EPN-NetVLAD: SE(3)-Invariant Place Recognition for 3D Point Clouds

Anonymous Author(s)
Affiliation
Address
email

Abstract: This paper reports on a new 3D point cloud-based place recognition framework using SE(3)-invariant global descriptors. We discover that, unlike existing methods, learned SE(3)-invariant global descriptors are more robust to matching inaccuracy and failure in severe rotation and translation configurations. Mobile robots undergo arbitrary rotational and translational movements. The SE(3)-invariant property ensures the learned descriptors are robust to the rotation and translation changes in the robot pose and can represent the intrinsic geometric information of the scene. Furthermore, we have discovered that the attention module aids in the enhancement of performance while allowing significant downsampling. We evaluate the performance of the proposed framework on real-world data sets. The experimental results show that the proposed framework outperforms state-of-the-art baselines in various metrics, leading to a reliable point cloud-based place recognition network.

Keywords: Place Recognition, SE(3)-Invariant, Equivariant Representation Learning, 3D Point Clouds

1 Introduction

Place recognition can be defined as linking the sensor's in-situ observations and the prebuilt reference map. Among numerous 2D (RGB, thermal, and event-triggered) and 3D (stereo, LiDAR, and RGB-D) sensors [1], 3D sensors are gaining popularity and have recently been extensively researched. Modern service robots, autonomous cars [2], and drones [3] are widely equipped with consumer-level 3D sensors due to their better environment perception ability and decreasing prices. Thus, place recognition techniques with 3D data can be used in estimating the agent's location in scenarios such as self-driving vehicles, autonomous indoor navigation, or scientific exploration. Additionally, place recognition, also known as loop closure detection, is a critical component in Simultaneous Localization and Mapping (SLAM). It enables a robot to determine if it has seen a place before and provides loop closure candidates [4]. With a correct loop closure, the SLAM system is able to eliminate accumulated drift from the odometry and improve the mapping accuracy [5].

Extracting consistent features from 3D data is an important research topic but remains underexplored and unsolved [6]. Although many pieces of research have been conducted on visual place recognition [7, 8], place recognition methods for 3D point clouds are limited. 3D point clouds encode geometry information and can overcome illusion and appearance-changing challenges that image-based methods are facing [9]. However, point clouds are sparse, unordered, and change drastically when the sensor is in different locations or orientations. These limitations hinder extracting consistent features from point clouds.

One key issue in present place recognition methods is that they do not consider transformation changes in data or expect robustness via simple data augmentation [10]. Thus, their performance is sensitive to transformation diversity in the training and testing point clouds samples. The extracted global descriptors change substantially when the point clouds are rotated or translated, resulting place recognition failure as descriptors matched incorrectly. To solve the transformation-sensitive global descriptors problem in previous works, this research aims at designing a rotation and

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

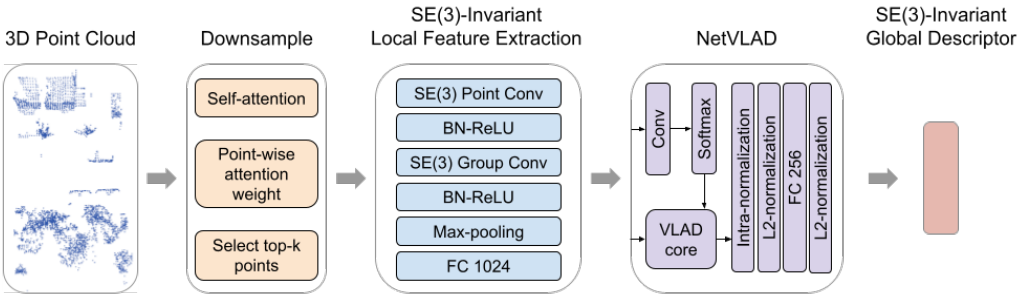


Figure 1: Overview of the proposed SE(3)-invariant place recognition network. In the first stage, point clouds are downsampled using self-attention weights. SE(3)-invariant local feature extraction consists of SE(3) point convolution, SE(3) group convolution, batch normalization followed by leaky ReLU activation, and pooling layer which makes equivariant features invariant. Global descriptors are computed by aggregating local features using NetVLAD. The output global descriptors are SE(3)-invariant and can perform place recognition tasks.

translation-invariant global descriptor for point clouds which we formally define as SE(3)-invariant feature.

