Computers & Graphics (2020)

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Attention-based relation and context modeling for point cloud semantic segmentation

Zhiyu Hu[a,1], Dongbo Zhang[a,1], Shuai Li[a,b], Hong Qin[c,*]

[a] *State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China*
[b] *Beijing Advanced Innovation Center for Biomedical Engineering, Beihang University, China*
[c] *Department of Computer Science, Stony Brook University, USA*

ARTICLE INFO

Article history:
Received May 7, 2020

Keywords:
Point cloud,
Deep learning,
Semantic segmentation,
Contextual fusion

ABSTRACT

Semantic segmentation of point cloud is a fundamental problem in scene-level understanding. Despite advancement in recent years by leveraging capabilities of Neural Networks and massive labeling datasets available, providing fine-grained semantic segmentation for point cloud is still challenging, given the fact that point cloud is usually unstructured, unordered and sparse. In this paper, we achieve semantic point cloud labeling by adaptively exploring semantic relation and aggregating contextual information between points. Specifically, we first introduce an attention-based local relation learning module for collecting local features, which can capture semantic relation in a manner of anisotropy. And we then design a novel context aggregation module guided by multi-scale supervision to obtain long-range dependencies between semantically-correlated points and enhance the distinctive ability of points in feature space. In addition, a gated propagation strategy is adopted instead of skip links to conditionally concatenate local point features in different layers. We empirically evaluate our method on public benchmarks (S3DIS and ShapeNetPart), and demonstrate our performance is on par or better than state-of-the-art methods.

1. Introduction

Semantic segmentation of point cloud plays an important role in scene understanding [1], and has shown extensive usage in many down-streaming applications including robot [2], autonomous driving [3], virtual/augmented reality [4], to name a few. Given a point cloud, semantic segmentation is to accurately assign a semantic label for each point. Different from 2D images, directly applying convolution kernels on point cloud is ill-suited due to its intrinsic irregular properties. Although recent works have achieved remarkable success, fine-grained point cloud semantic segmentation is still challenging to the research community.

Learning-based methods have made measurable progress in 3D semantic segmentation task. Maturana et al. [5] directly voxelized 3D data and extended convolution kernels from 2D to 3D to learn a compact representation. Limited by hardware resources, voxelized modality can not represent 3D data with high-resolution. To alleviate resource consumption, some methods [6, 7, 8] solve it by skipping empty spaces generated by voxelization. An alternative solution is to express 3D data with multi-view representation and take advantages of existing powerful 2D network architectures to learn deep representation [9, 10, 11]. Nevertheless, as irreversible modalities, voxelization and multi-view representations still inevitably suffer from considerable loss of information. With the burgeoning development of scanning facilities especially the consumer-level depth sensors, point cloud as a raw representation of 3D data has attracted more and more research interests in recent years. To directly apply neural network on 3D point cloud, Qi et al. [12]

*Corresponding author:
e-mail: qin@cs.stonybrook.edu (Hong Qin)
[1] These two authors contribute equally to this research work.

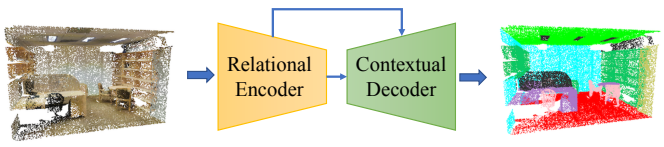


Fig. 1. The pipeline of our method. We handle point cloud semantic segmentation by a hierarchical encoder-decoder architecture For encoding part, our relational encoder attempts to learn the semantic relation between neighbouring points. For decoding part, our contextual decoder flexibly captures long-range dependencies between points even far away in spatial location.

first proposed a novel network architecture, named PointNet, to extract features for each point individually. However, the learned point-wise features fall short of capturing local structural information. Following PointNet, a lot of network architectures are proposed to learn deep features in a hierarchical way, such as PointNet++ [13] and PointCNN [14], with the purpose of enhancing the ability of modeling local structure. Although these methods have achieved favourable performance, they fail to explore the semantic relation and contextual information between points, which is crucial for 3D semantic segmentation task.

In this paper, we handle point cloud semantic segmentation by exploring semantic relation between neighbouring points and aggregating contextual information via a hierarchical encoder-decoder architecture (see Fig. 1). To achieve this, there are two main issues that should be thoughtfully considered for our point cloud semantic segmentation network architecture : 1) how to adaptively explore semantic relation inherent in point cloud; 2) how to selectively aggregate contextual information in both short and long spatial range. On one hand, to learn representative local feature in encoding stage, we propose an attention-based local relation learning module, which is guided by spatial information and feature attributes, to adaptively learn semantic relation between points in local neighborhood. On the other hand, to further enhance the distinguishing ability of features, a multi-scale context-guided aggregation module based on self-attention mechanism is designed to aggregate contextual information in the feature space, thus capturing long-range dependencies between points and allowing more expressive feature representation. In contrast to representing point cloud by each point itself (XYZ coordinates only), we also take its local geometric information into account to enrich point representation so that each point is able to be aware of its local structural information. Besides, we adopt a gated propagation strategy with the purpose of filtering irrelevant and redundant information when propagating low-level features from encoder to decoder.

Benefiting from above modules, our method adequately explores semantic relation and contextual information between points in both short and long spatial range. Comprehensive experiments are elaborately conducted to demonstrate the performance of our proposed method. In summary, our main contributions can be summarized as follows:

1. We propose an attention-based local relation learning module for local feature pooling, which dynamically explores semantic relation in an anisotropic way;
2. We design a novel context-guided aggregation module guided by multi-scale supervision to further enhance the distinguishing ability of points in feature space;
3. We employ a gated propagation strategy instead of skip links between encoder and decoder to flexibly filter out irrelevant and redundant information.

2. Related Work

In this section, we briefly review the state-of-the-art learning-based approaches to analyzing 3D data in term of semantic segmentation.

2.1. 3D modalities

Leveraging deep learning techniques for 3D data semantic segmentation, one straightforward choice is to extend 2D convolution kernels on voxel-based modality [5, 10, 15]. However, limited by computational and memory resource, 3D convolution kernel fails to tackle high-resolution point clouds and is inefficient since voxel-based modality contains lots of empty space. To alleviate this, Riegler et al. [6] proposed OctNet, which resorted to an octree-based 3D representation, to learn deep 3D representation at high resolutions. The key idea of OctNet is to only focus computations on the relevant regions by adaptively representing 3D space with a set of unbalanced octrees. Following OctNet, advanced 3D convolutional neural networks (CNNs), O-CNN [7] and adaptive O-CNN [8] were designed to only operate 3D CNNs on the octants occupied by 3D data.

Compared with voxel-based modality, an alternative choice is to represent 3D data with multi-view modality. Taking advantages of the powerful 2D CNN architectures, there are existing methods attempting to learn deep 3D representation with multi-view modality [9, 10, 11]. The core insight of their methods is to convert the problem of 3D deep representation into multi-view 2D deep representation. Although these above methods have shown remarkable progress in terms of 3D deep representation, voxel-based and multi-view modalities, as intermediate representations, are inevitably accompanied by loss of geometric information.

