2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

SeqSphereVLAD: Sequence Matching Enhanced Orientation-invariant Place Recognition

Peng Yin[1], Fuying Wang[2], Anton Egorov[3], Jiafan Hou[4], Ji Zhang[1], Howie Choset[1]

***Abstract*—Human beings and animals are capable of recognizing places from a previous journey when viewing them under different environmental conditions (e.g., illuminations and weathers). This paper seeks to provide robots with a human-like place recognition ability using a new point cloud feature learning method. This is a challenging problem due to the difficulty of extracting invariant local descriptors from the same place under various orientation differences and dynamic obstacles. In this paper, we propose a novel lightweight 3D place recognition method, SeqSphereVLAD, which is capable of recognizing places from a previous trajectory regardless of the viewpoint and the temporary observation differences. The major contributions of our method lie in two modules: (1) the spherical convolution feature extraction module, which produces orientation-invariant local place descriptors, and (2) the coarse-to-fine sequence matching module, which ensures both accurate loop-closure detection and real-time performance. Despite the apparent simplicity, our proposed approach outperform the state-of-the-arts for place recognition under datasets that combine orientation and context differences. Compared with the arts, our method can achieve above 95% average recall for the best match with only 18% inference time of PointNet-based place recognition methods.**

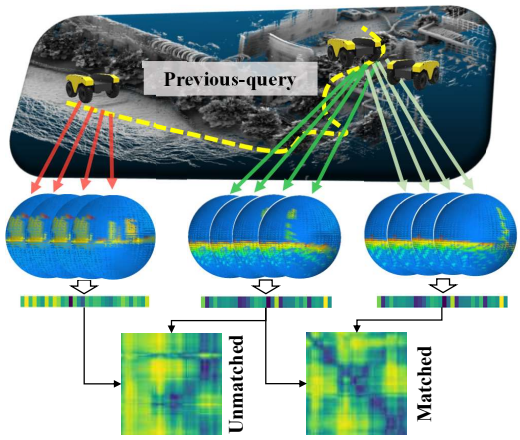

Fig. 1: Our place recognition method is based on spherical and sequenced feature extraction together. Instead of extracting place descriptors from 3D point clouds, our method project point cloud onto a spherical view where features are extracted. The approach then sequences the feature to form a place descriptor for matching places under viewpoint differences and measurement noise.

I. INTRODUCTION

Place recognition enables robots to recognize revisited locations, making it essential for loop closure detection in Simultaneous Localization and Mapping (SLAM) and global localization systems. Vision-based place recognition methods [1] usually suffer from illumination changing, dynamic observation and viewpoints changes. Comparing to a vision sensor, the LiDAR (Light Detection and Ranging) device is more robust to illumination variations, competitive in terms of ranging accuracy, and will have a more comparably price in the near future. However, recognizing the same place under orientation difference and temporary observation changes is a challenging task. Traditional place recognition methods rely heavily on the performance of 3D alignment algorithms, such as Iterative Closest Point (ICP).The performance of place recognition tasks via these traditional algorithms is sensitive to the viewpoint difference and computationally expensive in practice.

Recent studies on learning-based 3D data association [2] have brought light to the place recognition task [3, 4]. Instead of using 3D alignment on the raw data, the learning-based methods extract place descriptors directly from the raw point cloud, which makes them capable of achieving robust place recognition on public datasets. However, a common disadvantage of these methods is that they are sensitive to orientational differences, since they rely PointNet [2], which is sensitive to orientation changes.

To achieve orientation-invariant 3D place recognition while balancing the accuracy and efficiency simultaneously, we propose a lightweight place recognition framework, SeqSphereVLAD, which aims at orientation-invariant and real-time place identification as depicted in Figure. 1. The framework proposed by us mainly includes two modules: A Spherical Place Descriptor Extraction (*SPDE*) and a coarse-to-fine Sequence Matching module (*CFSM*). In the first module, instead of obtaining place descriptors from the raw data, *SPDE* extracts orientation-invariant place descriptors from a spherical perspective. To handle the uncertainty of a single observation, SeqSphereVLAD can achieve higher place recognition accuracy by utilizing previous coarse-to-