The attention mechanism of transformers [11] enables networks to learn the correlation between input features and attain the importance of each feature. Some place recognition frameworks like SOE-Net [12] include this technique to utilize the neighborhood context. However, they usually apply an attention mechanism to local features instead of directly on the 3D points. As a result, only the correlations between processed features are learned but not between 3D points.

In this paper, we propose a place recognition framework that utilizes SE(3)-invariant features to perform place recognition in challenging rotation and translation scenes. We first downsample point clouds using the self-attention mechanism. The downsampled point clouds are then fed into Equivariant Point Network (EPN) [13] to learn SE(3)-invariant local features. Finally, SE(3)-invariant global descriptors are learned by aggregating local features using NetVLAD [14]. We trained our network (see Figure 1) on Oxford RobotCar Dataset [15] and evaluate on Oxford and in-house benchmark [16]. We also validate our proposed framework for rotation and translation scenes. Experimental results show that our approach consistently outperforms existing state-of-the-art approaches. The experiment of the trained network on unseen data sets verifies the generalizability and scalability of our EPN-NetVLAD.

The main contributions of this work can be summarized as follows:

1. We propose EPN-NetVLAD, a learning-based place recognition algorithm that is the first to provide SE(3)-invariant descriptors using only geometric information from 3D point clouds.
2. The proposed method is robust against arbitrary rotation and translation of robot poses. It is generalizable and scalable to unseen data sets.
3. We apply a self-attention mechanism to downsample point clouds which maintains place recognition in high performance up to 50 % downsampling rate.
4. The code is open-sourced and will be publicly available after receiving the final decision.

2 Related Work

3D point clouds generated by LiDAR (light detection and ranging), stereo cameras, RGB-D cameras, or other sensors obtain rich and accurate environmental 3D geometric information. We first review place recognition works that utilize the geometric information from 3D point clouds in section 2.1. Then, we present existing works that utilize the point cloud descriptors with rotation-invariant or translation-invariant properties in section 2.2. In section 2.3, we provide brief introduction on designing attention mechanisms to improve place recognition performance.

2.1 Geometry-based Place Recognition

Previously, place recognition using point clouds relied on histograms or hand-engineered features such as M2DP [17] and Scan Context [18]. Fast Histogram [19] uses the similarity between LiDAR histograms to determine revisited places. M2DP [17] projects 3D point clouds to multiple 2D planes and constructs global descriptors from singular vectors of density signatures. Scan Context [18] represents the point cloud in the polar axis and encodes the height of the observed points into the representation.

PointNetVLAD [16] is a pioneering work to apply a learning-based feature extractor to place recognition tasks. It combines PointNet [20] and NetVLAD [14] to allow end-to-end representation training from a given 3D point cloud. OverlapNet [21] constructs range images to learn the overlap score and the yaw angle between two inputs. MinkLoc3D [22] presents sparse voxelized point cloud representation and sparse 3D convolutions. LPD-Net [23] proposes adaptive local feature extraction and graph-based neighborhood aggregation to construct a global descriptor. LCDNet [24] provides pose-estimation in addition to representation learning in the network. However, these algorithms are not robust to rotational and translational pose changes.

2.2 Exploiting Symmetry in Place Recognition

To perform place recognition more robustly and accurately, it is critical to consider that the observer may be in different orientations or locations. To address this challenge, researchers propose some hand-crafted features that are rotation-invariant to improve the performance of place recognition. Yin et al. [25] propose a heading-invariant feature that uses histograms of range in a LiDAR scan ring to deal with the change of heading angle of the vehicle. Scan Context [18] uses ring keys, the occupancy ratio of rings in scan context, as rotation-invariant features. It is further generalized in their later work Scan Context++ [26] to include lateral invariance by augmentation based on urban road assumption. LiDAR-Iris [27] encodes height information into eight-bit binary code and uses Fourier transform to estimate the translation between two LiDAR-Iris images to remove the rotational difference between LiDAR scans. Xu et al. [28] build polar grid height coding image descriptor which is rotationally invariant. While these hand-crafted features are rotation-invariant, some structural information is ignored when composing them. Later, deep learning features are widely used since their performances surpass hand-crafted features [29].