2.2. Deep learning on point cloud

Compared to voxel-based and multi-view modalities, point cloud holds the advantage of simplicity in representation and are therefore suitable for exploiting learning-based methods. Qi et al. [12] proposed the pioneering network architecture, named PointNet, which can directly consume raw point cloud. The core idea is to use multi-layer perceptrons (MLPs) to uplift low-level spatial location features to high-dimensional space, where points of different categories can be easily distinguished. Although PointNet achieved measurable progress in semantic segmentation, learning features for each point individually prevents it from capturing local structural information, which is

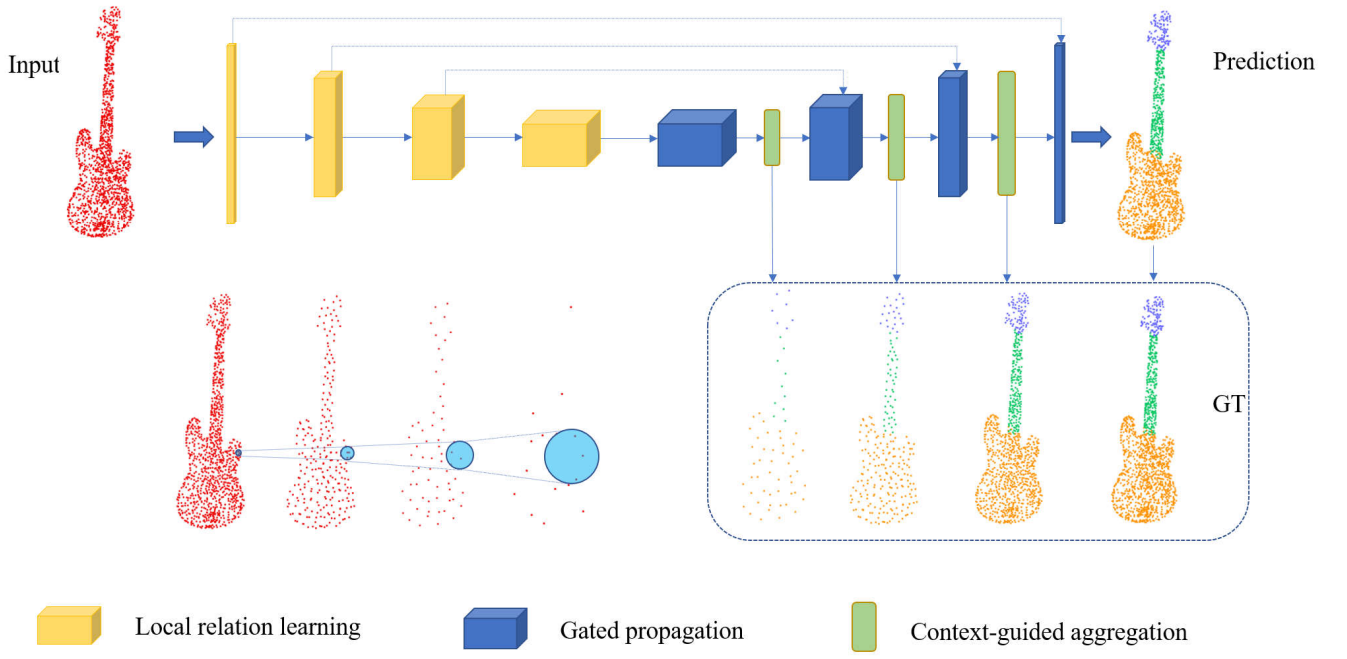


Fig. 2. The network architecture of our method. We employ a hierarchical encoder-decoder architecture, which mainly consists of local relation learning modules, gated propagation modules and context-guided aggregation modules. Our network is trained end-to-end and guided by the multi-scale loss.

crucial for semantic segmentation task. To address this, a subsequent version, PointNet++ [13], was introduced to characterize local structure in a hierarchical way. But still, it treats points in local neighborhood independently.

Inspired by PointNet/PointNet++, there have been a lot of advanced network architectures being proposed for various vision tasks, i.e., object/scene classification and segmentation [14, 16, 17, 18, 19, 20, 21, 22], object detection [3, 23, 24] and properties estimation [25, 26], etc. To overcome the irregularity of point cloud, Li et al. [14] proposed a novel architecture, named PointCNN, which aimed to compensate desertion of shape information and variance to point ordering when directly applying convolution kernels on point cloud. Building upon bilateral convolution layers [27], Su et al. [17] introduced a generic and flexible neural network architecture which mapped points into a high-dimensional sparse lattice to capture spatial patterns between neighboring points in a flexible manner. Komarichev et al. [20] proposed an annular convolution operator which specified the ring-shaped structures and directions in the computation to better capture the local neighborhood geometry of each point. Although these methods have achieved decent performance, to the best of our knowledge, none of them take into account both local relation and global context, which is important for exploring semantic information. In this paper, we aim to exploit the relation and context information for point cloud semantic segmentation.

2.3. Relation and context modeling

Recently, some methods were proposed to learn relation between points in local neighborhood [28, 19, 29, 30]. PointNet++ aggregated point features by max-pooling local features, but it did not touch on relation learning between local points. To better capture local geometric features, Wang et al. [28] proposed an Edge-Conv operation to dynamically pool features for a target point from its neighbors in a manner of weighted sum. At the same time, Liu et al. [29] and Wu et al. [19] both learned feature weights from extracted spatial information to get shape-aware representation. Later, Lan et al. [30] modeled the relation between points in a local neighborhood via vector decomposition to preserve the geometric structure in Euclidean space throughout the feature extraction hierarchy. Despite steady progress achieved by these methods in capturing local patterns, the learned local patterns only depict the relation between the target point and its neighbors, without involving interactions between all the points in local neighborhood. Therefore, to address this, our method learns deeper semantic relation between points in local neighborhood.

Context plays an important role in computer vision tasks, especially for scene parsing and understanding tasks. By exploring the proximity of certain point to other points in semantic space, semantically-correlated information can be harvested even if points are far away from each other, while irrelevant information can be suppressed even if points stay close. Recently, a lot of methods were proposed to exploit the context for image segmentation [31, 32, 33, 34]. As for point cloud, to the best of our knowledge, there have been fewer methods to dealing with context information. Xie et al. [35] proposed ShapeContextNet to learn point features in three similar steps to PointNet++, but the context it learned is limited to local regions. Jiang et al. [36] introduced an edge branch into PointNet++ architecture, which explored semantic similarity between point and its neighbors under strong supervision, but still, the contextual information is collected only from local regions. Inspired by [37], Wang et al. [38] applied spatial and channel attention on point cloud,

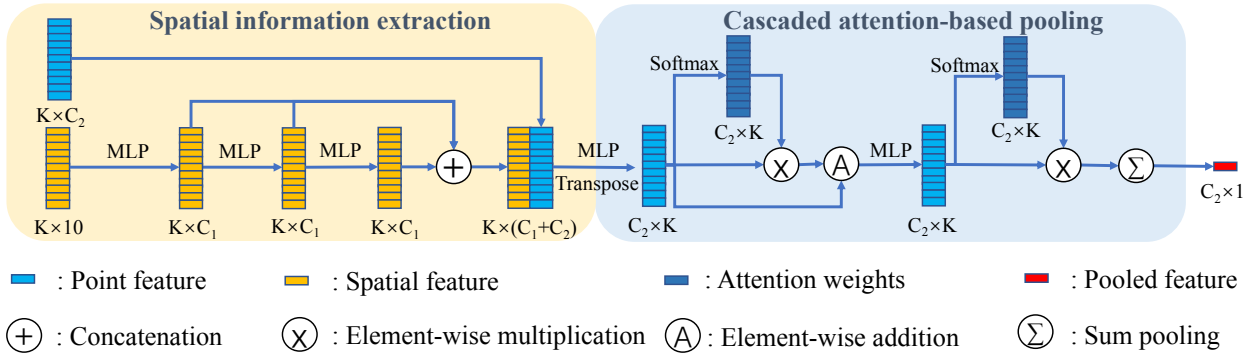


Fig. 3. Illustration of local relation learning module.

which brought about considerable computational cost. Different from these methods, our method learns contextual information in a multi-scale fashion, which can be viewed as implicitly global context but it reduces a large amount of computational consumption.

