2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

ReLoc-Aligner : Orientation-aware Scene Descriptor for Re-Localization within a 3D Point Cloud Map

SungJoon Cho[1,2] and Jun-Sik Kim[1,*]

***Abstract*— We propose a new orientation-aware scene descriptor *ReLoc-Aligner* for re-localization of a 3D point cloud. Re-localization within a 3D point cloud map is crucial for conducting Simultaneous Localization and Mapping (SLAM). Existing re-localization or place recognition methods of 3D LiDAR sensor data aim to estimate the current position of the sensor robustly to orientation changes. However, they do not determine the current orientation of the sensor within a 3D point cloud map, which limits their applications to re-localization or loop closing in SLAM. On the other hand, existing methods capable of orientation estimation tend to be slower than them. Our scene descriptor has a property of orientation awareness that enables us to extract the orientation difference between two scans directly from the descriptor. This is useful for the registration of point clouds from a good initial estimate, which leads to better re-localization of a scan. We propose a training method for the new descriptor. In addition, we develop fast querying and re-localization methods using the descriptors. Intensive experiments demonstrate that the proposed method is superior to the existing state-of-the-art methods in both place recognition and orientation estimation.**

I. Introduction

Robots and ground vehicles can perform simultaneous localization and mapping (SLAM) [1] using the 3D LiDAR scan data, which can be represented as a point cloud. To create an accurate map with SLAM, a loop closure detection process is essential, requiring the estimation of the current robot's pose. It is also required for a robot to estimate its pose in a map generated by another robot for integrating maps created by multiple robots, or collaborative SLAM [2]. Therefore, to estimate the current robot's position and orientation within a global point cloud map or global re-localization, is a commonly required task. For re-localization, the current scan point cloud is registered to the global map through point registration. The sensor pose is iteratively optimized from an initial pose estimate by maximizing the point registration. Thus, estimating accurate initial position and orientation or pose of the sensor is critical.

To find the current position on the global map, research on place recognition [3] has been conducted. Place recognition involves searching the database for the most similar scan data when revisiting previously traveled areas. For fast and accurate searching, the previous studies represent the 3D scan point cloud as a lightweight descriptor. They also focus on creating an orientation-invariant descriptor to robustly recognize its position even during reverse revisits.

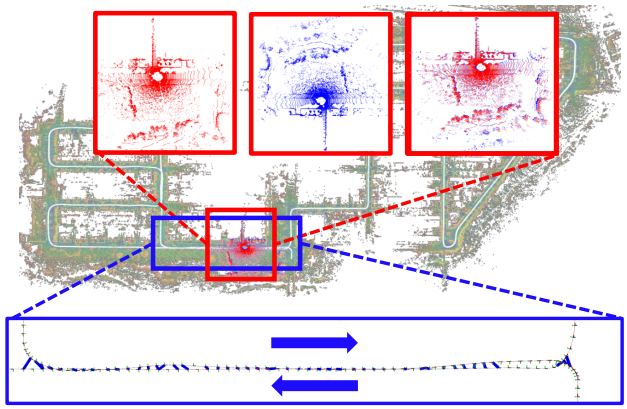


Fig. 1: *ReLoc-Aligner* achieves superior performance in the challenging KITTI08, which has many reverse revisit cases. This is achieved by accurately matching with close scans, as illustrated in the bottom figure with a blue boundary, and by performing successful point registration with accurate orientation estimation, as depicted in the upper figures with red boundaries.

PointNetVLAD [4] utilizes PointNet [5] to extract local features and NetVLAD [6] for aggregation, aiming to create a global descriptor that is invariant to the input order of 3D point clouds. Lin *et al.* [7] propose a method for learning geometric information using an SE(3)-equivariant encoder to ensure robustness to changes in robot rotation and translation. LPD-Net [8] suggests an adaptive local feature extraction module and the graph-based neighborhood aggregation module to create a discriminative global descriptor. Locus [9] employs SegMap-CNN [10] to extract features from segments and uses their topological and temporal relationships. OverlapTransformer [11] leverages a transformer network to extract discriminative features from range images and uses NetVLAD to produce yaw-angle-invariant global descriptors. LoGG3D-Net [12] applies a local consistency loss and utilizes quadruplet loss [13] to generate a discriminative global descriptor. It is also yaw-angle-invariant, which makes it robust to reverse revisits. The model achieves state-of-the-art performance on KITTI [14] and MulRan [15] datasets with low computing time cost. However, generating a yaw-angle-invariant global descriptor causes the loss of orientation information. To register the

[1]The authors are with the Center for Humanoid Research, KIST (Korea Institute of Science and Technology), Seoul, 02792, South Korea. {chosj, junsik.kim}@kist.re.kr
[2]SungJoon Cho is with the School of Electronic Engineering, Korea University, Seoul, 02841, South Korea.

This reseach was supported by the Challengeable Future Defense Technology Research and Development Program through the Agency For Defense Development(ADD) funded by the Defense Acquisition Program Administration(DAPA) in 2023(No. 915052101).