There are only a few works that try to encode rotation-invariant or translation-invariant features into learning-based place recognition algorithms. PointNetVLAD [16] and LCDNet [24] try to increase robustness by randomly rotating input point clouds during training. RINet [10] exploits additional semantic information and combines it with rotation equivariant convolution to achieve rotation-invariant. OverlapNet [21] and OverlapTransformer [30] use range images to make the feature yaw-angle-invariant. Lu et al. [31] propose RING descriptor that is translation-invariant after discrete Fourier transform procedure and is yaw-angle-invariant. Nevertheless, these strategies are not efficient in more challenging scenarios.

While only a small number of works take rotation-equivariant and translation-equivariant into consideration in place recognition tasks, there are a series of works that design network architectures with the equivariance property for general feature learning. Esteves et al. [32] propose Spherical CNNs (Convolutional Neural Networks) which map 3D models into spherical functions and use spherical convolutions to generate equivariant features maps. Vector Neuron [33] proposes a SO(3)-equivariant network that replaces scalars with 3-vectors in the neurons. Equivariant Point Network (EPN) [13] performs SE(3) separable convolution which separates 6D convolution into convolutions in the 3D Euclidean space and in SO(3). It enables SE(3)-equivariant feature learning in a computational affordable way, which we adopt in this work. In general, these networks address the rotation-equivariant feature learning problem in classification and segmentation tasks. We are the first to apply SE(3)-invariant feature learning (EPN in this work) to place recognition tasks.

2.3 Attention Mechanism in Place Recognition

Attention mechanism has been applied to some place recognition tasks to better utilize the neighborhood context. PCAN [34] predicts the significance of each local feature using an attention mechanism. SOE-Net [12] encodes a self-attention structure that explores the relationship between points. Retriever [35] builds an attention mechanism between local features and a latent code to construct

global descriptors. OverlapTransformer [30] includes Transformer to learn spatial relations of different features before feeding into NetVLAD. Among these applications, attention mechanisms are used to learn the importance of the local features. In this work, we explore applying the attention mechanism to the input 3D points.

3 Methodology

In this section, we detail our proposed EPN-NetVLAD framework for SE(3)-invariant place recognition using 3D point clouds. An overview of the proposed approach is depicted in Figure 1. The framework consists of three parts: attentive downsampling, SE(3)-invariant local feature extraction, and SE(3)-invariant global descriptor generation. We will fully discuss each component in the following subsections.

3.1 Attentive Downsampling

3D point clouds measured from LiDAR or RGB-D sensors may contain hundreds of thousands of points. To perform place recognition efficiently in neural networks, we exploit the attention mechanism to downsample point cloud measurements while preserving meaningful information. For a point cloud with N points $P \in \mathbb{R}^{N\times 3}$, we apply the multi-head attention module [11] using the PyTorch library [36] to learn the attention weights $W_{atten} \in \mathbb{R}^{N\times 3}$ from the input point cloud. Multi-head attention is defined as $\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, head_2, head_3)W^O$, where $\text{Concat}(\cdot)$ does the features concatenation. The number of parallel attention heads is set as 3. Each attention head is defined as $head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$. Here, we set query Q, key K, and value V as the input point cloud P, $Q = K = V = P$, to perform self-attention and learn correspondences between the input 3D points.

With the attention weight, we summarize over the feature space to obtain point-wise attention weights $W_{pw-atten} \in \mathbb{R}^N$, which represent the significance of each point. $W_{pw-atten} = \sum_{i=1,2,3} W^i_{atten}$, where $W^i_{atten} \in \mathbb{R}^N$ is the attention weight in dimension i. We select top-k attention weights and keep the corresponding points $P' = \mathbb{R}^{k\times 3}$.

3.2 Local SE(3)-Invariant Features based on Modified EPN

Learning equivariant representation from point clouds can provide efficiency and generalizability in challenging robot perception tasks. *Equivariance* is a form of symmetry for functions that the output's change of transformation is determined by the transformation of the input. Mathematically, a function $f_e : X \to X$ is *equivariant* to a set of transformations G, if for any $g \in G$, $g \cdot f_e(x) = f_e(g \cdot x), \forall x \in X$.

