Proceedings of The 2021 IEEE International
Conference on Real-time Computing and Robotics
July 15-19, 2021, Xining, China

Semantic Scan Context: Global Semantic Descriptor for LiDAR-based Place Recognition

Yuxiang Li§, Pengpeng Su§, Ming Cao, Haoyao Chen, Xin Jiang and Yunhui Liu

***Abstract*— Place recognition plays an important role in a typical simultaneously localization and mapping(SLAM) framework, which allows the autonomous mobile robot to identify the revisited places. Recently, emerging researches focus on incorporating geometry features to achieve accurate and view-invariant relocalization in outdoor environment. However, the ambiguity of geometry features occurs in the scenes with similar objects. To address this problem, we propose *Semantic Scan Context*, a noval global descriptor based on 3D LiDAR scans and static semantic information such as trunks, poles, traffic signs, buildings, roads and sidewalks. This descriptor not only records the geometrical structure of a 3D LiDAR scan, but also encodes the semantic distribution information. Furthermore, we introduce a coarse-to-fine hierarchical retrieval method to realize the efficient matching for the proposed descriptors. We define three distances to measure the similarity of two places. By weighted sum of these three distances, revisited places can be exactly determined. The recall-precision curve of proposed method is evaluated on public datasets and compared with existing methods. The experimental results proof that our method achieves a competitive re-identification performance.**

I. INTRODUCTION

Place recognition is a fundamental component in SLAM, which has been extensively studied for real applications. In order to run better in the real environment, robots need to recognize the revisited places as much as possible to optimize their trajectories and overcome incremental pose drift. The key problem for place re-identificaiton is to efficiently identify the effective structures of the scenes experienced by the robot. Popular methods for this problem can be classified into two categories: visual-based method like [1] or laser-based method such as [2], [3] and [4].

For common visual systems, the BoW(Bag-of-Word) is a typical solution. Since the BoW depends on low-level features, e.g. ORB [5], the great changes in the appearance of objects may cause a serious decline in identification. To solve the problem, convolutional neural networks are used to extract higher-level information from images, such as object-based method [6] and landmark-based method [7] [8]. In [7] and [8], the place recognition is converted into a graph matching problem by using the topological relationship between objects in an image. In addition, deep learning is used to better understand the pictures in [9] [10] [11] [12]. However, the low level of photosensitive range of the camera makes visual systems susceptible to light changes in outdoor environment(e.g. night).

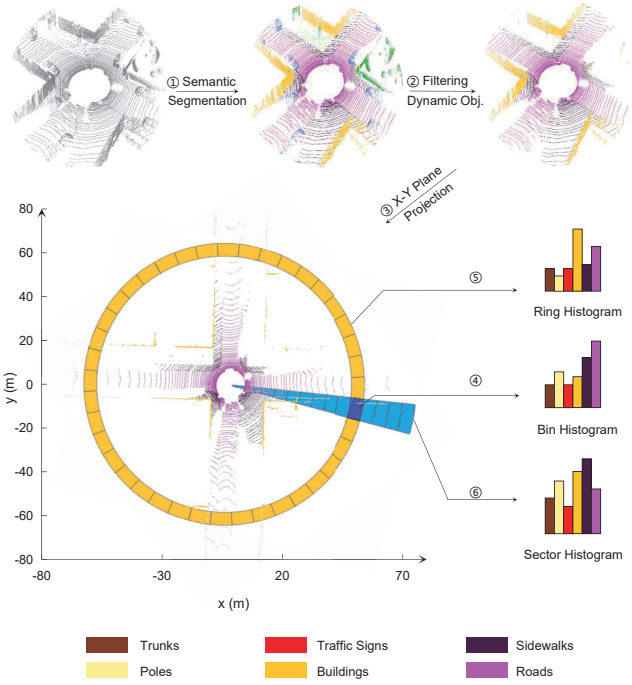


Fig. 1. Semantic Scan Context. Our proposed approach starts with a 3D LiDAR scan, every point is labeled through a semantic segmentation network (1). Then we only focus on 6 kinds of static objects by filtering dynamic objects (2) in the point cloud. By X-Y plane projection (3), we evenly devide the plane into sectors and rings as well as small bins. We then generate histogram for each bin (4) to contruct semantic scan context, which is defined as a global descriptor for each frame. Meanwhile, ring histogram(5) and sector histogram(6) are calculated to realize fast retrieval method for loop closure detection.