3. Method

The architecture of our proposed method is demonstrated in Fig. 2. The core insight of our method is to adaptively explore semantic relation and aggregate contextual information between points. To do this, we design our network as a hierarchical encoder-decoder architecture. In the encoder module, we adopt PointNet++ as our backbone to obtain a compact representation. Specifically, we use an attention-based pooling instead of original max-pooling operators in PointNet++ to identify local structure in an anisotropic way. As for the decoder module, we design a context-guided aggregation module cooperating with multi-scale supervision to adaptively grab global contextual information. In addition, to filter redundant information between encoder and decoder modules, we adopt a gated propagation strategy instead of the original skip links. All the details of our encoder and decoder modules will be shown in Section 3.1 and Section 3.2. And the gated propagation will be covered in Section 3.3. Finally in Section 3.4, we give the formulation of the loss function.

3.1. Local relation learning

Given 3D point cloud $P = \{p_1, ..., p_n \,|\, p_i \in R^3, i = 1, ..., n\}$, where n is the cardinality of P, we first we first sample a subset of points $P_S = \{p_1, p_2, ..., p_s\}$ by farthest point sampling. For each sampled point $p_i \in P_S$, we then utilize ball query to find the neighbors of p_i (p_i itself included), which form a customized local neighborhood $N(p_i)$ for p_i.

Spatial information extraction. For each local neighborhood centered at a certain point p_i, we learn the relation between p_i and its neighbors from local spatial information. Specifically, for p_i and one of its neighbor $p_j \in N(p_i)$, we follow RS-CNN [29] to explicitly create a vector $h_{i,j} \in R^{10}$ as below:

$$h_{i,j} = \{x_i, x_j, x_i - x_j, \|x_i - x_j\|\}, \quad (1)$$

where x_i, x_j denote the 3-D coordinates of p_i and p_j, and $\|\cdot\|$ denotes the Euclidean norm. Such a 10-D vector $h_{i,j}$ explicitly contains the spatial and geometric information so that relation between p_i and p_j can be implicitly encoded by MLPs, which is illustrated by *spatial information extraction* module in Fig. 3. It is worth mentioning that the relation learning process is carried out over all the neighbors of p_i in the local neighborhood, but here we only take one neighbor p_j to explain for the sake of brevity. The MLPs map $h_{i,j}$ to high-level features with C_1 dimension and we concatenate these features together, which is then concatenated with the point feature (either raw RGB and normalized coordinates or learned intermediate feature, with C_2 dimension) to get an enhanced feature $\hat{h}_{i,j} \in R^{C_1+C_2}$.

Cascaded attention-based pooling. Suppose there are K points in the local neighborhood of p_i, in which case the feature $\hat{h}_{i,j}$ can be expanded to $H_i \in R^{K\times(C_1+C_2)}$ when all the points in $N(p_i)$ are considered. To generate attention weights for K points, the expanded feature H_i is first passed through a MLP layer and get transposed, then we apply softmax operation to it so that attention weights for K points add to 1. After that, element-wise multiplication is performed on the feature and attention weights, which gives us a reassigned feature. It is worth mentioning that this attention-based feature reassignment allows the interaction between points and thus captures richer information. We duplicate this process to better learn the feature distribution of neighbors and guarantee adequate interaction between points. At last, we pool the features by summation to get the output feature $F_{out} \in R^{C_2}$, which is the customized local feature for p_i. The cascaded attention-based pooling module in Fig. 3 illustrates the pooling process.

3.2. Multi-scale context-guided aggregation

Inspired by the non-local neural networks[39], we leverage a similar structure to model the context. Based on self-attention mechanism, the non-local operation in [39] was originally designed to capture the long-range spatial-temporal dependencies in images and videos. By computing a affinity matrix, which depicts the similarity between any two data points, non-local networks learn a query-specific context and apply this learned context to each query-point to capture long-range dependencies between these points and eventually augment the feature representations. However, despite its success, non-local operation

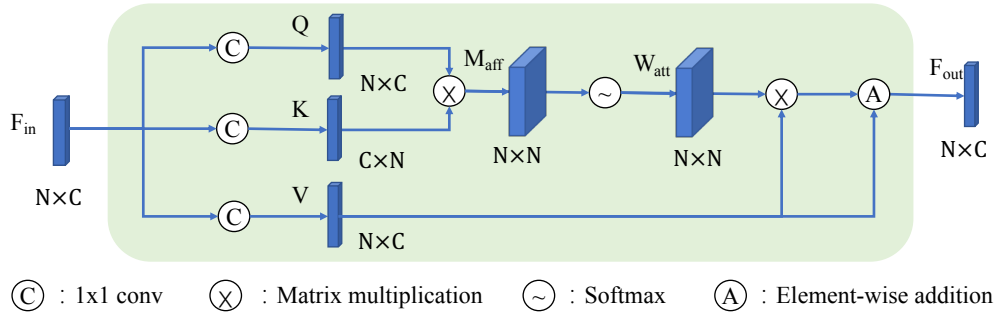


Fig. 4. Architecture of the context-guided aggregation module.

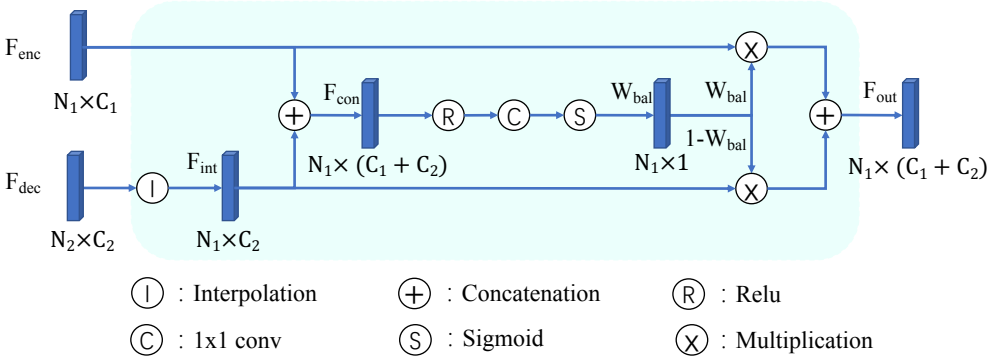


Fig. 5. Architecture of the gated propagation strategy.

involves matrix multiplication with a quadratic computational complexity $O(CN^2)$, which leads to bottlenecks when scaling up to deal with complex real-world problems, especially large-scale scene-level point cloud segmentation. To tackle this problem, instead of placing the context-guided aggregation module after the last feature propagation layer, we integrate these modules into the early layers of the decoder, which is of lower resolution, such that the computation and memory consumption can be largely reduced without sacrificing the performance.

