SelFLoc: Selective Feature Fusion for Large-scale Point Cloud-based Place Recognition

Qibo Qiu[a,b], Wenxiao Wang[a], Haochao Ying[a], Dingkun Liang[b], Haiming Gao[b,*], Xiaofei He[a]

[a]*Zhejiang University*
[b]*Zhejiang Lab*

Abstract

Point cloud-based place recognition is crucial for mobile robots and autonomous vehicles, especially when the global positioning sensor is not accessible. LiDAR points are scattered on the surface of objects and buildings, which have strong shape priors along different axes. To enhance message passing along particular axes, Stacked Asymmetric Convolution Block (SACB) is designed, which is one of the main contributions in this paper. Comprehensive experiments demonstrate that asymmetric convolution and its corresponding strategies employed by SACB can contribute to the more effective representation of point cloud feature. On this basis, Selective Feature Fusion Block (SFFB), which is formed by stacking point- and channel-wise gating layers in a predefined sequence, is proposed to selectively boost salient local features in certain key regions, as well as to align the features before fusion phase. SACBs and SFFBs are combined to construct a robust and accurate architecture for point cloud-based place recognition, which is termed SelFLoc. Comparative experimental results show that SelFLoc achieves the state-of-the-art (SOTA) performance on the Oxford and other three in-house benchmarks with an improvement of 1.6 absolute percentages on mean average recall@1.

Keywords: Autonomous vehicle, Localization, Place recognition, Asymmetric convolution, Feature fusion

*Corresponding author
Email addresses: qiuqibo_zju@zju.edu.cn (Qibo Qiu), ghm@mail.nankai.edu.cn (Haiming Gao)

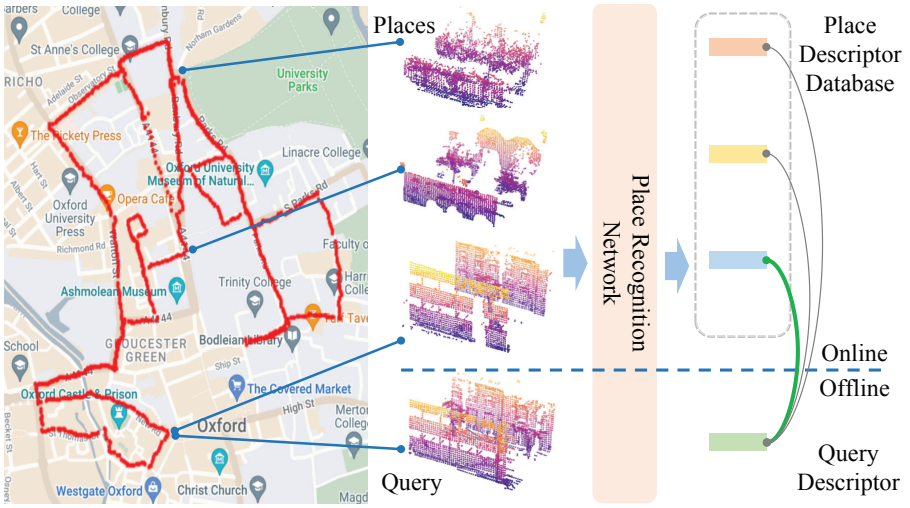


Figure 1: Point cloud-based place recognition in large-scale urban environments. The place recognition network extracts global descriptors from point clouds in different locations, which are subsequently stored in a database. Once the vehicle reaches a new location, the closest match (**green**) can be retrieved if the distance between the queried and recorded global descriptor is the shortest.

1. Introduction

Due to the frequent absence of global positioning signals, localization based on environmental perception is becoming increasingly vital in the navigation system for both mobile robots and autonomous vehicles [1, 2, 3]. Place recognition attempts to determine whether a place has been visited, which contributes to localization in GPS-denied environments. According to the type of sensor input, place recognition can be splited into image-based methods [4, 5, 6] and point cloud-based methods. Point cloud-based methods are more robust to the lighting and seasonal changes [7] when compared with image-based ones. In the remainder of this paper, *place recognition* refers to LiDAR point cloud-based place recognition. As shown in Fig. 1, the offline stage of place recognition collects the global descriptors of previously visited places, which forms the place descriptor database. The online stage extracts the global descriptor of current query LiDAR scan and ranks the LiDAR scans of previously visited places based on the Euclidean distances between the query descriptor and the descriptors in the database.

With the development of deep learning, the representation of global descriptor for place recognition has progressed greatly. PointNetVLAD [7] first creates the place recognition benchmarks and proposes a framework based on PointNet [8] and NetVLAD [9]. Different from images, point clouds are inherently unordered and irregular [10], which makes it difficult for PointNet-based models to capture the spatial relationships among the local neighboring points [11]. To this end, k-nearest neighbor (k-NN) [12, 13, 14] and sparse voxelization-based 3D convolution [15, 16, 17] are mostly utilized to capture the locality and hierarchy. Furthermore, transformer [18, 19] is also employed for local feature enhancement [13, 20, 16, 17]. Nevertheless, aforementioned methods are proposed based on symmetrical sampling, neglecting the fact that point clouds for various objects disperse differently along each axis.

In this paper, informative features are extracted by enhancing message passing along particular axes. Specifically, the Stacked Asymmetric Convolution Block (SACB) is designed, where multiple 1D asymmetric convolutions are stacked to extract local features, and the convolution numbers and strategies can vary with each axis to strengthen feature aggregation along particular axes. For example, additional convolution along x-axis can be performed to strengthen the feature in the driving direction, and dilation strategy can be further applied to expand the effective receptive field. In addition, the number of parameters can be reduced drastically compared with the traditional 3D convolution.

On the other hand, to fuse multi-scale local features, Minkloc3D [21] is the first place recognition method that built on Feature Pyramid Network (FPN) to fuse multi-scale local features. The success of Minkloc3D-based models comes at fully use of the locality and hierarchy of 3D convolution, and shows that reliable place recognition is more dependent on local features. Salient local features locate at certain key regions, and feature semantics (channels) should be aligned before the fusion phase. Different from simple addition or concatenation, this paper argues that multi-scale features should be fused selectively, in which way local features can be boosted. Specifically, Selective Feature Fusion Block (SFFB) is introduced by stacking point- and channel-wise context gating layers to reweight the local features. Note that SFFB can be integrated as a plugin before any feature fusion phase. The contributions of this paper are

threefold:

- This paper proposes SACB to leverage the strong shape priors of the point cloud, which is stacked by 1D asymmetric convolutions equipped with different strategies. In addition, it reduces the parameters, which contributes to deployment.
- SFFB block is introduced to fuse multi-scale features selectively, according to the point- and channel-wise context. Ablation experiments show that SFFB is beneficial to feature semantic alignment and key region enhancement, which further contributes to accurate global descriptor matching.
- Comprehensive experiments show that both SACB and SFFB are effective for place recognition, supported by superior performance on the Oxford and three in-house datasets.