Differ from vision sensors, lidar sensors have no limit of light change or measuring range, which are even tackable in the environment lacking texture features. Despite the fine resolution and rich details captured by lidar sensors, there are still challenges for robust and efficient place recognition using point clouds. Some artifical descriptors of point cloud can achieve relatively good re-identification performance, such as M2DP [2], SC [3], DELIGHT [13] and so on. While these approaches tend to make the full use of local details rather than high-level features. There are also some

*This work was supported in part by the National Natural Science Foundation of China under Grant U1713206 as well as the Shenzhen Science and Innovation Committee under JCYJ20200109113412326, JCYJ20180507183837726 and JCYJ20180507183456108. (Corresponding author: Haoyao Chen.)

§ Y.X. Li and P.P. Su contributed equally.

Y.X. Li, P.P. Su, M. Cao, X. Jiang and H.Y. Chen* are with the School of Mechanical Engineering and Automation, Harbin Institute of Technology Shenzhen, P.R. China, e-mail: hychen5@hit.edu.cn.

Y.H. Liu are with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, P.R. China, e-mail: yhliu@mae.cuhk.edu.hk.

2021 IEEE International Conference on Real-time Computing and Robotics (RCAR) | 978-1-6654-3678-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/RCAR52367.2021.9517367

learning-based methods for point cloud based place recognition. PointNetVLAD [11], combined PointNet [14] and NetVLAD [15], can generate a pointcloud descriptor with permutation invariance. Unlike PointNetVLAD's borrowing metric learning, SegMatch [16] and SegMap [17] introduce an encoder-decoder system to incorporate a descriptor into its SLAM framework so as to obtain robust loop-closure detection. However, these learning-based methods require enough static objects exsiting in the environment and need to build a dense local map progressively.

In this paper, we propose a global semantic signature based on 3D laser data as well as a three-stage retrieval approach to realize fast place recognition. First of all, the point cloud data generated from 3D LiDAR sensor is processed by a semantic segmentation network to obtain semantic labels. In order to overcome the scene changes in the environment, we only select the static features (such as trunks, poles, traffic signs, buildings, roads and sidewalks) that are stable over time through prior semantic information. To take advantage of these static and robust point clouds, we establish a semantic scan context to incorporate semantic information for re-identification and propose a distance score to measure the similarity between two semantic scan contexts. Secondly, to realize fast retrieval for semantic scan context, we design histogram-based ring key and sector key, as well as their distance metrics. Finally, according to these three distances defined by the semantic scan context, ring key and sector key, we assign different weight for each distance and combine them into a discriminant function to determine the revisited place.

The remaining sections of this paper are organized as follows. The generating process of semantic scan context and similarity metric are presented in Section II. Then, we design a three-stage coarse-to-fine retrieval stragy in Section III. Experiment of the proposed method on the KITTI dataset is carried out in Section V. In the end, Section VI summarizes this paper.

II. GLOBAL DESCRIPTOR AND DISTANCE METRICS

In this section, we present how to construct the semantic scan context and explain how to take advantage of semantic information for place recognition. We outline the details of our proposed method, as the block illustrated in Fig.1. We firstly input the point cloud recieved from 3D LiDAR into the semantic segmentation network. Then the labeled point cloud is used to generate a semantic scan context. In addition, a similarity score is proposed to measure the distance between two semantic scan contexts, which can be used for further place recognition.

