HPGCNN: Hierarchical Parallel Group Convolutional Neural Networks for Point Clouds Processing

Jisheng Dang, Jun Yang*

Lanzhou Jiaotong University, Lanzhou 730070, China

Abstract. To achieve high performance but less complexity for point clouds processing, we introduce HPGCNN, an efficient and lightweight neural architecture. The key component in our approach is the Hierarchical Parallel Group Convolution(HPGConv) operation. It can capture both the discriminative independent single-point features and local geometric features of point clouds at the same time to enhance the richness of the features with less redundant information by designing two hierarchical parallel group convolutions, which is helpful to recognize elusive shapes. To significantly further reduce complexity and natively prevent overfitting, we use global average pooling and a full connected layer instead of the traditional three full connected layers for classification. Moreover, to further capture the contextual fine-grained features with higher-level semantics, we introduce a novel multi-semantic scale strategy to progressively increase the receptive field of each local area through the information communication of local areas of different scales. Extensive experiments show that our HPGCNN clearly surpasses state-of-the-art approaches for point clouds classification dataset ModelNet40 and large scale semantic segmentation datasets ShapeNet Parts, S3DIS, vKITTI and SemanticKITTI in terms of accuracy and complexity.

1 Introduction

3D point clouds recognition and segmentation is the key technology of 3D point clouds processing and analysis, which is widely used in the fields of autonomous driving [1, 2], intelligent robotics [3, 4], environment perception[5, 6], shape synthesis and modeling [7], etc. However, the irregularity and disorder of 3D point clouds have hindered the development of 3D point clouds recognition and segmentation technology. 3D point clouds recognition and segmentation have become a hot and difficult research topic in the field of computer vision and computer graphics. In order to improve the accuracy of 3D point clouds recognition and segmentation, a large number of methods, mostly composed of traditional methods and deep learning methods, have been proposed. Traditional methods [8, 9] are used to design the feature descriptors of 3D point clouds manually, researchers rely on their existing domain knowledge to extract the features of the

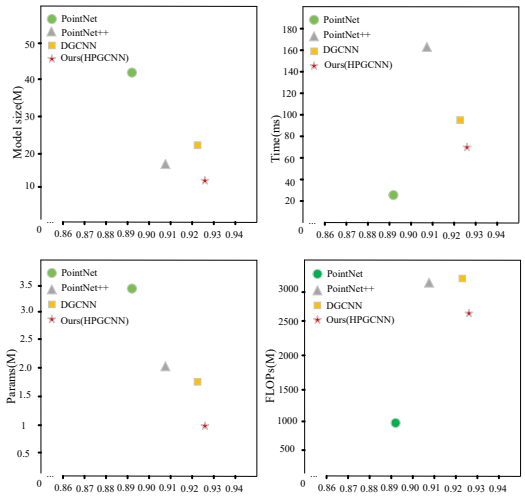

Fig. 1. Comparison with the state-of-the-art approaches in terms of accuracy, parameters, forward time and model size.

3D point clouds which are used further for the task of point clouds recognition and segmentation. Although these methods are effective when used on small datasets, their generalization ability is too poor to be suitable for other large scale datasets. Over the past few years, Convolutional Neural Network(CNN) has demonstrated its powerful abstraction ability of semantic information in computer vision field. Due to the complexity of the internal structure of 3D point clouds, deep learning methods used for 3D models recognition and segmentation still face with great challenges, there have been many achievements in this research area though.

However, currently most deep learning methods focus mainly on how to improve the recognition accuracy with little attention being paid to and few people considering such important issues as the model complexity, the computational complexity and eliminating redundancy of convolution operation is rarely mentioned for point clouds processing and analysis. In the point clouds processing domain and especially for some resource-constrained applications, such as autonomous driving which needs to process large scale point clouds with limited resources, the less redundancy information there is the better. Besides, as mentioned earlier, existing convolution kernels lack the ability to capture both the discriminative local geometric features and the independent single-point features of the point clouds at the same time, resulting in the features extracted that are inadequate. In fact, the local geometric features fully mine the fine-grained details of the local area, but it only focuses on the relative relationship between the point pairs, ignoring the absolute position relationship of each point in the 3D space, which destroys the independent single-point structure features of each point extracted from the most original three-dimensional coordinates informa-

tion of the independent points without considering the neighborhood points. Therefore, a powerful convolution kernel should simultaneously take into account both discriminative local geometric features and independent single-point features to significantly improve the integrity of the features. We are most interested in improving the ability of the convolution kernel to extract more complete features while reducing the redundancy of the convolution kernel, which is beneficial to achieve a better balance among three aspects: model complexity, computation complexity and recognition accuracy. The redundancy comes from two extents: the local geometric features of local point clouds and the independent single-point features of each independent point. Therefore, we present a novel module built of a stack of blocks called Hierarchical Parallel Group Convolution (HPGConv). Our main contributions are summarized as follows:

- We propose an effective convolution operation, the hierarchical parallel group convolution, which can reduce the redundancy of the convolution kernel, and has the ability to encode both the discriminative local geometric features and the independent single-point features of the point clouds at the same time to enhance the completeness of features.

- Instead of adopting the traditional three fully connected layers used for classification in CNN, we adopt the global average pooling with a fully connected layer strategy to reduce the complexity and natively prevent overfitting in the overall structure.

- We introduce a novel multi-semantic scale strategy to progressively increase the receptive field of local areas of different scales, thereby effectively capturing contextual fine-grained features with higher level semantics.

- We propose a hierarchical parallel group convolutional neural network aimed at both in depth redundancy reduction and in width exploitation of the more discriminative independent single-point features and local geometric features for point clouds analysis. Extensive experiments on classification and segmentation tasks verify the effectiveness of our approach.

2 Related Work

(1) Volumetric-based Methods. In literatures [10–13], the irregular point clouds is transformed into regular 3D volumetric grids, and then 3DCNN is used to extract feature descriptor directly from the 3D volumetric grids to complete the point clouds classification and segmentation tasks. However, the sparsity of the 3D volumetric grids results in a large amount of memory consumption. To solve the problem of data sparsity, [14–16] adopt sparse structures like octrees or hash-maps that allow larger grids to enhance performance. Although they achieve leading results on point clouds classification and segmentation, their primary limitation is the heavy computation cost, especially when processing large-scale point clouds.