2. Related Work

2.1. Point Cloud-based Place Recognition

Traditional point cloud-based place recognition methods are depend on hand-crafted features [22, 23, 24, 25], which are well-designed to produce a discriminative global descriptor. Recently, the representation of discriminative feature is remarkably enhanced by deep learning methods. PointNetVLAD is the pioneering deep learning-based method proposed for place recognition, where the local features are extracted by PointNet. Different from images, it is difficult for PointNet-based models to capture the local response, since point clouds are inherently unordered and irregular [10]. With the help of predefined local geometric features, LPD-Net [12] can enhance local features by a graph-based aggregation operation. While NDT-Transformer [26] transforms the point cloud into Normal Distribution Transform (NDT) cells, thus point-wise features can be boosted, and it is also the first one to make use of transformer for globality capturing. Furthermore, PPT-Net [13] designs a pyramid point cloud transformer to capture globality spatially on different clustering granularities.

Different from the aforementioned methods, 3D convolution can also be employed to extract local features. In particular, Minkloc3D [21] and its inherited versions are

among the most successful 3D convolution-based place recognition models, which are built on the ResNet and FPN architectures. However, the proposed SelFLoc decomposes a 3D convolution into 1D convolutions to take advantage of the strong shape priors in point clouds, in addition to introducing novel attention mechanisms for selective feature fusion.

2.2. Asymmetric Convolution

Asymmetric convolution is originally designed to improve parameter efficiency. [27] argues that any $n \times n$ kernel can be replaced by a $1 \times n$ asymmetric convolution followed by a $n \times 1$ convolution, and the computation cost can be greatly reduced. The hypothesis that a 2D kernel with a rank of one equals a sequence of 1D convolutions supports asymmetric convolution, while ranks of a 2D kernel cannot be guaranteed to be one. To this end, [28] represents a 2D kernel (matrix) of rank k as the outer product of a sequence of 1D convolutions (vectors). Supported by this low-rank approximation, the non-bottleneck module in ResNet [29] is redesigned by ERFNet [30], and this factorization considerably decreases the kernel size and enables real-time operation. ACNet [31] provides a novel application of asymmetric convolution, which leverages 1D convolution to enhance the model robustness to rotational distortions. The SACB in proposed SelFLoc is not only intended for feature enhancement along particular axes and parameter reduction, but also equipping the asymmetric convolutions with different strategies.

2.3. Attention and Context Gating Mechanisms

Context gating as a reweighting operation designed to enhance the more informative features. There are both spatial and channel-wise attention mechanisms in image convolution networks [32, 33, 34]. Likewise, point- and channel-wise attention mechanisms are both utilized in point feature aggregation [35]. In particular, PCAN [36] presents a 3D point-wise attention map for place recognition and retrieval of point clouds, which is inspired by CRN [34]. On the other hand, the ECA [32] module is introduced to place recognition by Minkloc3Dv2 [15] and TransLoc3D [17], where the global information is aggregated by channel-wise attention.

Recently, transformer-based attention architecture has become popular in both natural language processing and computer vision. In addition, there are attempts to surpass the dominance of CNN and transformer by using MLP-like architecture [37, 10]. PointNet [8] is a pioneering trial in MLP-like architecture for point cloud. While PointMixer [10] initially utilizes MLP-Mixer [37] for point cloud understanding. In this paper, point- and channel-wise gating layers are employed for semantic alignment and key region enhancement, which further contribute to reliable place recognition.

3. Method

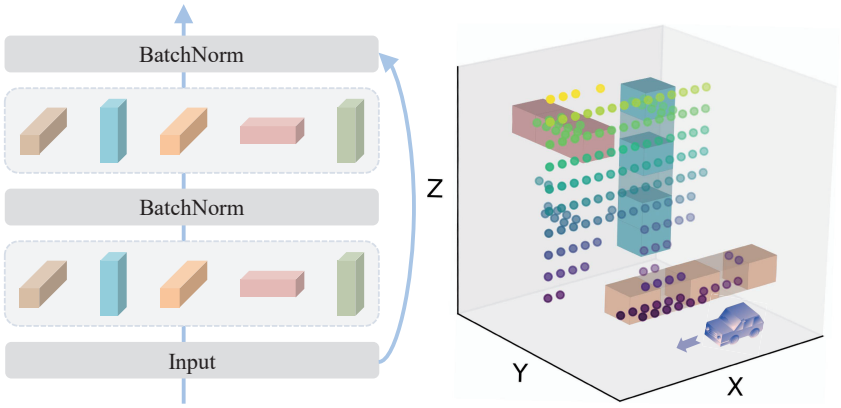
3.1. Asymmetric Convolution

Asymmetric convolution for point cloud. Initially, asymmetric convolution is created to reduce the computation cost and model size. ERFNet redesigns the non-bottleneck residual module of ResNet by introducing 1D asymmetric convolution, which contributes to traffic scene segmentation. In addition to reducing the computation cost, we replace the typical 3D convolution to decouple aggregation of features along each axis, which greatly enhances message passing along particular axes. Moreover, various strategies can be adapted for different 1D asymmetric convolution layers along different axes, e.g., dilation, deformation and stacking.

A 3D convolution can be separated into a sequence of 1D convolutions, as supported by the low-rank hypothesis [28]. Let $W \in \mathbb{R}^{d_{in} \times d_x \times d_y \times d_z \times d_{out}}$ denote the weights of a 3D convolution layer, where d_{in} and d_{out} are numbers of input and output planes, and $d_x \times d_y \times d_z$ indicates the kernel size. For convenience, d_x, d_y and d_z are set to the same value d. Given the i-th kernel in the 3D convolution layer $k^i \in \mathbb{R}^{d \times d \times d}$, it can be decomposed as follows:

$$k^i = \sum_{r=1}^{R} \alpha_r^i \otimes \beta_r^i \otimes \gamma_r^i, \tag{1}$$

where R is the rank of k^i, $\otimes$ is the outer product operation, and α_r^i, β_r^i, γ_r^i are 1D vectors having the same size of $1 \times d$. In this paper, R is set to 1 for the trade-off between accuracy and efficiency. Therefore, the kernel k^i with size of d^3 can be decomposed



(a) SACB (b) Asymmetric convolution

Figure 2: (a): An SACB is composed of two sub-blocks, each of which is formed by stacking a specified number of asymmetric convolutions in a predefined sequence. (b): Asymmetric convolutions equipped with different strategies, e.g., typical (**pink**), dilation (**orange**) and deformation (**blue**).

into three $1 \times d$ kernels, employing the decomposition on a 3D convolution layer with $3 \times 3 \times 3$ kernels can yield a 66% reduction in parameters.