A. Semantic Scan Context

In order to overcome the ambiguity of geometry information in the enviroment, we employ semantic segmentation network such as RangeNet++ [18] to assign semantic label to every point in each LiDAR scan. Then we can incorporate the semantic information into scan context for place recognition. Since the common static objects in urban environment can provide robust localization information, in this work we generate semantic scan context with six kinds of inherently static features, including trunks, poles, traffic signs, buildings, roads and sidewalks. After getting the label for each point in a 3D LiDAR scan, the labeled pointcloud can be projected into x-y plane in the sensor coordinate. Similar to Scan Context [3], we evenly devide the x-y plane along azimuthal and radial directions into N_r rings and N_s sectors separately, as shown in Fig.1. Let L_{max} be the maximum radius of the projected x-y plane, the points outside this circular area is omitted. Then the radial gap between rings is L_{max}/N_r and the central angle of a sector is equal to $2\pi/N_s$.

A LiDAR scan with n labeled points is defined as $P = \{p_1, p_2, ..., p_n\}$, and the label of the point p_k is denoted as l_k. Then each point p_k can be represented in polar coordinate as follows:

$$\begin{aligned} p_k &= [\rho_k, \theta_k, l_k] \\ \rho_k &= \sqrt{x_k^2 + y_k^2} \\ \theta_k &= arctan(\frac{y_k}{x_k}) \end{aligned} \tag{1}$$

Because the x-y plane is divided into integer intervals, each point needs to be assigned to its nearest bins. Thus we use rounding function to approximate its index, then each bin B_{ij} can be represented as:

$$\begin{aligned} B_{ij} &= \{p_k \in P |\ i \in [N_r],\ j \in [N_s]\} \\ i &= round(\rho_k) \\ j &= round(\theta_k) \end{aligned} \tag{2}$$

where the symbol $[N_s]$ represents $\{1, 2, ..., N_{s-1}, N_s\}$, the symbol $[N_r]$ is equal to $\{1, 2, ..., N_{r-1}, N_r\}$.

Inspired by semantic context mentioned in [19], we encode the semantic histogram information for each subspace after partitioning the points of a 3D LiDAR scan. Let L denote the total number of segmentation classes. Since we only use 6 classes of segmentation in this work, L is equal to 6. Let $b(p_k)$ be the function that returns the index of a semantic class for point p_k, thus the semantic histogram $\boldsymbol{h}_{ij}$ of the bin B_{ij} located at i th ring and j th sector can be defined as:

$$\boldsymbol{h}_{ij} = (h_{ij}^1\ h_{ij}^2\ ...\ h_{ij}^{L-1}\ h_{ij}^L) \tag{3}$$

where

$$h_{ij}^l = \frac{\#b(p) = l}{\#p \in B_{ij}},\ p \in B_{ij},\ l \in 1, ..., L \tag{4}$$

Here, each point p within the bin B_{ij} is sorted into a histogram by its semantic label. For each semantic class, the corresponding value of the histogram represents the proportion of points of this class within the bin to the total points that are inside the bin.

For stable approximation of the histogram, we threshold it into a binary vector $\hat{\boldsymbol{h}}_{ij}$ as follows:

$$\hat{\boldsymbol{h}}_{ij} = (\hat{h}_{ij}^1\ \hat{h}_{ij}^2\ ...\ \hat{h}_{ij}^{L-1}\ \hat{h}_{ij}^L) \tag{5}$$

where

$$\hat{h}_{ij}^{l} = \begin{cases} 1 & if\ h_{ij}^{l} \geq t_{bin} \\ 0 & if\ h_{ij}^{l} < t_{bin} \end{cases} \tag{6}$$

Here the threshold t_{bin} is selected to reduce the effect of arbitrary noise in histogram due to sporadic points and their semantic classes. The semantic scan context Ω is filled with semantic histogram binary vectors $\hat{\boldsymbol{h}}_{ij}$ by:

$$\Omega(i,j) = \hat{\boldsymbol{h}}_{ij},\ i \in [N_s],\ j \in [N_r] \tag{7}$$

Note that if $B_{ij} \in \varnothing (i.e.$, no scan data), then $\Omega(i,j) = 0$. Up to this point, the semantic scan context has been constructed as a gloabl signature Ω, which is a $N_r \times N_s$ matrix encoded with semantic distribution of the environment.

