Loop-Closure Detection Based on 3D Point Cloud Learning for Self-Driving Industry Vehicles

Zhe Liu, Chuanzhe Suo, Shunbo Zhou, Wen Chen, Hesheng Wang, and Yun-Hui Liu

Abstract—Self-driving industry vehicle plays a key role in the industry automation and contributes to resolve the problems of the shortage and increasing cost in manpower. Place recognition and loop-closure detection are main challenges in the localization and navigation tasks, especially when industry vehicles work in large-scale complex environments, such as the logistics warehouse and the port terminal. In this paper, we resolve the loop-closure detection problem by developing a novel 3D point cloud learning network, an active super keyframe selection method and a coarse-to-fine sequence matching strategy. More specifically, we first propose a novel deep neural network to extract a global descriptors from the original large-scale 3D point cloud, then based on which, an environment analysis approach is presented to investigate the feature space distribution of the global descriptors and actively select several super keyframes. Finally, a coarse-to-fine sequence matching strategy, which includes a super keyframe based coarse matching stage and a local sequence matching stage, is presented to ensure the loop-closure detection accuracy and real-time performance simultaneously. The proposed network is evaluated in different datasets and obtains a substantial improvement against the state-of-the-art PointNetVLAD in place recognition tasks. Experiment results on a self-driving industry vehicle validate the effectiveness of the proposed loop-closure detection algorithm.

I. INTRODUCTION

Autonomous navigation is paramount significant in robotic community such as helping self-driving vehicles [1] and unmanned aerial vehicles [2] achieve full autonomy. Place recognition, in particular, represents one of the crucial challenges of accurate navigation, since it provides candidates for loop-closure, which is essential for drift-free localization and globally consistent mapping [3]. Current solutions for place recognition mainly fall into two categories, image-based and 3D point cloud-based. Many successful image-based approaches were proposed in the literature, due to the feasibility of extracting visual feature descriptors (e.g. SURF [4] or ORB [5]). These visual feature descriptors can be aggregated into a global descriptor of the scene for place recognition [6]. Unfortunately, this kind of solutions are unreliable due to its non-robustness under different season or weather conditions, and under different viewpoints [7].

3D point cloud-based solutions, on the other hand, does not suffer as much as vision solutions when changes in viewpoint are present. This paper therefore considers 3D point cloud for their potential to provide robust place recognition and loop-closure detection in large-scale environments. However, compared to feature extraction algorithms for visual images, there is no similar approach designed for point clouds that can reach the same level of maturity [8]. This is the fundamental issue that the existing point cloud-based methods have been trying to overcome. Most existing methods tackle this problem by using a coarse localization result from the Global Positioning System (GPS), followed by point cloud registration methods for loop-closure check. Therefore, point cloud-based method is largely neglected since GPS might not always be available, and point cloud registration is usually computationally expensive, so that real-time performance can not be guaranteed.

In this paper we resolve the large-scale place recognition problem based on 3D point cloud learning. A newly designed deep neural network based on local feature extraction and graph-based feature aggregation was adopted to extract the discriminative and generalizable global descriptor of the environment. Inspired by SeqSLAM [9], the global descriptor output from the network will be used for loop-closure detection. The contributions of our work can be summarized as follows:

1) 3D point cloud learning: We propose a novel neural network to extract a global descriptor from the original large-scale 3D point cloud. Local spatial distribution features of each point are well handled and local structure and neighborhood relations of each part in the point cloud are adequately revealed, these ensure the discrimination, generalization and classification performance of the proposed global descriptor.
2) Active matching: Based on the extracted global descriptor, an environment analysis approach is presented first to actively select super key-frames. Then a coarse-to-fine sequence matching approach is developed to achieve the accurate loop-closure detection with a feasible online searching time.
3) Thorough validation in real-world applications: The proposed algorithm is evaluated in different datasets and presents a substantial improvement against the state-of-the-art approach PointNetVLAD [8]. More specially, we test the proposed algorithm in a real industry park as well as in the campus of The Chinese University of Hong Kong, which validate the effectiveness and the practical applicability of our method.
II. RELATED WORK

