Scan Context acquires rotational invariance in a forcible way that worsens JS. Therefore, PGHCI performs better than Scan Context. The LiDAR IRIS obtains a binary signature image for each point cloud after several LoG-Gabor filtering and thresholding operations on the LiDAR-IRIS image. Eight LoG-Gabor filters with different parameters were tested on a validation dataset for proper parameters. However, there is no reasonable explanation for the selected parameters in the paper. We find that LiDAR IRIS performs worst for the datasets that are downsampled from the original datasets. For KITTI sequences 05 and 08 and our own dataset, we pick a frame every three frames. This means that the distances between two adjacent frames increase, increasing the difficulty of place recognition. The QCAT dataset remains unchanged, so that the distances between two adjacent frames in the QCAT dataset are smaller. LiDAR IRIS performs better than Scan Context for the QCAT dataset, indicating that LiDAR IRIS with selected parameters in the paper performs well on the simple dataset.

4.3. *Evaluation of place recognition solution in existing heterogeneous 3D maps*

In this section, the place recognition solution using virtual LiDAR and PGHCI descriptors in existing heterogeneous 3D maps is evaluated. The two existing heterogeneous 3D maps described in Section 4.1.1 are used to evaluate our solution. The 16-line LiDAR (Velodyne VLP-16C) is selected as the localization sensor. It is used to obtain real scans in the same scene as the above existing 3D map. The ground truth of real scans can be obtained by registration in a manual way. The corresponding place of real scans can be obtained through our solution.

The first step of our solution is to generate a series of virtual scans that are similar to the real scan of the localization sensor (here, VLP-16C) from two existing maps. Six and seven requisite control points are selected to generate the locations of virtual LiDAR in the indoor and outdoor maps, respectively. The interval range of each location of virtual LiDAR is set to 3 m. In the existing heterogeneous 3D map, 114

Fig. 19. The locations of real scans, virtual scans, and control points. In the figure, the blue point is the location of the virtual scan; the red point is the location of the picked control point; the origin is the location of the real scan. To display the location more clearly, the points of ground and roof are removed in the indoor scene and the points of ground are removed in the outdoor scene. (a) A top view of the 3D map of the indoor scene; (b) top view of the 3D map of the outdoor scene. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

locations are generated to set virtual LiDAR. There are 26 locations of virtual LiDAR in the indoor scene and 88 locations of virtual LiDAR in the outdoor scene, as shown in Fig. 19. The postures of virtual LiDAR are set to zero. The parameters of VLP16 are as follows: measurement range 100 m, vertical FOV 30°, resolution of vertical angle 2° and resolution of horizontal angle 0.3°. The parameters of virtual LiDAR are set according to the above parameters.

After obtaining the location, posture and parameters of virtual LiDAR, we can generate virtual scans based on existing heterogeneous 3D maps. A real scan and a virtual scan are shown in Fig. 20. The size of the original heterogeneous 3D indoor map from Navvis m3 is 680.2 Mb, and the size of the corresponding virtual scans is 1.8 Mb. The size of the original heterogeneous 3D outdoor map from TLS is 3.45 Gb, and the size of the corresponding virtual scans is 5.0 Mb. Although our virtual scans cover a partial passable region of a heterogeneous 3D outdoor map, an enormous gap in the size of the data shows that virtual LiDAR efficiently compresses the data size.

The second step of our solution is to match the real scan with all of the virtual scans. To find the corresponding virtual scan, generation of PGHCI descriptors, rotation transformation and similarity estimation are conducted. The minimum distance between the PGHCI descriptors of the real scan and virtual scans can be obtained after repeating the above procedures. In addition, the Euclidean distance corresponding to the