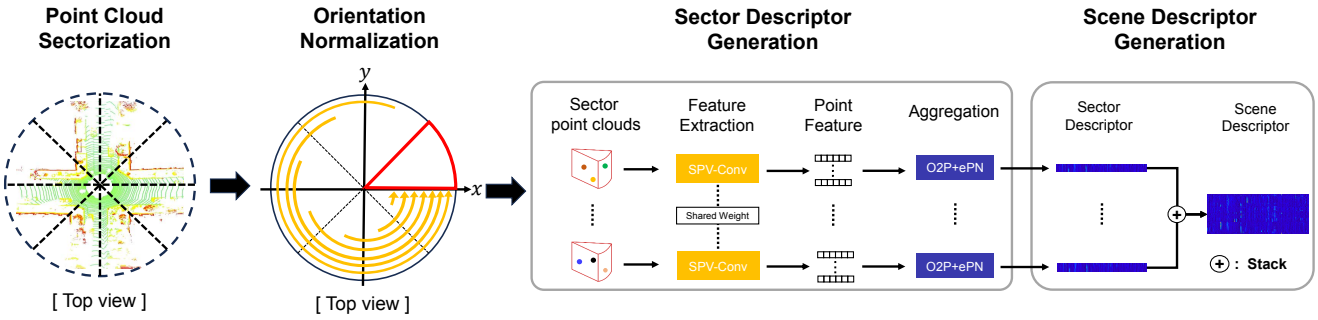


Fig. 2: The overall structure of the proposed orientation-aware scene descriptor generation from a scan. A point cloud from a 3D LiDAR scan is divided into N_S sectors radially. The orientation of each sector point cloud is normalized to the orientation of the red sector. Points in each sector are described as a single sector descriptor by point feature extraction and aggregation. A scene descriptor is finally built by stacking N_S sector descriptors in order.

scan, additional orientation estimation process is required.

Unlike these models, OverlapNet [16] estimates the difference of orientations between scans, contributing to the loop closure process. Oreos [17] suggests a network for orientation estimation using a regression loss function. IRIS [18] is generated by encoding the height information from the surrounding point cloud. Using the Fourier transform to estimate the translation between IRIS images helps in estimating the orientation difference between scans. Scan Context [19] preserves orientation information when representing a scan as a descriptor. It robustly recognizes positions during reverse revisits and estimates the orientation difference between the current scan and the most similar one in the database, demonstrating state-of-the-art performance across various datasets. Intensity Scan Context [20] also estimates the orientation difference between a scan pair using geometrical and intensity information.

However, the models that preserve orientation information show significantly low accuracy of place recognition in the cases of revisiting with positional offsets, and the search process is time-consuming. Therefore, it is required to have a scene descriptor that not only preserves orientation information but also enables fast and accurate estimation of both position and orientation as shown in Fig. 1.

The main contributions of our paper are summarized as follows:

- We propose a new orientation-aware scene descriptor, *ReLoc-Aligner*, which describes a scene as an ordered set of descriptors for radially divided sections.
- We propose a training method for the new descriptor for features in each sector to be consistent under some positional offset, while they are still sufficiently distinctive to those in the other sectors.
- We propose a method for fast re-localization that aligns the orientation between scans in advance using the orientation-aware property of the proposed descriptor. We can further accelerate the search for the closest scan across the scan database by analyzing the patterns of the proposed descriptors.
- We demonstrate the superior performance of the proposed *ReLoc-Aligner* to the current state-of-the-art methods in both a place recognition task and a final re-localization task through intensive experiments with the KITTI and MulRan datasets. Notably, we show that our descriptor achieves excellent results compared to others in KITTI08 which includes many reverse and orthogonal revisit cases.

II. METHOD

In this section, we propose a novel orientation-aware scene descriptor, *ReLoc-Aligner*, and a training method for the descriptor. We also present two fast re-localization methods to estimate a vehicle's position and orientation within a global map.

A. Orientation-aware Scene Descriptor

Extracting an orientation-aware scene descriptor *ReLoc-Aligner* from a raw point cloud takes four steps, as shown in Fig. 2. We divide the point cloud into radial sections, which are called as sectors. Each point cloud in each sector is transformed into a sector coordinate system to consistently describe its original geometry. Then, geometric features are extracted from each point cloud to create a sector descriptor. We build a orientation-aware scene descriptor by stacking the sector descriptors in order.

1) Point Cloud Sectorization: Scan data from a 3D LiDAR can be represented as a 3D point cloud $\mathcal{P} \in \mathbb{R}^{N_\mathcal{P} \times 3}$, consisting of $N_\mathcal{P}$ points with x, y, and z position values. We divide a point cloud radially into N_S equal-sized sections, referred to as sectors, as shown Fig. 2. We set N_S as 60. Each sector's point cloud is referred to as a sector point cloud $\mathcal{P}_{S^i}$. The sector index i starts from 0 at the heading of the 3D LiDAR, and increments by 1 in the counterclockwise direction. Each point cloud $\mathcal{P}_{S^i}$ in sector i is used to generate a sector descriptor $S(P_{S^i})$.

This sectorization idea is inspired by *Scan Context* [19]. While it separates the point cloud into multiple bins by dividing it in both azimuthal and radial directions and extracts a single representative value from each bin, we describe all the points in a radial sector to accurately encode the geometry in the sector.

2) Sector Orientation Normalization: To preserve the local geometry information, we normalize the orientation of each sector point cloud $\mathcal{P}_{S^i}$. A sector point cloud is normalized as, $\mathcal{P}'_{S^i} = R_{\theta_i} \cdot \mathcal{P}_{S^i}$, where R_{θ_i} is the rotation matrix for the yaw rotation $\theta_i = \frac{2\pi}{N_S} \cdot i$.

3) Sector Descriptor Generation: Each normalized sector point cloud is compactly represented as a sector descriptor. We use a network-based sector descriptor generation method, which encodes the point cloud into high-dimensional features and compresses these into a lightweight sector descriptor. The descriptor generation method from [12] is utilized for generating a sector descriptor. A SparseConv U-Net [21] is used to extract a d-dimensional geometrical feature $f(p) \in \mathbb{R}^d$ for each point p. Then, second-order pooling and eigenvalue power normalization (O2P+ePN) [22]–[24] is used to compress the point features to a d^2-dimensional sector descriptor $S(\mathcal{P}_{S^i}) \in \mathbb{R}^{d^2}$. We set d as 16.