Fig. 4 illustrates the architecture of the context aggregation module. Given an input feature $F_{in} \in R^{N\times C}$, which represents N points with C-channel features, 1×1 convolutions are first employed to transfer F_{in} to different embeddings $Q \in R^{N\times C}$, $K \in R^{N\times C}$, $V \in R^{N\times C}$, which denote query, key and value, respectively. Then K is transposed and left-multiplied by Q to obtain the affinity matrix $M_{aff} \in R^{N\times N}$, to which softmax operation is applied to calculate the attention weights $W_{att} \in R^{N\times N}$,

$$w_{i,j} = \frac{\exp\left(Q_i \cdot K_j^T\right)}{\sum_{i=1}^{N} \exp\left(Q_i \cdot K_j^T\right)}, i, j = 1, 2, 3, ..., N \qquad (2)$$

where $w_{i,j} \in W_{att}$ measures the similarity between i-th position and j-th position in input points. Finally, we perform matrix multiplication on attention weights W_{att} and value V to aggregate contextual information from value points for each query point, which gives us the augmented feature $F_{aug} \in R^{N\times C}$, which will be fed to the next feature propagation layer in the network.

We employ context-guided aggregation module between every two feature propagation layers. To facilitate the training process, each of these context-guided aggregation modules is supervised by an auxiliary loss function, forming a multi-scale context-guided aggregation module as a whole, which we will elaborate in section 3.4.

3.3. Gated propagation

To further suppress irrelevant information flow between encoder and decoder, we introduce a gated propagation module into the feature propagation layer, which aims to seek a balance between encoder features and decoder features before fuse them together.

Fig. 5 illustrates this gated propagation process. For input feature $F_{enc} \in R^{N_1\times C_1}$ and $F_{dec} \in R^{N_2\times C_2}$, inverse distance weighted interpolation is first employed to interpolate F_{dec} to $F_{int} \in R^{N_1\times C_2}$, which is of equal resolution with F_{enc} so that we can concatenate them together to obtain $F_{con} \in R^{N_1\times(C_1+C_2)}$. After that, we successively apply relu activation, 1×1 convolution and sigmoid function to the concatenated feature F_{con} to get the balancing weights $W_{bal} \in R^{N_1}$, whose elements $w_i \in [0, 1]$ gives a weight for i-th position, which helps balance F_{enc} against F_{int}. Finally, we multiply W_{bal} with F_{enc} and $(1-W_{bal})$ with F_{dec}, and concatenate these two features together to get our output feature $F_{out} \in R^{N_1\times(C_1+C_2)}$.

3.4. Loss function

To enable the network with the capability of exploring semantic relation and contextual information, the loss function

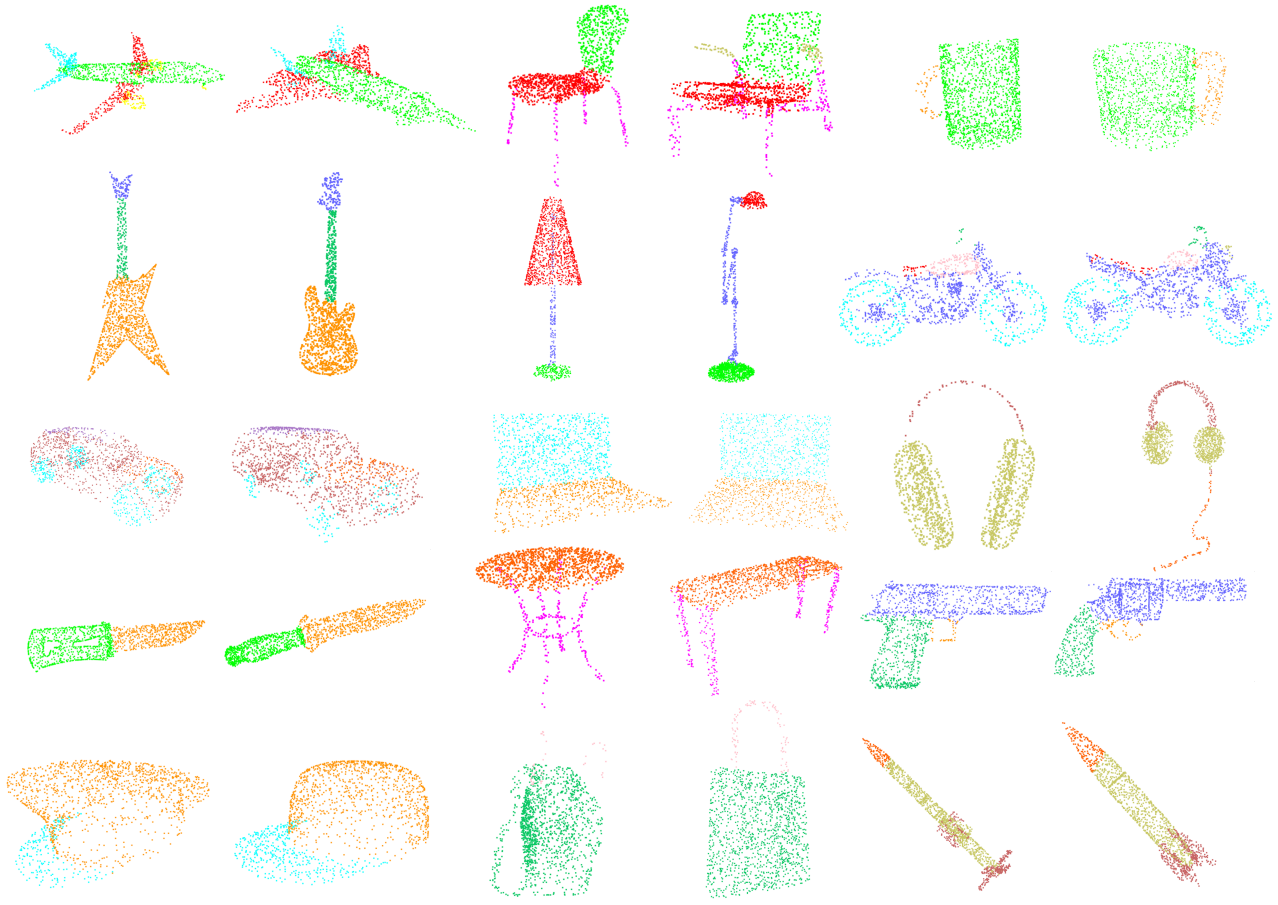


Fig. 6. Visualization of part segmentation results on ShapeNetPart dataset.

should be carefully designed. In our paper, multi-scale cross entropy losses are adopted to facilitate the training process. We define our loss function as follow:

$$L = \sum_{i=1}^{M} \lambda_i L_i, \tag{3}$$

where L_i is the cross entropy loss on i-th scale and M is set 4 in our paper. λ_i represents trad-off factor to balance multi-scale cross entropy losses and we empirically set $\lambda_i = 1$ by default.

4. Experiments

In this section, we will provide a thorough evaluation to demonstrate the performance of our proposed method both qualitatively and quantitatively. For sake of fairness, we compare our proposed method with state-of-the-art methods under two public datasets: Stanford 3D Indoor Semantics Dataset (S3DIS) [40], and ShapeNetPart [41].