Loop-closure detection using 3-D point cloud data is an significant but still open problem, which restricts the ability of robot localization and mapping in a complex environment, and thus hinders the robot from acquiring full autonomy. Extensive study has been carried out to tackle this problem. We identified the current solutions into three main trends: 1) approaches based on handcrafted local and global features, 2) approaches based on planes or objects, 3) approaches based on 3D point cloud learning.

Handcrafted local features, such as Shape context [11] and SHOT [10], first find a keypoint, divide neighboring points into different bins, and encode a histogram with the defined pattern of neighboring bins. Moreover, FPHF [12], present for 2.5D scans, which is a point-wise histogram based 3D feature descriptor. Global descriptors of the local point cloud were also presented. M2DP was proposed in [13], which projects the original point cloud to several planes and extracts a global representation. ESF [14] implemented the concatenation of histograms generated from shape functions. Local features usually require normal vectors of keypoints and thus are not suitable for large-scale place recognition tasks in outdoor scenarios. Global descriptors are usually tailored to specific tasks and therefore have poor generalization features.

Taking into account the limitations of the local and global descriptors, some researches have also presented to utilize planes or objects for place recognition tasks. A plane-based place recognition method was proposed in [15], and the covariance of the plane parameters are adopted for matching. The drawback of this approach is that it only applies to small and indoor environments, and the plane model assumption is not valid in some practical scenes. Recently, SegMatch [16] presented a matching method based on segments. This is a high-level perception but the points are required to be represented in a global coordinates, and the enough static objects assumption will not always be satisfied for real applications.

In order to solve the above problems, deep neural network was introduced for 3D point cloud feature learning and achieved state-of-the-art performance. Some work attempts to convert point cloud input to a regular 3D volume representation to alleviate the orderless problem of the point cloud, such as the volumetric CNNs [17] and 3D ShapeNets [18]. They are suitable for point cloud-based classification and recognition, respectively. On the other hand, different from volumetric representation, Multiview CNNs [19] projects the 3D data into 2D images so that 2D CNN can be performed. Additionally, by achieving the permutation invariance in the network, PointNet [20] makes it possible to learn features directly from the raw point cloud data. Although PointNet has achieved superior performance on small-scale shape classification and recognition tasks, it did not scale well for large-scale place recognition problem. PointNetVLAD [8] is the first work that directly applies 3D point cloud to large-scale place recognition, but this method does not consider local feature extraction adequately, and the network is heavy in memory when the feature dimension is huge.

III. SYSTEM FRAMEWORK

In this paper, the objective is to extract global descriptors of an input large-scale 3D point cloud, and based on which, to further resolve the loop-closure detection problems.

As shown in Figure 1, the proposed system framework can be divided into three parts:

- 3D point cloud learning: The original 3D laser point cloud is used as system input directly. For each point $p_i$ in the point cloud, we first extract its local spatial distribution features and also transform its original coordinates $(x_i, y_i, z_i)$ into a unified viewpoint, the local features as well as the transformed coordinates are utilized as the neural network input. Then we present a new deep neural network to generate a global descriptor, in the form of a 256-dimensional vector, to uniquely describe the input large-scale point cloud.

- Mapping and environment analysis: All the generated global descriptors will be stored with corresponding position information in order to generate a descriptor map. Then we investigate the feature space distribution characteristics of the global descriptors to generate several descriptor clusters and select out one typical place in each cluster, the corresponding global descriptors of these selected places will be defined as the super keyframes.

- Loop-closure detection: A coarse-to-fine matching strategy is proposed for loop-closure detection in order to ensure the real-time performance. In the coarse matching stage, the global descriptor of the new input point cloud is compared with all the super keyframes firstly to find out the matched cluster. Then in the fine matching stage, local sequence matching strategy is utilized around each place in the matched cluster to find out the accurate location of the input point cloud, thus achieving the loop-closure detection task.