4) ReLoc-Aligner Generation: By stacking the sector descriptors, a single scene descriptor $g(\mathcal{P})$, which is called *ReLoc-Aligner* is generated. The order of stacking follows the sequence of yaw angles from the sector point clouds, aligning them in a counterclockwise direction. The ith row of the scene descriptor represents the ith sector descriptor $g(\mathcal{P})_i = S(P'_{S^i})$, ranging from 0 to $N_S - 1$.

The structure of the scene descriptor ensures both the preservation of a geometrical feature of each sector point cloud and the order of sector orientation. Compared to other models' global descriptors [4], [9], [12], the proposed scene descriptor has capabilities not only for place recognition but also for orientation retrieval. We will explain how to use the proposed descriptor efficiently for the re-localization in Section II-C.

B. Training ReLoc-Aligner

To achieve precise re-localization, each sector descriptor should be consistent under orientation differences and should be distinct to those of the other sectors. We propose sector-based loss functions for point features and a scene descriptor, refined from the loss function described by [12].

To train *ReLoc-Aligner*, a training set consisting of an anchor scan ${}^{\mathtt{a}}\mathcal{P}$, a set of positive scans $\{\mathcal{P}_{pos}\}$, a set of negative scans $\{\mathcal{P}_{neg}\}$ and one of the other negative scans ${}^{\circ}\mathcal{P}$ from a 3D LiDAR scan sequence is required. The ground truth positions of an anchor, positive, negative, and the other negative scan are represented as $\mathtt{x_a}$, $\mathtt{x}_{pos}, \mathtt{x}_{neg}$ and $\mathtt{x_o}$, respectively. A scan is classified as a positive scan if its location is closer than τ_{pos} from the anchor scan, i.e., $\mathcal{D}(\mathtt{x_a}, \mathtt{x}_{pos}) < \tau_{pos}$. A scan is classified as a negative scan if its location from the anchor scan exceeds τ_{neg}, i.e., $\mathcal{D}(\mathtt{x_a}, \mathtt{x}_{neg}) > \tau_{neg}$. A scan is classified as the other negative scan if $\tau_{pos} < \mathcal{D}(\mathtt{x_a}, \mathtt{x_o}) < \tau_{neg}$. In this paper, we set τ_{pos} as 3 and τ_{neg} as 20.

1) Point Feature Loss in a Sector: A point corresponding to the same location is trained to have a consistent feature $f(p)$, regardless of the sensor's orientation in the global map. Each point should also have a distinctive feature even from other points within the same sector. To achieve this, we use the Hardest-Contrastive loss [25] between two nearby point clouds.

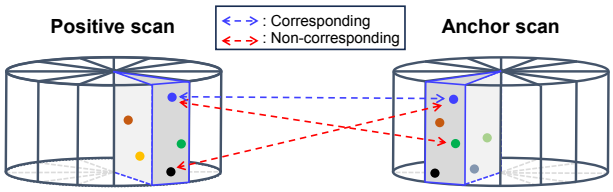


Fig. 3: The process for obtaining point feature loss. The corresponding and non-corresponding points of the blue point from the anchor scan are selected within the corresponding sector of the positive scan.

Specifically, an anchor point cloud ${}^{\mathtt{a}}\mathcal{P}$ and a positive point cloud ${}^{\mathtt{p}}\mathcal{P}$, which is randomly selected from the positive set $\{\mathcal{P}_{pos}\}$ are used for training the point feature. To find the pairs of correspondences between two point clouds, Iterative Closest Point (ICP) [26] is used to align the point clouds ${}^{\mathtt{a}}\mathcal{P}$ and ${}^{\mathtt{p}}\mathcal{P}$, utilizing initial ground truth relative pose.

An anchor point ${}^{\mathtt{a}}p_i$ from ${}^{\mathtt{a}}\mathcal{P}$ finds a corresponding point ${}^{\mathtt{p}}p_j$ from ${}^{\mathtt{p}}\mathcal{P}$ if $\mathcal{D}({}^{\mathtt{a}}p_i, {}^{\mathtt{p}}p_j) < \rho$, where ρ represents the distance threshold. In Fig. 3, the blue points in the positive and anchor scans are corresponding points. We define the set of point correspondences C between an anchor point cloud and a positive point cloud. We also define a set of points' index Q_i, randomly selected from the sector that contains ${}^{\mathtt{p}}p_i$. Based on a corresponding pair set C and a set of the possible Q, the point feature loss can be defined as follows:

$$
\begin{aligned}
L_p = \sum_{(i,j)\in C} \Bigg\{ & \left[\|f({}^{\mathtt{a}}p_i) - f({}^{\mathtt{p}}p_j)\|_2^2 - m_{pos} \right]_+ / |C| \\
& + \lambda I_i \left[m_{neg} - \min_{k \in \mathcal{Q}_j} \|f({}^{\mathtt{a}}p_i) - f({}^{\mathtt{p}}p_k)\|_2^2 \right]_+ / N_i \\
& + \lambda I_j \left[m_{neg} - \min_{k \in \mathcal{Q}_i} \|f({}^{\mathtt{p}}p_j) - f({}^{\mathtt{a}}p_k)\|_2^2 \right]_+ / N_j \Bigg\}
\end{aligned} \tag{1}
$$

where m_{pos}, m_{neg} and λ are the positive, negative scalar margins and a scalar weight, respectively. We set these hyperparameters as 0.1, 2.0, and 0.5. The indicator function I_i is 1 when $\mathcal{D}({}^{\mathtt{a}}p_i, {}^{\mathtt{p}}p_k) > \rho$ and 0 otherwise. The total count of valid I_i cases is represented as N_i, while N_j accounts for the I_j cases. The ReLU function $[\cdot]_+$ prevents loss from falling below zero.