Buildings and objects are often aligned in the driving direction, and LiDAR points are typically distributed on their surfaces. What is more, point clouds scattered on various surfaces exhibit distinct correlations along different axes. For instance, assuming that the x-axis direction corresponds with vehicle orientation in a traffic scene, points on the lamp-post are distributed along the z-axis. Whereas points on the building wall are primarily scattered on the plane perpendicular to the y-axis. In general, correlation on the x- and y-axis is significant for points on the corners of buildings.

To enhance the feature aggregation for a given scene, the ratio among the numbers of 1D convolutions along x-, y- and z-axis can be customized to strengthen message passing along particular axes. As demonstrated in Fig. 2 (a), each sub-block is created by stacking a series number of 1D convolutions along different axes in a predefined sequence. By the above decomposition, convolution operations along the x-, y- and z-axis can be conducted independently. In addition, different strategies can be applied to these 1D convolutions along different axes, as illustrated in Fig. 2 (b).

3.2. Selective Feature Fusion

In this paper, point- and channel-wise context gating layers are stacked to form the SFFB, which is a kind of mixed attention mechanism that contributes to local feature enhancement by leveraging two attention modes. SFFB works as a plugin placed before low- and high-level feature fusion, and the detailed design will be described as follows.

Point-wise context gating. It has been verified that point-wise context gating is beneficial to place recognition [36]. Point clouds have plenty of low-level visual cues, e.g., edges, planes and corners, which are shaped by the corresponding key points. Moreover, certain regions with salient geometric shapes contribute the most to place recognition, e.g., doors, windows, lamp-posts and their spatial relationships. We apply the point-wise attention to selectively boost the features of key regions. In addition, the operation contributes to key point estimation, which is crucial for the geometric verification task that follows the place recognition.

Given a feature map $X \in \mathbb{R}^{N \times C}$, N and C indicate the number of feature points and channels, respectively. A point-wise gating layer F_{point} conducts a reweighting operation on each point:

$$F_{point}(X_n) = X_n \cdot \sigma\left(MLP_{point}\left(X_n\right)\right), \tag{2}$$

where X_n indicates the n-th point of X, σ is the sigmoid activation. Given X_n with size of $1 \times C$, the output size of MLP_{point} is 1. The point-wise layer has the same size of input and output, which pays more attention to key regions.

Channel-wise gating. A channel-wise gating layer conducts a reweighting operation on each channel, then channel dependencies can be exploited. Another concern is that place recognition usually aggregates local features into a global descriptor to compute the Euclidean distance. Channel-wise gating as a kind of semantic refinement further makes the semantics (channels) to be more comparable and more diverse, which contributes to accurate distance compution. The channel-wise gating layer $F_{channel}$ is represented as follows:

$$F_{channel}(X_n) = X_n \odot \sigma\left(MLP_{channel}\left(AvgPool\left(X\right)\right)\right), \tag{3}$$

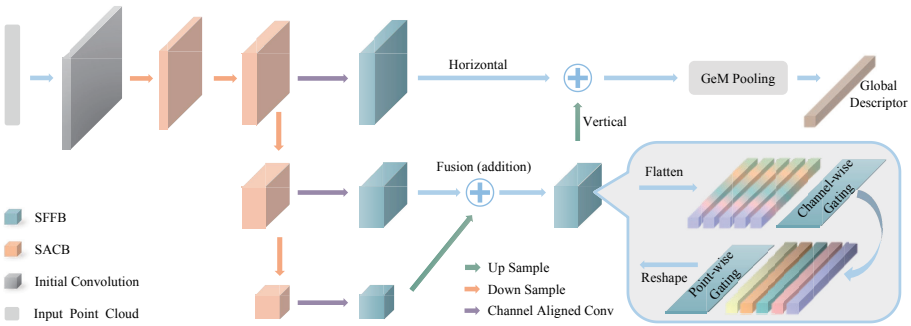


Figure 3: The architecture of SelFLoc implemented in an encoder-decoder style. In encoder stage, down sampling layers are utilized to reduce the resolutions of feature maps. each of which is followed by an SACB. Low- (horizontal) and high-level (vertical) features are fused (addition) during the decoder stage. Note that an SFFB is placed prior to the local feature fusion phase, which is intended for point- and channel-wise selective fusion refinement.

where $X_n \in \mathbb{R}^{1\times C}$ indicates the n-th point of $X \in \mathbb{R}^{N\times C}$, $\odot$ and σ are the Hadamard product (element-wise multiplication) and sigmoid activation, respectively. Notably, the value of N is not fixed, due to the difference between the numbers of points in each frame, and an global average pooling operation is required to ensure the input size of $MLP_{channel}$ is $1 \times C$.

Original SENet [33] uses global average pooling to squeeze global spatial information. Then fully-connected (FC) layers with dimensionality reduction are used to take advantage of the information. On the other hand, ECA reweights each channel by considering its nearest neighbors, thus only local cross-channel interaction is captured. Both of the above methods bring side effects on Euclidean distance computation. In this paper, FC layers without dimensionality reduction are employed in $MLP_{channel}$, which are more suitable for reliable place recognition. For a comprehensive comparison, $MLP_{channel}$ in SENet, ECA-Net and MLP-Mixer styles are also implemented in our ablution experiments.

3.3. Overall Architecture.

As illustrated in Fig. 3, the overall architecture mainly consists of an initial convolution, SACBs and SFFBs, which is implemented in an encoder-decoder style. The encoder stage, in particular, adapts an initial convolution to project 3D LiDAR points into a deep feature space. Then multi-scale feature maps can be produced by a series

of SACB blocks, placed after down sampling operations. Low- and high-level features are fused during the decoder stage. And the SFFB is naturally placed prior to the local feature fusion phase.

Initial convolution is conducted by a $K_0 \times K_0 \times K_0$ 3D sparse convolution for primary deep feature extraction. Given an input point cloud $P = \{p_1, p_2, ..., p_N\}$, initial feature $X = \{x_1, x_2, ..., x_N\}$ can be extracted. p_n indicates the coordinate of n-th point in the point cloud, whose size is 1×3, and the size of corresponding feature x_n is $1 \times C_0$. As described previously, the asymmetric convolution is more suitable for place recognition task than symmetric 3D convolution. However, asymmetric convolution is not guaranteed to work well on low-level layers [27], typical 3D sparse convolution is still required for this phase.

SACB consists of two sub-blocks, each sub-block has three asymmetric convolution layers operating on x-, y- and z-axis, respectively, as well as an additional layer along a particular axis. Furthermore, to verify the influence of additional convolution along different axes, three version are created: *SelFLoc_X*, *SelFLoc_Y* and *SelFLoc_Z*. The dilation strategy is also introduced into the additional asymmetric convolution. For a fair comparison, only one SACB block of a specific depth employs dilation strategy for each trial.