Equivariance generalizes the concept of *invariance* which means that the output of functions is independent of the transformations applied onto the input. Mathematically, a function $f_{inv} : X \to X$ is *invariant* to a set of transformations G, if for any $g \in G$, $f_{inv}(x) = f_{inv}(g \cdot x), \forall x \in X$.

In this work, we leverage and modify the Equivariant Point Network (EPN) [13] to learn the SE(3)-equivariant feature and capture the inherent symmetry of 3D point cloud data. In the original EPN [13], given a 3D point x, a rotation g, a feature representation function $\mathcal{F} : \mathbb{R}^3 \times \text{SO}(3) \to \mathbb{R}^D$, and a kernel $h : \mathbb{R}^3 \times \text{SO}(3) \to \mathbb{R}^D$, the discretized SE(3)-equivariant convolutional operator is defined as the dot product between the translated and rotated kernel and the function $\mathcal{F}$:

$$(\mathcal{F} * h)(x, g) = \sum_{x_i \in \mathcal{P}} \sum_{g_j \in G} \mathcal{F}(x_i, g_j) h(g^{-1}(x - x_i), g_j^{-1} g) \tag{1}$$

where P and G are the discretized sets corresponding to $\mathbb{R}^3$ and $\text{SO}(3)$ respectively. To reduce the computation cost in 6D convolution, the authors separate the kernel h into two smaller kernels h_1, h_2 that represent SE(3) point convolution and SE(3) group convolution respectively.

$$(\mathcal{F} * h_1)(x, g) = \sum_{x'_i \in \mathcal{P}} \mathcal{F}(x - x'_i, g) h_1(g x'_i, \mathbf{I}) \tag{2}$$

$$(\mathcal{F} * h_2)(x, g) = \sum_{g_j \in G} \mathcal{F}(x, g^{-1} g) h_2(0, g_j) \tag{3}$$

This design preserves SE(3)-equivariant features from the input point cloud. Global pooling is then applied to extract SE(3)-invariant features. Different from the group attentive pooling in EPN [13], we directly apply a max-pooling in the rotational dimension for each spatial point to generate SE(3)-invariant features. The last part of the local feature extractor is a linear layer to map the SE(3)-invariant features to the desired dimension. See Figure 1 for an illustration.

3.3 Global SE(3)-Invariant Place Representation

Global descriptors are computed by aggregating local features using NetVLAD. NetVLAD learns cluster centers of VLAD (Vector of Locally Aggregated Descriptors) in a CNN framework. The output descriptors V are adopted for describing the places and are given in (4). This equation shows j-th dimensions of the i-th descriptor, where x is the local feature. w_k, b_k, and c_k are trainable parameters to learn the center of cluster k.

$$V(j,k) = \sum_{i=1}^{N} \frac{\exp^{w_k^T x_i + b_k}}{\sum_{k'} \exp^{w_k^T x_i + b_{k'}}} (x_i(j) - c_k(j)) \tag{4}$$

To learn discriminative and generalizable global descriptors for performing place recognition task, we use lazy quadruplet loss proposed by Uy and Lee [16]. For each iteration of training, there are an anchor point cloud P_a, a "positive" point cloud P_{pos} that is similar to the anchor point cloud, some "negative" point clouds $\{P_{neg}\}$ that are dissimilar to the anchor point cloud, and a random point cloud in the training set P_{neg^*}. The lazy quadruplet loss defined in (5) can minimize the L1 distance between anchor and positive representation $\delta_{pos} = d(f(P_a), f(P_{pos}))$ while maximizing the distance between anchor and some negative representation $\delta_{neg_j} = d(f(P_a), f(P_{neg_j})), P_{neg_j} \in \{P_{neg}\}$. α and β are constant parameters giving the margin.

$$L_{LazyQuad}(P_a, P_{pos}, P_{neg}, P_{neg^*}) = \max_{\mathbf{j}}([\alpha + \delta_{pos}\delta_{neg_j}]_+) + \max_{\mathbf{k}}([\beta + \delta_{pos}\delta_{neg_k^*}]_+) \tag{5}$$

4 Experimental Results and Discussion

We construct SE(3)-invariant place recognition descriptors using the described method. In this section, we examine the design of attentive downsampling, SE(3)-invariant properties, and the performance of place recognition.