4.1. Implementation configurations

Our proposed module is implemented in TensorFlow on a workstation with an Intel Core i7-4790 CPU (3.60GHz, 16GB memory) and a GeForce GTX 1070 GPU (8GB memory, CUDA 8.0).

The whole network is trained in an end-to-end manner using the Adam optimizer [42] with batch size 12, base learning rate 0.001 and momentum 0.9. For S3DIS dataset, we train the network for 100 epochs and decay the learning rate by 0.5 for every 300k iterations. For ShapeNetPart dataset, we train the network for 200 epochs and decay the learning rate by 0.7 for every 200k iterations. Our method achieves 0.56/0.04 second per batch for training/inference on this setting.

4.2. Shape-level segmentation on ShapeNetPart dataset

We carry out semantic part segmentation experiments on the ShapeNetPart dataset to evaluate our performance in 3D shape segmentation. Being a subset of ShapeNet, ShapeNetPart contains 16, 681 models from 16 shape categories, each annotated with 2 to 6 parts and there are 50 different parts in total. We follow the official train/test split and formulate the task as predicting a per point part label, given 3D shape point clouds and their category labels. For evaluation, part-averaged intersection of union (pIoU) is calculated along with detailed per-class pIoU.

The quantitative results of our method on the ShapeNetPart dataset are shown in Table 1. We compare our method

Table 1. Semantic part segmentation results on ShapeNetPart dataset ("e-p" denotes earphone and "s-b" denotes skateboard).

Method	Year	pIoU	aero	bag	cap	car	chair	e-p	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	s-b	table
PointNet [12]	2017	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++ [13]	2017	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
PointCNN [14]	2018	86.1	84.1	86.5	86.0	80.8	90.6	79.7	92.3	88.4	85.3	96.1	77.2	95.3	84.2	64.2	80.0	83.0
DGCNN [28]	2019	85.2	84.0	83.4	86.7	77.8	90.6	74.7	91.2	87.5	82.8	95.7	66.3	94.9	81.1	63.5	74.5	82.6
PointConv [19]	2019	85.3	82.6	79.6	85.2	79.4	90.8	70.7	91.0	87.1	83.7	95.6	73.9	94.9	82.6	59.1	75.0	82.6
DPA [21]	2019	86.1	84.3	81.6	89.1	79.5	90.9	77.5	91.8	87.0	84.5	96.2	68.7	94.5	81.4	64.2	76.2	84.3
A-CNN [20]	2019	85.9	83.9	86.7	83.5	79.5	91.3	77.0	91.5	86.0	85.0	95.5	72.6	94.9	83.8	57.8	76.6	83.0
RS-CNN [29]	2019	86.2	83.5	84.8	88.8	79.6	91.2	81.1	91.6	88.4	86.0	96.0	73.7	94.1	83.4	60.5	77.7	83.6
Ours	**2020**	**85.8**	**83.4**	**81.9**	**84.2**	**78.4**	**91.0**	**74.5**	**91.0**	**84.8**	**84.3**	**95.7**	**69.7**	**95.2**	**81.3**	**63.0**	**77.0**	**84.2**

Table 2. Semantic segmentation results on S3DIS dataset evaluated on Area 5. Note that all existing methods couldn't perform well on the category *beam*, which only accounts for 0.029% points of the total ("-" denotes no results reported in the original paper).

Method	Year	OA	mAcc	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [12]	2017	-	48.98	41.09	88.80	97.33	69.80	0.05	3.92	46.26	10.76	58.93	52.61	5.85	40.28	26.38	33.22
SegCloud [1]	2017	-	57.35	48.92	90.06	96.05	69.86	0.00	18.37	38.35	23.12	70.40	75.89	40.88	58.42	12.96	41.60
PointNet++ [13]	2017	84.67	59.14	51.22	90.38	97.31	76.20	0.00	6.58	48.83	21.65	67.76	73.27	29.34	59.30	51.61	43.57
SPGraph [16]	2018	86.38	66.50	58.04	89.35	96.87	78.12	0.00	42.81	48.93	61.58	84.66	75.41	69.84	52.60	2.10	52.22
PCCN [43]	2018	-	67.01	58.27	92.26	96.20	75.89	0.27	5.98	69.49	63.45	66.87	65.63	47.28	68.91	59.10	46.22
PointCNN [14]	2018	85.91	63.86	57.26	92.31	98.24	79.41	0.00	17.60	22.77	62.09	74.39	80.59	31.67	66.67	62.05	56.74
PointConv [19]	2019	86.06	59.87	52.06	92.44	97.42	78.31	0.00	16.03	35.65	30.27	70.66	80.1	16.79	63.42	48.16	47.46
PointWeb [44]	2019	86.97	66.64	60.28	91.95	98.48	79.39	0.00	21.11	59.72	34.81	76.33	88.27	46.89	69.30	64.91	52.46
GACNet [18]	2019	87.79	-	62.85	92.28	98.27	81.90	0.00	20.35	59.07	40.85	78.54	85.80	61.70	70.75	74.66	52.82
PointEdge [36]	2019	87.18	68.30	61.85	91.47	98.16	81.38	0.00	23.34	65.30	40.02	75.46	87.70	58.45	67.78	65.61	49.36
ELGS [38]	2019	88.43	-	60.06	92.80	98.48	72.65	0.01	32.42	68.12	28.79	74.91	85.12	55.89	64.93	47.74	58.22
Ours	**2020**	**88.50**	**66.09**	**59.66**	**93.38**	**98.45**	**81.50**	**0.00**	**7.00**	**55.14**	**48.61**	**77.16**	**87.81**	**50.68**	**65.54**	**57.76**	**52.57**

with state-of-the-art methods (PointNet [12], PointNet++ [13], PointCNN [14], DGCNN [28], DPA [21], A-CNN [20] and RS-CNN [29]). As we can see from Table 1, the performance of our proposed method achieves comparable results with other methods. Specifically, we achieve 85.8% pIoU, which is 2.1% and 0.7% higher than PointNet and PointNet++. In Fig. 6 we give the visualization of part segmentation results.

4.3. Indoor scene semantic segmentation on S3DIS dataset

For evaluating our performance in indoor real-world scene scans, we conduct semantic segmentation on S3DIS dataset. As a large-scale indoor scene dataset, S3DIS dataset contains 3D scans of 271 rooms in 6 areas, obtained by a Matterport scanner. Each point in the scene is represented by a 9-dimensional vector (XYZ, RGB and normalized coordinates as to the room) and accompanied with a semantic label among 13 categories (see Fig. 8).

To prepare our dataset, we follow the procedure in [12] and split the rooms into 1m×1m overlapped blocks on the ground plane, and randomly sample 4096 points from each block on-the-fly. In our experiments, we train our network on Area 1, 2, 3, 4 and 6 in S3DIS dataset and test it on Area 5.

For the evaluation metrics, we use overall point-wise accuracy (OA), mean of class-wise accuracy (mAcc) and mean of class-wise intersection over union (mIoU).