(2) Projection-based Methods. Projection-based methods [17–20] first project/flatten the 3D point clouds onto 2D views, and then the 2D views are input to classic 2D deep learning networks to extract their features to leverage the success of 2D CNNs. For large-scale scene point clouds segmentation, these methods have some problems such as surface occlusion and density variation. Tatarchenko et al. [21] projects the local neighborhood onto the local tangent plane, and then processes them with two-dimensional convolution. However, projection-based methods have the problems of data redundancy and geometric structure information loss.

(3) Methods based on geometric deep learning. Geometric deep learning [22–24] is the approach for processing non-Euclidean structural point clouds using deep neural networks. Qi et al. [25] proposes the PointNet which uses Multi-Layer Perceptron (MLP) to extract the features of each point in the point clouds and then, in preparation for achieving the goal of 3D point clouds recognition, obtains the global feature descriptor by aggregating the features of all the points through a max pooling layer. In literature [26], PointNet++ hierarchically extracts features by arranging local point clouds. In literature [27], the local geometric features can be captured while the permutation invariance is guaranteed by establishing the dynamic graph CNN. The PointCNN network proposed in literature [28] learns an X-transformation from the input point clouds which is a generalization of the CNN leveraging of the spatial-local correlation from the data representing in the point clouds. A novel convolutional network structure called SpiderCNN [29] extracts deep semantic features by extending convolutional operations from regular grids to irregular point sets. PCNN [30] provides a flexible framework for adapting standard image-based CNNs to the point clouds setting using extension and restriction operations. Liu et al. [31] proposes the relation-shape convolution operation to encode the geometric relations of the points explicitly, thus resulting in good shape awareness. The graph convolutional neural network proves the advantage of graph representation method in non-Euclidean data processing tasks, it mainly contains of spectral representation [32–34] and surface representation [23, 26, 35–37]. These methods combine the features of local surface patches and are not affected by patch deformation in Euclidean space, so local geometric features can be fully explored. For large scale scene point clouds processing, Hu et al. [38] adopts random sampling and local features aggregation module which significantly speed up the processing of large scale point clouds. SPG [39] preprocesses the large scale point clouds as super graphs to learn per super-point semantics.

3 Hierarchical Parallel Group Convolutional Neural Network

In this section, we explain the architecture of our Hierarchical Parallel Group Convolution (HPGConv for short) which is shown in Figure 2 and Hierarchical Parallel Group Convolutional Neural Network(HPGCNN for short) which is shown in Figure 3.

3.1 Hierarchical Parallel Group Convolution

Our HPGconv consists of two group convolutions: the Hierarchical MLP Group Convolution (HMGConv for short) and the Hierarchical Graph Group Convolution (HGGConv for short).

Hierarchical MLP group convolution. In the HMGConv, the parameters of MLP are divided into respective M groups, and it can be denoted as $\mathbf{W}^k = \{\mathbf{W}^k_{11}, \mathbf{W}^k_{22}, ..., \mathbf{W}^k_{MM}\}$. In order to overcome the problem that information cannot communicate between the different-level feature maps of different groups of the standard group convolution and leverage the inter-group information more effectively, we hierarchically fuse feature maps of different groups. Specifically, we fuse the feature map $\mathbf{x}^k$ of $(m+1)^{th}$ group with the output $\mathbf{W}^k_{mm}\mathbf{x}^k$ of m^{th} group on the channel dimension after the feature map of the m^{th} group directly go through the $\mathbf{W}^k_{mm}$, and then feed the feature map into $\mathbf{W}^k_{(m+1)(m+1)}$. The output $\mathbf{x}^{k+1}$ of k^{th} layer HMGConv to capture independent single-point features is given as follows,

$$\begin{aligned}\mathbf{x}^{k+1}_M = diag(\mathbf{W}^k_{mm}{}^{M}_{m=1}) \times \mathbf{x}^k = \begin{bmatrix} \mathbf{W}^k_{11} & 0 & 0 & 0 \\ 0 & \mathbf{W}^k_{22} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \mathbf{W}^k_{MM} \end{bmatrix} \times \\ \begin{bmatrix} \mathbf{x}^k & 0 & 0 & 0 \\ 0 & \mathbf{x}^k = \mathbf{x}^k + \mathbf{W}^k_{11}\mathbf{x}^k & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \mathbf{x}^k = \mathbf{x}^k + \mathbf{W}^k_{(M-1)(M-1)}\mathbf{x}^k \end{bmatrix}\end{aligned} \tag{1}$$

where k denotes the k^{th} convolution layer. We fuse the features of all the groups sufficiently to facilitate information communication by stacking group convolutions together and then through shuffling the feature channels. As a result of information communication, each group of the updated groups contains information from the other groups. We carry out the channel shuffle $\mathbf{P}^T$ by performing matrix reconstruction and matrix transpose operations in turns.

$$\mathbf{y}_M = \mathbf{P}^T\mathbf{x}^{k+1}_M = R_P * T_P * R_P * \mathbf{x}^{k+1}_M \tag{2}$$

where R_P and T_P indicate the matrix reconstruction and the matrix transpose operations respectively.