4.1 Data Set

We train our EPN-NetVLAD algorithm on Oxford RobotCar [15] benchmark created by Uy and Lee [16]. Oxford benchmark contains 45 sequences of a vehicle taking measurements using SICK LMS-151 2D LiDAR in similar routes for different times, days, and seasons. Each point cloud is a submap of a pre-built map. The points of the ground are removed and the point clouds are normalized to be zero mean and inside the range of [-1, 1]. Training and testing sets are geographically split with the ratio of 70 % and 30 %. For creating training tuples, a ground truth location within 10 m is considered a positive pair, while a location larger than 50 m is considered a negative sample. We train with 21,711 sub-maps.

In addition to the Oxford benchmark, the proposed method is evaluated on in-house data sets. In-house data sets are generated by Uy and Lee [16] and are constructed from Velodyne-64 LiDAR scans. It includes three different regions: university sector (U.S.), residential area (R.A.), and business district (B.D.).

We trained and tested our method on a system equipped with Intel i9-10900K CPU with a 3.7 GHz processor and an Nvidia GeForce RTX 3090. Point clouds are downsampled to 2048 points using the attention mechanism. We construct EPN-NetVLAD with two layers of EPN, one with 32 local features, and one with 64 local features. In EPN, we set the number of discrete rotation groups as 60. EPN is followed with max-pooling and a linear layer to map local features to 1024 dimensions. Then, we use NetVLAD to learn global descriptors with dimensions of 256. The network is trained for 30 epochs with a learning rate of 5×10^{-5}. Each training tuple consists of one query point

Table 1: Experiment result showing the average recall (%) at top 1% and at top 1 for each of the models on Oxford benchmark.

	Average Recall (%) at top 1%	Average Recall (%) at top 1
EPN-NetVLAD (Ours)	**89.17 %**	**77.69** %
PointNetVLAD [16]	84.94 %	71.39 %
Scan Context [18]	32.91 %	22.89 %
M2DP [17]	34.69 %	23.14 %

Table 2: Generalizability experiment result showing the average recall (%) at top 1% for each of the models. Both methods are only trained on Oxford and tested on U.S., R.A., and B.D. data sets.

Datasets	Oxford	U.S.	R.A.	B.D.
EPN-NetVLAD (Ours)	**89.17 %**	**87.82 %**	**81.98 %**	**76.91 %**
PointNetVLAD [16]	84.94 %	80.79 %	73.86 %	69.29 %

cloud, one "positive" point cloud, one "negative" point cloud, and another random point cloud. The hyper-parameters in lazy quadruplet loss in set as $\alpha = 0.5, \beta = 0.2$.

4.2 Place Recognition Evaluation

We first evaluate the performance of the proposed method on the Oxford benchmark. In place recognition tasks, precision and recall are the two well-established evaluation metric [7]. Precision is the percentage of true loop closures among all the places we recognize. Recall is the percentage of places we recognize among all true loop closures. The definition is shown in (6), where TP is the number of true-positive cases, FP represents the number of false-positive cases, and FN stands for the number of false-positive cases. The F1 score is introduced and defined in the same equation to obtain a balancing metric between precision and recall.