Quantitatively, semantic segmentation results evaluated on S3DIS Area 5 are shown in Table 2. Compared to state-of-the-art methods, ours yields the highest score in terms of OA. Specifically, we achieve 88.50% on OA, which surpasses all previous methods. Besides, we achieve 66.09% and 59.66% on mAcc and mIoU, respectively, which is on par with these approaches. Qualitatively, we present visualization of semantic segmentation results in Fig. 7. It can be observed that our method segments the input scenes with high-quality even in complex scenes, i.e., office and conference room, which manifests the effectiveness of our method. Compared with PointNet++, the results produced by our method are much more accurate and consistent, especially for categories like door, sofa, table and clutter, etc (see circled places in Fig. 7).

4.4. Feature visualization

To demonstrate the capability of our method in terms of enhancing the point's representative power in feature space, we present the t-SNE visualization of features in Fig. 8. In detail, we randomly pick 100 points from each category and perform t-SNE on the features of these points to reveal how well these features are separated in high-dimensional space. From left to right in Fig. 8 are visualizations of input features, features learned by PointNet++ and features learned by our method, which are 6-dimensional, 128-dimensional and 128-dimensional vectors, respectively. From the Fig. 8, we can observe that the features learned by our method are more discriminative than the PointNet++ counterparts.

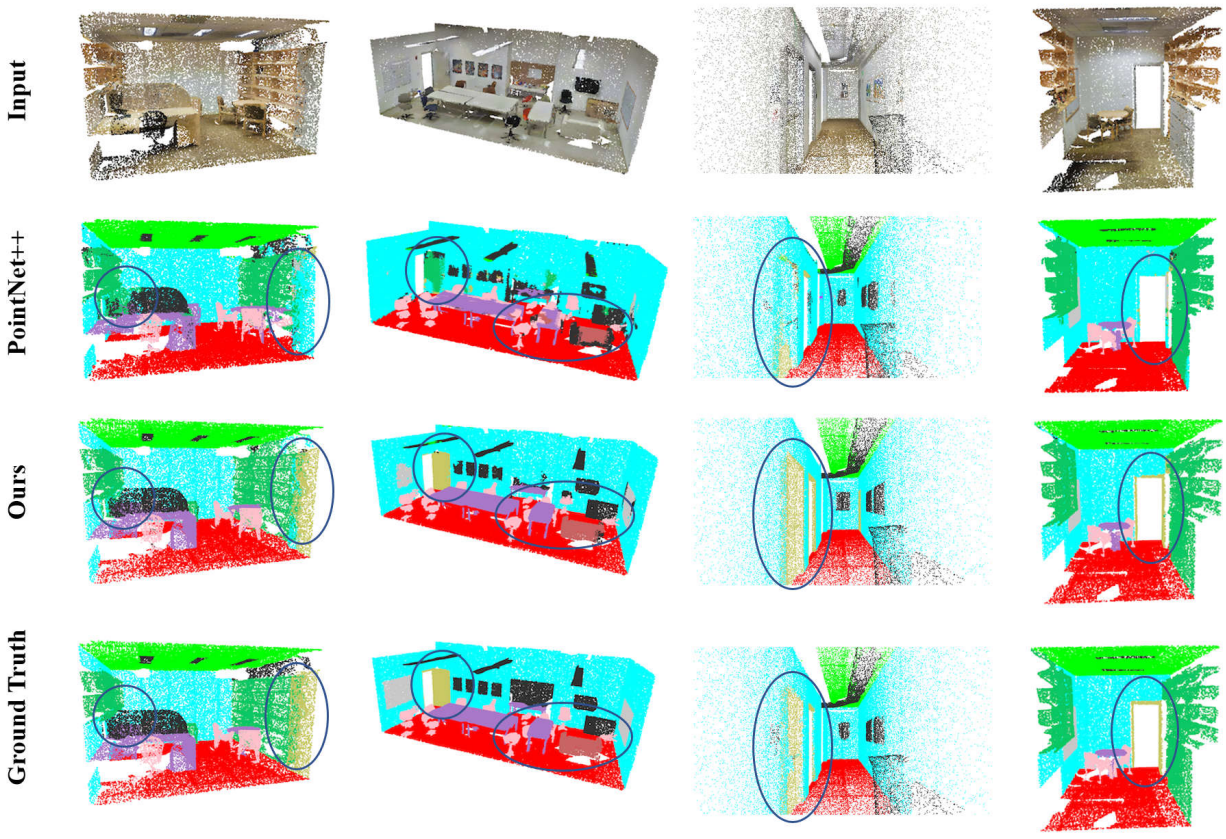
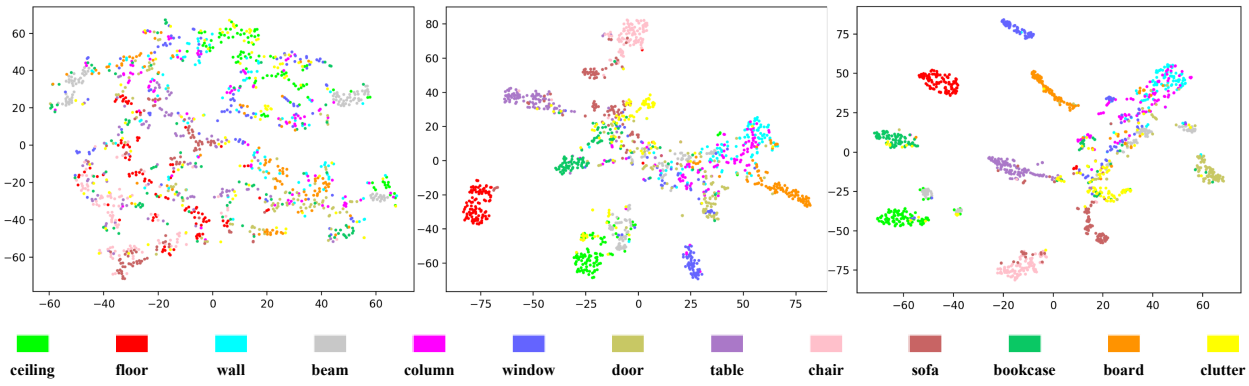


Fig. 7. Visualization of semantic segmentation results on S3DIS dataset.



(a) Input feature **(b) PointNet++** **(c) Ours**

Fig. 8. t-SNE visualization of input feature, features learned by PointNet++ and features learned by our method. (Best viewed in color)

To quantitatively measure the discrimination of the learned features, we design two metrics: the intra-class distance $Dist_{intra}$ and inter-class distance $Dist_{inter}$ to manifest the superiority of our method. Specifically, the $Dist_{intra}$ measures the average distance between each point to the center point of their category on the 2D t-SNE map, the $Dist_{inter}$ measures the average distance between any two center points of 13 categories. The experiment is conducted for 100 times each to overcome the non-uniqueness of t-SNE. As shown in Table 3, our method achieves higher $Dist_{inter}$ and lower $Dist_{intra}$ than PointNet++, which means that features learned by our method are more discriminative.

Table 3. Comparison between PointNet++ and our method in terms of t-SNE quantitative evaluation.

Method	$Dist_{intra}$	$Dist_{inter}$
PointNet++	21.50	62.25
Ours	**20.36**	**63.62**

Table 4. Ablation results on the S3DIS dataset Area 5.