Hierarchical graph group convolution. Although our HMGConv fully explores the discriminative independent single-point features, it lacks the ability to capture the local geometric features, so we propose the HGGConv. A local area is constructed by searching the k nearest points, neighboring the sampling point, to calculate edge features $\mathbf{y}_j$ and fusion features $\mathbf{y}_{ij}$:

$$\mathbf{y}_j = (\mathbf{x}_{ij} - \mathbf{x}_i), \mathbf{x}_i \in \mathbb{R}^F; \forall \mathbf{x}_{ij} \in Neighbors(\mathbf{x}_i) \tag{3}$$

$$\mathbf{y}_{ij} = (\mathbf{x}_i, \mathbf{x}_{ij} - \mathbf{x}_i), \mathbf{x}_i \in \mathbb{R}^F; \forall \mathbf{x}_{ij} \in Neighbors(\mathbf{x}_i) \tag{4}$$

Similarly, the parameters of the graph convolution are divided into G groups to extract the discriminative local geometric features in each group. We also hierarchically fuse the extracted high-level features of the previous group with the input features of the next group to enhance the interaction of feature maps between different groups,

$$\mathbf{x}_G^{k+1} = diag(\mathbf{W}_{gg\,g=1}^{k\ \ G}) \times \mathbf{x}^k = \begin{bmatrix} \mathbf{W}_{11}^k & 0 & 0 & 0 \\ 0 & \mathbf{W}_{22}^k & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \mathbf{W}_{GG}^k \end{bmatrix} \times \begin{bmatrix} \mathbf{x}_i^k & 0 & 0 & 0 \\ 0 & \mathbf{y}_j^k = \mathbf{y}_j^k + \mathbf{W}_{11}^k \mathbf{x}_i^k & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \mathbf{y}_{ij}^k = \mathbf{y}_{ij}^k + \mathbf{W}_{(G-1)(G-1)}^k \mathbf{y}_j^k \end{bmatrix} \tag{5}$$

Mentioned earlier, the channel shuffling facilitates information communication between different groups.

$$\mathbf{y}_G = \mathbf{P}^T \mathbf{x}_G^{k+1} = R_P * T_P * R_P * \mathbf{x}_G^{k+1} \tag{6}$$

Finally, we fuse the discriminative independent single-point features with the local geometric features to enhance the feature richness.

$$\begin{aligned} \mathbf{x}^{k+1} = \mathbf{y} = Concat(\mathbf{y}_M, \mathbf{y}_G) = Concat(\mathbf{P}^T(\mathbf{x}_M^{k+1} + \mathbf{x}_G^{k+1}) = \\ Concat(\mathbf{P}^T(diag(\mathbf{W}_{mm\,m=1}^{k\ \ M}) + diag(\mathbf{W}_{gg\,g=1}^{k\ \ G}) \times \mathbf{x}^k \end{aligned} \tag{7}$$

In summary, our HPGConv block can be formulated as

$$\mathbf{x}^{k+1} = \mathbf{x}^k(\mathbf{W}_M + \mathbf{W}_G), \tag{8}$$

where $\mathbf{W}_M$ and $\mathbf{W}_G$ denote the parameters of the HMGConv and the HGGConv. We let $\mathbf{W} = (\mathbf{W}_G + \mathbf{W}_M)$ be the composite convolution kernel, then we get the following formula,

$$\mathbf{x}^{k+1} = \mathbf{W}\mathbf{x}^k \tag{9}$$

which implies that an HPGConv block is equivalent to a regular convolution with the convolution kernel being the product of two sparse kernels. The computational complexity(FLOPs) and model complexity (parameters) of the HPGConv can be calculated by equations 10 and 11 respectively,

$$\begin{aligned} params &= \frac{F_{input} \times W}{g} \times \frac{F_{output}}{g} \times g \\ &= \frac{F_{input} \times F_{ouput} \times W}{g} \end{aligned} \tag{10}$$

$$FLOPs = \frac{N \times (1+k) \times F_{input} \times F_{output} \times W}{g} \tag{11}$$

Since we group the input channel and output channel separately, both of the input channel and the output channel are reduced to 1/g. Then, the increase of results will be limited by the fusion operation to only g times. Therefore, the final output of the parameters and FLOPs will be reduced to 1/g. Our HPGConv is an efficient and universal convolution operation with fewer redundant parameters that can enhance the encoding of more discriminative information to capture both the local geometric features and the independent single-point structural features at the same time for accomplishing accurate elusive shape. Our HPGConv can be integrated into multiple existing pipelines for point clouds processing easily.

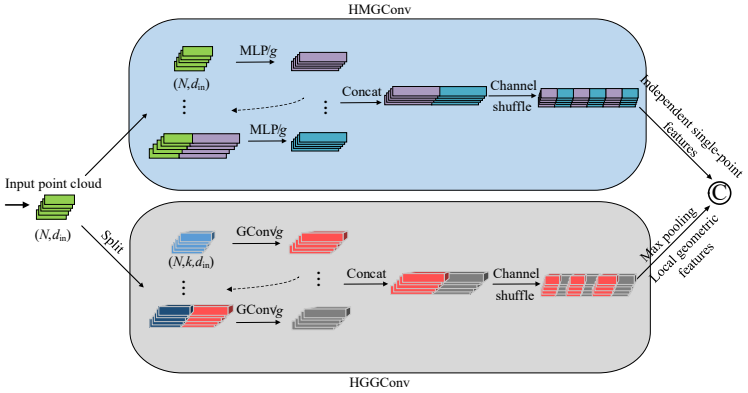

Fig. 2. HPGConv operation.

3.2 Hierarchical Parallel Group Convolutional Neural Network

Our Hierarchical Parallel Group Convolutional Neural Network(HPGCNN) architecture for point clouds classification and segmentation is shown in Figure 3. In addition to HPGConv, it also contains two components: (1) multi-semantic scale, (2) global average pooling with a fully connected layer.

Multi-Semantic Scale. Similar to PointNet++ [27], we adopt the Farthest Point Sampling (FPS) to iteratively select M, (M<N), points from the point clouds. Then, we use the KNN algorithm to build local areas by searching a fixed number of nearest neighboring point for each sampling point according to the sorted Euclidean distance in 3D space. In order to capture the multi-scale local geometric features, for each sample point in the point clouds, we construct multi-scale local area structure by finding the top $[K_1, ..., K_t, ..., K_T]$ nearest neighbors. Taking into account the fact that there is no information interaction between local areas of different scales, we introduce multi-semantic scale strategy

to input the high-level features extracted from the previous scale local area into the next scale local area to extend its receptive field. This allows the local areas of different scales to have larger receptive field which in turn leads to obtaining the higher semantic-level features that are beneficial to the acquirement of the multi-level contextual semantic information.