2) Scene Loss: We propose a loss function based on the comparison between sectors for training a scene descriptor, refined from quadruplet loss [13]. The purpose of the loss function is to ensure that an anchor and its positive scans generate similar scene descriptors under orientation differences, while an anchor and its non-positive scans generate distinct scene descriptors. We consider not only the ground truth position but also the orientation to enhance the scene descriptor's capability for place recognition and its orientation awareness.

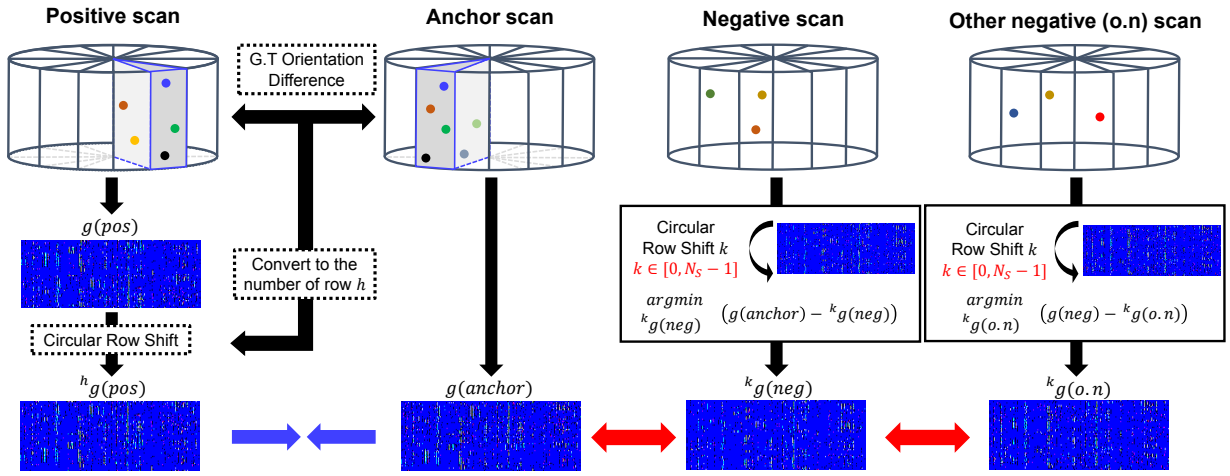


Fig. 4: Overall process to obtain scene descriptor loss for training *ReLoc-Aligner*. The feature distance between the anchor and its positive scan, aligned by the ground truth orientation difference, is used for the training. For the non-positive pairs, the smallest feature distance among whole distance cases of circular row shifts is used.

The methods for calculating the feature distance between the anchor and the positives and for the non-positives (negative and other-negative) scene descriptors are different. The anchor and positive point clouds are scanned within a sufficiently close distance of τ_{pos}, and the point registration between those is possible. By registering those point clouds to each other, the orientation difference θ_p between the two point clouds can be determined. As shown in Fig. 4, the anchor and positive scene descriptors are aligned by circularly shifting the rows with the ground truth orientation difference θ_p, where $h = \frac{N_S \cdot \theta_p}{2\pi}$ is the amount of row shift. The feature distance for the anchor scene descriptor $\mathrm{g}(\mathcal{P}_1)$ and positive scene descriptor $\mathrm{g}(\mathcal{P}_2)$ is calculated as

$$d(h, \mathrm{g}(\mathcal{P}_1), \mathrm{g}(\mathcal{P}_2)) = \frac{1}{R} \sum_{r=0}^{N_S-1} \left[V \left\| \mathrm{g}(\mathcal{P}_1)_r - {}^{h}\,\mathrm{g}(\mathcal{P}_2)_r \right\|_2^2 \right] \tag{2}$$

where $V(r, \mathrm{g}(P_1), \mathrm{g}(P_2))$ is a validity indicator V that indicates the availability of the sensor data when some part of the sensor data can not be used by self-occlusion with parts such as a sensor fixture. Unavailable sensor data are set to be zero, and the indicator function is 0 when the inner product of the r-th rows in the two scene descriptors is 0, 1 otherwise. The sum of all the validity indicators V is represented as R.

The feature distance between the anchor and non-positive scene descriptors is defined as the smallest feature distance among all comparison cases of circular row shifts, as shown in Fig. 4. The purpose of training is to distinguish the anchor and negative scene descriptors, even when the difference is as minimal as possible.

$$d_n(\mathrm{g}(\mathcal{P}_1), \mathrm{g}(\mathcal{P}_2)) = \min_{k \in [0, N_S-1]} \left\{ d(k, \mathrm{g}(\mathcal{P}_1), \mathrm{g}(\mathcal{P}_2)) \right\} \tag{3}$$

Among the positive set $\{\mathcal{P}_{pos}\}$, a positive scan which has the largest feature distance with the anchor is defined as the hardest positive scan P_{hp}, as follows

$$\mathcal{P}_{hp} = \underset{\mathcal{P}_{p^i} \in \{\mathcal{P}_{pos}\}}{\mathrm{argmax}}\ d(h_i, \mathrm{g}({}^{\mathrm{a}}\mathcal{P}), \mathrm{g}(\mathcal{P}_{p^i})) \tag{4}$$

where h_i is the amount of circular row shift between $g({}^{\mathrm{a}}\mathcal{P})$ and $g(\mathcal{P}_{p^i})$.