Method	OA	mAcc	mIoU	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
Ours w/o LRL	87.14	63.69	56.54	92.59	98.39	78.89	0.00	4.58	52.26	35.13	74.25	84.29	46.06	65.51	53.30	49.74
Ours w/o MSCGA	87.48	64.74	57.31	92.91	98.12	79.90	0.00	8.71	51.58	38.69	75.48	83.75	43.83	64.70	56.98	50.36
Ours w/o GP	88.17	64.85	57.56	93.03	98.41	80.94	0.00	9.36	50.86	42.86	77.48	84.96	38.48	66.42	53.36	52.10
Baseline	85.25	59.86	52.42	91.73	97.88	75.98	0.00	2.70	48.66	23.26	71.74	77.06	33.78	60.60	51.79	46.31
Ours	**88.50**	**66.09**	**59.66**	**93.38**	**98.45**	**81.50**	**0.00**	**7.00**	**55.14**	**48.61**	**77.16**	**87.81**	**50.68**	**65.54**	**57.76**	**52.57**

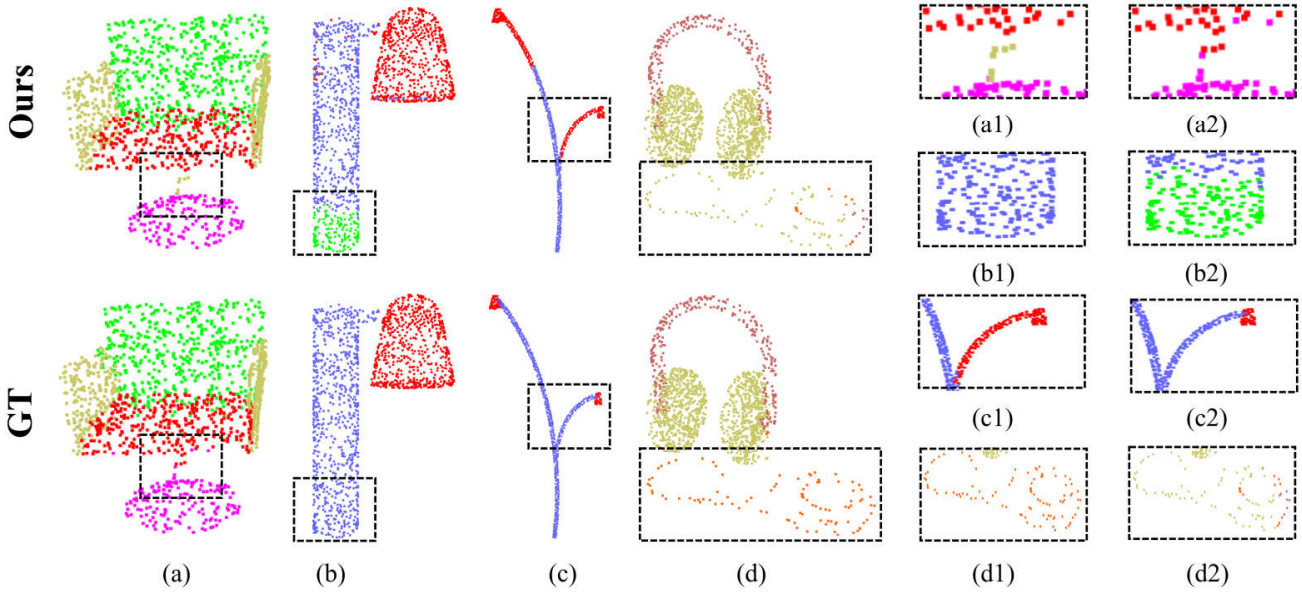


Fig. 9. Visualization of failure cases on ShapeNetPart.

4.5. Ablation study

In this section, we conduct ablation studies to verify the efficacy of each module in our framework. We unload each module from our full pipeline and compare the performance with and without each module. All the experiments are carried out on S3DIS dataset and evaluated on Area 5 and under the multi-scale supervision. As shown in Table 4, our method improves the performance over the baseline, which is simply PointNet++, by a large margin. Specifically, we can observe a 3.25%, 6.23% and 7.24% performance lifting on OA, mAcc and mIoU, respectively. The more details of our quantitative results are illustrated in Table 4.

Local relation learning (LRL). To further evaluate our LRL module, a comparison is conducted by removing it from our full pipeline and replacing it with max-pooling. As shown in Table 4 (first row and bottom row), our method without LRL module has obviously degenerated, with 1.36%, 2.40% and 3.12% performance dropping on OA, mAcc and mIoU, respectively. Moreover, for per-category IoU, we can obviously see that the IoU of the door category drops from 48.61% to 35.13%. Since the max-pooling module couldn't capture local structure, we suppose that the door is often mistakenly predicted as other categories, i.e., the wall. Therefore, leveraging its intrinsic anisotropy, our method can effectively capture structures and patterns in local neighborhood.

Multi-scale context-guided aggregation (MSCGA). Besides LRL module, another key module is MSCGA module with the aim of aggregating contextual information. From Table 4 (second row and bottom row) we can see that, removing our MSCGA module also results in performance decrease. Specifically, the OA, mAcc and mIoU suffer a 1.02%, 1.35% and 2.35% degradation, respectively. This indicates that with the context we adaptively collect, the distinctiveness of learned features is improved by our MSCGA module.

Gated propagation (GP). The segmentation performance undergoes 0.33%, 1.24% and 2.10% decrease after we replace our GP strategy with concatenation-based skip links (see Table 4 (third row and bottom row)). The mIoU and mAcc sustain a much more severe degradation than OA, which indicates that our GP strategy contributes to the discrimination of learned features with less irrelevant and redundant information passed to decoder, thus improves the segmentation performance. Particularly, we see a remarkable performance gap on the sofa category, which lose 12.20% IoU without GP strategy.

4.6. Limitation

Although our proposed method achieves decent results in point cloud semantic segmentation, it still suffers from several limitations. As illustrated in Fig. 9, we exhibit some failure cases. As a supervised learning method, our method fails to handle the model of unseen category in training stage. For the swivel (see Fig. 9 (a)) between seat and base, our network

wrongly predicts it as armrest. And for the lamp model (see Fig. 9 (b)), our network habitually adds a base part. Besides, our prediction is still imperfect when it comes to thin and sparse structures (see Fig. 9 (c) and (d)).

Currently, our pipeline is still not quite lightweight enough to perform scene-level semantic segmentation in real-time, which might impede its industry-level applications. In the future, we would like to further reduce complexity of our network architecture to achieve better efficiency without sacrificing the performance. In addition, we also plan to extend our framework to fine-grained object part segmentation and instance-level scene semantic segmentation.

5. Conclusion

In this paper, we presented a novel hierarchical encoder-decoder network architecture for point cloud semantic segmentation. Different from existing methods, our method can adaptively explore semantic relation inherent in points by incorporating our local relation learning module. Moreover, we proposed a novel multi-scale context-guided aggregation module to further capture long-range dependencies, and a gated propagation strategy is adopted to replace skip links with the aim of filtering out irrelevant and redundant information. Extensive experiments and comparisons on public benchmarks have shown that our method outperforms the baseline method significantly and achieves competitive results compared with the state-of-the-art methods.

Acknowledgments

This research is supported in part by the National Key R&D Program of China under Grant No. 2017YFF0106407, National Natural Science Foundation of China under Grant No. 61532002, the Applied Basic Research Program of Qingdao under Grant No. 161013xx, National Science Foundation of USA under Grant IIS-1715985 and IIS-1812606, the Capital Health Research and Development of Special under Grant No. 2016-1-4011, Fundamental Research Funds for the Central Universities, and Beijing Natural Science Foundation-Haidian Primitive Innovation Joint Fund under Grant No. L182016.