Global Average Pooling with A Fully Connected Layer. To achieve classification more efficiently, we first introduce the strategy of global average pooling with a fully connected layer instead of traditional three fully connected layer. Specifically, the average value of feature maps is invariant to the order of input points and we use use global average pooling as the symmetric function. One advantage of the global average pooling is that it combines the global information of the feature maps by calculating the average value which strengthens the correspondence between the feature maps and the category, and is closer to the semantic category information. Another advantage is that the removal of the two fully connected layers results in a significant reduction in the model complexity and the computational complexity, thus overfitting the overall structure is natively avoided at this layer, and the cumbersome dropout layer [40] is no longer needed. As a final tribute to the global average pooling, we can say that by summing out the spatial information, it is more robust to the spatial translations of the point clouds.

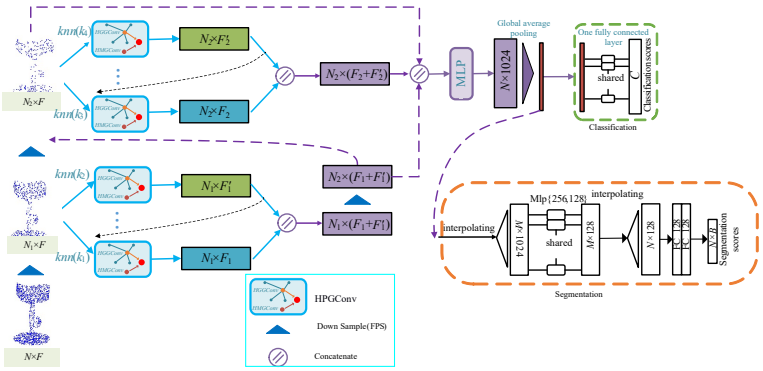

Fig. 3. Overview of HPGCNN.

4 Experiments

4.1 Shape recognition.

For the 3D point clouds recognition task, we carry out our experiments on standard public dataset ModelNet40 [10]. Table 1 shows the comparison results,

obtained out of the ModelNet40 dataset, between our model and the recent state-of-the-art methods in terms of both accuracy and complexity, HGGC represents hierarchical graph group convolution. Figure 1 shows the efficiency of our model compared to other state-of-the-art methods. As can be seen from Table 1, our model outperforms the advanced DGCNN approach by 0.4% in terms of accuracy and reduces the amount of parameters, FLOPs, forward time and model size by 44.4%, 18.8%, 24.9% and 45.2% respectively. The effectiveness of our HPGCNN is evident from the results. We can also observe that our strategy of replacing the traditional three fully connected layers with the global average pooling and one fully connected layer is effective because the strategy successfully reduces the model and the computation complexity while maintaining a considerable recognition accuracy. Although our HGGC achieves the best performance in model complexity and computational complexity, it only focuses on local geometric features, drowning independent single point features, and weakens feature richness, leading to a decline in recognition ability. It is also worth mentioning that combining the HPGConv with the Deep Convolution (DConv) produces better results. In Figure 4, we adopt T-distributed Stochastic Neighbor Embedding (T-SNE) to show that our HPGConv has the ability to extract more discriminative features, points with the same color belong to the same category. It can be seen that the extracted features by our HPGConv are much more discriminative than original point clouds and features extracted by DGCNN after training 250 epochs.

Table 1. Recognition results on the ModelNet40 dataset

Methods	mA (%)	OA (%)	params (Million)	size (MB)	FLOPs (Million)	time(ms)
VoxNet [11]	83.0	85.9	-	-	-	-
PointNet [25]	86.0	89.2	3.4	41.8	918.6	**24**
PN++ [26]	-	90.7	2.0	17.7	3136.0	163.2
KC-Net [41]	-	91.0	-	-	-	-
SpecGCN [42]	-	91.5	2.1	-	1112.0	11254.0
Kd-Net [43]	-	91.8	-	-	-	-
DGCNN [27]	90.2	92.2	1.8	22.1	3212.0	94.6
PCNN [30]	-	92.3	8.2	93.8	-	-
SpiderCNN [29]	90.7	92.4	-	36.9	-	-
PointCNN [28]	88.1	92.2	0.7	43.6	1682.0	-
Ours(HPGC-3FC)	90.1	92.4	1.6	22.9	2621	72.7
Ours(HGGC)	88.6	91.9	**0.5**	**6.5**	**771**	43.8
Ours(HPGC)	90.3	92.5	0.6	7.3	1079	59.6
Ours(HPGC+DC)	**90.4**	**92.6**	1.0	12.1	2609	71.0

Ablation study. We have also carried out experiments on the ModelNet40 dataset with various settings of our model taken into consideration. Table 2

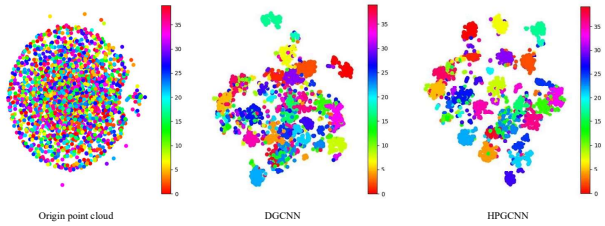

Fig. 4. Visualization of original point clouds and extracted discriminative 1024-dimensional features by HPGConv.

Table 2. Comparison of accuracy and complexity of different numbers of the groups

		OA	FLOPs	time	size	params
G=1	M=1	92.3	4039	92.2	24.0	1.8
G=2	M=2	92.6	2609	71.0	12.1	1.0
G=3	M=3	92.4	500	50.0	7.2	0.35

shows the performance of our model with different numbers of group of HMGConv and HGGConv. It can be observed that, when the numbers of group of HMGConv or the numbers of group of HGGConv increases till it reaches specified number and then decreases, our model has higher recognition accuracy OA and lower model complexity and computation complexity, this is because of the fact that the presence of multiple groups helps capture more useful discriminative information with fewer parameters. In the case of our model (G=2, M=2), the FLOPs, the parameters, the model size and forward time are reduced by 35.4%, 55.6%, 49.6% and 23.0% respectively and the OA increases by 0.3% when compared to no grouping operation(G=1, M=1), which verifies the effectiveness of our HPGConv. However, we notice that the presence of the groups in a much larger number (G=3, M=3) degenerates the performance of our model. The reason is probably that only limited useful feature information is encoded by each individual group when we split the feature channels to too many groups. Beside the aforementioned settings, we try out our model with other different settings and the experimental results are shown in Table 3. It can be seen that the performance of HPGCNN is better when we adopt hierarchical strategy. Because it can increase the information interaction between groups to take advantage of the inter-group information. Multi-semantic scale strategy is better than traditional multi-scale strategy, which can increase the receptive field of local point clouds of different scales to capture contextual features with higher level semantic information, help to recognize fine-grained details.