Notice that the numbers and strategies of asymmetric convolutions along different axes can be varied and a study on these combinations may lead to superior outcome than the ones presented in this research. However, such studies are out of the scope of this research, and the above models are chosen for a good balance between accuracy and efficiency.

SFFB is placed prior to each additive fusion phase, which includes a point-wise context gating layer and a channel-wise one. There are two different stacking orders: point-wise gating first and channel-wise gating first. We implement these two model types, moreover, models with only point- or channel-wise gating layers are also implemented for comparison.

Overall forward process is elaborated in **Algorithm 1**. The down and up sampling depths are denoted as D_d and D_u, respectively, which should be predefined for a specific place recognition task. Note that the $ChannelAlignedConv$ is used for

Algorithm 1 The Overall Forward Process of SelFLoc.

Require: Point cloud P, down sampling depth D_d, up sampling depth D_u.
Ensure: Global descriptor G of point cloud P.

Initialize lower level feature maps $F \leftarrow \emptyset$
$X \leftarrow InitialConvolution(P)$
for $d = 0 \rightarrow D_d - 1$ **do**
    $X \leftarrow DownSample(X)$
    $X \leftarrow SACB(X)$
    if $D_d - D_u - 1 \leq d < D_d - 1$ **then**
        $F \leftarrow F \cup X$
    end if
end for
for $d = 0 \rightarrow D_u - 1$ **do**
    $X \leftarrow UpSample(X)$
    $X \leftarrow SFFB(X)$
    Get the last d-th feature map Y of F
    $Y \leftarrow ChannelAlignedConv(Y)$
    $Y \leftarrow SFFB(Y)$
    $X \leftarrow X + Y$
end for
$G \leftarrow GeMPool(X)$
Return G

channel number alignment, which is inspired by Minkloc3dv2 and plays a crucial role for feature addition phase. In addition, GeM pooling [38] is employed to aggregate local features into a global descriptor G in accordance with typical place recognition methods. As discussed above, Euclidean distance employed in point cloud retrieval for place recognition demands the amplitudes of each channel are comparable. GeM, as a generalized form of harmonic and quadratic mean, can further align the channels (semantics).

3.4. Probability Model of Point Cloud-based Place Recognition

Given the training dataset $D = \{(q, i, j) \mid q \in \mathcal{Q}\}$, where $\mathcal{Q}$ is the query set, and i and j indicate sampled point cloud frames. The training objective $\mathcal{O}$ is to maximize the posterior probability defined as follows:

$$\begin{aligned} \mathcal{O} &= \ln p\left(\Theta \mid D\right) \\ &= \ln p\left(D \mid \Theta\right) p(\Theta), \end{aligned} \tag{4}$$

where Θ indicates the parameters of SelFLoc. According to reference [39], we introduce a common assumption that $p(\Theta)$ is a normal distribution: $p(\Theta) \sim N(0, \lambda_\Theta I)$. $p\left(D \mid \Theta\right)$ is the likelihood function, which can be rewriten as follows:

$$\ln p\left(D \mid \Theta\right) = \ln \prod_{(q,i,j)\in D} p\left(i >_q j \mid \Theta\right)^{\delta((q,i,j)\in D_O)} \cdot (1 - p\left(i >_q j \mid \Theta\right))^{\delta((q,i,j)\notin D_O)}, \tag{5}$$

where δ is the indicator function. $p\left(i >_q j \mid \Theta\right)$ represents the probability that the point cloud frame i is closer to the query frame q compared to the frame j. D_O is a set containing observed preferences, which can be defined as:

$$D_O = \{(q, i, j) \mid q \in \mathcal{Q} \wedge i \in S_P \wedge j \in S_N\}, \tag{6}$$

where S_P and S_N represent the positive and negative sets, respectively. Moreover, the probability can be calculated by the corresponding global descriptors as:

$$p(i >_c j|\Theta) = \sigma(\hat{G}_{qij}(\Theta)) \tag{7}$$

$$\hat{G}_{qij}(\Theta) = d(q, i) - d(q, j) \tag{8}$$

$$\sigma(x) := \frac{1}{1 + e^{-x}}, \tag{9}$$

where $d(q,i)$ is the Euclidean distance between the global descriptors of q and i. Combining the above equations, we rewrite the objective function as follows:

$$\begin{aligned}
\mathcal{O} &= \ln \prod_{(q,i,j)\in D} p\left(i >_q j \mid \Theta\right) p(\Theta) \\
&= \ln \prod_{(q,i,j)\in D} \sigma(\hat{G}_{qij}(\Theta))^{\delta((q,i,j)\in D_O)} (1-\sigma(\hat{G}_{qij}(\Theta)))^{\delta((q,i,j)\notin D_O)} p(\Theta) \\
&= \sum_{(q,i,j)\in D_O} \ln \sigma(\hat{G}_{qij}(\Theta)) + \sum_{(q,i,j)\notin D_O} \ln(1-\sigma(\hat{G}_{qij}(\Theta))) + \ln p(\Theta) \\
&= \sum_{(q,i,j)\in D_O} \ln \sigma(\hat{G}_{qij}(\Theta)) + \sum_{(q,i,j)\notin D_O} \ln(1-\sigma(\hat{G}_{qij}(\Theta))) - \lambda_\Theta ||\Theta||^2 \\
&\approx \sum_{(q,i,j)\in D_O} \ln \sigma(d(q,i)-d(q,j)) + \sum_{(q,i,j)\notin D_O} \ln(1-\sigma(d(q,i)-d(q,j))).
\end{aligned} \tag{10}$$

According to Jensen's inequality, the objective function has a lower bound:

$$\begin{aligned}
\mathcal{O} &\geq \ln \sum_{(q,i,j)\in D_O} \sigma(d(q,i)-d(q,j)) + \ln \sum_{(q,i,j)\notin D_O} (1-\sigma(d(q,i)-d(q,j))) \\
&= \ln \mathcal{O}_1 + \ln(|D-D_O| - \mathcal{O}_2).
\end{aligned} \tag{11}$$

According to the monotonicity of Equation 11, maximizing the lower bound is equivalent to maximizing $\mathcal{O}_1$ and minimizing $\mathcal{O}_2$. Training the proposed SelFLoc using the Smooth-AP [40, 15] loss function can achieve this goal.