ReLoc-Aligner loss is defined as

$$L_g = \sum_{i=1}^{n} \Big\{ [d(\mathrm{g}({}^{\mathrm{a}}\mathcal{P}), \mathrm{g}(\mathcal{P}_{hp})) - d_n(\mathrm{g}({}^{\mathrm{a}}\mathcal{P}), \mathrm{g}(\mathcal{P}_{neg^i})) + \alpha]_+ + [d(\mathrm{g}({}^{\mathrm{a}}\mathcal{P}), \mathrm{g}(\mathcal{P}_{hp})) - d_n(\mathrm{g}({}^{\mathrm{o}}\mathcal{P}), \mathrm{g}(\mathcal{P}_{neg^i})) + \beta]_+ \Big\} \tag{5}$$

where n is the number of $\{\mathcal{P}_{neg}\}$, α and β are scalar margins. We set n as 2, α as 0.5 and β as 0.3.

Then, the total loss is defined as

$$L = L_p + L_g \tag{6}$$

C. Re-localization Method

Re-localization is the process of determining the position and yaw angle within a global map. This involves identifying a position by searching for the most similar scan in the database whose scene descriptor should be closely matched to the scene descriptor of the query scan. The search for the closest resemblance includes performing a circular row shift on each reference descriptor and calculating the cosine distance between it and the query. This process is repeated N_S times for each reference, referred to as a full shift-based comparison. For each reference, the smallest cosine distance across all comparison cases indicates the difference between the two descriptors. The full shift-based comparison is carried out for all references, and the one with the least difference from the query scene descriptor becomes the matching candidate.

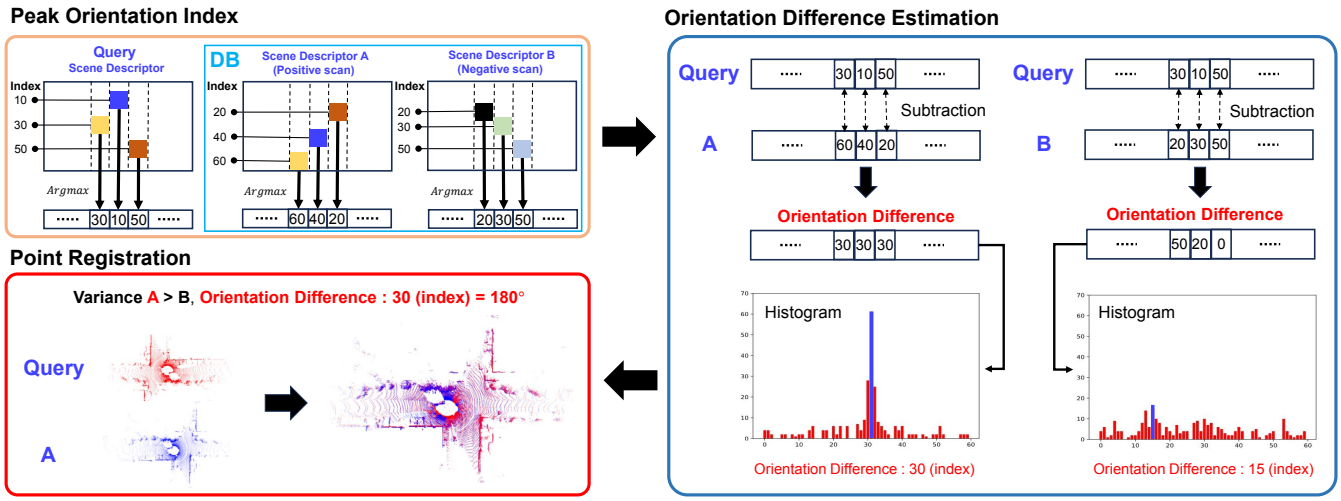


Fig. 5: Whole process of searching for the most similar scene descriptor in the database to the query scene descriptor and estimating the relative orientation angle. First, a peak orientation index (POI) is created for each scene descriptor using the column-wise argmax information. The difference between the POIs of the query and those of the database is then converted into histograms. The most occurred index difference implies the optimal orientation difference. Since the variance of the occurred index differences of the scan **A** is the largest in the database, the query and **A** point clouds are registered using the orientation difference angle.

Thus, the difference between two scene descriptors $\mathrm{g}(\mathcal{P}_1)$ and $\mathrm{g}(\mathcal{P}_2)$ is defined as:

$$\min_{k\in[0,N_S-1]} \left\{ 1 - \frac{1}{R} \sum_{r=0}^{N_S-1} \left[V \frac{\mathrm{g}(\mathcal{P}_1)_r \cdot^k \mathrm{g}(\mathcal{P}_2)_r}{\|\mathrm{g}(\mathcal{P}_1)_r\| \cdot \|^k \mathrm{g}(\mathcal{P}_2)_r\|} \right] \right\} \quad (7)$$

where R, V follows Eq. (2). The amount of shifted rows k can be converted into yaw angle $\theta_{yaw} = \frac{2\pi k}{N_S}$. The full shift-based comparison method is inspired by Scan Context [19] and shows good accuracy for our model as well. However, repeating the process N_S times to obtain a minimum value is time-consuming and thus inefficient. In the KITTI00, it takes 18 seconds for a query to search through a database containing 2,127 descriptors. We develop faster search methods to reduce this computational burden in the full shift-based comparison.

1) Peak Orientation Indices: We observe that the element-wise sectional maximums in the proposed scene descriptor appear in similar patterns under orientation variations and small positional changes. As illustrated in the left top in Fig. 5, we extract the orientation indices of the peak descriptor values.

The peak orientation indices (POI) is formed as a d^2-dimensional vector as :

$$m(g) = (r(c_0), ..., r(c_{d^2-1})), \quad (8)$$

where $r : c_i \rightarrow \mathbb{R}$. $r(c_i)$ extracts argmax value of each column value set c_i as follows,

$$r(c_i) = \underset{j\in\{0,1,..,N_S-1\}}{\mathrm{argmax}} \; c_{ij}, \quad (9)$$

where c_{ij} means a jth value of c_i. Each scene descriptor has a fixed POI vector.