Table 3. Ablation study of our model

hierarchical	multi-scale	multi-semantic scale	OA(%)
			92.3
	√		92.4
		√	92.5
√	√	√	92.6

4.2 Semantic segmentation.

In order to test the effectiveness of our HPGCNN further for the fine-grained shape analysis of point clouds, we evaluate the performance of our model on the four public large scale semantic segmentation datasets ShapeNet Parts [44], S3DIS [45], vKITTI [46] and SemanticKITTI [47]. In all experiments, we implement the models with Tensorflow on one RTX 2080 GPU.

(1) Evaluation on ShapeNet Parts. Table 4 presents the quantitative results of different approaches on ShapeNet Parts. It can be seen that our HPGCNN has certain advantages in terms of the mIoU and FLOPs, specially the FLOPs is significantly lower than all other mainstream approaches. This because our HPGConv can significantly reduce the redundancy of the convolution kernel to encode more discriminative information.

Table 4. Comparison of segmentation accuracy of different approaches on the ShapeNet Parts dataset

Methods	mIoU(%)	FLOPs(Million)
PointNet [25]	83.7	17841
PointNet++ [26]	85.1	-
DGCNN [27]	85.1	9110
LDGCNN [48]	85.1	28656
HPGCNN(Ours)	**85.4**	**4527**

(2) Evaluation on S3DIS. Table 5, 6 and 7 show quantitative comparison of our HPGCNN with other state-of-the-art methods on large scale point clouds dataset, we can also see that our HPGCNN performs better when the tasks involved are large-scale point clouds processing. The OA of our HPGCNN when invoked on the S3DIS dataset reaches 90.0%, surpassing all the existing advanced approaches by a large margin and the mIoU is second only to KPconv. Notably, HPGCNN also achieves superior performance on six out of the twelve classes. The qualitative results are visualized in Figure 5, it is easy to observe that prediction results of our HPGCNN is closer to Ground Truth. Compared to the most advanced approach SSP+SPG, our HPGCNN effectively improves the ability to recognize fine-grained details of elusive small object categories, such

as the edges of the boards and the beams. Although global overall structural information and local geometric details of these elusive classes are very similar, to correctly recognize them requires a combination of global single point features and local geometric features.

Table 5. Comparison of segmentation accuracy of different approaches on the S3DIS dataset(6-fold cross validation)

Methods	OA(%)	mIoU(%)	ceil	floor	wall	beam	col	wind	door	chair	table	book	sofa	boad	clut.
PointNet [25]	78.5	47.6	88.0	88.7	69.3	42.4	23.1	47.5	51.6	42	54.1	38.2	9.6	29.4	35.2
MS+CU [49]	79.2	47.8	-	-	-	-	-	-	-	-	-	-	-	-	-
G+RCU [49]	81.1	49.7	90.3	92.1	67.9	44.7	24.2	52.3	51.2	47.4	58.1	39	6.9	30	41.9
PointNet++ [26]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
DGCNN [27]	84.4	56.1	-	-	-	-	-	-	-	-	-	-	-	-	-
3P-RNN [50]	86.9	56.3	-	-	-	-	-	-	-	-	-	-	-	-	-
RSNet [51]	-	56.5	92.5	92.8	78.6	32.8	34.4	51.6	68.1	60.1	59.7	50.2	16.4	44.9	52.0
SPG [39]	85.5	62.1	89.9	95.1	76.4	62.8	47.1	55.3	68.4	73.5	69.2	63.2	45.9	8.7	52.9
LSANet [52]	86.8	62.2													-
PointCNN [28]	88.1	65.4	94.8	97.3	75.8	63.3	51.7	58.4	57.2	71.6	69.1	39.1	61.2	52.2	58.6
PointWeb [53]	87.3	66.7	-	-	-	-	-	-	-	-	-	-	-	-	-
ShellNet [54]	87.1	66.8	-	-	-	-	-	-	-	-	-	-	-	-	-
HEPIN [55]	88.2	67.8	-	-	-	-	-	-	-	-	-	-	-	-	-
KPConv [56]	-	**70.6**	93.6	92.4	**83.1**	63.9	54.3	**66.1**	**76.6**	57.8	64	**69.3**	**74.9**	**61.3**	60.3
SSP+SPG [57]	87.9	68.4	91.7	95.5	80.8	62.2	**54.9**	58.8	68.4	78.4	69.2	64.3	52.0	54.2	59.2
RandLA-Net [38]	87.2	68.5	-	-	-	-	-	-	-	-	-	-	-	-	-
Ours	**90.3**	69.2	**95.4**	**97.5**	81.2	**73.7**	44.8	55.6	71.3	**86.5**	**76.3**	68.9	30.0	52.3	**66.0**

(3) Evaluation on vKITTI. As shown in Table 6, our HPGCNN outperforms all existing advanced methods in terms of both mIoU and OA by a large margin on vKITTI dataset. Compared with the most advanced approach SSP+SPG, the OA and mIoU of HPGCNN improves by 7.4% and 9.1% respectively. Figure 6 illustrates our qualitative visualization of six samples by randomly selecting one scene from each of the six city scene sequences. It can be seen that our HPGCNN has achieved surprising segmentation results out of all the six scene sequences. In particular, our approach can correctly recognize the boundary between the two elusive categories terrain and road, although the overall shape of these items is greatly similar and they differ in terms of their local details. The reason is that our HPGConv can capture both global single-point structural features and fine-grained local geometric features to complement each other, making the extracted features more sufficient.