Relation to Smooth-AP Loss. Smooth-AP is a commonly used training loss function in the field of place recognition. Specifically, it calculates AP_q for each query frame q as follows:

$$\begin{aligned}
AP_q &\approx \frac{1}{|S_P|} \sum_{i\in S_P} \frac{1+\sum_{h\in S_P} \sigma(d(q,i)-d(q,h))}{1+\sum_{h\in S_P} \sigma(d(q,i)-d(q,h)) + \sum_{j\in S_N} \sigma(d(q,i)-d(q,j))} \\
&= \frac{1}{|S_P|} \sum_{i\in S_P} \frac{1+\mathcal{O}_2}{1+\mathcal{O}_2+\mathcal{O}_1}.
\end{aligned} \tag{12}$$

Therefore, the loss for each batch can be calculated as:

$$\mathcal{L}_{AP} = \frac{1}{m} \sum_{q=1}^{m} (1 - AP_q), \tag{13}$$

where m is the number of queries in one batch. Comparing Equation 11, 12, and 13, we can see that maximizing $\mathcal{L}_{AP}$ is equivalent to maximizing the lower bound of objective function $\mathcal{O}$.

4. Experiments

This section verifies the performance of proposed SelFLoc by conducting experiments on a variety of benchmark datasets. Additionally, ablation studies are performed to analyze the introduced blocks and strategies.

4.1. Experiments Setting

Benchmark and evaluation. PointNetVLAD originally created four benchmark datasets for evaluating point cloud-based place recognition networks: Oxford is a partial set of Oxford RobotCar dataset [41], U.S., R.A. and B.D. are respective in-house datasets of a university sector, a residential area and a business district. These datasets are obtained using a LiDAR sensor installed on a car that regularly drives over each of the four regions at different times, and collects data under varying environmental conditions.

The locations are sampled at a specified interval (shown in Table 1) along continuous tracks of the vehicle, and the corresponding submaps are constructed by dividing the LiDAR scans and erasing non-informative ground planes. Each submap is down sampled to 4096 points using a voxel filter and tagged with the Universal Transverse Mercator (UTM) coordinate of its centroid. During training tuple generation, a point cloud pair is considered positive if the distance is less than 10m and negative if the distance is greater than 50m. In order to evaluate various place recognition methods, the query point cloud is successfully localized when the retrieved point cloud is within 25m.

Table 1: The number of submaps for training and testing, * approximate value.

	Training Set	Testing Set	Submaps /Run	Intervals (training)	Intervals (testing)
Oxford	21711	3030	120-150	10m	20m
U.S.			400*		
R.A.	6671	4542	320*	12.5m	25m
B.D.			200*		

Our evaluation follows the baseline training pipeline introduced in [7]. Specifically, the network is trained using the training set of the Oxford dataset and tested on the testing sets of the Oxford and three in-house datasets. Table 1 shows the details of training and testing datasets, the average recall@1% (AR@1%) and average recall@1 (AR@1) metrics are primarily adopted for evaluation.

Implementation details. Implementation of the proposed network SelFLoc is based on the MinkLoc3Dv2 codebase, where the sparse convolution on point clouds is implemented by Minkowski Engine [42].

To be fair, the size of input point clouds is 4096×3, which is the same as PointNetVLAD. Each down sampling layer decreases the spatial resolution by two. In our experiments, there are 4 down sampling layers (each is placed before an SACB), and 2 up sampling layers (each is followed by an SFFB). The training loss, stays the same with MinkLoc3Dv2.

4.2. Main Results

To verify the quantitative performance of the proposed SelFLoc method, we conduct comprehensive experiments on the benchmark datasets in [7]. Specifically, we compare SelFLoc with a lot of advanced methods, including RI_STV [14], MinkLoc3Dv2 [15], HiBi-Net [43], SVT-Net [16], PPT-Net [13], NDT-transformer [26], EPC-Net [44], LPD-Net [12], PCAN [36] and the pioneering PointNetVLAD [7]. Table 2 reports the AR@1% and AR@1 metrics of aforementioned methods trained by the baseline pipeline. Although many excellent methods have be proposed for place recognition, and there is little room for improvement [15, 17]. SelFLoc can still has a remarkable performance, outperforming the most sophisticated methods [14, 15] by 1.6 absolute percentages on AR@1 (90 vs. 91.6) and 1 absolute percentage on AR@1% (95.1 vs.

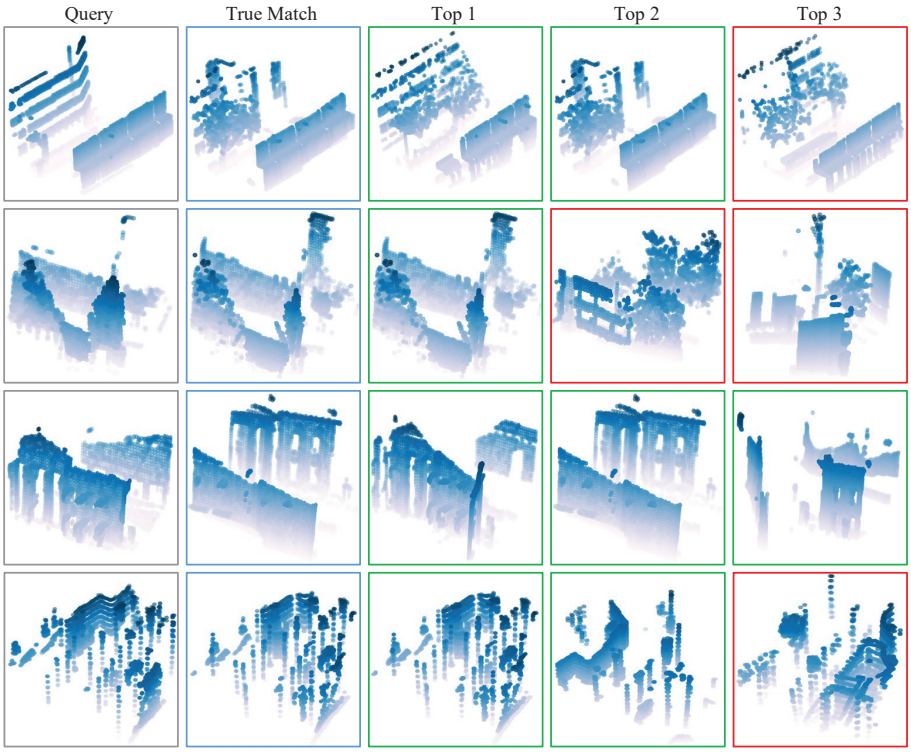


Figure 4: Query (gray) and top 3 retrieved frames (green: successful, red: failed). Moreover, one of the true (blue) matches is displayed for comparison. SelFLoc successfully finds the closest match even when the perspective changes (row 3).

96.1). Moreover, the in-house datasets in Table 2 have not be trained and the AR@1 metric has been improved by 2.3%, 2.3% and 2.1%, respectively. Compared with MinkLoc3Dv2, SelFLoc has a better capability for generalization in addition to its high accuracy. This is crucial for mobile robots because their operation scenarios are diverse, and it is difficult to collect sufficient training data for each scenario. Note that 3D convolution-based methods have higher metrics than kNN-based and NDT-based methods in the mass, which demonstrates that voxel-based convolution is still an effective technique to capture the locality and hierarchy of point cloud.