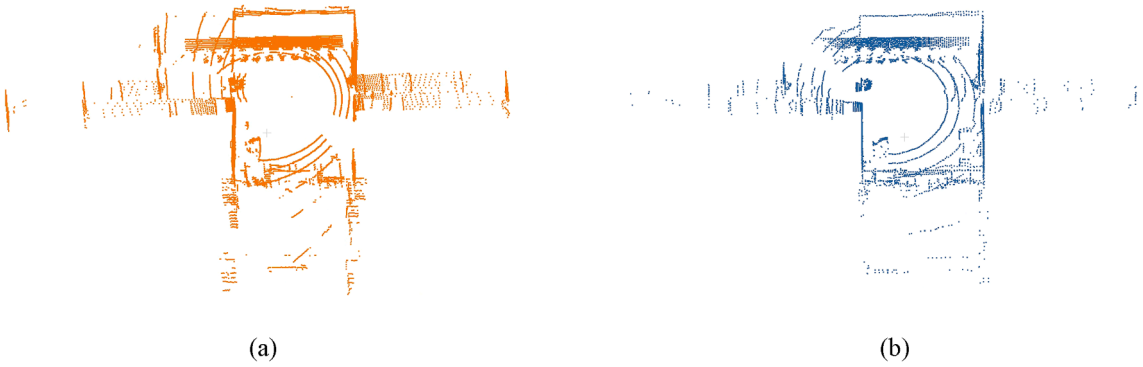

Fig. 20. (a) A real scan from VLP-16C; (b) a virtual scan from virtual LiDAR.

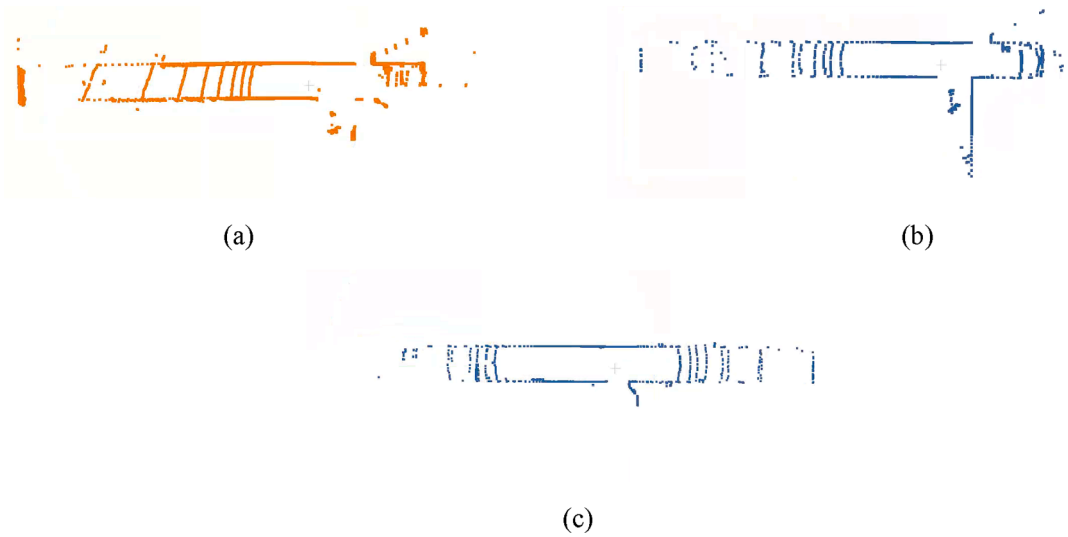

Fig. 21. (a) Top view of real scan 4 that missed partial 3D information of scene; (b) top view of the virtual scan at the location nearest to the location of real scan 4; (c) top view of the virtual scan under the corridor scene.

Table 1
Success rate of the proposed solution under indoor and outdoor scenes.

Scene	Scan	Minimum distance between two PGHCI	Corresponding Euclidean distance(m)	Success rate
indoor	real scan1	0.07	1.26	100%
	real scan2	0.13	0.53	100%
	real scan3	0.26	1.27	100%
	real scan4	0.17	6.4	6.67%
outdoor	real scan1	0.33	0.14	100%
	real scan2	0.27	0.21	100%
	real scan3	0.26	0.07	100%
	real scan4	0.28	0.95	100%

minimum distance can also be obtained. Table 1 lists the related details.

The interval range of each virtual scan is set to 3 m. Therefore, the Euclidean distance used to calculate the success rate was set to 1.5 m to ensure that each real scan had only one corresponding virtual scan. In other words, if the corresponding Euclidean distance is smaller than 1.5 m, a unique virtual scan corresponding to a real scan can be found. This indicates that a 100% success rate can be obtained. The figures for the difference value (DV) of distances are obtained as shown in Figs. 22 and 23. The DV is the difference value between all distances and the minimum distance.