The proposed deep neural network ensures the discrimination, generalization and classification performance of the generated global descriptor, which greatly facilitates the place recognition task in large-scale environment. Furthermore, the presented coarse-to-fine matching strategy contributes to reduce the computational and storage complexity. Based on the proposed active super keyframe selection method and the local sequence matching strategy, both the accuracy and the real-time performance of the loop closure detection in large-scale complex environments can be guaranteed.

IV. 3D POINT CLOUD LEARNING

PointNet and PointNet++ have good performance in small-scale object point cloud learning, and PointNetVLAD [8] extends PointNet to solve the place recognition problem in large-scale environments. However, both PointNet and PointNetVLAD can not be used to solve the loop-closure problem of self-driving applications since their place recognition accuracies are not satisfied (these will be validated.
in our experiments). PointNet and PointNetVLAD do not consider local feature extraction adequately, and does not reveal the spatial distribution of the input point cloud. In our opinions, these two points are of great importance in SLAM applications and may greatly affect the place recognition performance in large-scale environments.

The proposed network architecture is shown in Figure 2. In order to improve the place recognition accuracy, we present two new modules: local feature extraction and graph-based feature neighborhood aggregation. The former one aims to extract local features around each point, while the latter one aims to reveal the spatial distributions of the similar local structures.

### A. Local Feature Extraction

Existing point cloud based network, including the PointNet [20] and PointNetVLAD [8], only consider the original point coordinates \((x_i, y_i, z_i)\) as network input, local spatial distribution features calculated in the neighborhood of each point have not been taken into consideration. However, several previous robotics researches have validated that local features are effective in 3D outdoor scene interpretation tasks and large-scale localization and mapping tasks in self-driving applications [3], [16], [21], [22]. So in this paper, \(k\) nearest neighboring points are considered to describe the local 3D structure around each point \(p_i\) and two types of local features are calculated:

- Local features based on Z-axis statistics: the maximum height difference \(\Delta Z_{i, \text{max}}\) of all the \(k\) neighbors around point \(p_i\) and their height variance \(\sigma Z_{i, \text{var}}\).
- Local features arising from the projection of the 3D point cloud onto the horizontal plane: 2D scattering \(S_{i}^{2D} = \lambda_{i, 1}^{2D} + \lambda_{i, 2}^{2D}\) and 2D linearity \(\frac{\lambda_{i, 1}^{2D}}{\lambda_{i, 2}^{2D}}\). \(\lambda_{i, 1}^{2D}\) and \(\lambda_{i, 2}^{2D}\) are eigenvalues of the 2D covariance matrix which is calculated by projecting the \(k\) neighboring points onto the 2D horizontal plane.

As shown in Figure 3, we also consider the original coordinates \((x_i, y_i, z_i)\) of each point \(i\) as network input, but in order to unify the viewpoint, the coordinates are transformed by an Input Transformation Net [20] to ensure the rotational translation invariance. Then the transformed coordinates and the above four local features of each point
are concatenated and used as the network input.

B. Graph-Based Feature Neighborhood Aggregation

Different with the object point cloud, a large-scale point cloud mostly consists of rich 3D structures of the surrounding environments and their spatial distribution relationships, such as the relative orientation between two buildings with cube point cloud shapes, or the relative distance between two trees with point cloud clusters. Similar local point cloud structures in different locations usually have similar local features, which can be utilized as a main judgment for place recognition.

In this paper, we introduce the Graph Neural Network to investigate the intrinsic relationships between each composition in the point cloud. In Section IV.A, we have merged the neighborhood information into the local feature of each point, so each point can be regarded as the feature description of the surrounding neighborhood. Then the Graph Neural Network can be utilized to aggregate the local features in feature space and extract the description vector of the whole point cloud.