B. Similarity Score between Semantic Scan Contexts

Given a pair of semantic scan contexts, we then need to calculate a distance to measure the similarity of two places. For two semantic scan contexts Ω^q and Ω^c, acquired from a query point cloud and a candidate point cloud respectively, we can compute the semantic distance of two corresponding binary vectors:

$$d(\Omega_{i,j}^{q}, \Omega_{i,j}^{c}) = d_{ham}(\hat{\boldsymbol{h}}_{ij}^{q}, \hat{\boldsymbol{h}}_{ij}^{c}) \tag{8}$$

where $d_{ham}(\boldsymbol{a}, \boldsymbol{b})$ is the Hamming distance between the two binary vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. Smaller distance value between two semantic vectors means a better similarity in local area. Hence we can get the distance between two semantic scan contexts by adding up all the distances of corresponding binary vectors. Therefore, the distance function is:

$$d(\Omega^q, \Omega^c) = \sum_{i=1}^{N_r} \sum_{j=1}^{N_s} d(\Omega_{i,j}^{q}, \Omega_{i,j}^{c}) \tag{9}$$

However, if the viewing angle of a LiDAR changes in the same place, the column of the candidate semantic scan context may be shifted. This makes the distance $d(\Omega^q, \Omega^c)$ unable to exactly describe the similarity of two semantic scan contexts. To detect the viewing angle change, we have to calculate distance with all possible column-shifted semantic scan contexts and find the minimum distance. The distance D_c between two semantic scan contexts is calculated by:

$$D_c(\Omega^q, \Omega^c) = \min_{n \in [N_s]} d(\Omega_n^q, \Omega^c) \tag{10}$$

$$n_c^* = \underset{n \in [N_s]}{argmin}\ d(\Omega_n^q, \Omega^c) \tag{11}$$

where Ω_n^q is obtained by shifting n columns. Meanwhile, we can identify the best matched Ω_n^q with column-shifts of n_c^* that can be used to correct viewing angle change.

III. PLACE RE-IDENTIFICATION

The purpose of constructing semantic scan context is to realize fast place recognition by mathcing current frame with historical frames. One simple way is to directly campare two semantic scan contexts by using the distance above. However, due to its large computational cost, this method only works with a small number of historical frames. It becomes much more time-consuming and finall loses real-time performance in the long run. To reduce the computational cost and speed up the searching process, in this section we present a three-stage coarse-to-fine retrieval strategy for matching semantic scan contexts.

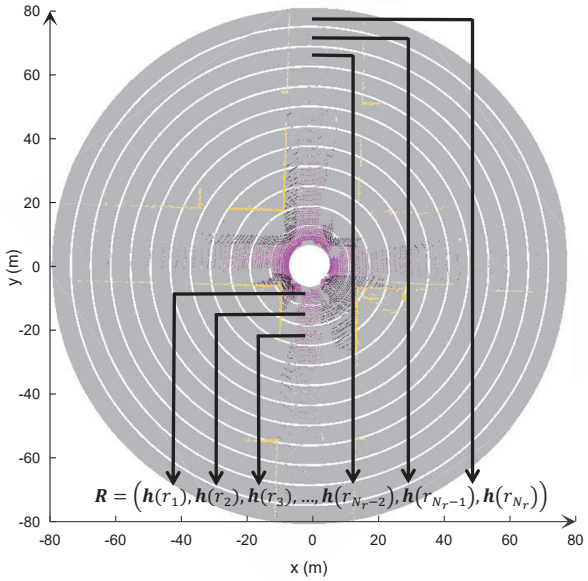


Fig. 2. The ring key generation for the fast retrieval

A. Fast Candidate Re-identification

Similar to scan context [3], we introduce ring key to realize fast retrieval as shown in Fig.2. Since it is meaningless to calculate the average of binary vectors stored in each bin, we turn to construct the semantic histogram for the points located in each ring. Let B_r denote the points located in the r th ring as:

$$B_r = \{B_{ij} |\ i = r,\ j \in [N_r]\} \tag{12}$$

where the symbol $[N_r]$ represents $\{1, 2, ..., N_{r-1}, N_r\}$. Then we count the points number for each semantic class within the r th ring, the semantic histogram $\boldsymbol{h}(r)$ of the ring r can be defined as:

$$\boldsymbol{h}(r) = (h_r^1\ h_r^2\ ...\ h_r^{L-1}\ h_r^L) \tag{13}$$

where

$$h_r^l = \frac{\#b(p) = l}{\#p \in B_r},\ p \in B_r,\ l \in 1, ..., L \tag{14}$$