References

[1] L. Tchapmi, C. Choy, I. Armeni, J. Gwak, S. Savarese, Segcloud: Semantic segmentation of 3d point clouds, in: Proceedings of the International Conference on 3D Vision, IEEE, 2017, pp. 537–547.

[2] J. Chen, Y. K. Cho, Z. Kira, Multi-view incremental segmentation of 3-d point clouds for mobile robots, IEEE Robotics and Automation Letters 4 (2) (2019) 1240–1246.

[3] S. Shi, X. Wang, H. Li, Pointrcnn: 3d object proposal generation and detection from point cloud, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 770–779.

[4] A. Stanescu, P. Fleck, D. Schmalstieg, C. Arth, Semantic segmentation of geometric primitives in dense 3d point clouds, in: Proceedings of the IEEE International Symposium on Mixed and Augmented Reality Adjunct, IEEE, 2018, pp. 206–211.

[5] D. Maturana, S. Scherer, Voxnet: A 3d convolutional neural network for real-time object recognition, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2015, pp. 922–928.

[6] G. Riegler, A. Osman Ulusoy, A. Geiger, Octnet: Learning deep 3d representations at high resolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3577–3586.

[7] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, X. Tong, O-cnn: Octree-based convolutional neural networks for 3d shape analysis, ACM Transactions on Graphics 36 (4) (2017) 1–11.

[8] P.-S. Wang, C.-Y. Sun, Y. Liu, X. Tong, Adaptive o-cnn: A patch-based deep representation of 3d shapes, ACM Transactions on Graphics 37 (6) (2018) 1–11.

[9] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, Multi-view convolutional neural networks for 3d shape recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 945–953.

[10] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, L. J. Guibas, Volumetric and multi-view cnns for object classification on 3d data, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 5648–5656.

[11] K. Genova, F. Cole, D. Vlasic, A. Sarna, W. T. Freeman, T. Funkhouser, Learning shape templates with structured implicit functions, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 7154–7164.

[12] 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.

[13] C. R. Qi, L. Yi, H. Su, L. J. Guibas, Pointnet++: Deep hierarchical feature learning on point sets in a metric space, in: Proceedings of the Advances in Neural Information Processing Systems, 2017, pp. 5099–5108.

[14] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, B. Chen, Pointcnn: Convolution on x-transformed points, in: Proceedings of the Advances in Neural Information Processing Systems, 2018, pp. 820–830.

[15] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao, 3d shapenets: A deep representation for volumetric shapes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1912–1920.

[16] L. Landrieu, M. Simonovsky, Large-scale point cloud semantic segmentation with superpoint graphs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4558–4567.

[17] H. Su, V. Jampani, D. Sun, S. Maji, E. Kalogerakis, M.-H. Yang, J. Kautz, Splatnet: Sparse lattice networks for point cloud processing, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2530–2539.

[18] L. Wang, Y. Huang, Y. Hou, S. Zhang, J. Shan, Graph attention convolution for point cloud semantic segmentation, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 10296–10305.

[19] W. Wu, Z. Qi, L. Fuxin, Pointconv: Deep convolutional networks on 3d point clouds, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9621–9630.

[20] A. Komarichev, Z. Zhong, J. Hua, A-cnn: Annularly convolutional neural networks on point clouds, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 7421–7430.

[21] J. Liu, B. Ni, C. Li, J. Yang, Q. Tian, Dynamic points agglomeration for hierarchical point sets learning, in: Proceeding of the IEEE International Conference on Computer Vision, 2019, pp. 7546–7555.

[22] J. Mao, X. Wang, H. Li, Interpolated convolutional networks for 3d point cloud understanding, in: Proceeding of the IEEE International Conference on Computer Vision, 2019, pp. 1578–1587.

[23] C. R. Qi, W. Liu, C. Wu, H. Su, L. J. Guibas, Frustum pointnets for 3d object detection from rgb-d data, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 918–927.

[24] C. R. Qi, O. Litany, K. He, L. J. Guibas, Deep hough voting for 3d object detection in point clouds, in: Proceeding of the IEEE International Conference on Computer Vision, 2019, pp. 9277–9286.

[25] P. Guerrero, Y. Kleiman, M. Ovsjanikov, N. J. Mitra, Pcpnet learning local shape properties from raw point clouds, in: Computer Graphics Forum, Vol. 37, Wiley Online Library, 2018, pp. 75–85.

[26] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, P.-A. Heng, Pu-net: Point cloud upsampling network, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2790–2799.

[27] V. Jampani, M. Kiefel, P. V. Gehler, Learning sparse high dimensional filters: Image filtering, dense crfs and bilateral neural networks, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recog-

nition, 2016, pp. 4452–4461.
[28] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, J. M. Solomon, Dynamic graph cnn for learning on point clouds, ACM Transactions on Graphics 38 (5) (2019) 1–12.
[29] Y. Liu, B. Fan, S. Xiang, C. Pan, Relation-shape convolutional neural network for point cloud analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 8895–8904.
[30] S. Lan, R. Yu, G. Yu, L. S. Davis, Modeling local geometric structure of 3d point clouds using geo-cnn, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 998–1008.
[31] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, A. Agrawal, Context encoding for semantic segmentation, in: Proceeding of the IEEE conference on Computer Vision and Pattern Recognition, 2018, pp. 7151–7160.
[32] Y. Yuan, J. Wang, Ocnet: Object context network for scene parsing, arXiv preprint arXiv:1809.00916.
[33] D. Lin, Y. Ji, D. Lischinski, D. Cohen-Or, H. Huang, Multi-scale context intertwining for semantic segmentation, in: Proceeding of the European Conference on Computer Vision, 2018, pp. 603–619.
[34] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, G. Wang, Semantic correlation promoted shape-variant context for segmentation, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 8885–8894.
[35] S. Xie, S. Liu, Z. Chen, Z. Tu, Attentional shapecontextnet for point cloud recognition, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4606–4615.
[36] L. Jiang, H. Zhao, S. Liu, X. Shen, C.-W. Fu, J. Jia, Hierarchical point-edge interaction network for point cloud semantic segmentation, in: Proceeding of the IEEE International Conference on Computer Vision, 2019, pp. 10433–10441.
[37] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.
[38] X. Wang, J. He, L. Ma, Exploiting local and global structure for point cloud semantic segmentation with contextual point representations, in: Proceeding of the Advances in Neural Information Processing Systems, 2019, pp. 4573–4583.
[39] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.
[40] I. Armeni, A. Sax, A. R. Zamir, S. Savarese, Joint 2d-3d-semantic data for indoor scene understanding, arXiv preprint arXiv:1702.01105.
[41] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al., Shapenet: An information-rich 3d model repository, arXiv preprint arXiv:1512.03012.
[42] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[43] S. Wang, S. Suo, W.-C. Ma, A. Pokrovsky, R. Urtasun, Deep parametric continuous convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2589–2597.
[44] H. Zhao, L. Jiang, C.-W. Fu, J. Jia, Pointweb: Enhancing local neighborhood features for point cloud processing, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5565–5573.