Fig. 4 displays the query point cloud and the point clouds retrieved by SelFLoc. The cases are challenging and SelFLoc successfully retrieves the closest match by leveraging the most discriminative feature. As depicted in Fig. 3, SFFB can be plugged

Table 2: Evaluation results of the advanced place recognition methods.

	Oxford		U.S.		R.A.		B.D.		Mean	
	AR@1	AR@1%	AR@1	AR@1%	AR@1	AR@1%	AR@1	AR@1%	AR@1	AR@1%
PointNetVLAD [7]	62.8	80.3	63.2	72.6	56.1	60.3	57.2	65.3	59.8	69.6
PCAN [36]	69.1	83.8	62.4	79.1	56.9	71.2	58.1	66.8	61.6	75.2
LPD-Net [12]	86.3	94.9	87.0	96.0	83.1	90.5	82.5	89.1	84.7	92.6
EPC-Net [44]	86.2	94.7	-	96.5	-	88.6	-	84.9	-	91.2
SOE-Net [20]	-	96.4	-	93.2	-	91.5	-	88.5	-	92.4
HiBi-Net [43]	87.5	95.1	87.8	-	85.8	-	83.0	-	86.0	-
MinkLoc3D [21]	93.0	97.9	86.7	95.0	80.4	91.2	81.5	88.5	85.4	93.2
NDT-Transformer [26]	93.8	97.7	-	-	-	-	-	-	-	-
PPT-Net [13]	93.5	98.1	90.1	97.5	84.1	93.3	84.6	90.0	88.1	94.7
SVT-Net [16]	93.7	97.8	90.1	96.5	84.3	92.7	85.5	90.7	88.4	94.4
TransLoc3D [17]	95.0	98.5	-	94.9	-	91.5	-	88.4	-	93.3
MinkLoc3Dv2 [15]	**96.3**	**98.9**	90.9	96.7	86.5	93.8	86.3	91.2	90	95.1
RI_STV [14]	-	98.5	-	97.3	-	93.0	-	91.7	-	95.1
SelFLoc (ours)	96.0	98.8	**93.2**	**98.3**	**88.8**	**94.8**	**88.4**	**92.4**	**91.6**	**96.1**

into both horizontal and vertical feature branches, we further visualize the point-wise attention maps in these branches before the last fusion phase. As shown in Fig. 6, point-wise gating layer in horizontal SFFB mainly pays attention to the major structure of the point cloud, while point-wise attention in vertical branch can be selectively allocated to the points isolated but salient.

4.3. Ablation Study

Additional axes. To verify the impact of different additional layers, asymmetric convolution layers along x-, y- and z-axis are respectively added to the sub-blocks of SACBs every time. Fig. 5 displays the quantitative results with different additional layers, and *SelFLoc_X* achieves better accuracy and robustness, demonstrating that feature extraction along the x-axis is the most effective for the place recognition task. This may inspire researchers in the field of mobile robotics to optimize axis-oriented point cloud processing methods for specific scenarios. For example, in the field of self-driving, there may be more sophisticated methods developed in the future for enhancing the transmission of point cloud features along X-axis.

Dilation strategy. The effective receptive field can be expanded by utilizing the dilation strategy. Results of testing dilation influence on different depths of SACBs in Fig. 5 indicate that models with lower level dilated convolutions have higher metrics. Dilated convolutions in lower layers contribute more to the capture of local response. On the contrary, dilated convolutions in higher layers help to capture the interrelationship between semantics at higher levels, however the performance will be negatively

affected if levels of dilated convolutions are excessively high. The performance of dilated convolution layers in different depths evidences that place recognition relies more on local features, and applicable depth is crucial for dilation strategy, e.g., $Depth = 1$.

Channel-wise attention mechanisms. In order to compare the effectiveness of different attention mechanisms for channel-wise feature aggregation, we conduct experiments with various attention mechanisms. Specifically, FC layers without dimensionality reduction (SelFLoc-FC), channel-wise gating in the SENet (SelFLoc-SE), channel-wise gating in the ECA-Net (SelFLoc-ECA) and channel-wise attention proposed in MLP-Mixer (SelFLoc-Mixer) are implemented, as shown in Table 3. The comparison between SelFLoc-Mixer and other methods shows that the squeeze phase is crucial for both robustness and accuracy. Moreover, FC attention without dimensionality reduction or neighbor limitation further improves the robustness by a large margin.

Table 3: Evaluation results (AR@1) of models with different channel-wise attention mechanisms.

Model Type	Oxford	U.S.	R.A	B.D.
SelFLoc-ECA	**96.53**	90.72	87.37	87.35
SelFLoc-SE	96.39	90.47	87.50	87.05
SelFLoc-Mixer	94.59	88.73	86.30	84.47
SelFLoc-FC(Ours)	96.04	**93.23**	**88.84**	**88.38**

Table 4: Evaluation results (AR@1) of different gating orders.

First Layer	Second Layer	Oxford	U.S.	R.A	B.D.
P	P	96.11	92.33	85.89	86.84
P	C	**96.21**	91.76	86.76	86.31
C	P	96.04	**93.23**	**88.84**	**88.38**
C	C	94.95	90.85	88.17	87.19

Table 5: Ablation study results (AR@1) of different D1 and D2.

D1	D2	Oxford	U.S.	R.A	B.D.
128	128	94.88	87.75	86.76	83.92
256	128	95.29	90.47	83.94	84.57
256	256	96.04	93.23	**88.84**	**88.38**
256	512	**96.24**	92.65	87.83	86.72
512	512	96.10	**93.43**	88.44	87.90