Therefore, the ring key can be expressed as a N_r-dimensional vector:

$$\boldsymbol{R} = (\boldsymbol{h}(r_1), \boldsymbol{h}(r_2), ..., \boldsymbol{h}(r_{N_r}) \tag{15}$$

Totally, there are $N_r \cdot L$ elements in vector $\boldsymbol{R}$. The vector $\boldsymbol{R}$ encodes semantic distribution information for each ring and is independent of the orientation of the sensor. Thus the ring key can be used as a global signature for fast candidate identificaiton. Then we use Manhattan distance to measure the distance between the corresponding semantic histograms

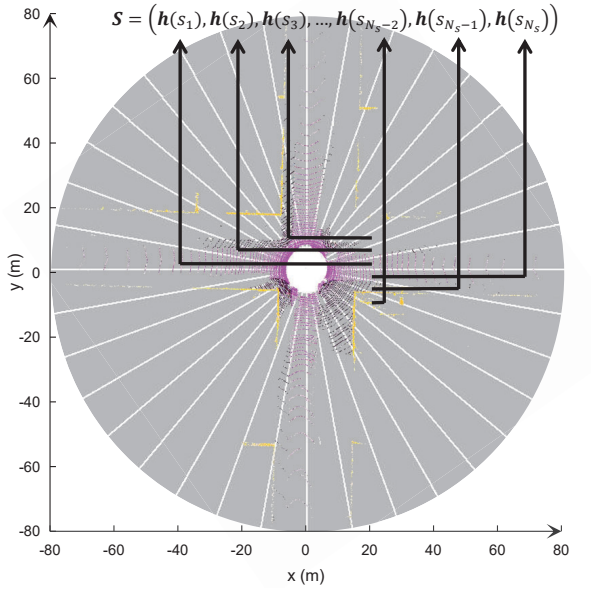


Fig. 3. The sector key generation for the fast alignment

of the query frame and the candidate:

$$D_r(\boldsymbol{R}^q, \boldsymbol{R}^c) = \sum_{r=1}^{N_r} d_{man}(\boldsymbol{h}(r^q), \boldsymbol{h}(r^c)), \quad c \in \mathcal{C} \tag{16}$$

where $\mathcal{C}$ is a set of indices of candidates extracted from KD-tree, $\boldsymbol{h}(r^q)$ and $\boldsymbol{h}(r^c)$ are the semantic histograms in the r th ring of the query frame and the candidate separately, $d_{man}(\boldsymbol{a}, \boldsymbol{b})$ denotes the Manhattan distance between two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. Further we resize the ring key as a single row vector such that it can be used to build KD-tree directly and search similar candidates efficiently. Then we can find the set $\mathcal{C}$ containing top similar keys and their corresponding distances. The number of top similar keys is a user-defined parameter. These candidates can be used for further scene recognition.

B. Fast Candidate Alignment

Owing to the rotation-invariance of the ring keys, we still need to detect the viewing angle between two frames. We construct the sector keys in the way similar to ring keys, as shown in Fig.3. Because sector is devided in azimuthal direction, two frames can be quickly aligned by shifting sector keys. Here we generate sector keys by using the sematic histogram of the points located in each sector. Let B_s denote the points located in the s th sector as:

$$B_s = \{B_{ij} |\ i \in [N_s],\ j = s\} \tag{17}$$

where the symbol $[N_s]$ represents $\{1, 2, ..., N_{s-1}, N_s\}$. Then we count the points number for each semantic class within the s th sector, the semantic histogram $\boldsymbol{h}(s)$ of the sector s can be defined as:

$$\boldsymbol{h}(s) = (h_s^1\ h_s^2\ ...\ h_s^{L-1}\ h_s^L) \tag{18}$$

where

$$h_s^l = \frac{\#b(p) = l}{\#p \in B_s},\ p \in B_s,\ l \in 1, ..., L \tag{19}$$