(4) Evaluation on SemanticKITTI. In Table 7, we compare HPGCNN with other state-of-the-art methods on SemanticKITTI dataset. Our approach takes fewer points as input, outperforms others by a large margin with fewer parameters, and obtains the best results in 6 of 19 categories. Note the high quality results on our method in relevant elusive classes such as fence, as well as in challenging classes such as motorcyclist. Furthermore, Figure 7 illustrates our qualitative visualization of four samples on the validation split. It can be seen that our HPGCNN predict perfectly. The reason is that our HPGConv is designed to encode more of the discriminative fine-grained independent single-point features and the local geometric features simultaneously, which can en-

Table 6. Comparison of segmentation accuracy of different approaches on the vKITTI dataset with 6-fold cross validation

Methods	OA(%)	mIoU(%)
PointNet [25]	79.7	34.4
Engelmann et al. in [58]	79.7	35.6
G+RCU [49]	80.6	36.2
3P-RNN [50]	87.8	41.6
SSP+SPG [57]	84.3	52.0
Ours	**91.7**	**61.1**

hance feature richness, and that the multi-semantic scale strategy can exploit the contextual fine-grained information along with the rich semantics.

Table 7. Comparison of segmentation accuracy of different approaches on the SemanticKITTI dataset [47]. Only the recent published methods are compared and all scores are obtained from the online single scan evaluation track. Accessed on 23 June 2020

Methods	input	mIoU(%)	Params	road	sidewalk	parking	other-ground	building	car	truck	bicycle	motorcycle	other-vehicle	vegtation	trunk	terrain	person	bicyclist	motorcyclist	fence	pole	traffific-sign
PointNet [25]	50k pnt	14.6	3.00	61.6	35.7	15.8	1.4	41.4	46.3	0.1	1.3	0.3	0.8	31.0	4.6	17.6	0.2	0.2	0.0	12.9	2.4	3.7
SPG [39]		17.4	**0.25**	45.0	28.5	1.6	0.6	64.3	49.3	0.1	0.2	0.2	0.8	48.9	27.2	24.6	0.3	2.7	0.1	20.8	15.9	0.8
SPLATNet [16]		18.4	0.8	64.6	39.1	0.4	0.0	58.3	58.2	0.0	0.0	0.0	0.0	71.1	9.9	19.3	0.0	0.0	0.0	23.1	5.6	0.0
PointNet++ [26]		20.1	6	72.0	41.8	18.7	5.6	62.3	53.7	0.9	1.9	0.2	0.2	46.5	13.8	30.0	0.9	1.0	0.0	16.9	6.0	8.9
TangentConv [21]		40.9	0.4	83.9	63.9	33.4	15.4	83.4	90.8	15.2	2.7	16.5	12.1	79.5	49.3	58.1	23.0	28.4	8.1	49.0	35.8	28.5
RandLA-Net [38]		50.3	0.95	90.4	67.9	56.9	15.5	81.1	**94.0**	**42.7**	19.8	21.4	**38.7**	78.3	60.3	59.0	**47.5**	48.8	4.6	49.7	44.2	38.1
SqueezeSeg [59]	64*2048pixels	29.5	1	85.4	54.3	26.9	4.5	57.4	68.8	3.3	16.0	4.1	3.6	60.0	24.3	53.7	12.9	13.1	0.9	29.0	17.5	24.5
SqueezeSegV2 [60]		39.7	1	88.6	67.6	45.8	17.7	73.3	81.8	13.4	18.5	17.9	14.0	71.8	35.8	60.2	20.1	25.1	3.9	41.1	20.2	36.3
DarkNet21Seg [47]		47.4	25	91.4	74.0	57.0	26.4	81.9	85.4	16.6	**26.2**	26.5	15.6	77.6	48.4	63.7	31.8	33.6	4.0	52.3	36.0	50.0
DarkNet53Seg [47]		49.9	50	**91.8**	**74.6**	**64.8**	27.9	84.1	86.4	25.5	24.5	**32.7**	22.6	78.3	50.1	64.0	36.2	33.6	4.7	55.0	38.9	**52.2**
PointASNL [61]	**8k pnt**	46.8	-	87.4	74.3	24.3	1.8	83.1	87.9	39.0	0.0	25.1	29.2	84.1	52.2	**70.6**	34.2	**57.6**	0.0	43.9	**57.8**	36.9
HPGCNN(Ours)	10k pnt	**50.5**	0.8	89.5	73.6	58.8	**34.6**	**91.2**	93.1	21.0	6.5	17.6	23.3	**84.4**	**65.9**	70.0	32.1	30.0	**14.7**	**65.5**	45.5	41.5

5 Conclusion

In this paper, we present hierarchical parallel group convolutional neural network that, in addition to reducing the redundancy problem of the standard convolution operation, exploits the local and global representations in the depth and width of the network so that they can be complementary to each other and enhance the ability to recognize elusive classes. Furthermore, the strategy, we put forward, of combining the global average pooling with a fully connected layer to complete the classification task also proved effective and successful in further reducing the complexity as well as in avoiding overfitting. A multi-semantic scale strategy is also introduced to effectively capture context fine-grained information with higher level semantics. Extensive experiments on multiple benchmarks demonstrate the high efficiency and the state-of-the-art performance of our approach.

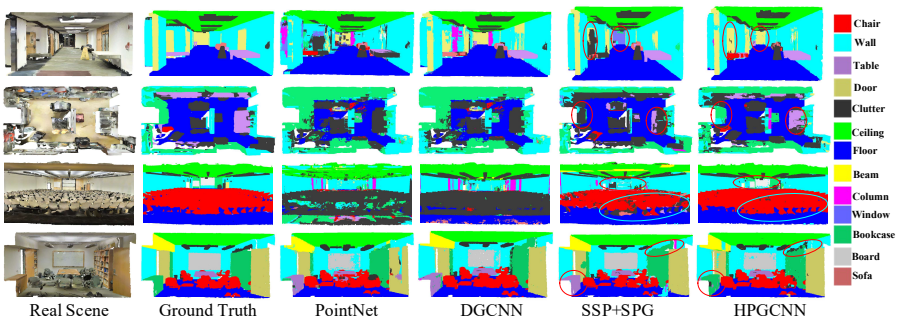

Fig. 5. Visualization of semantic segmentation results on the S3DIS dataset.