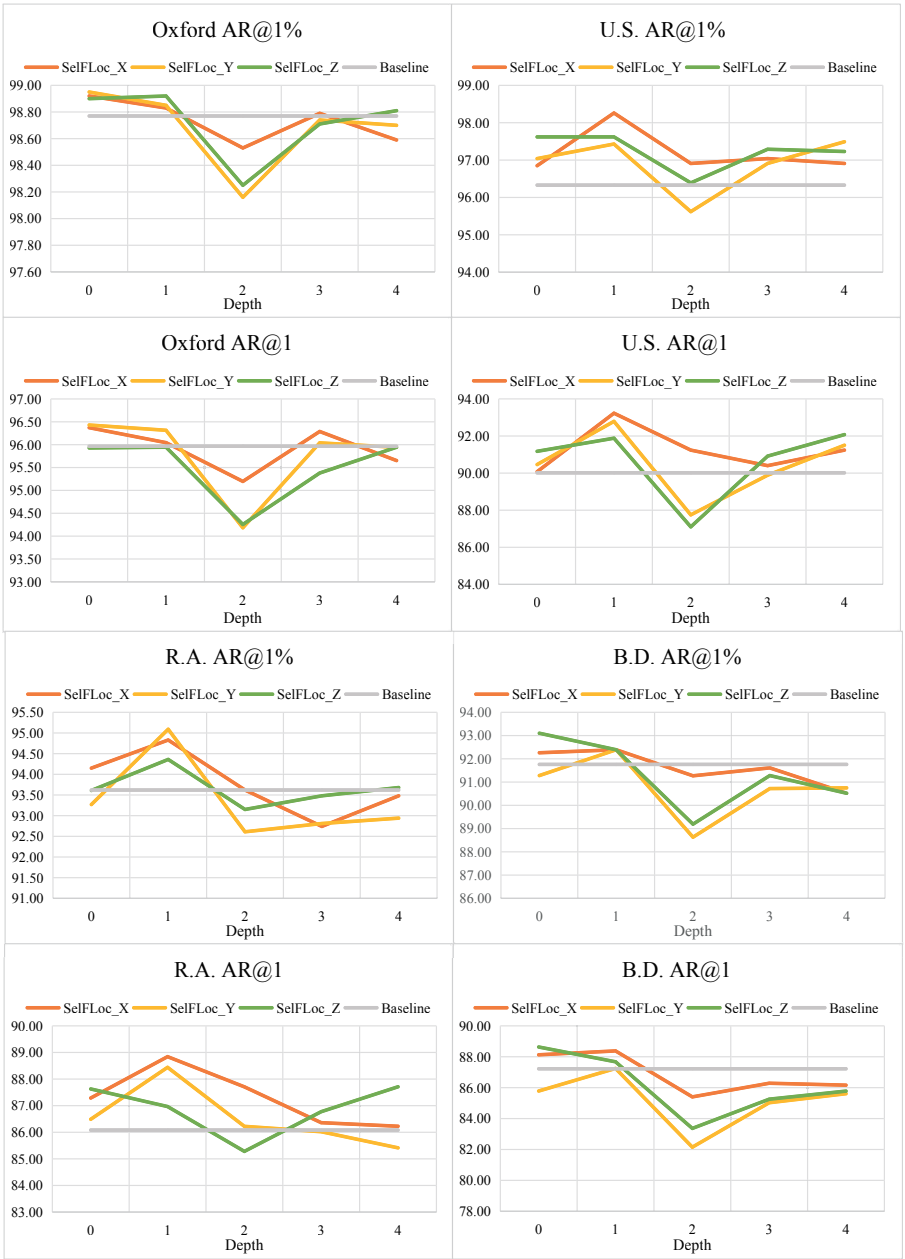


Figure 5: Horizontal axis represents the depth of which SACB equipped with dilation strategy. There are 4 SACBs employed in our experiments, and $Depth = 0$ indicates that no SACB is equipped with dilation strategy. *SelFLoc_X*, *SelFLoc_Y* and *SelFLoc_Z* represents the models with one additional layer along x-, y- and z-axis, respectively, while model without additional asymmetric convolution layer is regarded as the **baseline**. Note that the dilation strategy is only applied on the additional layer of each sub-block in the SACB.

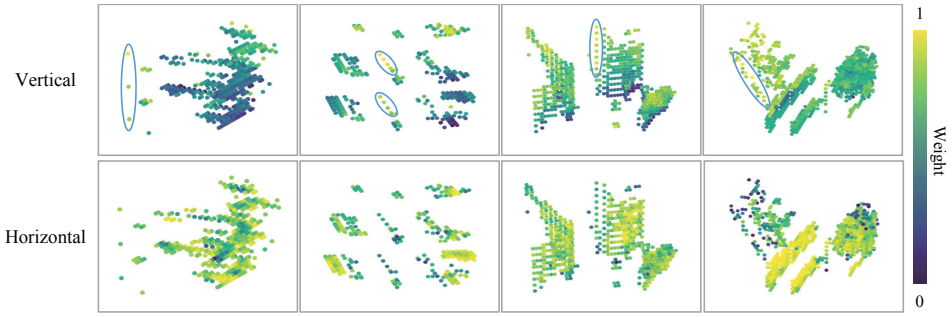


Figure 6: Point-wise attention in horizontal and vertical SFFBs. Note that the horizontal point-wise attention is generated by lower-level local features, while the vertical one is selectively enhanced by semantics from higher level. Points in blue circles are isolated but salient.

Combinations of point- and channel-wise gating layers. An SFFB is composed of a point-wise and a channel-wise gating layer. Table 4 shows performance of different combinations of gating layers. From the table, it can be seen that SFFBs with only point or channel-wise gating layer have worse performance than SFFBs with both point- and channel-wise gating layers. In addition, the best performance can be obtained when the first layer is channel-wise and the second layer is point-wise. As mentioned previously, channel-wise gating is employed for semantic (channel) alignment, therefore earlier alignment can result in better performance.

Dimension influence. Here we study the impact of model size. In particular, we denote D1, D2 as the output dimensions of last and penultimate SACB blocks, respectively. Note that D2 is equal to the numbers of channels in each fusion phase, as well as the size of global descriptor for retrieval. As shown in Table 5, expanding the model size can lead to higher accuracy, while overfitting will reduce its robustness. The model with $D1 = 256$ and $D2 = 256$ achieves an empirically optimal trade-off between accuracy and robustness.

5. Conclusion

In this paper, a novel architecture named SelFLoc is proposed for point cloud-based place recognition, which takes advantage of the strong shape priors of point clouds by stacking 1D asymmetric convolutions equipped with different strategies. In addition,

features from different scales are refined by selective point- and channel-wise gating layers before the fusion phase. Comprehensive experiments on the Oxford dataset and three in-house datasets demonstrate that SelFLoc can achieve SOTA performance in terms of both accuracy and robustness.

Limitation and Future Work. The decomposition of 3D convolution, enhancement of axis-oriented features, and selective feature fusion in this research are all tailored for scenarios involving self-driving. However, the introduction of these strategies relies on the experience of researchers. To make them more applicable to other robotic studies, such as drones and bipedal robots, we will explore novel methods to endow robots with the ability to autonomously learn these strategies. One potential approach is to integrate the selection of strategies into the learnable world model [45, 46].

6. Acknowledgements

This research is supported in part by the National Natural Science Foundation of China under Grant 62303428, and in part by Zhejiang Provincial Natural Science Foundation of China under Grant LQ23F030010.

References

[1] J. L. Matez-Bandera, J. Monroy, J. Gonzalez-Jimenez, Efficient semantic place categorization by a robot through active line-of-sight selection, Knowledge-Based Systems 240 (2022) 108022.

[2] C. Wang, X. Chen, C. Li, R. Song, Y. Li, M. Q.-H. Meng, Chase and track: Toward safe and smooth trajectory planning for robotic navigation in dynamic environments, IEEE Transactions on Industrial Electronics 70 (1) (2022) 604–613.