2) Orientation-first matching using POI: Under orientation variation, the scene descriptor becomes circular row-shifted as described in Section II-C, and the POI should have the same offset in each element, ideally. The offset implies the relative orientation difference.

The element-wise offset ${}^d c_i$ between the query and reference POI is calculated by

$${}^d c_i = \begin{cases} m(g_1)_i - m(g_2)_i + N_S & \text{if } m(g_1)_i - m(g_2)_i < 0, \\ m(g_1)_i - m(g_2)_i & \text{otherwise.} \end{cases} \quad (10)$$

where i is an index of the column. To find the maximum likely offset between two POIs, we count the occurrences of offsets ${}^d C = \{{}^d c_0, {}^d c_1, ..., {}^d c_{f^2}\}$ as a histogram resulting in an N_S-dimensional vector. The offset with the highest frequency tends to be the amount of row shift for the reference, and it represents the orientation difference between two scans, as shown on the right in Fig. 5. This simple process is executed in a batch for all reference descriptors at once. Each reference descriptor is directly aligned by circularly shifting with the offset. This reduces the computational cost significantly by removing the N_S-repeating circular row shifts in the full shift-based comparison, without loss of accuracy.

3) Faster Re-Localization Method: The orientation-first matching using the POI reduces the computational cost a lot. However, it still requires performing the circular row shifts and measuring the cosine distances for all descriptors in the database. To achieve even faster querying, we introduce a method for selecting the best candidate based on the variance of the POI histogram.

If two scene descriptors generated at nearby positions differ only by the amount of yaw angle, the resulting column-

	KITTI							MulRan					
	00	02	05	06	07	08	mean	K1	K2	K3	D3	R2	mean
ScanContext [19]	0.966	0.871	0.914	0.985	0.698	0.610	0.841	0.954	0.969	0.994	0.893	0.826	0.916
PointNetVLAD [4]	0.909	0.637	0.859	0.924	0.171	0.437	0.656	0.952	0.856	0.979	0.685	0.868	0.868
Locus [9]	0.983	0.762	**0.981**	0.992	**1.000**	0.931	0.942	0.938	0.874	0.969	0.718	**0.994**	0.899
LoGG3D-Net [12]	0.953	0.872*	0.976	0.977	**1.000**	0.876*	0.942	0.966	0.938	0.991	**0.977**	0.969	0.968
ReLoc-Aligner	**0.987**	**0.884**	0.972	**0.998**	0.943	**0.970**	**0.959**	**0.995**	**0.993**	**0.998**	0.950	0.972	**0.982**
ReLoc-Aligner (Fast)	0.982	0.883	0.965	0.989	0.926	0.946	0.949	0.982	0.958	**0.998**	0.876	0.976	0.958

TABLE I: **Result of place recognition with $F1_{max}$ on the KITTI and MulRan datasets.** *For each query scan pose, we determined whether a revisit case exists within 3 meters of the query pose in the database poses and re-evaluated accordingly. We used the pre-trained models provided by the authors.

wise offsets ${}^{d}C$ will exhibit consistently similar values. Occurrence counts or the histogram of the column-wise offsets are depicted in Fig. 5. For a matched pair, the significant peak is observed in the histogram as illustrated by $\mathbf{A}$ in Fig. 5. In contrast, it is not observed for unmatched pair as shown in $\mathbf{B}$. To measure the consistency of the column-wise offsets, we simply calculate the variance of their occurrence counts, which tends to be greater for matched pairs. We select the scan with the highest variance of the occurrence counts as a candidate match. By performing a row shift only once and measuring cosine distance for the best candidate scene descriptor, faster re-localization than previous methods becomes achievable. The computational cost is significantly reduced by removing the full comparison in the database.

4) Pose Refinement: The position and orientation in the global map can be refined by point registration between the query and the candidate scans. Because the proposed retrieval methods can choose the scan close enough as well as its relative orientation, a simple ICP can be used for accurate point registration with a good initial guess as shown on the left bottom in Fig. 5.

III. EXPERIMENTAL EVALUATION

A. Dataset

KITTI : The KITTI dataset consists of 11 sequences, numbered 00 through 10. Each contains point cloud data scanned with the Velodyne HDL-64E along with ground truth scan poses. Sequences include forward, orthogonal, and reverse revisit cases. Specifically, sequence 08 includes multiple reverse and orthogonal revisit cases, making it particularly useful for assessing the robustness of reverse revisit. We train our model on 10 of these sequences and use the remaining one sequence for evaluation. Model evaluation is carried out on 6 sequences: 00, 02, 05, 06, 07, and 08. Therefore, we trained six models for evaluation, each with a different sequence.

MulRan : The MulRan dataset includes ground truth scan poses and point cloud data scanned with the Ouster OS1-64. It features occlusions, resulting in a lower overlap ratio between scans during reverse revisits. Additionally, it contains revisit cases with both rotation and translation applied, presenting a more challenging localization problem compared to other datasets. For model training, we utilize the DCC 01, 02, and Riverside 01, 03 sequences, while for evaluation, we use the KAIST 01, 02, 03, DCC 03, and Riverside 02 sequences.

B. Place Recognition Result

The place recognition capability of the proposed descriptor is evaluated using the $F1_{max}$ metric, which leverages precision and recall, as shown in Table I. A pair of a query and the candidate descriptor is defined as positive if the cosine distance is less than or equal to threshold, and negative otherwise. The criteria for true and false positives are determined based on the physical pose distances between the query and candidates, whether it is less than or equal to 3 meters or more than 20 meters, respectively. True and false negatives are determined based on whether a revisit case exists within 3 meters of the query pose. The database used for comparison with each query includes scan data before the query timestamp, excluding the t seconds prior to the query. For KITTI evaluation, t is set to 30, while for MulRan, it is set to 90. We evaluates the capability using our model, Scan Context [19], PointNetVLAD [4], Locus [9], and LoGG3D-Net [12], as conducted in the experiment of [12]. Our model, *ReLoc-Aligner* is evaluated in two versions: the default and the fast version. The default version from Section II-C.2 involves measuring cosine distances of all descriptors with a row shift after orientation estimation, while the fast version from Section II-C.3 selects a single candidate based on the variance of the POI offset histogram.