Peng Yin, Ji Zhang and Howie Choset are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. (pyin2, jizhang, choset@andrew.cmu.edu)

Fuying Wang is in the department of Electronic Engineering at Tsinghua University, Beijing, 100084, China. (thuwfy15@gmail.com)

Anton Egorov, Skolkovo Institute of Science and Technology, Moscow, 121205, Russia. (Anton.Egorov@skoltech.ru)

Jiafan Hou is in the School of Science and Engineering at The Chinese University of Hong Kong, Shenzhen, Shenzhen, 518172, China. (116010072@link.cuhk.edu.cn)

fine sequence matching approach [5], i.e., using sequence of frames to locate the best-matched target instead of one frame. However, transitional sequence matching is computationally expensive, and thus it can not be directly applied in real applications.

We conduct an extensive experimental analysis to evaluate our method using public datasets [6, 7] and data collected by ourselves. Notably, experiment results show that our method is remarkably more robust to orientation changes than state-of-the-art works. For each orientation differences in the place recognition task, our method guarantees robust and real-time place retrieval, while all other methods failed. The linked video[1] gives better visual results of the proposed method.

II. Related Work

In this section, we will first briefly introduce the related works on place recognition, then investigate recent developments in place matching approaches with existing place descriptors.

A. 3D Place Recognition

Place recognition is a well studied problem in the computer vision area [1]. With the development of convolution-based visual feature extraction, Arandjelovic *et al.* proposed a visual place feature learning framework NetVLAD [8], which enables end-to-end learning by aggregating global descriptors from local visual features. With the development of PointNet-based [2, 9] feature learning techniques and the instinctive advantage of LiDAR to illuminations changes, recent 3D place recognition approaches [3, 4] have made significant progress. Mikaela *et al.* [3] combined the feature extraction ability of PointNet [2] to obtain translation-invariant 3D place descriptor. Compared with traditional point cloud-based place recognition methods, such as ICP [10], PointNetVLAD [3] can directly obtain place descriptors from raw point clouds. Based on Mikaela's work, LPDNet [4] further improved place recognition accuracy by enhancing the local feature extraction ability with PointNet++ [9], which is designed to capture different scaled point features.

However, since PointNet is sensitive to rotation changes, PointNet-based approaches [3, 4] can not obtain invariant place descriptors under orientation differences. Esteves *et al.* proposed SphereCNN [11], which is computationally efficient and can capture orientation-equivariant features in the spherical domain. Inspired by Esteves' work, our SeqSphereVLAD captures orientation-invariant local descriptors via extracting orientation-equivariant features in spherical harmonics and clusters local features into global order-invariant descriptors.

B. Place Matching

The traditional place recognition methods usually apply Bag-of-Visual-Words (BoW) [12] to encode place descriptors into a tree-like structure, then use a single scan to retrieve the same places. FABMAP [13] uses a Bayesian filtering approach to achieve long-term place recognition over a 1000 km trajectory [13]. However, since the single scan usually contains measurement noise and observable context difference caused by the spatial viewpoint differences, this effect will lead to the place descriptor uncertainty in both BoW and FABMAP methods. Instead of relying on the single scan observation, SeqSLAM [5] utilizes a sequence of observations to improve the matching robustness, which significantly improves the place recognition accuracy under variant measurement noise and environmental conditions. However, since the brute-force searching procedure in sequence-based place recognition is time-consuming, these methods can not be directly applied in real-world applications. In our previous work [14], we introduce a coarse-to-fine searching approach, which balances the matching efficiency and accuracy at the same time.

SeqSphereVLAD can achieve robust and efficient place recognition in 3D by adopting the same coarse-to-fine matching approach in our previous work [14].

III. SeqSphereVLAD

In this work, the robot is assumed to move in the large-scale outdoor 3D environment. As sensory inputs, our method relies on the 3D point cloud generated from a LiDAR device and odometry estimated by Inertial Measurement Unit (IMU) or Visual Odometry. The proposed method runs three threads in parallel, i.e., Spherical Place Descriptor Extraction (*SPDE*), Coarse-to-Fine Sequence Matching (*CFSM*), and Global Reference Queries Maintaining (*GRQM*). Since the geometry features from one LiDAR scan are too sparse to distinguish, we apply a LiDAR odometry [15] method to accumulate continuous LiDAR inputs into a dense local map for the 3D place recognition task. In this section, we will introduce the system overall and the above three threads in details.