As shown in Fig. 22 and Table. 1, we find that the success rate of place recognition is 100% with real scan1, real scan2 and real scan3 in the indoor scene. This indicates that place recognition is accomplished through our solution in real scan1, real scan2 and real scan3. Place recognition failed with real scan4 in the indoor scene. When the unique virtual scan corresponding to real scan4 is found, there are 14 error matches. Therefore, the success rate of real scna4 is 6.67%. As a person stands close to the LiDAR with a distance of less than 0.5 m, the laser is occluded partially during the scanning process. As a result, real scan 4 missed partial 3D information of the scene, making the whole points similar to a corridor, as shown in Fig. 21(a). Insufficient 3D data result in incorrect matching pairs with the virtual scans generated in the corridor scene, as shown in Fig. 21(c). The virtual scan at the location nearest to the location of real scan 4 is shown in Fig. 21(b).

As shown in Fig. 23 and Table. 1, we observe that the success rate of place recognition is 100% for all real scans in the outdoor scene. This indicates that place recognition is accomplished in the outdoor scene through our solution. Outdoor scenes have more features than indoor scenes. There are more similar structures in the indoor scene. Generally, it is feasible to use virtual LiDAR for place recognition in existing heterogeneous 3D maps.

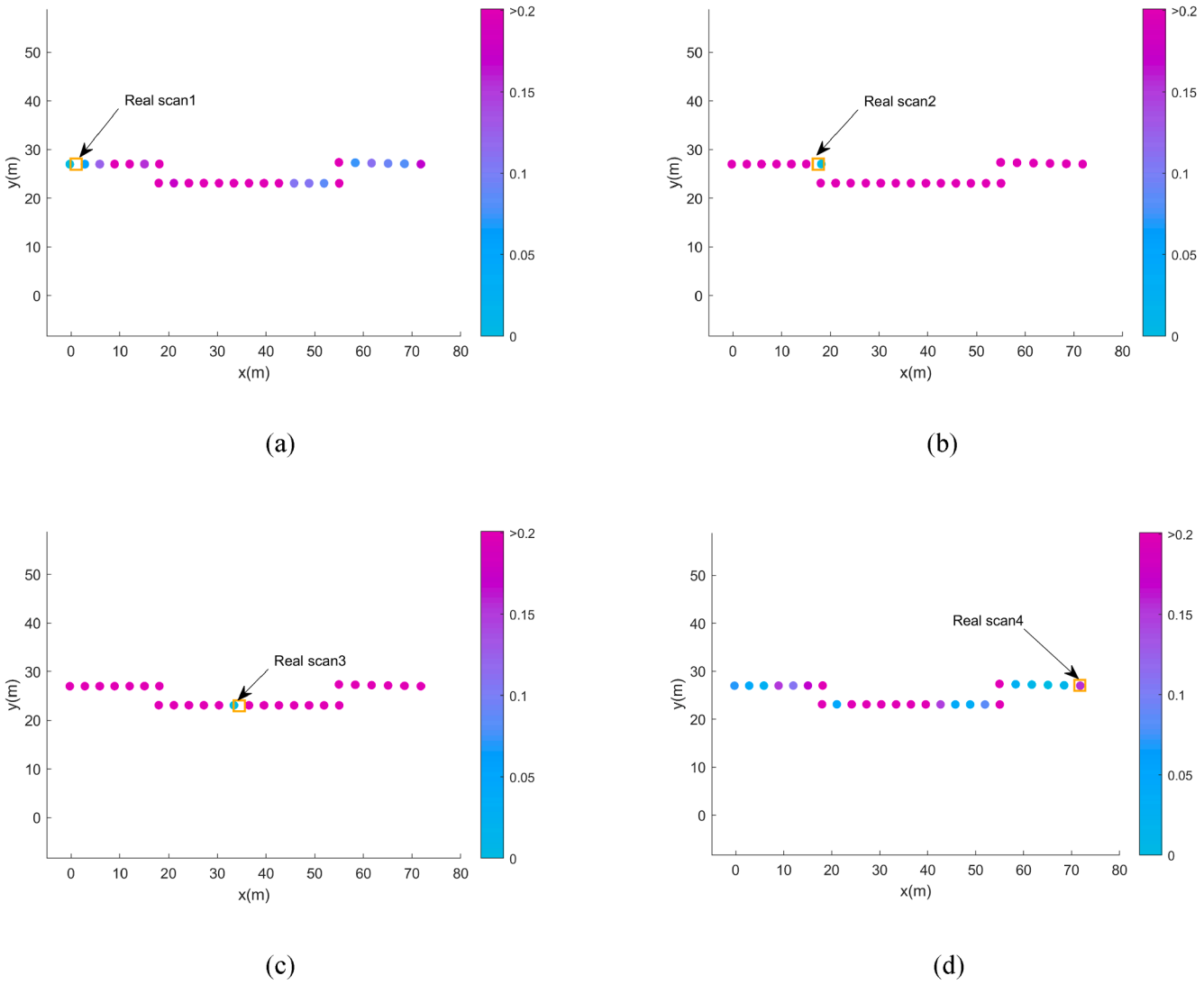

Fig. 22. DV figures of real scans in the indoor scene. The locations of virtual scans are shown by circles and locations of real scans are shown by squares. (a) DV figure of real scan1; (b) DV figure of real scan2; (c) DV figure of real scan3; (d) DV figure of real scan4;

5. Conclusions

The current place recognition methods require the use of the same type of sensors in the mapping and localization processes. Therefore, the heterogeneous 3D map cannot be used for place recognition directly with the existing method. This requirement restricts the applicability of existing methods, as it is highly likely that different types of sensors are used in mapping and localization processes. This paper presented a solution using virtual LiDAR and a polar grid height coding image descriptor for place recognition in the existing heterogeneous 3D map.