More specifically, we first use an alignment network on point features and calculate a transformation matrix in the feature space, then the point features are transformed into a unified viewpoint and thus the rigid motion invariance can be guaranteed. As shown in Figure 4, a KNN aggregation is implemented on each point to find \( k \) nearest neighbors in the new feature space. These feature space relations are further utilized to build the graph relations in the previous feature space and then we aggregate the neighbor features into each point. Finally, two MLP Networks are used to update feature space neighbor relations and a Max Pooling Operation is implemented to aggregate \( k \) edge information of each point into a vector. The feature transformation module utilized in this paper is similar with that in PointNet [20], however, please note that we only use the graph relations generated in the transformed feature space and the actual feature aggregations are performed in the original feature space. This operation is not the case in PointNet [20], but contributes to improve the place recognition accuracy greatly.

In the proposed graph-based feature aggregation, similar local point cloud structures in different locations will be aggregated in the feature space since they usually have similar local features. What’s more, in the proposed graph, the edge relation is defined as the combination of \( p_i - p_j \) and \( p_j \), since the original coordinate information is also considered in the feature vector \( p_i \), the spatial distributions and the relative relationships can also be considered in feature aggregation. So the proposed network can reveal the spatial distributions of the similar local structures in the input point cloud adequately.

C. NetVLAD and Output Aggregation

Similar with [8], the NetVLAD is introduced into the proposed network to aggregate local feature descriptors and generate the final global descriptor vector. NetVLAD has been proven to be permutation invariant and achieved improved performance in large-scale place recognition task. Also similar with [8], the lazy quadruplet loss based on metric learning is utilized as the network loss function, which aims to reduce the positive sample distance as well as enlarge the negative sample distance during the training process. The final global descriptor is generated in the form of a 256-dimensional vector and can be used to uniquely describe the input large-scale point cloud.

V. LOOP-CLOSURE DETECTION

We first analyze the whole environment by investigating the feature space distribution characteristics of the global descriptors and select out the super keyframes. Then we utilize a coarse-to-fine matching strategy for loop-closure detection in order to ensure the accuracy and real-time performance simultaneously. Please note that the super keyframes are selected in the feature space (descriptor distribution space) without considering their Cartesian space locations, but in each local sequence matching, the Cartesian space location of each place in the sequence and the corresponding location relations are considered in order to achieve an accurate result.

A. Active Super Keyframe Selection

Firstly, we investigate the feature space distribution of the global descriptors and generate descriptor clusters. Canopy based approach and K-means based approach (or their combination) are two classes of promising approaches for high dimensional space clustering tasks and we find that, in our case, K-means based method is much better since the clustering performance of Canopy depends largely on the initial cluster centers, so we choose K-means++ clustering method [24] in this paper. More specifically, we evaluate the sum of distortions under different cluster number \( k \) and utilize the Elbow method to determine the optimal \( k \) value. What’s more, we introduce an additional constraint which requires that the \( L_2 \) distance from each global descriptor to
its corresponding cluster center is lower than \( D \), where \( D \) is an environment related parameter which defines the \( L_2 \) distance threshold of two global descriptors which can be recognized as the similar places.

Secondly, in each cluster, the global descriptor with the nearest \( L_2 \) distance to the cluster center is selected as the super keyframe and other global descriptors are restored in a descriptor index which corresponds to this super keyframe. Then we can obtain \( k \) super keyframes and \( k \) global descriptor indices. The selected super keyframes contain all the typical places in the whole environment and each type of typical places has at least one super keyframe. These will be validated in the experiments.

B. Coarse-to-Fine Matching

In the coarse matching stage, the global descriptor of the new input point cloud is compared with all the super keyframes firstly to find out the matched cluster by calculating the \( L_2 \) distances. Then in the fine matching stage, local sequence matching strategy is utilized around each place in the corresponding global descriptor index of the matched cluster to find out the accurate location of the input point cloud, thus achieving the loop-closure detection task.