A. System Overall

In the *SPDE* thread, dense local maps are first transformed into the spherical views, then SphereVLAD extracts the orientation-invariant place descriptors via a spherical convolution operation and a feature clustering approach. In section III-B, we will investigate details of our SphereVLAD method, orientation-equivariant property of local descriptors extracted by spherical convolution, and orientation-invariant property of global place descriptors.

Given a temporary sequence S_{lt} and global reference sequences S_{gr}, the *CFSM* thread applies a coarse-to-fine sequence matching approach to retrieve the best match among thousands of potential candidates. In section III-C, we will investigate the coarse-to-fine sequence matching method, which is a multi-resolution sequence matching mechanism on the cached trajectories. *GRQM* is served as an assisting thread for *SPDE* and *CFSM* threads. When new place descriptors arrive from *SPDE*, *GRQM* accumulates local observations into temporary sequence S_{lt}, and restores the sequence

[1] https://youtu.be/MB3CF2yy2EU

Algorithm 1: SeqSphereVLAD

Input : O_t = Dense 3D Observation at timestamp t
Output: Matching Status

begin
  / SPDE thread */*;
  $Sph_t = MLG(O_t)$ *// Multi-layer Generation*
  $f_t = SphConv(Sph_t)$ *// Spherical Convolution*
  $d_t = FeatAgg(f_t)$ *// Feature Clustering*
  $[S_{lt}, S_{gr}] = GetQuery(d_t)$ *// Obtain Queries*
  / CFSM thread */*;
  $PT_t = Particle(S_{lt}, S_{gr})$ *// Particles Initialization*
  $score_t$ =MRS_Matching(PT_t);
  if $score_t \geq Match\_thresh$ **then**
    Local refinement;
    / GMQM thread */*;
    Update Query;
  else
    / GMQM thread */*;
    Add new Query;
  end
  return Matching Status;
end

into cached global reference sequences S_{gr}. Then *GRQM* extracts out the paired $[S_{lt}, S_{gr}]$ and feeds them to the *CFSM* thread. If loop closure is detected by *CFSM*, *GRQM* will update the orders of existing queries in the global reference sequence. Algorithm. 1 demonstrates the whole procedure of our SeqSphereVLAD method, which updates the global reference trajectory and infers best matches simultaneously. In the following sections, we will mainly investigate the *SPDE* thread and the *CFSM* thread respectively.

B. Spherical Place Descriptor Extraction

In this section, we describe the details of our Spherical Place Descriptor Extraction (*SPDE*) module in three steps. First, we define a multi-layer spherical projection for point clouds in Section III-B1. Then, we explain the orientation-equivariance of spherical convolution in Section III-B2. Finally, we demonstrate that our feature clustering operation can extract orientation-invariant place descriptors from the output of spherical convolution in Section III-B3.

1) Multi-Layer Spherical Projection: 3D input data first need to be converted to spherical functions for feeding into our spherical convolution. For mesh inputs, [11] used a two-channel spherical representation and achieved excellent 3D object classification and retrieval performance. However, for a point cloud, this projection method will weaken the neighboring association of each point and reduce the distinctness of the point cloud.

To capture more abundant geometric structures of point clouds, we propose an approach to generate multi-layer spherical perspectives. In practice, we equally divide the distance from center to spherical boundary into N ranges and extract the two-channel spherical signal within each range $L_k, 0 \leq k \leq N-1$. We concatenate spherical representations of each distance range and obtain the overall spherical representation for each point cloud. Note that our multi-layer spherical representation is equivariant to orientations, which is an essential prerequisite for extracting orientation-invariant features by spherical convolution.

2) Orientation-equivariant Local Descriptors Extraction: By analogy with 2D convolution that shows translation-equivariance, we introduce a spherical convolution network that equips the capability of extracting orientation-equivariant features from spherical representations of point clouds. To avoid space-varying distortions caused by planar projection, we convolve spherical signals into the harmonic domain instead of directly using the planar projections. The mathematical model of spherical convolution into harmonic domain shows its orientation-invariance.