The proposed virtual LiDAR allows for the use of different types of LiDAR in the processes of 3D map creation and localization. Through generating virtual scans that are similar scans of real LiDAR, this method overcomes the limitation of the existing place recognition methods that require the use of the same types of sensors in the mapping and localization processes. In addition, virtual LiDAR can compress the data size of existing heterogeneous 3D maps by generating virtual scans. We compare the size of all virtual scans with that of the original 3D map and find that our method significantly reduces the data size.

The proposed PGHCI descriptor adopts a polar grid and encodes a 3D point cloud into a 2D image. The polar grid endows the PGHCI descriptor with rotational invariance and compensates for the insufficient information due to the sparsity of far points. The value of each bin is assigned according to the height distribution of the points in the bin. All of the above considerations provide more descriptiveness for the descriptor, promoting place recognition. The adoption of JS divergence further increases the robustness of the descriptor, particularly under changed viewpoints. Comprehensive experiments demonstrated that the PGHCI descriptor has higher descriptiveness and is robust to the changed viewpoint scenes, as shown by the comparison of the PR curves using datasets with multiple scenes. We also find that the JS method is a more accurate method for estimating the rotation transform between two PGHCI descriptors. In addition, the applicability of two weighted distances for different types of sensors with two weighted distances is analyzed. The analysis can serve as prior knowledge for place recognition using different LiDAR sensors.

We integrated virtual LiDAR and PGHCI descriptors into the proposed solution for place recognition in the heterogeneous 3D map. The experiments show that the proposed place recognition solution has a high success rate of up to 100% in the evaluation, except in the exceptional cases where scanning is mostly occluded in the localization process and place recognition may fail completely. Therefore, the proposed place recognition solution is feasible for achieving robust place recognition using a heterogeneous 3D map, given a satisfactory scanning condition.