Therefore, the sector key can be expressed as a N_s-dimensional vector:

$$\boldsymbol{S} = (\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_{N_s}) \tag{20}$$

Then we use Manhattan distance to measure the distance between the corresponding semantic histograms of the query frame and the candidate:

$$d(\boldsymbol{S}^q, \boldsymbol{S}^c) = \sum_{s=1}^{N_s} d_{man}(\boldsymbol{h}(s^q), \boldsymbol{h}(s^c)) \tag{21}$$

where $\boldsymbol{h}(s^q)$ and $\boldsymbol{h}(s^c)$ are the semantic histograms in the s th sector of the query frame and the candidate separately. Hence the query sector key can be shifted to match with the candidate as follows:

$$D_s(\boldsymbol{S}^q, \boldsymbol{S}^c) = \min_{n \in [N_s]} d(\boldsymbol{S}_n^q, \boldsymbol{S}^c) \tag{22}$$

$$n_s^* = \underset{n \in [N_s]}{argmin}\ D_s(\boldsymbol{S}_n^q, \boldsymbol{S}^c) \tag{23}$$

where $\boldsymbol{S}_n^q$ is obtained by shifting n columns. At the same time, we can identify the inital value of column-shifts that allows the acceleration of matching process.

C. Fine Candidate Matching

By fast candidates re-identification and candidates alignment, we can get top similar candidates and their initial viewing angles. These two steps substantially reduce the search space for semantic scan context matching. Next we can make a fine matching between the query frame and candidates using the distance D_c. With the initial viewing angle n_s^* calculated by equation (11), the column-shifts range can be narrowed down. Then the equation (10) and (11) can be redefined as:

$$D_c'(\Omega^q, \Omega^c) = \min_{n \in [n_s^* \pm t_s]} d(\Omega_n^q, \Omega^c) \tag{24}$$

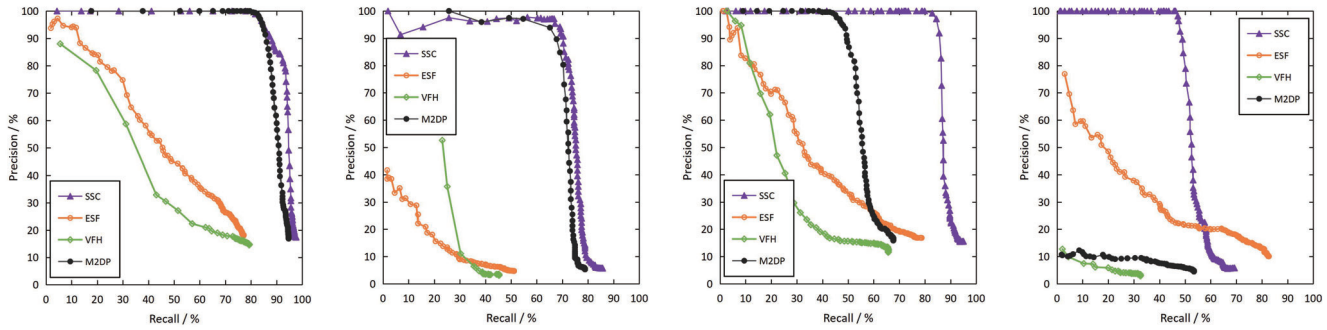
$$n^* = \underset{n \in [n_s^* \pm t_s]}{argmin}\ d(\Omega_n^q, \Omega^c) \tag{25}$$

where $[n_s^* \pm t_s]$ denotes $[n_s^* - t_s, n_s^* - t_s + 1,\ ...\ , n_s^* + t_s - 1, n_s^* + t_s]$, t_s represents the search radius around n_s^*, n^* is the viewing angle of the best match.