Spherical convolution of $SO(3)$ signals f and h (f, h are functions: $SO(3) \rightarrow \mathbb{R}^K$) in the rotation group $SO(3)$ are defined as:

$$[f \star_{SO(3)} h](\mathbf{R}) = \int_{SO(3)} f(\mathbf{R}^{-1}\mathbf{Q}) h(\mathbf{Q}) d\mathbf{Q}. \quad (1)$$

where $\mathbf{R}, \mathbf{Q} \in SO(3)$. As the proof in [16], spherical convolution is shown to be orientation-equivariant:

$$[f \star_{SO(3)} [L_Q h](\mathbf{R}) = [L_{\mathbf{Q}}[f \star_{SO(3)} h]](\mathbf{R}) \quad (2)$$

where $L_{\mathbf{Q}}(\mathbf{Q} \in SO(3))$ is a rotation operator for spherical signals.

Practically, the convolution of two spherical signals f and h are computed by three steps. We first expand f and h to their spherical harmonic basis, then compute the point-wise product of harmonic coefficients, and finally invert the spherical harmonic expansion. (see more details in [17])

3) Orientation-invariant Place Descriptors: We leverage a feature clustering operation to convert the output of spherical convolution into an orientation-invariant place descriptor. Intuitively, there exists spatial similarity in output local descriptors of spherical convolution. Therefore, we cluster the local descriptors and take a sum of residuals (difference vector between descriptor and its corresponding cluster center) as global place descriptor.

Formally, given N D-dimensional local descriptors $\{\mathbf{x}_i\}$ as input, and K cluster centers ("visual words") $\mathbf{c}_k$ as VLAD parameters, the (j,k) element of output global descriptor $\mathbf{V}$ is:

$$\mathbf{V}(j,k) = \sum_{i=1}^{N} \overline{a}_k(\mathbf{x}_i)(x_i(j) - c_k(j)) \quad (3)$$

where $x_i(j)$ and $c_k(j)$ are the j-th dimension of the i-th descriptor and k-th cluster center, respectively.

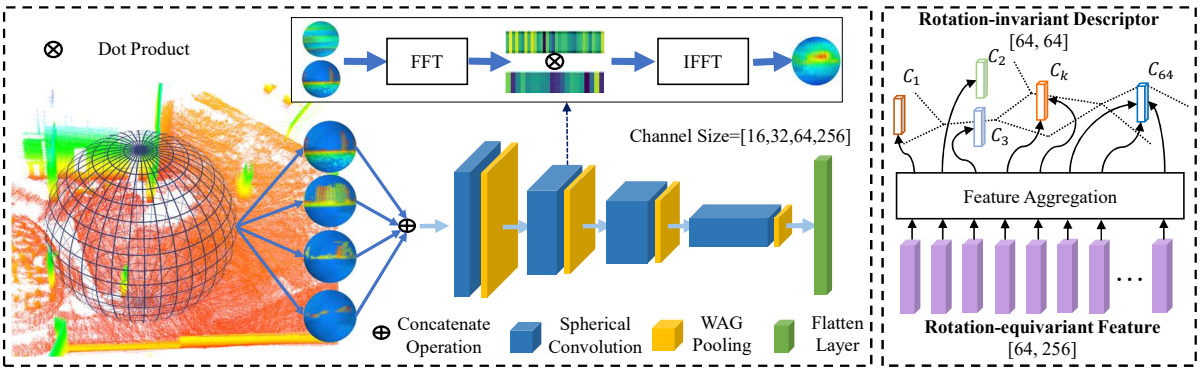

Fig. 2: Network structure of the SphereVLAD.. Given multi-layer spherical perspectives, SphereVLAD utilizes a spherical convolution module to obtain orientation-equivariant local features, and clusters such local features into the orientation-invariant place descriptor.

Here $\overline{a}_k(\mathbf{x}_i)$ has the following definition:

$$\overline{a}_k(\mathbf{x}_i) = \frac{e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}}{\sum_k e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}} \tag{4}$$

where vector $\mathbf{w}_k$ and scalar b_k are two learnable parameters.