Similar to other 3D point cloud-based place recognition solutions,

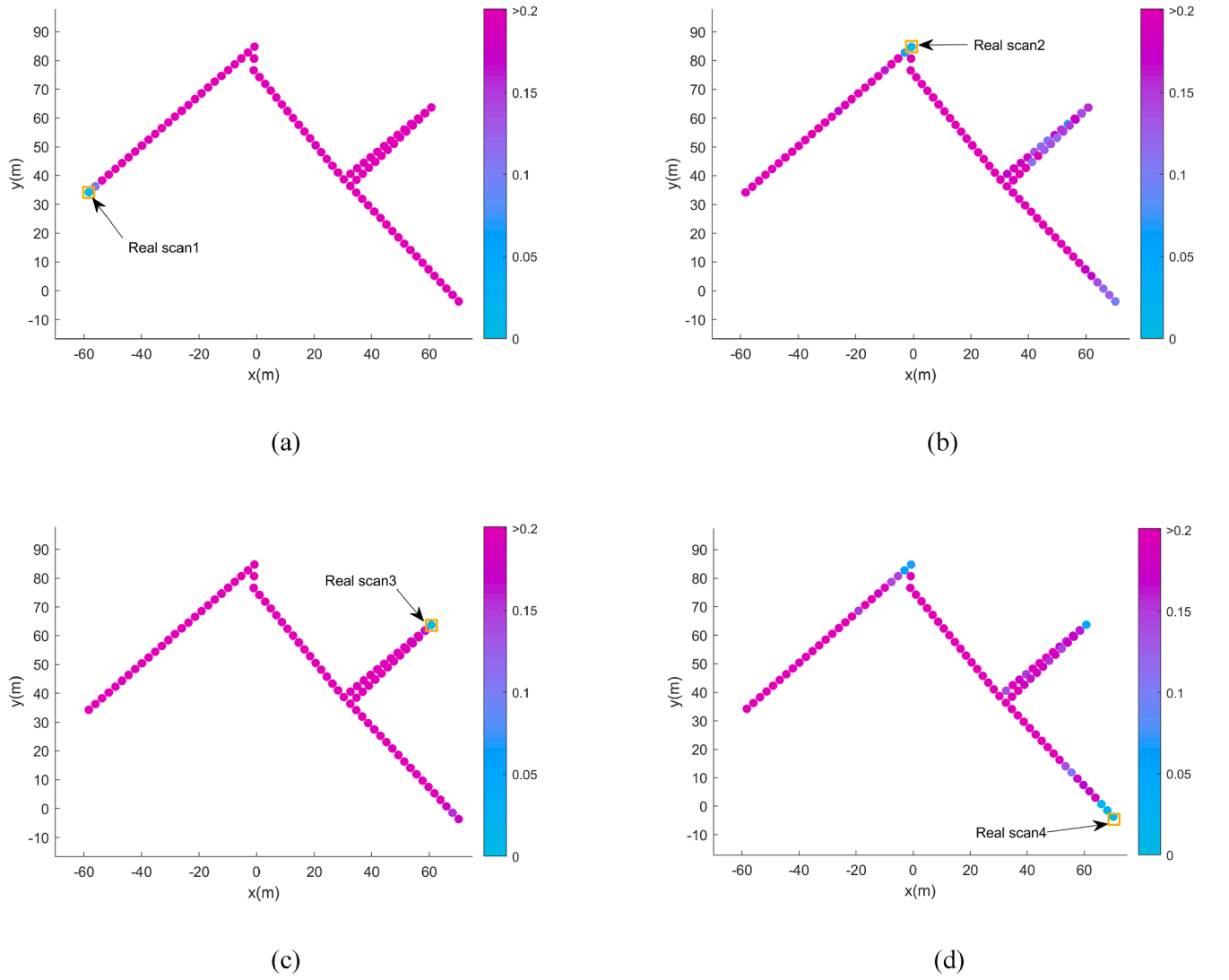

Fig. 23. DV figures of real scans of the outdoor scene. The positions of virtual scans are shown by circles and the locations of real scans are shown by squares. (a) DV figure of real scan1; (b) DV figure of real scan2; (c) DV figure of real scan3; (d) DV figure of real scan4;

the proposed solution is not sufficiently robust in scenes that have similar structures. For example, when we conduct place recognition in a building that has multiple floors with similar structures, false results may be obtained. The supplementation of texture information may help to improve the robustness of place recognition in this condition. We will integrate texture information into the generation of descriptors in future work. In addition, the approach of global localization will be developed based on our place recognition results. Our method can be used to provide an initial position for coarse registration and fine registration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study was supported in part by the Natural Science Fund of China with Project No. 41874031 and 42111530064, the National Key Research Development Program of China with project No.2016YFB0502204, and Academy of Finland with Project No. 337656.

References

Wu, T., Liu, J., Li, Z., Liu, K., Xu, B., 2018. Accurate smartphone indoor visual positioning based on a high-precision 3D photorealistic map. Sensors (Switzerland) 18 (6). https://doi.org/10.3390/s18061974.

Wang, Z., Zhang, Q., Li, J., Zhang, S., Liu, J., 2019. A computationally efficient semantic SLAM solution for dynamic scenes. Remote Sens. 11 (11), 1–19. https://doi.org/10.3390/rs11111363.