Due to the low resolution of semantic scan context, some information of the labeled pointcloud is inevitably lost. Thus direct thresholding the distance D_c' to filter unmatched pairs may lead to false positive. To alleviate this problem, we combine three distances above by calculating their weighted sum:

$$\begin{aligned} D(P^q, P^c) = a_1 \cdot D_r(\boldsymbol{R}^q, \boldsymbol{R}^c) \\ + a_2 \cdot D_s(\boldsymbol{S}^q, \boldsymbol{S}^c) \\ + a_3 \cdot D_c'(\Omega^q, \Omega^c) \end{aligned} \tag{26}$$

where P^q and P^c are the labeled point clouds of query frame and the candidate, seperately. a_1, a_2 and a_3 are the weights for each distance. Finally, given the set $\mathcal{C}$ of candidates



(a) KITTI00 (b) KITTI02 (c) KITTI05 (d) KITTI08

Fig. 4. Precision-recall curves for the evaluation dataset.

retrieved from the KD-tree of ring key, the closest candidate is supposed to be the revisited place if the distance is less than a threshold τ from the query:

$$c^* = \underset{c \in \mathcal{C}}{argmin}\ D(P^q, P^c), \quad \mathbf{s.t}\ \ D < \tau \tag{27}$$

where c^* is the index of the final place determined to be a loop closure.

IV. EXPERIMENT

In this section, the proposed method is validated on public datasets in urban scenes. We perform a comparision with other state-of-the-art methods: VFH, ESF and M2DP. Our semantic scan context(SSC) are implemented in C++. To unify the test platform, we use the C++ implementation of VFH and ESF in the Point Cloud Library (PCL), and we embed the Python codes of M2DP in our C++ test program for comparison experiment. The performance of these algorithms are analyzed using the KITTI dataset [20]. All the evaluations are running on a mobile laptop equipped with an Intel Core I5-9400 and 16GB RAM using Ubuntu 16.04 system.

A. Dataset and Experiment Settings

Semantic KITTI is a common choice for verifying the performance of place recognition. There are multiple sequences along different trajectories in the dataset and various environmental changes pose great challenge to loop closure detection. These sequences are recorded on a running car mounted with a 64-ray LiDAR, and the ground truth of trajectories is available. We consider it as a correct loop closure, if the ground truth distance between the query location and the matched location is less than 4m.

For each scan, we calculate these four descriptors and build KD-tree separately to search the matching candidates. The number of nearest neighbours is set to 10 for all tests. To avoid matching neighboring frames, 100 recent frames are excluded. We use the default parameters mentioned in M2DP [2], and we fix the normal radius of VFH to $0.03m$. As for the parameters of semantic scan context, we set $L_{max} = 80m$, $N_r = 20$ and $N_s = 60$. The weighting factors a_1, a_2, and a_3 of the discriminant function are empirically set to 6, 0.09 and 0.02 respectively.

B. Re-identification Performance Evaluation

In this work, we use precision and recall metrics to evaluate the performance of our proposed method and comparison methods. By adjusting the distance threshold of the nearest neighbour search, the recall and precision curve can be generated to compare the performance of different methods. We test these methods on 4 sequences including KITTI00, KITTI02, KITTI05, and KITTI08. The sequence 08 has only reverse loops, and others have loop events with the same directions. The sequence 02 contains both forward and reverse visit and is considered more challenging for place recognition. The recall-precision curves with respect to different methods are plotted in Fig.4. Note that the semantic scan context significantly outperform ESF, VFH and M2DP in all the datasets. The VFH only focuses on the angles between point normals and the centroid direction, which leads to the lose of point location information. While the ESF make full use of shapes of point cloud, ignoring other geometrical information. As for our method, it concentrates on the semantic histograms of static objects for each bins, which overcomes the impacts of dynamic objects and demonstrates the superiority over these state-of-the-art loop detection algorithms.

TABLE I
TIME COSTS ON KITTI00.