Our feature clustering operation is invariant to orientation because it well respects the permutation invariance of input local descriptors (order of $\{\mathbf{x}_i\}$). Therefore, for the same point cloud with different orientations, output place descriptors of SphereVLAD are theoretically the same.

C. Coarse-to-Fine Sequence Matching

Given the extracted orientation-invariant place descriptor, we apply a sequence-matching approach to improve place recognition accuracy against the measurement noise and temporary observation differences. Given a sequence of global reference descriptors S_{gr} and a sequence of temporary descriptors S_{lt}, the proposed *CFSM* module can locate the best match via a particle filter-based on the global searching manner. We first down-sample both sequence descriptors S_{gr} and S_{lt} into the lowest resolution level with a skipping interval $v = V_0$; then particles are generated uniformly within the reference descriptors S_{gr}, where each particle represents a potential match between S_{lt} and S_{gr}. By updating the particle in a coarse-to-fine manner, we can locate the best match iteratively. We will introduce the particle initialization, particle and map updating, and complexity analysis respectively.

1) Particle Initialization: At the lowest resolution level, particles are sampled uniformly along the whole frame sequence. We define an overlapped area, which controls the overlapping ratio between temporary sequence S_{lt} and the global reference sequence S_{gr}. Then the initial number of particles P_{init} can be estimated by

$$P_{init} = \frac{M}{N} \cdot \frac{1}{1 - R_{Overlap}}, \tag{5}$$

where M and N are the sequence length of reference frames and temporary frames respectively, $R_{Overlap}$ is the overlap ratio, which ranges within $[0, 1]$. The entire particle sets have the following format

$$\begin{aligned} P &= \{p_t^{[1]}, p_t^{[2]}, p_t^{[3]}, ..., p_t^{[P_{init}]}\} \\ p_t^i &= [id_t^i, w_t^i], \end{aligned} \tag{6}$$

where id_t^i and w_t^i represent the index of predicted reference sequence and its corresponding weight for particle p_t^i.

2) Particle & Map Updating: For each particle, we evaluate its corresponding matching score by following the SeqSLAM [5] procedure. Please refer to the original paper for detailed explanation. The new particle weighting is obtained by $\hat{\omega}_k^i = \omega_{k-1}^i \times \frac{1}{1+e^{\ score_i}}$. After updating all particles, the particles' weights are further updated with a normalization operation $\omega_k^i = \frac{\hat{\omega}_k^i}{\sum \hat{\omega}_k^i}$. Based on the new particles' weighting,the effectiveness score of new particles P is calculated by $\hat{N}_{eff} = 1/(\sum (\omega_k^i)^2)$. If the $\hat{N}_{eff}$ is smaller than the given threshold $thresh_{eff}$, resampling on the new particles' distribution will be triggered.

The particles will converge to potential matching targets. We determine whether to change the sequence resolution level by evaluating an active coverage score $M_{cover} = \frac{M_{active}}{M_{active}+M_{negative}}$, If the convergence rate satisfies $M_{cover} \leq 50\%$, sequences S_{lt} and S_{gr} will be updated into a higher resolution level. Please note, we will not generate new particles within the negative areas, and only half of the particles will be kept to avoid the increasing computation consumption for a single particle.

3) Complexity Analysis: Given M reference frames and N temporary frames, for SeqSLAM, the complexity is $O(MN)$. For our *CFSM* method in map resolution level i with P_{init} initial particles, the complexity is $O\left(\frac{P_{init}}{2^i} N_i\right)$, where N_i is the number of testing frames on the i-th resolution level. Assume l_{max} is the maximum resolution