He, L., Wang, X., Zhang, H., 2016. M2dp: A novel 3D point cloud descriptor and its application in loop closure detection. IEEE Int. Conf. Intell. Robot. Syst. vol. 2016-Novem, 231–237. https://doi.org/10.1109/IROS.2016.7759060.

Kim, G., Kim, A., 2018. Scan Context: Egocentric Spatial Descriptor for Place Recognition Within 3D Point Cloud Map. IEEE Int. Conf. Intell. Robot. Syst. 4802–4809. https://doi.org/10.1109/IROS.2018.8593953.

Wang, Y., Sun, Z., Yang, J., Kong, H., 2019. LiDAR Iris for loop-closure detection. arXiv 5769–5775.

Guo, J., Borges, P.V.K., Park, C., Gawel, A., 2019. Local Descriptor for Robust Place Recognition Using LiDAR Intensity. IEEE Robot. Autom. Lett. 4 (2), 1470–1477. https://doi.org/10.1109/LSP.2016.10.1109/LRA.2019.2893887.

Cop, K.P., Borges, P.V.K., Dube, R., 2018. Delight: An Efficient Descriptor for Global Localisation Using LiDAR Intensities. Proc. - IEEE Int. Conf. Robot. Autom. 3653–3660. https://doi.org/10.1109/ICRA.2018.8460940.

Liu, J., Liang, X., Hyyppä, J., Yu, X., Lehtomäki, M., Pyörälä, J., Zhu, L., Wang, Y., Chen, R., 2017. Automated matching of multiple terrestrial laser scans for stem mapping without the use of artificial references. Int. J. Appl. Earth Obs. Geoinf. 56, 13–23. https://doi.org/10.1016/j.jag.2016.11.003.

Liu, J., Xu, D., Hyyppa, J., Liang, Y., 2021. A survey of applications with combined BIM and 3D laser scanning in the life cycle of buildings. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 14, 5627–5637. https://doi.org/10.1109/JSTARS.2021.3068796.