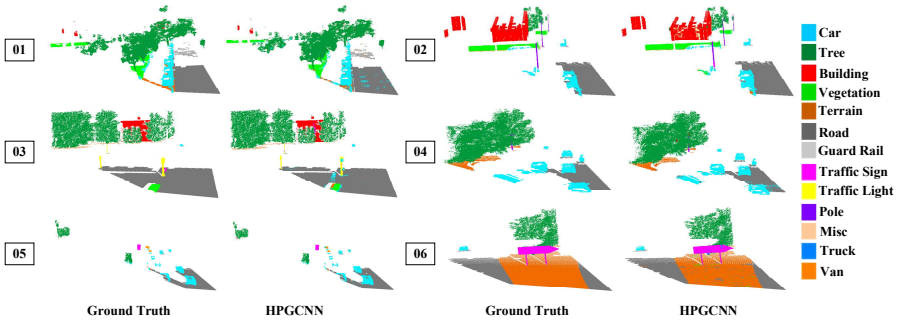

Fig. 6. Visualization of semantic segmentation results on the vKITTI dataset.

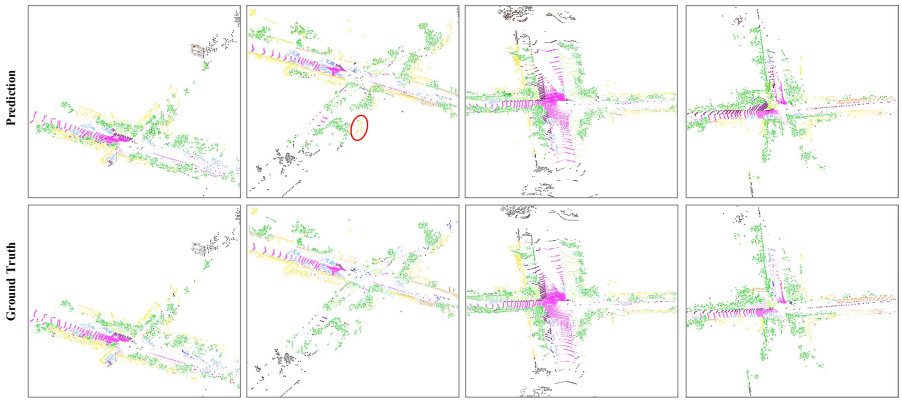

Fig. 7. Visualization of semantic segmentation results on the SemanticKITTI [47]. Red boxes show the failure cases.

References

1. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. arXiv preprint arXiv:1901.09346 (2017)
2. Liu, Z., Chen, H., Di, H., Tao, Y., Gong, J., Xiong, G., Qi, J.: Real-time 6d lidar slam in large scale natural terrains for ugv. IEEE Intelligent Vehicles Symposium (IV) (2018) 662–667
3. Rusu, R.B., Marton, Z., Blodow, N., Dolha, M.E., Beetz, M.: Towards 3d point cloud based object maps for household environments. Robotics and Autonomous Systems **56** (2008) 927–941
4. Biswas, J., Veloso, M.: Depth camera based indoor mobile robot localization and navigation. In Robotics and Automation (ICRA) (2012) 1697–1702
5. Zhou, Y., Tuzel, O.: Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 4490–4499
6. Golovinskiy, A., Kim, V.G., Funkhouser, T.: Shape-based recognition of 3d point clouds in urban environments. In Computer Vision (2009) 2154–2161
7. Golovinskiy, A., Kim, V.G., Funkhouser, T.: Shape-based recognition of 3d point clouds in urban environments. (2009)
8. Chua, C.S., Jarvis, R.: Point signatures: A new representation for 3d object recognition. International Journal of Computer Vision **25** (1997) 63–85
9. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms for 3d registration. In ICRA (2009) 1848–1853
10. Zhirong Wu, Shuran Song, A.K.: 3d shapenets: a deep representation for volumetric shape modeling. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015) 1912–1920
11. Maturana, D., Scherer, S.: Voxnet: a 3d convolutional neural network for real-time object recognition. 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2015) 922–928
12. Yizhak, B.S., Michael, L., Anath, F.: 3dmfv: 3d point cloud classification in real-time using convolutional neural network. IEEE Robotics Automation Letters (2018) 3145–3152
13. Xavier Roynard, J.E.D.: Classification of point cloud scenes with multi scale voxel deep network. arXiv preprint arXiv, 1804.03583 (2018)
14. Riegler, G., Ulusoy, A.O., Geiger, A.: Octnet: Learning deep 3d representations at high resolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 6620–6629
15. Graham, B., Engelcke, M., Der Maaten, L.V.: 3d semantic segmentation with submanifold sparse convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 9224–9232
16. Su, H., Jampani, V., Sun, D., Maji, S., Kalogerakis, E., Yang, M., Kautz, J.: Splatnet: sparse lattice networks for point cloud processing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 2530–2539
17. Su, H., Maji, S., Kalogerakis, E., Learnedmiller, E.: Multi-view convolutional neural networks for 3d shape recognition. Proceedings of the IEEE International Conference on Computer Vision (2015) 945–953
18. Alexandre Boulch, B.L.S.: Unstructured point cloud semantic labeling using deep segmentation networks. In Proceedings of the Workshop on 3D Object Retrieval (2017)