In particular, the fine matching algorithm is tailored based on SeqSLAM [9], a robust visual SLAM framework, to make it suitable for point cloud data. The basic idea of fine matching is that, instead of finding the a global best match frame relative to the current frame, we look for the best candidate matching frame within every local sequence. To do this, the fine matching process is divided into two components: local best recognition and sequence matching. Local best recognition towards to find all the frames within local neighborhoods that are the best match for the current frame, which is conducted by calculating the difference between two frames based on the \( L_2 \) distances of the global descriptors that extracted by our network, and a difference matrix would be generated as shown in Figure 7. Then to match the target place sequences, a search is performed through the current difference matrix with a searching window. At each reference frame, the search projects several trajectories based on different possible velocities. The trajectory velocity is ranged from \( V_{\text{min}} \) to \( V_{\text{max}} \). The \( L_2 \) distance based difference score is calculated in each trajectory line. The trajectory with the minimum score is the best match.

VI. EXPERIMENTS

As shown in Figure 2, the proposed deep neural network has three main modules: local feature exaction (LFE), graph-based feature aggregation (GFA) and NetVLAD. In LFE, we select \( k = 20 \) nearest points to generate the local neighborhood of each point. In GFA, the \( k \) in KNN aggregation is also set to 20. NetVLAD is the same as that in [8], where the lazy quadruplet loss parameters are set to \( \alpha = 0.5, \beta = 0.2, P_{\text{pos}} = 2, P_{\text{neg}} = 18 \). All experiments are conducted with a 1080Ti GPU on TensorFlow.

A. Place Recognition Results in Robotcar Datasets

We train and evaluate the proposed network on the Oxford Robotcar dataset [23]. In Oxford RoboCar dataset, the 3D point cloud submap is made up of point clouds within the car’s 20m trajectory. The point cloud of each submaps contains 4096 points and is normalized to the range of \([-1, 1]\). We use 44 sets, 21,711 training submaps to train the proposed network. In the evaluation process, we randomly selected four scenes from the 44 sets data of the Oxford dataset for evaluation. The data collection of 44 sets is in different seasons, different times and different weathers, and we querying the same scene in these sets for place recognition. Such place recognition with large time span and light changes is not possible with images. We use Recall to evaluate the ability of place recognition to see if there is a real scene in the top \( N \) scenes closest to it. We compare it with Average Recall@1 and Average Recall@1%.

![Average recall under different networks.](image)

**TABLE I**

Comparison results of the average recall (%) at top 1% (@1%) and at top 1 (@1) under different network structures.

|                | Ave recall @1% | Ave recall @1 |
|----------------|----------------|---------------|
| PN STD         | 46.52          | 31.87         |
| PN MAX         | 73.87          | 54.16         |
| PN-VLAD baseline | 81.01         | 62.76         |
| PN-VLAD refine  | 80.71          | 63.33         |
| LFE-VLAD (our) | 86.99          | 72.76         |
| GFA-VLAD (our)  | 86.91          | 74.16         |
| LFE-GFA-VLAD (our) | 89.55 | 77.92         |

We compare our approach to PointNet with the maxpool layer (PN MAX) and PointNet trained for object classification in ModelNet (PN STD) to study whether the network trained on small-scale dataset can be directly used to large-scale point clouds. Moreover, we also compare our network with the state-of-the-art PN-VLAD baseline, and PN-VLAD refine [8] to show the performance of our algorithm. Comparison results with PointNetVLAD are shown in Figure 7 where LFE-VLAD represents the network without graph-based feature aggregation, GFA-VLAD represents the network without local feature extraction.