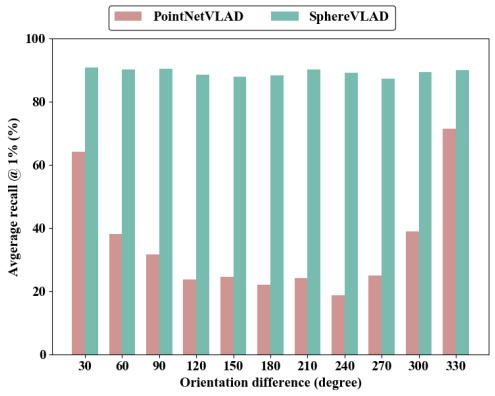

Fig. 3: Average recall (%) @1% of PointNetVLAD and SphereVLAD under 11 cases with different orientation. (*Campus* dataset) PointNetVLAD performs reasonably well when point clouds are orientationally aligned (within 30°), but the performance drops significantly as angular difference becomes larger (60° 300°). However, SphereVLAD shows orientation-invariance and outperforms PointNetVLAD under tested cases.

level, we will have $N_i = \frac{N}{2^{l_{max}\ i}}$ testing frames. Then,

$$\begin{aligned} \mathcal{C}_{\frac{Seq}{MRS}} =& \frac{O(MN)}{O\left(\sum_{i=0}^{l_{max}} \frac{P_{init}}{2^i} \cdot N_i\right)} \\ =& \frac{O(MN)}{O\left(\sum_{i=0}^{l_{max}} \frac{1}{2^i} \cdot \frac{M}{N} \cdot \frac{1}{1-R_{overlap}} \cdot \frac{N}{2^{l_{max}\ i}}\right)} \\ =& N \cdot (1 - R_{overlap}) \cdot \frac{2^{l_{max}}}{l_{max}}. \end{aligned} \tag{7}$$

where $\mathcal{C}_{\frac{Seq}{MRS}}$ is the computation complexity ratio between SeqSLAM and our*CFSM* method. If we set $l_{max} = 3$ and $R_{overlap} = 0.5$, the computation complexity ratio will be $\mathcal{C}_{\frac{Seq}{MRS}} = 1.33N$.

IV. Experiments

In this section, we show quantitative results in Section. IV-A to compare our method with state-of-the-art [3] for 3D place recognition. Also, we conduct a real-world 3D mapping experiment in Section. IV-B.

A. Place Recognition Results

1) Comparison with state-of-the-art: To assess benefits of our approach, we query the same point cloud in 6 different orientation cases for place recognition tasks. Specifically, we create 6 rotated point clouds (along z axis) for each one, where rotated angles are uniformly sampled from $[30°, 180°]$. Additionally, 5° random rotation noises are added to each point cloud to reduce discrepancies between the simulation and real world. We utilize SphereVLAD to generate place descriptors of point clouds, and find best candidate place descriptor by our coarse-to-fine sequence matching module. Similar to [3, 4], we use Average Recall @N and Average Recall @1% to evaluate place recognition accuracy.

TABLE I: Comparison results of *KITTI*, *Campus*, and *City* datasets under different networks. Here "(seq)" represents sequence matching and others are single frame matching. We use average (%) of Average Recall @1 under 6 different orientation cases to evaluate place recognition accuracy.

	KITTI	*Campus*	*City*
PN-STD	0.46	4.20	3.79
PN-MAX	0.69	2.75	7.38
PN-VLAD baseline	13.75	17.88	15.96
PN-VLAD refine	18.93	32.11	31.16
SPH-VLAD baseline	77.91	89.28	79.06
SPH-VLAD refine	88.63	91.40	81.58
PN-STD (seq)	2.27	8.64	5.76
PN-MAX (seq)	3.02	9.69	8.15
PN-VLAD baseline (seq)	34.31	20.07	23.82
PN-VLAD refine (seq)	43.54	56.25	46.12
SPH-VLAD baseline (seq)	99.70	98.82	97.01
SPH-VLAD refine (seq)	**99.93**	**98.88**	**99.04**

Comparison results are shown in Table. I. We divide 3D place recognition methods into single frame-based matching and sequence matching. In each of them, we compare our SphereVLAD (SPH-VLAD) with the original PointNet architecture with the max-pool layer (PN-MAX) and the PointNet trained for object classification in ModelNet (PN-STD) [2]. We also compare our method with the state-of-the-art PN-VLAD baseline and PN-VLAD refine with the same configuration as in [3]. Average Recall @1 in Table. I shows our SeqSphereVLAD outperforms other methods on *Campus* dataset, *KITTI* dataset and *City* dataset. In these evaluation cases, SeqSphereVLAD shows the best performance in all tested 6 orientation-different cases for each dataset.