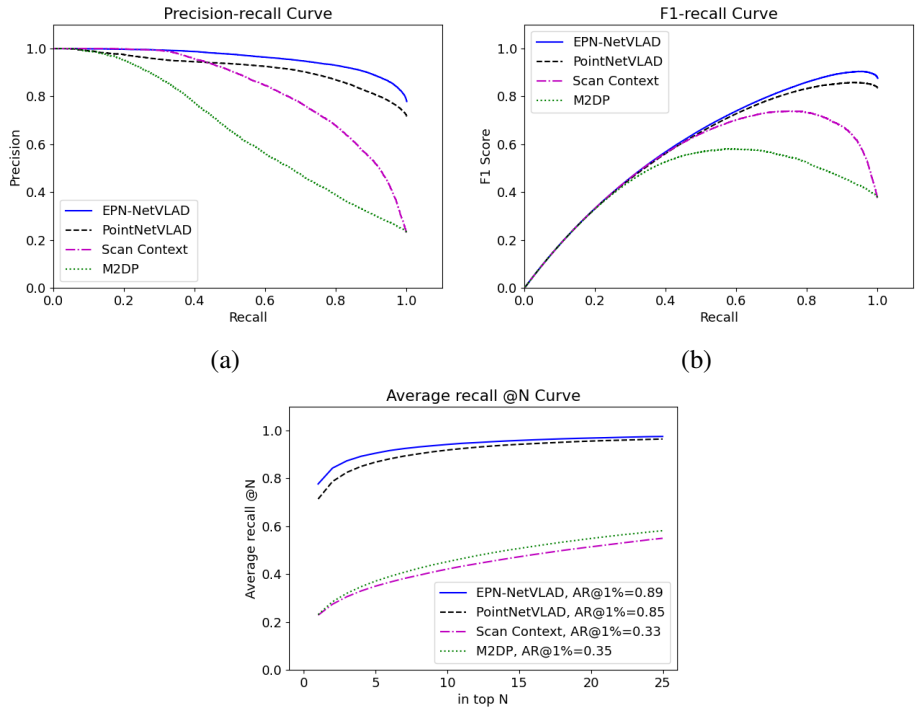
$$\text{precision} = \frac{TP}{TP + FP};\ \text{recall} = \frac{TP}{TP + FN};\ \text{F1} = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

When performing place recognition, we could select N candidates that are similar to the current query input. The average recall curve represents the model performance for the top 25 matches. With these evaluation metrics, precision-recall curves, F1-recall curves, and average recall curve of the proposed method and other state-of-the-art methods are shown in Figure 2. Both EPN-NetVLAD (ours) and PointNetVLAD [16] are trained on the same Oxford benchmark training set. Where Scan Context [18], and M2DP [17] construct hand-engineered features to perform place recognition. From the figure, the proposed network EPN-NetVLAD outperforms other baselines in all metrics. Table 1 shows the average recall at top 1% and at top 1. Regardless of selecting several or only one loop closure candidate, our method consistently performs better than others.

To show the generalizability of the proposed method, we evaluate EPN-NetVLAD on in-house data sets with three kinds of regions that are unseen to the network, including the university sector (U.S.), residential area (R.A.), and business district (B.D.). The result is shown in Table 2 and our method achieves the best performance among all the data sets that we did not train on.

4.3 Experiment with SE(3) transformation

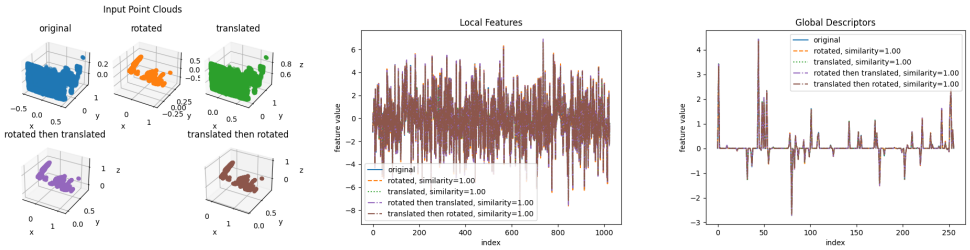
In addition to place recognition experiments on Oxford and in-house benchmark, we construct simulated data to test the model performance with severe rotation and translation. First, we visualize the local features and global descriptors when the input point cloud is transformed under rotation, translation, rotated then translated, and translated then rotated. Figure 3 shows the results and the cosine similarity score between each transformed feature/descriptor and the original feature/descriptor. We can see that even if the point cloud is rotated or translated, the output features and descriptors remain the same and has 100 % similarity to the original one.



(a) (b)

(c)

Figure 2: Experimental results of proposed method (EPN-NetVLAD, blue line), state-of-the-art approaches PointNetVLAD [16], Scan Context [18], and M2DP [17] on Oxford benchmark.



(a) point clouds under transformations

(b) SE(3)-invariant local features

(c) SE(3)-invariant global features

Figure 3: Visualization of the input point clouds, local features, and global descriptors under different transformation.