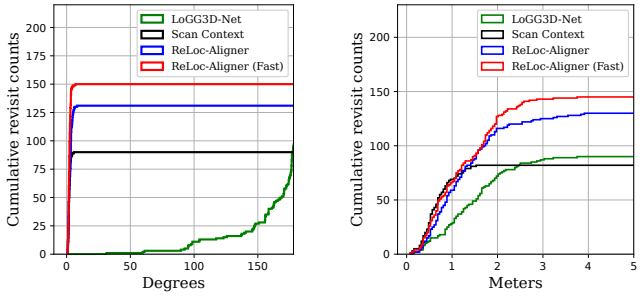
ReLoc-Aligner achieved the best mean score on both KITTI and MulRan datasets. It showed significantly better results in KITTI08, which consists only of reverse and orthogonal revisits, with a score of 0.970. This is about 4% and 10% better than the state-of-the-art methods Locus and LoGG3D-Net, respectively. For the KAIST 01-03 sequences of MulRan dataset, which consists of many rotational and translational revisits, the proposed methods also achieved results exceeding 0.99. The fast version of *ReLoc-Aligner* showed comparable results to existing state-of-the-art models in both the KITTI and MulRan datasets. In KITTI08, it achieved an excellent score of 0.946, outperforming other models. We demonstrate that both versions of our model, *ReLoc-Aligner*, are robust in recognizing position regardless of orientation and positional differences compared to previous scan data.

TABLE II: Result of relative rotation and translation errors between matched scans on the KITTI dataset.

	Sequence 00			Sequence 08		
	Success	RE [deg]	TE [m]	Success	RE [deg]	TE [m]
LoGG3D-Net [12]	96.1%	6.128	2.757	0 %	159.657	35.907
Scan Context [19]	**100%**	**0.010**	**0.008**	86.3 %	9.616	14.22
ReLoc-Aligner	99.7%	0.018	0.445	96.7%	**0.083**	5.471
ReLoc-Aligner (Fast)	99.9%	0.017	0.151	**98.5%**	1.383	**1.890**

TABLE III: Result of relative rotation and translation errors between matched scans on the MulRan dataset.

Approach	KAIST 03			DCC 03		
	Success	RE [deg]	TE [m]	Success	RE [deg]	TE [m]
LoGG3D-Net [12]	98.5%	0.803	0.909	92.6%	11.218	1.690
Scan Context [19]	**100%**	**0.008**	**0.003**	81%	5.623	46.343
ReLoc-Aligner	**100%**	0.009	0.154	96.9%	0.125	5.229
ReLoc-Aligner (Fast)	99.9%	0.010	0.317	**99.7%**	**0.029**	**0.784**



(a) Relative Rotation Error. (b) Relative Translation Error.

Fig. 6: Cummulative counts of detected revisits on rotation and translation error. It was measured between the query and the positive candidates from KITTI08.

C. Pose Refinement Accuracy

In this section, we evaluate the accuracy of the relative transformation between the query and a candidate scan as estimated by each model. The sequences used for evaluation, KITTI00 and KAIST03 are the sequences with a relatively high number of forward revisits. KITTI08 is composed of revisits in reverse and orthogonal directions, and DCC03 involves a significant number of revisits with reverse and translation.

The results for KITTI are detailed in Table II, and for MulRan in Table III. The success rate refers to the percentage of the correctly identified pairs of a query and candidate scan as positive pairs. For a correct pair, its physical pose distance is less than 3 meters, its relative rotation error is less than $5°$, and its translation error is less than 2 meters. The relative rotation error (RE) and the relative translation error (TE) are measured as the average difference between the 6D ground truth and the estimated 6D relative transformation after ICP registration between a query and its positive match.

The comparison of RE and TE results includes both the default and fast versions of the *ReLoc-Aligner*, as well as state-of-the-art models. This includes the LoGG3D-Net, known for its superior place recognition capabilities, and the Scan Context, noted for its exceptional ability to estimate the yaw angle between a scan pair. LoGG3D-Net conducts ICP using the identity matrix as the initial transformation. For our model and Scan Context, the estimated relative yaw angle between the pairs is provided as the initial transformation for ICP.

Our model and Scan Context show success rates close to 100% in KITTI00 and KAIST03 mainly composed of forward revisit cases. Notably, both the default and fast versions of *ReLoc-Aligner* exhibit significantly lower RE and TE results in KITTI08 and DCC03 compared to other models. In KITTI08, the fast version demonstrates about $8°$ and over 13 meters lower RE and TE results than Scan Context, and about $6°$ and 46 meters lower results in DCC03. Moreover, it demonstrates exceptional performance across all four sequences, achieving less than $1.5°$ of the RE and less than 2 meters of the TE. When executing SLAM in an unknown environment to the robot, the map is built based on the estimated re-localization results, without prior knowledge of the correct result. Lower RE and TE values indicate that it contributes to accurate loop closure detection and the creation of integrated maps using multi-robots during SLAM operations.

We also evaluate the result of the relative rotation and translation error before point registration within KITTI08, as shown in Fig. 6. It shows the cumulative counts of the detected revisits, which are used for initial poses for the ICP registration. This result demonstrates that the proposed methods chose more and better candidates even in their detection stage. Specifically, both versions of our model could find more positives with relative rotations of less than 5 degrees than the other state-of-the-art methods. For the evaluation of relative translation error, both versions of our model could find more positives within a small error margin of less than 2 meters than the others, with the majority of the error distribution existing within 3 meters.