From Table I we can find that GFA-VLAD is better than PointNetVLAD, which increases the place recognition accu-
We first conduct experiments in an industry park (about 120m × 80m), as shown in Figure 6, to test the effectiveness of the proposed approach both in indoor and outdoor environments. The testing route are shown in Figure 7, the vehicle is commanded to track the outdoor-indoor route for two loops, and in the second loop, some parts of the route are designed to have some deviations from those in the first loop. The desired vehicle velocity is set to $V_d = 3m/s$ in both loops. The trajectory velocity bounds in sequence searching are set to $V_{min} = 0.8V_d$ and $V_{max} = 1.2V_d$, and the sequence searching window size is set to 10 point cloud frames. In our platform, the computation time in global descriptor generation and loop-closure detection is about 150ms. Figure 7 shows the loop-closure detection results, all the detected loop-closure locations assemble three matching route segments, i.e., the segment from location 1 to location 2, the segment from location 3 to location 4 and the segment from location 5 to location 6. From the left of Figure 7, we can find that the route segment from location 2 to location 3 in the second loop is different with that in the first loop, and in the right of Figure 7, the proposed point cloud learning and loop-closure detection approaches also divide these into different place sequences successfully. The similar results can also be found in the segment from location 4 to location 5. Furthermore, from location 3 to location 4, the vehicle trajectory in the second loop has a slightly offset compared with that in the first loop, however, the proposed approach can recognize these two segments as the same place sequences successfully, this validates the robustness of the proposed approaches to viewpoint variations.

We then conduct experiments in the campus of The Chinese University of Hong Kong to validate the proposed approach in large-scale outdoor environments. Note that the university campus is built on the mountain area, the slope terrain brings large difficulties in the mapping and loop-closure tasks. The map built by the state-of-the-art LoGo-LOAM [25] is show in the middle of Figure 8, we can find that the map can not be loop-closed due to the slope terrain and large distances. However, in the right of Figure 8, the proposed approach finds out three matching segments successfully, and in particular, the same route segment with the opposite moving directions can also be recognized (as shown in the segment from location 5 to location 4 in the first loop and in the segment from location 4 to location 5 in the second loop), these validate the effectiveness and robustness of the proposed approaches, and will greatly facilitate the practical applications. Figure 9 gives an example to show the corresponding point cloud frames and video image frames from a matched sequence.

C. Discussion

In our datasets, such place recognition with large time span and light changes is not possible with images. However, our point cloud-based approach has achieved excellent results in both industry park and university campus datasets. The results demonstrate that our network has superior advantages for point cloud-based place recognition in a large-scale environment, far exceeding PointNetVLAD and reaching the state-of-art. Even if the model trained on oxford dataset is directly migrated to the datasets collected in the industry warehouse and university campus, it can achieve good results without training. Also please note that we do not use any global position information or odometry data in the loop-closure detection. These shows that the proposed approach has great place recognition and loop-closure performance in complex large-scale environments. In the future, we will incorporate the proposed loop-closure detection results with SLAM approaches to build the closed point cloud map of the university campus and resolve the life-long SLAM problem.

VII. Conclusion

In this paper, we propose a loop-closure detection solution based on 3D point cloud learning and active sequence matching. A novel deep neural network is designed to extract the discriminative and generalizable global descriptor from the original large-scale 3D point cloud data. Building on top of these descriptors, an environment analysis approach is developed to actively select super key-frames, then a
Fig. 7. Experiment results in the industry park. Left: The laser point cloud map and the vehicle route, where the vehicle route is recorded by the proposed localization system, the star-points represent the selected super keyframes, the colored points represent all the point cloud frames and their colors represent their belonged clusters. Right: The $L_2$ distances between the generated global descriptors along the whole point cloud frame sequence, where the color scale of the point is darker the $L_2$ distance is smaller, the red markers belong to the first loop and the black markers belong to the second loop.

Fig. 8. Experiment results in the university campus. Left: The campus environment in Google Map and the commanded vehicle route. Middle: The trajectory recorded in the point cloud map built by LeGO-LOAM [25] and two video frames recorded at the same place in the different experiment loops, where the red markers belong to the first loop and the black markers belong to the second loop. Right: The $L_2$ distances between the generated global descriptors along the whole point cloud frame sequence.