Furthermore, we construct a simulated data set to include place recognition examples of different transformations. It contains the point clouds that are transformed under purely SO(3)-rotation, purely 3D translation, and with both rotation and translation. With original point clouds in a range between [-1, 1], 3D rotations are applied randomly, and 3D translations are applied with a standard deviation of 1.0. We use 440 point clouds where each of them has two positive pairs. We then use the model trained on the Oxford benchmark to perform place recognition on this simulated data set. The result is shown in Table 3, the proposed network performs significantly better in severe transformation.

4.4 Attentive Downsampling

We design an experiment to test the performance of the downsampling point cloud using an attention mechanism. Following the place recognition task experiment, we study the performance of the proposed network with random downsampling and attentive downsampling method. In this experiment,

Table 3: Experiment result reports average recall at top 1% for performing place recognition task on different scenes where the point clouds are transformed under rotation or/and translation.

Rotation	Translation	PointNetVLAD	EPN-NetVLAD (Ours)
✓		6.60 %	**98.74** %
	✓	3.21 %	**100.00** %
✓	✓	2.96 %	**99.43** %

Table 4: Experiment result showing the average recall (%) at top 1% when the input point cloud is downsampled with different percentage and different method. This table compares the result of random downsampling and attentive downsampling which utilize attention mechanism to downsample.

Number of Points	Downsampling Rate	Random Downsampling	Attentive Downsampling
4096	0 %	71.66 %	71.66 %
3000	27 %	63.34 %	**71.65** %
2048	50 %	57.29 %	**71.05** %
1600	61 %	53.19 %	**66.22** %
1024	75 %	43.17 %	**57.97** %

the network is constructed with only one layer of EPN with 64 local features to simplify the task. The result of different downsampling rates is presented in Table 4. It shows that using an attention mechanism to downsample point clouds can maintain high place recognition performance up to 50 % downsampling rate.

4.5 Runtime Performance

The inference time for each point cloud with 4096 points is 2.05 seconds. The default setup of the network uses 60 discrete rotation groups to compute rotation-equivariant features. We also tested our framework with only 20 discrete rotation groups, and the network can run for 0.53 seconds per point cloud.

5 Limitation

The major limitation of the proposed framework is the relatively slow runtime due to the large computational loads, which is a common drawback of equivariant neural networks [13]. However, with recent progress on more efficient equivariant network designs [37] and the development of more powerful computing hardware, we expect this limitation to be largely resolved in the near future.

6 Conclusion

We have designed a place recognition framework that exploits SE(3)-equivariant representation learning. In particular, SE(3)-invariant features learned from 3D point clouds improve robustness to large transformations and generalizability in place recognition tasks. In addition, we propose using an attention mechanism in place recognition to downsample the input point cloud while maintaining high performance. Our experimental results on real-world data sets show the proposed method performs well in various metrics. Future work includes a light-weight design of the equivariant encoder for real-time onboard applications and the extension of this framework to stereo cameras where image data can also be incorporated into the learned representation.

References

[1] T. Barros, R. Pereira, L. Garrote, C. Premebida, and U. J. Nunes. Place recognition survey: An update on deep learning approaches. *arXiv preprint arXiv:2106.10458*, 2021.

[2] C. Häne, L. Heng, G. H. Lee, F. Fraundorfer, P. Furgale, T. Sattler, and M. Pollefeys. 3D visual perception for self-driving cars using a multi-camera system: Calibration, mapping, localization, and obstacle detection. *Image and Vision Computing*, 68:14–27, 2017.

[3] M. Sanfourche, V. Vittori, and G. Le Besnerais. eVO: A realtime embedded stereo odometry for MAV applications. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2107–2114. IEEE, 2013.

[4] H. Johannsson, M. Kaess, M. Fallon, and J. J. Leonard. Temporally scalable visual SLAM using a reduced pose graph. In *2013 IEEE International Conference on Robotics and Automation*, pages 54–61. IEEE, 2013.

[5] 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*, 32(6):1309–1332, 2016.

[6] X. Zhang, L. Wang, and Y. Su. Visual place recognition: A survey from deep learning perspective. *Pattern Recognition*, 113:107760, 2021.

[7] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2015.

[8] Z. Zeng, J. Zhang, X. Wang, Y. Chen, and C. Zhu. Place recognition: An overview of vision perspective. *Applied Sciences*, 8(11):2257, 2018.