D. Computation Time

We measured the time cost for offline and online processes with the KITTI00. For the offline test, all the scans from the sequence were used as candidates. For the online test, we ran the modified SLAM implementation of [27] and selected keyframe scans every 2 meters or in every $10°$. In the common description task for the two tests, it took 175ms

TABLE IV: Analysis for re-localization computation time : Average time costs on KITTI00 (in ms).

	Offline Test	Online Test
ReLoc-Aligner	352	65
ReLoc-Aligner (Fast)	15	4

to complete. When performing a re-localization task, the computation time for the default and fast versions of *ReLoc-Aligner* are noted in Table IV. The default and faster versions of our methods took 352ms and 15ms, respectively, in the offline tests. In the online SLAM tests, it operates much faster with times of 65ms and 4ms, respectively. In both tests, we used a system with an i9-10900 CPU, 64GB RAM and a GPU of a RTX 4090Ti. We implemented the algorithm in Python, and we expect significant speed-up when processed in C++.

IV. CONCLUSION

In this paper, we propose a new orientation-aware scene descriptor *ReLoc-Aligner*, which describes a scene as an ordered set of local descriptors for radially divided sectors. The existing scene descriptors for place recognition are usually designed to be rotation-invariant, but it makes the point registration and re-localization significantly harder. The orientation awareness of the proposed descriptor is useful to register two 3D scan point clouds without additional matching or registration. We propose a training method for the new descriptor to be consistent under orientation variation, while it can still have enough distinctiveness to each other. We also develop fast re-localization methods using the proposed descriptor by analyzing its patterns under rotation and translation. Various and intensive tests show that the proposed descriptor can achieve better performance on place recognition and make it possible to re-localize scans in various revisit directions by point registration.

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] P.-Y. Lajoie, B. Ramtoula, F. Wu, and G. Beltrame, "Towards collaborative simultaneous localization and mapping: a survey of the current research landscape," *arXiv preprint arXiv:2108.08325*, 2021.

[3] P. Yin, S. Zhao, I. Cisneros, A. Abuduweili, G. Huang, M. Milford, C. Liu, H. Choset, and S. Scherer, "General place recognition survey: Towards the real-world autonomy age. sep. 2022," *arXiv preprint arXiv:2209.04497*, 2023.

[4] M. A. Uy and G. H. Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4470–4479.

[5] 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.

[6] 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.

[7] C. E. Lin, J. Song, R. Zhang, M. Zhu, and M. Ghaffari, "Se (3)-equivariant point cloud-based place recognition," in *Conference on Robot Learning*. PMLR, 2023, pp. 1520–1530.

[8] 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/CVF international conference on computer vision*, 2019, pp. 2831–2840.

[9] K. Vidanapathirana, P. Moghadam, B. Harwood, M. Zhao, S. Sridharan, and C. Fookes, "Locus: LiDAR-based Place Recognition using Spatiotemporal Higher-Order Pooling," *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 5075–5081, 2021.

[10] R. Dube, A. Cramariuc, D. Dugas, H. Sommer, M. Dymczyk, J. Nieto, R. Siegwart, and C. Cadena, "Segmap: Segment-based mapping and localization using data-driven descriptors," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 339–355, 2020.

[11] J. Ma, J. Zhang, J. Xu, R. Ai, W. Gu, and X. Chen, "Overlaptransformer: An efficient and yaw-angle-invariant transformer network for lidar-based place recognition," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6958–6965, 2022.

[12] K. Vidanapathirana, M. Ramezani, P. Moghadam, S. Sridharan, and C. Fookes, "LoGG3D-Net: Locally Guided Global Descriptor Learning for 3D Place Recognition," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2022.

[13] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: a deep quadruplet network for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 403–412.

[14] 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.

[15] G. Kim, Y. S. Park, Y. Cho, J. Jeong, and A. Kim, "Mulran: Multimodal range dataset for urban place recognition," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6246–6253.

[16] X. Chen, T. Läbe, A. Milioto, T. Röhling, O. Vysotska, A. Haag, J. Behley, and C. Stachniss, "Overlapnet: Loop closing for lidar-based slam," *arXiv preprint arXiv:2105.11344*, 2021.

[17] L. Schaupp, M. Bürki, R. Dubé, R. Siegwart, and C. Cadena, "Oreos: Oriented recognition of 3d point clouds in outdoor scenarios," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 3255–3261.

[18] Y. Wang, Z. Sun, C.-Z. Xu, S. E. Sarma, J. Yang, and H. Kong, "Lidar iris for loop-closure detection," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5769–5775.

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

[20] H. Wang, C. Wang, and L. Xie, "Intensity scan context: Coding intensity and geometry relations for loop closure detection," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2095–2101.

[21] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han, "Searching efficient 3d architectures with sparse point-voxel convolution," in *European Conference on Computer Vision*, 2020.

[22] P. Koniusz, F. Yan, P. Gosselin, and K. Mikolajczyk, "Higher-order occurrence pooling for bags-of-words: Visual concept detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 313–326, 2017.

[23] P. Koniusz and H. Zhang, "Power normalizations in fine-grained image, few-shot image and graph classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.

[24] P. Li, J. Xie, Q. Wang, and W. Zuo, "Is second-order information helpful for large-scale visual recognition?" in *Proceedings of the IEEE International Conference on Computer Vision*, dec 2017, pp. 2089–2097.

[25] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 8958–8966.

[26] 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, 1992.

[27] G. Kim, S. Yun, J. Kim, and A. Kim, "Sc-lidar-slam: A front-end agnostic versatile lidar slam system," in *2022 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE, 2022, pp. 1–6.