19. Lawin, F.J., Danelljan, M., Tosteberg, P., Bhat, G., Khan, F.S., Felsberg, M.: Deep projective 3d semantic segmentation. In International Conference on Computer Analysis of Images and Patterns (2017) 95–107
20. Feng, Y., Zhang, Z., Zhao, X., Ji, R., Gao, Y.: Gvcnn: group-view convolutional neural networks for 3d shape recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 264–272
21. Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.: Tangent convolutions for dense prediction in 3d. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 3887–3896
22. Bronstein, M.M., Bruna, J., Lecun, Y., Szlam, A., Vandergheynst, P.: Geometric deep learning: going beyond euclidean data. IEEE Signal Processing Magazine **34** (2017) 18–42
23. Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J.: 3d object recognition in cluttered scenes with local surface features: a survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **36** (2014) 2270–2287
24. Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J.: An integrated framework for 3d modeling, object detection, and pose estimation from point clouds. IEEE Transactions on Instrumentation and Measurement **64** (2015) 683–693
25. Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J.: Pointnet: deep learning on point sets for 3d classification and segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 77–85
26. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: deep hierarchical feature learning on point sets in a metric space. Advances in Neural Information Pprocessing Systems (2017)
27. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics **38** (2019) 146
28. Li Y, Bu R, S.M.: Pointcnn: convolution on x-transformed points. Advances in Neural Information Processing Systems (2018) 820–830
29. Xu, Yifan, F.T.X.M.Z.L.Q.Y.: Spidercnn: deep learning on point sets with parameterized convolutional filters. Proceedings of the European Conference on Computer Vision (ECCV) (2018) 87–102
30. Atzmon, M., Maron, H., Lipman, Y.: Point convolutional neural networks by extension operators. arXiv preprint arXiv, 1803.10091 **37** (2018) 71
31. Liu, Y., Fan, B., Xiang, S., Pan, C.: Relation shape convolutional neural network for point cloud analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019) 8895–8904
32. Zhang, Y., Rabbat, M.: A graph cnn for 3d point cloud classification. IEEE International Conference on Acoustics (2018) 6279–6283
33. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In Advances in Neural Information Processing Systems, (2016) 3844–3852
34. Yi, L., Su, H., Guo, X., Guibas, L.J.: Syncspeccnn: synchronized spectral cnn for 3d shape segmentation. In Conference on Computer Vision and Pattern Recognition (2017) 6584–6592
35. Masci, J., Boscaini, D., Bronstein, M.M., Vandergheynst, P.: Geodesic sonvolutional neural networks on riemannian manifolds. In Proceedings of the IEEE International Conference on Computer Vision Workshops (2015) 832–840
36. Simonovsky, M., Komodakis, N.: Dynamic edge-conditioned filters in convolutional neural networks on graphs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)

37. Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., Bronstein, M.M.: Geometric deep learning on graphs and manifolds using mixture model cnn. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 5425–5434
38. Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A.: Randla-net: efficient semantic segmentation of large-scale point clouds. arXiv preprint arXiv (2019)
39. Landrieu, L., Simonovsky, M.: Large-scale point cloud semantic segmentation with superpoint graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 4558–4567
40. Wang, C., Samari, B., Siddiqi, K.: Local spectral graph convolution for point set feature learning. In Proceedings of the European Conference on Computer Vision (ECCV) (2018) 56–66
41. Shen, Y., Feng, C., Yang, Y., Tian, D.: Mining point cloud local structures by kernel correlation and graph pooling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018) 4548–4557
42. Wang, C., Samari, B., Siddiqi, K.: Local spectral graph convolution for point set feature learning. In Proceedings of the European Conference on Computer Vision (ECCV) (2018) 52–66
43. Klokov R, L.V.: Escape from cells: deep kd-networks for the recognition of 3d point cloud models. Proceedings of the IEEE International Conference on Computer Vision (2017) 863–872
44. Yi, L., Kim, V.G., Ceylan, D., Shen, I., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., Guibas, L.J.: A scalable active framework for region annotation in 3d shape collections. ACM Transactions on Graphics (TOG) **35** (2016) 210
45. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016) 1534–1543
46. Gaidon, A., Wang, Q., Cabon, Y., Vig, E.: Virtual worlds as proxy for multi-object tracking analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
47. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: Semantickitti: a dataset for semantic scene understanding of lidar sequences. In IEEE International Conference on Computer Vision (2019) 9297–9307
48. Zhang, K., Hao, M., Wang, J., De Silva, C.W., Fu, C.: Linked dynamic graph cnn: learning on point cloud via linking hierarchical features. arXiv: Computer Vision and Pattern Recognition (2019)
49. Engelmann, F., Kontogianni, T., Hermans, A., Leibe, B.: Exploring spatial context for 3d semantic segmentation of point clouds. IEEE International Conference on Computer Vision Workshop (2017) 716–724
50. Ye, X., Li, J., Huang, H., Du, L., Zhang, X.: 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In Proceedings of the European Conference on Computer Vision (ECCV) (2018) 415–430
51. Huang, Q., Wang, W., Neumann, U.: Recurrent slice networks for 3d segmentation of point clouds. Proceedings of the IEEE International Conference on Computer Vision (2018) 2626–2635
52. Chen, L., Li, X., Fan, D., Cheng, M., Wang, K., Lu, S.: Lsanet: feature learning on point sets by local spatial attention. arXiv: Computer Vision and Pattern Recognition (2019)

53. Zhao, H., Jiang, L., Fu, C., Jia, J.: Pointweb: enhancing local neighborhood features for point cloud processing. Proceedings of the IEEE International Conference on Computer Vision (2019) 5565–5573
54. Zhang, Z., Hua, B., Yeung, S.: Shellnet: efficient point cloud convolutional neural networks using concentric shells statistics. Proceedings of the IEEE International Conference on Computer Vision (2019) 1607–1616
55. Li Jiang, Hengshuang Zhao, S.L.: Hierarchical pointedge interaction network for point cloud semantic segmentation. In ICCV (2019) 1607–1616
56. Thomas, H., Qi, C.R., Deschaud, J., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: flexible and deformable convolution for point clouds. arXiv: Computer Vision and Pattern Recognition (2019)
57. Landrieu, L., Boussaha, M.: Point cloud oversegmentation with graph-structured deep metric learning. arXiv: Computer Vision and Pattern Recognition (2019)
58. Engelmann, F., Kontogianni, T., Schult, J., Leibe, B.: Know what your neighbors do: 3d semantic segmentation of point clouds. In ECCV (2018) 395–409
59. Wu, B., Wan, A., Yue, X., Keutzer, K.: Squeezeseg: convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In ICRA (2018) 1887–1893
60. Wu, B., Zhou, X., Zhao, S., Yue, X., Keutzer, K.: Squeezesegv2: improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. arXiv: Computer Vision and Pattern Recognition (2018)
61. Yan, X., Zheng, C., Li, Z., Wang, S., Cui, S.: Pointasnl: robust point clouds processing using nonlocal neural networks with adaptive sampling. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)