Fig. 9. An example of the matched sequences in our loop-closure detection results, where the video image frames and the corresponding point cloud frames recorded in the first experiment loop are shown in the Upper, while the frames recorded in the second loop are shown in the Lower. All the point clouds have been projected into the horizontal plane for better visualization.

The proposed network is evaluated in RoboCar datasets and presents a substantial improvement against the current approach. The results for place recognition increased the existing state-of-the-art result (PointNetVLAD) from 81.01% to 89.55%. Furthermore, real-world experiments were also carried out on terrains where the traditional methods are very challenging. The robustness of our algorithm to viewpoint variations were validated in a complex indoor and outdoor industrial park. In the large-scale university campus environment with the slope terrain, the proposed loop-closure detection approach successfully recognized the matching route segments which can not be detected by the LeGO-LOAM algorithm, this also shows the effectiveness and practical applicability of the proposed approach. What’s more, the
model trained on the RoboCar dataset can be directly applied to the real-world applications without any further training, which greatly facilitates the practical applications.

REFERENCES

[1] T. Ort, L. Paull, and D. Rus, “Autonomous vehicle navigation in rural environments without detailed prior maps,” in Proc. IEEE Int. Conf. Robot. and Automat., Brisbane, Australia, pp. 2040-2047, 2018.

[2] K. Sun, K. Mohta, B. Pfommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, “Robust stereo visual inertial odometry for fast autonomous flight,” in IEEE Robot. and Automat. Lett., vol. 3, no. 2, pp. 965-972, 2018.

[3] Giseop Kim and Ayoungh Kim, “Scan Context: Egocentric Spatial Descriptor for Place Recognition within 3D Point Cloud Map,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots and Sys., Madrid, Spain, pp. 4802-4809, 2018.

[4] H. Bay, T. Tuytelaars, and L. V. Gool, “Surf: Speeded up robust features,” in Proc. Eur. Conf. Comput. Vision, Graz, Austria, pp. 404-417, 2006.

[5] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: An efficient alternative to sift or surf,” in Proc. IEEE Int. Conf. Comput. Vision, Barcelona, Spain, pp. 2564-2571, 2011.

[6] S. Lowry, N. Snderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, “Visual place recognition: A survey,” in Proc. IEEE Int. Conf. Comput. Vision and Pattern Recog., Honolulu, Hawaii, pp. 652-660, 2017.

[7] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in Proc. IEEE Conf. Comput. Vision Pattern Recog., Honolulu, Hawaii, pp. 6526-6534, 2017.

[8] M. Weinmann, J. Boris, and M. clément, “Semantic 3d scene interpretation: a framework combining optimal neighborhood size selection with relevant features,” in ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 2, no. 3, 2014.

[9] H. Su, S. Maji, F. Tombari, and L. Di Stefano, “SHOT: Unique signatures of which greatly facilitates the practical applications.

[10] M. Weinmann, J. Boris, and M. clément, “Semantic 3d scene interpretation: a framework combining optimal neighborhood size selection with relevant features,” in ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 2, no. 3, 2014.

[11] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, “1 year, 1000km: the oxford robotcar dataset,” Inter. J. Robot. Res., vol. 36, no. 1, pp. 3-15, 2017.

[12] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3D object detection network for autonomous driving,” in Proc. IEEE Int. Conf. Comput. Vision Pattern Recog., Honolulu, Hawaii, pp. 6526-6534, 2017.

[13] A. David, and S. Vassilvitskii, “k-means++: the advantages of careful seeding,” Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2007.

[14] T. Shan, and B. Englot, “LeGO-LOAM: lightweight and ground-optimized lidar odometry and mapping on variable terrain,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots and Sys., Madrid, Spain, pp. 4758-4765, 2018.