2) Time and memory efficiency analysis: In Table. II, we examine time and memory efficiency of SphereVLAD and PointNetVLAD. In comparison with PointNetVLAD, our method consumes about 30% GPU memory for training and takes about 20% time for extracting place descriptor for a point cloud, which make it more suitable to be embedded in mobile robots.

TABLE II: Comparison result of time and memory requriements of PointNetVLAD and SphereVLAD.

Method	Training GPU memory	Run-time per frame
PN-VLAD	$7711M$	$55.00ms$
SPH-VLAD	**2459M**	**10.50ms**

3) Orientation-invariance analysis: In Figure. 3 which plots Average Recall @ 1% of PointNetVLAD [3] and SphereVLAD under 11 cases with different orientation, we verify the orientation-invariance of SphereVLAD. Average recall of PointNetVLAD drops significantly as orientation difference increases. However, SphereVLAD presents stable performance in all 11 orientation-different cases.

B. Real-world Mapping results

Our mobile robot platform contains a LiDAR device (Velodyne-VLP 16), an inertial measurement unit (Xsense

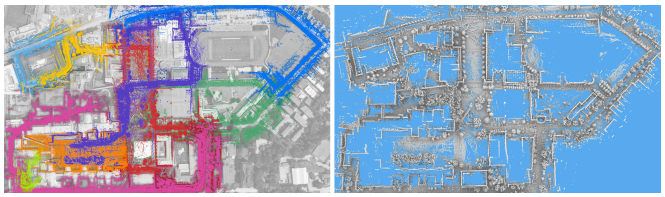

Fig. 4: Map merging from the corresponding sub-maps.

MTi 30, 0.5° error in roll/pitch, 1° error in yaw, $550mW$), a mini PC ($i7$ Intel NUC, 3.5 GHz, 28W), and an embedded GPU device (Nvidia Xavier, 8G memory). As shown in the Figure. 4, we built a 3D global map of Carnegie Mellon University campus to validate SeqSphereVLAD in large-scale outdoor environments.

V. Conclusions

In this paper, we proposed SeqSphereVLAD, a sequence matching enhanced orientation-invariant 3D place recognition method. We introduced the SphereVLAD place descriptor, which can extract orientation-invariant place descriptors from spherical representations of point clouds. Based on such descriptors, we design a coarse-to-fine sequence matching module to improve 3D place identification accuracy. The results on both public and self-generated datasets show that our method notably outperforms state-of-the-art in point cloud-based place recognition tasks. Furthermore, we embed this place recognition framework into our mobile robot platform for map merging of the campus area, which demonstrates the feasibility and usability of SeqSphereVLAD for the changing viewpoints and large-scale SLAM problems.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.

[3] M. Angelina Uy and G. Hee Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[4] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, and Y.-H. Liu, "Lpd-net: 3d point cloud learning for large-scale place recognition and environment analysis," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2831–2840.

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

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[7] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2016.

[8] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5297–5307.

[9] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in neural information processing systems*, 2017, pp. 5099–5108.

[10] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb 1992.

[11] C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis, "Learning so (3) equivariant representations with spherical cnns," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 52–68.

[12] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, Sept 2012.

[13] M. Nowakowski, C. Joly, S. Dalibard, N. Garcia, and F. Moutarde, "Topological localization using Wi-Fi and vision merged into FABMAP framework," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 3339–3344.

[14] P. Yin, R. A. Srivatsan, Y. Chen, X. Li, H. Zhang, L. Xu, L. Li, Z. Jia, J. Ji, and Y. He, "Mrs-vpr: a multi-resolution sampling based global visual place recognition method," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7137–7142.

[15] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and Systems*, vol. 2, 2014, p. 9.

[16] T. S. Cohen, M. Geiger, J. Köhler, and M. Welling, "Spherical cnns," *arXiv preprint arXiv:1801.10130*, 2018.

[17] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3 d shape descriptors," in *Symposium on geometry processing*, vol. 6, 2003, pp. 156–164.