Zhang, J., Singh, S., 2017. Low-drift and real-time lidar odometry and mapping. Auton. Robots 41 (2), 401–416. https://doi.org/10.1007/s10514-016-9548-2.
Gusmão, G.F., Barbosa, C.R.H., Raposo, A.B., 2020. Development and validation of lidar sensor simulators based on parallel raycasting. Sensors (Switzerland) 20 (24), 1–18. https://doi.org/10.3390/s20247186.
Bosché, F., Guillemet, A., Turkan, Y., Haas, C.T., Haas, R., 2014. Tracking the built status of MEP works: Assessing the value of a Scan-vs-BIM system. J. Comput. Civ. Eng. 28 (4), 1–14. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000343.
Bosché, F., Ahmed, M., Turkan, Y., Haas, C.T., Haas, R., 2015. The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components. Autom. Constr. 49, 201–213. https://doi.org/10.1016/j.autcon.2014.05.014.
Bosché, F., Guenet, E., 2014. Automating surface flatness control using terrestrial laser scanning and building information models. Autom. Constr. 44, 212–226. https://doi.org/10.1016/j.autcon.2014.03.028.
Ma, L., Sacks, R., Zeibak-Shini, R., Aryal, A., Filin, S., 2016. Preparation of Synthetic As-Damaged Models for Post-Earthquake BIM Reconstruction Research. J. Comput. Civ. Eng. 30 (3), 1–12. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000500.
X. Yue, B. Wu, S. A. Seshia, K. Keutzer, and A. L. Sangiovanni-Vincentelli, "A LiDAR point cloud generator: From a virtual world to autonomous driving," *ICMR 2018 - Proc. 2018 ACM Int. Conf. Multimed. Retr.*, pp. 458–464, 2018, doi: 10.1145/3206025.3206080.
Hanke, T., et al., 2018. Generation and validation of virtual point cloud data for automated driving systems. IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC vol. 2018-March, 1–6. https://doi.org/10.1109/ITSC.2017.8317864.
Zhao, J., Li, Y., Zhu, B., Deng, W., Sun, B., 2021. Method and applications of LiDAR modeling for virtual testing of intelligent vehicles[J]. IEEE Transactions on Intelligent Transportation Systems 22 (5), 2990–3000.
Linnhoff, C., Rosenberger, P., Holder, M.F., Cianciaruso, N., Winner, H., 2020. Highly Parameterizable and Generic Perception Sensor Model Architecture.
Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., 2014. 3D object recognition in cluttered scenes with local surface features: A survey. IEEE Trans. Pattern Anal. Mach. Intell. 36 (11), 2270–2287. https://doi.org/10.1109/TPAMI.3410.1109/TPAMI.2014.2316828.
Yang, J., Zhang, Q., Cao, Z., 2016. The effect of spatial information characterization on 3D local feature descriptors: A quantitative evaluation. Pattern Recognit. 66 (January) https://doi.org/10.1016/j.patcog.2017.01.017.
X.-F. Han, S.-J. Sun, X.-Y. Song, and G.-Q. Xiao, 3D point cloud descriptors in hand-crafted and deep learning age: State-of-the-art[J]. arXiv preprint arXiv:1802.02297, 2018.
Yang, J., Quan, S., Wang, P., Zhang, Y., 2020. Evaluating Local Geometric Feature Representations for 3D Rigid Data Matching. IEEE Trans. Image Process. 29 (8), 2522–2535. https://doi.org/10.1109/TIP.2019.2959236.
B. Steder, M. Ruhnke, S. Grzonka, and W. Burgard, "Place recognition in 3D scans using a combination of bag of words and point feature based relative pose estimation," pp. 1249–1255, 2011, doi: 10.1109/iros.2011.6094638.
Yu, Z., 2009. "Intrinsic shape signatures: A shape descriptor for 3D object recognition", *2009 IEEE 12th Int.* Conf. Comput. Vis. Work. ICCV Work. 2009, 689–696. https://doi.org/10.1109/ICCVW.2009.5457637.
Sipiran, I., Bustos, B., 2011. Harris 3D: A robust extension of the Harris operator for interest point detection on 3D meshes. Vis. Comput. 27 (11), 963–976. https://doi.org/10.1007/s00371-011-0610-y.
Mian, A., Bennamoun, M., Owens, R., 2010. On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. Int. J. Comput. Vis. 89 (2–3), 348–361. https://doi.org/10.1007/s11263-009-0296-z.
Dong, Z., Yang, B., Liu, Y., Liang, F., Li, B., Zang, Y., 2017. A novel binary shape context for 3D local surface description. ISPRS J. Photogramm. Remote Sens. 130, 431–452. https://doi.org/10.1016/j.isprsjprs.2017.06.012.
Tombari, F., Salti, S., Di Stefano, L. Unique signatures of histograms for local surface description. In: *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6313 LNCS, no. PART 3, pp. 356–369, 2010, doi: 10.1007/978-3-642-15558-1_26.
Tao, W., Hua, X., Wang, R., Xu, D., 2020. Quintuple local coordinate images for local shape description. Photogramm. Eng. Remote Sensing 86 (2), 121–132. https://doi.org/10.14358/PERS.86.2.121.
Boroson, E.R., Ayanian, N., 2019. 3D keypoint repeatability for heterogeneous multi-robot SLAM. Proc. - IEEE Int. Conf. Robot. Autom. vol. 2019-May, 6337–6343. https://doi.org/10.1109/ICRA.2019.8793609.