[3] Y. Shi, R. Yang, Z. Wu, P. Li, C. Liu, H. Zhao, G. Zhou, City-scale continual neural semantic mapping with three-layer sampling and panoptic representation, Knowledge-Based Systems 284 (2024) 111145.

[4] Q. Zhang, Z. Xu, Y. Kang, F. Hao, Z. Ren, J. Cheng, Distilled representation using patch-based local-to-global similarity strategy for visual place recognition, Knowledge-Based Systems 280 (2023) 111015.

[5] J. Yu, C. Zhu, J. Zhang, Q. Huang, D. Tao, Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition, IEEE Transactions on Neural Networks and Learning Systems 31 (2) (2019) 661–674.

[6] Y. Wang, Y. Qiu, P. Cheng, J. Zhang, Transformer-based descriptors with fine-grained region supervisions for visual place recognition, Knowledge-Based Systems 280 (2023) 110993.

[7] M. A. Uy, 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.

[8] C. R. Qi, H. Su, K. Mo, 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.

[9] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, 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.

[10] J. Choe, C. Park, F. Rameau, J. Park, I. S. Kweon, Pointmixer: Mlp-mixer for point cloud understanding, in: European Conference on Computer Vision, Springer, 2022, pp. 620–640.

[11] X. Li, X. Zhang, X. Zhou, I.-M. Chen, Upg: 3d vision-based prediction framework for robotic grasping in multi-object scenes, Knowledge-Based Systems 270 (2023) 110491.

[12] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, 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.

[13] L. Hui, H. Yang, M. Cheng, J. Xie, J. Yang, Pyramid point cloud transformer for large-scale place recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2021, pp. 6098–6107.

[14] D. Kong, X. Li, W. Hu, J. Hu, Y. Hu, Q. Xu, X. Song, Explicit points-of-interest driven siamese transformer for 3d lidar place recognition in outdoor challenging environments, IEEE Transactions on Industrial Informatics (2023).

[15] J. Komorowski, Improving point cloud based place recognition with ranking-based loss and large batch training, arXiv preprint arXiv:2203.00972 (2022).

[16] Z. Fan, Z. Song, H. Liu, Z. Lu, J. He, X. Du, Svt-net: Super light-weight sparse voxel transformer for large scale place recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 551–560.

[17] T.-X. Xu, Y.-C. Guo, Y.-K. Lai, S.-H. Zhang, Transloc3d: Point cloud based large-scale place recognition using adaptive receptive fields, arXiv preprint arXiv:2105.11605 (2021).

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

[19] Y. Shen, L. Hui, Flowformer: 3d scene flow estimation for point clouds with transformers, Knowledge-Based Systems 280 (2023) 111041.

[20] Y. Xia, Y. Xu, S. Li, R. Wang, J. Du, D. Cremers, U. Stilla, Soe-net: A self-attention and orientation encoding network for point cloud based place recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2021, pp. 11348–11357.

[21] J. Komorowski, Minkloc3d: Point cloud based large-scale place recognition, in: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 2021, pp. 1790–1799.

[22] L. He, X. Wang, H. Zhang, M2dp: A novel 3d point cloud descriptor and its application in loop closure detection, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2016, pp. 231–237.

[23] G. Kim, S. Choi, A. Kim, Scan context++: Structural place recognition robust to rotation and lateral variations in urban environments, IEEE Transactions on Robotics 38 (3) (2021) 1856–1874.

[24] F. Cao, F. Yan, S. Wang, Y. Zhuang, W. Wang, Season-invariant and viewpoint-tolerant lidar place recognition in gps-denied environments, IEEE Transactions on Industrial Electronics 68 (1) (2020) 563–574.

[25] L. Luo, S.-Y. Cao, Z. Sheng, H.-L. Shen, Lidar-based global localization using histogram of orientations of principal normals, IEEE Transactions on Intelligent Vehicles 7 (3) (2022) 771–782.

[26] Z. Zhou, C. Zhao, D. Adolfsson, S. Su, Y. Gao, T. Duckett, L. Sun, Ndt-transformer: Large-scale 3d point cloud localisation using the normal distribution transform representation, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2021, pp. 5654–5660.

[27] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.

[28] J. Alvarez, L. Petersson, Decomposeme: Simplifying convnets for end-to-end learning, arXiv preprint arXiv:1606.05426 (2016).

[29] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[30] E. Romera, J. M. Alvarez, L. M. Bergasa, R. Arroyo, Erfnet: Efficient residual factorized convnet for real-time semantic segmentation, IEEE Transactions on Intelligent Transportation Systems 19 (1) (2017) 263–272.

[31] X. Ding, Y. Guo, G. Ding, J. Han, Acnet: Strengthening the kernel skeletons for powerful cnn via asymmetric convolution blocks, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 1911–1920.

[32] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu, Eca-net: Efficient channel attention for deep convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 11534–11542.

[33] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.

[34] H. Jin Kim, E. Dunn, J.-M. Frahm, Learned contextual feature reweighting for image geo-localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2136–2145.

[35] G. Chen, L. Wan, L. Song, Z. Liu, 3d perception arithmetic of random environment based on rgb enhanced point cloud fusion, Knowledge-Based Systems (2023) 110710.

[36] W. Zhang, C. Xiao, Pcan: 3d attention map learning using contextual information for point cloud based retrieval, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 12436–12445.

[37] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, et al., Mlp-mixer: An all-mlp architecture for vision, Advances in Neural Information Processing Systems 34 (2021) 24261–24272.

[38] F. Radenović, G. Tolias, O. Chum, Fine-tuning cnn image retrieval with no human annotation, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (7) (2018) 1655–1668.

[39] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, Bpr: Bayesian personalized ranking from implicit feedback, arXiv preprint arXiv:1205.2618 (2012).

[40] A. Brown, W. Xie, V. Kalogeiton, A. Zisserman, Smooth-ap: Smoothing the path towards large-scale image retrieval, in: European Conference on Computer Vision, Springer, 2020, pp. 677–694.

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

[42] C. Choy, J. Gwak, S. Savarese, 4d spatio-temporal convnets: Minkowski convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3075–3084.

[43] D. W. Shu, J. Kwon, Hierarchical bidirected graph convolutions for large-scale 3-d point cloud place recognition, IEEE Transactions on Neural Networks and Learning Systems (2023).

[44] L. Hui, M. Cheng, J. Xie, J. Yang, M.-M. Cheng, Efficient 3d point cloud feature learning for large-scale place recognition, IEEE Transactions on Image Processing 31 (2022) 1258–1270.

[45] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, K. Goldberg, Daydreamer: World models for physical robot learning, in: Conference on Robot Learning, PMLR, 2023, pp. 2226–2240.

[46] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, H. Li, End-to-end autonomous driving: Challenges and frontiers, arXiv preprint arXiv:2306.16927 (2023).