	Descriptor Calculating	Average Loop Searching	Maximum Loop Searching
SSC	0.0094	0.0042	0.0072
VFH	0.1470	0.0041	0.0093
ESF	0.0393	0.0089	0.0185
M2DP	0.5080	0.0031	0.0066

The computation cost of generating descriptor and nearest neighbour searching of all methods are assessed on KITTI 00. The results are listed in Table I, in which the time cost is measured in seconds. In particular, we record the maximum

time cost of searching time because it costs more time for KD-tree to search candidates with the growing number of frames. The table shows that M2DP is the most time-saving method for searching loop cloosure, while the time cost of searching semantic scan context is a little higher. It's mainly because the three-stage retrieval takes much more time. Nevertheless, the construction of semantic scan context is efficient enough, which is even faster than the others. Especially, a high-performance semantic network, such as the RangeNet++ [18] which only takes up to $82\ ms$ to process a LiDAR scan, can help the semantic scan context detect loop closure in real time.

V. Conclusion

In this work, we have presented a novel descriptor for performing place recognition with semantic information. The global descriptor is based on the semantic histogram of points received from 3D LiDAR sensor. Meanwhile, ring key and sector key are generated to realize fast retrieval and alignment. Furthermore, three kinds of distance are combined to better determine the matched pairs. Experiments on the benchmark dataset KITTI proof that the proposed method can achieve a competitive re-identificaiton performance and execute efficiently with a high-performance semantic segmentation network. It is still promising to combine much more information into the semantic scan context to adapt to more complex place recognition.

References

[1] D. Gálvez-López and J. D. Tardós, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[2] L. He, X. Wang, and H. Zhang, "M2dp: A novel 3D point cloud descriptor and its application in loop closure detection," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2016-Novem, pp. 231–237, 2016.

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

[4] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain," *IEEE International Conference on Intelligent Robots and Systems*, pp. 4758–4765, 2018.

[5] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[6] L. S.-v. Berlin, "Visual place recognition using landmark distribution descriptors," vol. 10114, pp. 487–502, 2017.

[7] P. Gao and H. Zhang, "Long-term loop closure detection through visual-spatial information preserving multi-order graph matching," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 369–10 376.

[8] ——, "Long-Term Loop Closure Detection through Visual-Spatial Information Preserving Multi-Order Graph Matching," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, pp. 10 369–10 376, 2020. [Online]. Available: https://aaai.org/ojs/index.php/AAAI/article/view/6604

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

[10] J. L. Schonberger, M. Pollefeys, A. Geiger, and T. Sattler, "Semantic Visual Localization," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 6896–6906, 2018.

[11] M. Angelina, U. Gim, and H. Lee, "PointNetVLAD : Deep Point Cloud Based Retrieval for Large-Scale Place Recognition Supplementary Material," no. c, pp. 1–2, 2018.

[12] E. Stenborg, C. Toft, and L. Hammarstrand, "Long-Term Visual Localization Using Semantically Segmented Images," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 6484–6490, 2018.

[13] K. P. Cop, P. V. Borges, and R. Dube, "Delight: An Efficient Descriptor for Global Localisation Using LiDAR Intensities," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3653–3660, 2018.

[14] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, no. January 2017, pp. 77–85, 2017.

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

[16] R. Dube, D. Dugas, E. Stumm, J. Nieto, R. Siegwart, and C. Cadena, "SegMatch: Segment based place recognition in 3D point clouds," *Proceedings - IEEE International Conference on Robotics and Automation*, no. May, pp. 5266–5272, 2017.

[17] R. Dubé, A. Cramariuc, D. Dugas, J. Nieto, R. Siegwart, and C. Cadena, "SegMap: 3D Segment Mapping using Data-Driven Descriptors," 2018.

[18] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and Accurate LiDAR Semantic Segmentation," *IEEE International Conference on Intelligent Robots and Systems*, no. i, 2019. [Online]. Available: https://github.com/PRBonn/lidar-bonnetal.

[19] N. Kobyshev, H. Riemenschneider, and L. V. Gool, "Matching features correctly through semantic understanding," in *2014 2nd International Conference on 3D Vision*, vol. 1, 2014, pp. 472–479.

[20] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9297–9307.