N. Muhammad and S. Lacroix, "Loop closure detection using small-sized signatures from 3D LIDAR data," *9th IEEE Int. Symp. Safety, Secur. Rescue Robot. SSRR 2011*, pp. 333–338, 2011, doi: 10.1109/SSRR.2011.6106765.
Wang, H., Wang, C., Xie, L., 2020. "Intensity Scan Context: Coding Intensity and Geometry Relations for Loop Closure Detection" arXiv, 2095–2101.
Elbaz, G., Avraham, T., Fischer, A. 3D point cloud registration for localization using a deep neural network auto-encoder.In: *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 2472–2481, 2017, doi: 10.1109/CVPR.2017.265.
Gojcic, Z., Zhou, C., Wegner, J.D., Wieser, A. The perfect match: 3D point cloud matching with smoothed densities. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019, vol. 2019-June, pp. 5540–5549, doi: 10.1109/CVPR.2019.00569.
A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 199–208, 2017, doi: 10.1109/CVPR.2017.29.
Qi, C.R., Su, H., Mo, K., Guibas, L.J. PointNet: Deep learning on point sets for 3D classification and segmentation. In: *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 77–85, 2017, doi: 10.1109/CVPR.2017.16.
Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space. Adv. Neural Inf. Process. Syst. vol. 2017-Decem, 5100–5109.
Deng, H., Birdal, T., Ilic, S., 2019. 3D local features for direct pairwise registration. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. vol. 2019-June, 3239–3248. https://doi.org/10.1109/CVPR.2019.00336.
Wang, Y., Solomon, J., 2019. Deep closest point: Learning representations for point cloud registration. Proc. IEEE Int. Conf. Comput. Vis. vol. 2019-Octob, 3522–3531. https://doi.org/10.1109/ICCV.2019.00362.
Lu, W., Wan, G., Zhou, Y., Fu, X., Yuan, P., Song, S., 2019. DeepVCP: An end-to-end deep neural network for point cloud registration. Proc. IEEE Int. Conf. Comput. Vis. vol. 2019-Octob, 12–21. https://doi.org/10.1109/ICCV.2019.00010.
Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019. Dynamic graph Cnn for learning on point clouds. ACM Trans. Graph. 38 (5) https://doi.org/10.1145/3326362.
Liu, Z., et al., 2019. SeqLPD: Sequence Matching Enhanced Loop-Closure Detection Based on Large-Scale Point Cloud Description for Self-Driving Vehicles. IEEE Int. Conf. Intell. Robot. Syst. 1218–1223. https://doi.org/10.1109/IROS40897.2019.8967875.
Chen, X., Läbe, T., Nardi, L., Behley, J., Stachniss, C. Learning an Overlap-based Sensor Model for 3D LiDAR Localization. In: *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 4602–4608, 2020, [Online]. Available: http://www.ipb.uni-bonn.de/pdfs/chen2020iros.pdf.
Yin, H., Wang, Y., Ding, X., Tang, L.i., Huang, S., Xiong, R., 2020. 3D LiDAR-Based Global Localization Using Siamese Neural Network. IEEE Trans. Intell. Transp. Syst. 21 (4), 1380–1392. https://doi.org/10.1109/TITS.697910.1109/TITS.2019.2905046.
Sun, L., Adolfsson, D., Magnusson, M., Andreasson, H., Posner, I., Duckett, T. Localising Faster: Efficient and precise lidar-based robot localisation in large-scale environments. 4386–4392, 2020, doi: 10.1109/icra40945.2020.9196708.
Dube, R., Dugas, D., Stumm, E., Nieto, J., Siegwart, R., Cadena, C., 2017. SegMatch: Segment based place recognition in 3D point clouds. Proc. - IEEE Int. Conf. Robot. Autom. 5266–5272. https://doi.org/10.1109/ICRA.2017.7989618.
Sun Peng-peng, M.H., Xiang-mo, Z.H.A.O., Zhi-gang, X.U., 2018. Urban curb robust detection algorithm based on 3D-LIDAR. J. Zhejiang Univ. Sci. 52 (3), 504–514.
Peng, Q., Wen, C., Yan, H., 2008. Driving effects of speed increase on development of railway transportation in China. Xinan Jiaotong Daxue Xuebao/Journal Southwest Jiaotong Univ. 43 (6), 685–691.
Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the KITTI vision benchmark suite. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 3354–3361. https://doi.org/10.1109/CVPR.2012.6248074.
Bosse, M., Zlot, R., 2009. Continuous 3D scan-matching with a spinning 2D. In: laser [C]//2009 IEEE International Conference on Robotics and Automation, pp. 4312–4319.
Romero, A.R., Borges, P.V.K., Elfes, A., Pfrunder, A., 2016. "Environment-aware sensor fusion for obstacle detection", *IEEE Int.* Conf. Multisens. Fusion Integr. Intell. Syst. 114–121. https://doi.org/10.1109/MFI.2016.7849476.
Hess, W., Kohler, D., Rapp, H., Andor, D., 2016. Real-time loop closure in 2D LIDAR SLAM. Proc. - IEEE Int. Conf. Robot. Autom. vol. 2016-June, 1271–1278. https://doi.org/10.1109/ICRA.2016.7487258.
Dong, Z., Liang, F., Yang, B., Xu, Y., Zang, Y., Li, J., Wang, Y., Dai, W., Fan, H., Hyyppä, J., Stilla, U., 2020. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. ISPRS J. Photogramm. Remote Sens. 163, 327–342. https://doi.org/10.1016/j.isprsjprs.2020.03.013.