[9] K. Żywanowski, A. Banaszczyk, and M. R. Nowicki. Comparison of camera-based and 3D lidar-based place recognition across weather conditions. In *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 886–891. IEEE, 2020.

[10] L. Li, X. Kong, X. Zhao, T. Huang, W. Li, F. Wen, H. Zhang, and Y. Liu. RINet: Efficient 3D lidar-based place recognition using rotation invariant neural network. *IEEE Robotics and Automation Letters*, 7(2): 4321–4328, 2022.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[12] Y. Xia, Y. Xu, S. Li, R. Wang, J. Du, D. Cremers, and U. Stilla. SOE-Net: A self-attention and orientation encoding network for point cloud based place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11348–11357, 2021.

[13] H. Chen, S. Liu, W. Chen, H. Li, and R. Hill. Equivariant point network for 3D point cloud analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14514–14523, 2021.

[14] 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*, pages 5297–5307, 2016.

[15] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 year, 1000 km: The Oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[16] 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*, pages 4470–4479, 2018.

[17] L. He, X. Wang, and H. Zhang. M2dp: A novel 3D point cloud descriptor and its application in loop closure detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 231–237. IEEE, 2016.

[18] G. Kim and A. Kim. Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4802–4809. IEEE, 2018.

[19] T. Röhling, J. Mack, and D. Schulz. A fast histogram-based similarity measure for detecting loop closures in 3-D lidar data. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 736–741. IEEE, 2015.

[20] 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*, pages 652–660, 2017.

[21] X. Chen, T. Läbe, A. Milioto, T. Röhling, O. Vysotska, A. Haag, J. Behley, C. Stachniss, and F. Fraunhofer. OverlapNet: Loop closing for LiDAR-based SLAM. In *Proceedings of the Robotics: Science and Systems Conference*, 2020.

[22] J. Komorowski. Minkloc3d: Point cloud based large-scale place recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1790–1799, 2021.

[23] 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*, pages 2831–2840, 2019.

[24] D. Cattaneo, M. Vaghi, and A. Valada. LCDNet: Deep loop closure detection for LiDAR SLAM based on unbalanced optimal transport. *arXiv preprint arXiv:2103.05056*, 2021.

[25] H. Yin, Y. Wang, X. Ding, L. Tang, S. Huang, and R. Xiong. 3D lidar-based global localization using siamese neural network. *IEEE Transactions on Intelligent Transportation Systems*, 21(4):1380–1392, 2019.

[26] G. Kim, S. Choi, and A. Kim. Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments. *IEEE Transactions on Robotics*, 2021.

[27] Y. Wang, Z. Sun, C.-Z. Xu, S. Sarma, J. Yang, and H. Kong. Lidar iris for loop-closure detection. *arXiv preprint arXiv:1912.03825*, 2019.

[28] D. Xu, J. Liu, J. Hyyppä, Y. Liang, and W. Tao. A heterogeneous 3D map-based place recognition solution using virtual lidar and a polar grid height coding image descriptor. *ISPRS Journal of Photogrammetry and Remote Sensing*, 183:1–18, 2022.

[29] C. Masone and B. Caputo. A survey on deep visual place recognition. *IEEE Access*, 9:19516–19547, 2021.

[30] 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*, 2022.

[31] S. Lu, X. Xu, H. Yin, R. Xiong, and Y. Wang. One ring to rule them all: Radon sinogram for place recognition, orientation and translation estimation. *arXiv preprint arXiv:2204.07992*, 2022.

[32] 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*, pages 52–68, 2018.

[33] C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. Guibas. Vector neurons: A general framework for SO(3)-equivariant networks. *Proceedings of the IEEE International Conference on Computer Vision*, 2021.

[34] W. Zhang and C. Xiao. Pcan: 3D attention map learning using contextual information for point cloud based retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12436–12445, 2019.

[35] L. Wiesmann, R. Marcuzzi, C. Stachniss, and J. Behley. Retriever: Point cloud retrieval in compressed 3D maps. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[36] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[37] M. Zhu, M. Ghaffari, W. A. Clark, and H. Peng. E^2PN: Efficient SE(3)-equivariant point network. *arXiv preprint arXiv:2206.05398*, 2022.