

SASO: Joint 3D Semantic-Instance Segmentation via Multi-scale Semantic Association and Salient Point Clustering Optimization

Jingang Tan^{1,2}, Lili Chen^{1,2*}, Kangru Wang^{1,2}, Jingquan Peng^{1,2}, Jiamao Li^{1,2}, Xiaolin Zhang^{1,2,3}

¹ Bionic Vision System Laboratory, State Key Laboratory of Transducer Technology, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai, China.

² University of Chinese Academy of Sciences, Beijing, China.

³ School of Information Science and Technology, ShanghaiTech University, Shanghai, China.

* E-mail: llichen@mail.sim.ac.cn

ISSN 1751-8644
 doi: 0000000000
www.ietdl.org

1 Introduction

Scene perception plays a decisive role in many applications, such as autonomous driving, robot navigation and augmented reality. With the growth of computer technology and artificial intelligence in recent years, scene perception ability of intelligent devices has received increasing attention from both academia and industry, especially for the 3D scenes which can represent the real environment intuitively. Semantic segmentation and instance segmentation of 3D scenes are the fundamental and critical portions of 3D scene perception. Nevertheless, how to model the 3D space into digital shape to accomplish scene segmentation task is an indefinite problem. Various representations of 3D scenes have been investigated, such as depth maps, voxels, multi-views, meshes and point clouds. Based on these representations, a series of excellent works have been investigated to operate segmentation task, such as [1–18]. Among these representations, point clouds are the most compact and natural to the geometric distributions of real 3D scenes, which have been applied extensively in recent researches. In terms of semantic and instance segmentation tasks in 3D point clouds, based on the great success achieved in recent years [1, 2, 5, 6, 9–14, 19–21] for each single task, joint learning methods for both tasks [14–16] have opened up a new effective way to explore the 3D scene segmentation, which improved the performance and promoted further development. Compared with the method [14] exploiting similarity matrix, [15, 16] utilized clustering algorithm to generate instance segmentation result, which was proved to be more effective and flexible. Nevertheless, whether the convergence direction of the training process is consistent with the orientation of clustering algorithm was rarely considered. Additionally, the marginal points are usually harder to be distinguished than the central points, and in multiple objects case the internal points are easier to be distinguished than the boundary points across objects, as shown in Figure 3. To address this problem, we propose a Salient Point Clustering Optimization (*SPCO*) module to introduce clustering into the training process and saliently focus on the points that are harder to be distinguished in instance segmentation.

has a strong association, which can be further exploited. For example, when a point comes from table, it is highly possible that there will be some neighbor points belonging to chair other than from ceiling. The most common approach to explore the semantic associations is the Conditional Random Fields (CRF) algorithm [22], which utilizes normalization based on statistical global probability and has been proved to be effective in segmentation tasks. However, CRF is complex and consumes plenty of resources, how to sufficiently exploit the semantic associations more efficiently is an indefinite problem. Consequently, we propose a Multi-scale Semantic Association (*MSA*) module to fine tune the semantic segmentation results, which is based on the multiple scale semantic association maps generated by statistical analysis. In addition, because of the inherent structures of indoor scenes, the imbalance problem of the category distribution badly limits the performance of 3D scene perception. For example, wall and floor certainly exist in every room while other categories may not, such as sofa, sink, bookshelf, etc.. This leads to the numbers of points from wall and floor are much more than the one from other categories. The imbalance problem is rarely considered in previous works. Thus, we present an adaptive Water Filling Sampling (*WFS*) algorithm to address this problem by changing the sampling probabilities of each category adaptively. To summarize, our contributions are the following:

- We propose a Salient Point Clustering Optimization (*SPCO*) module to introduce clustering into the training process and saliently focus on the points that are harder to be distinguished in instance segmentation.
- We propose a Multi-scale Semantic Association (*MSA*) module based on statistical knowledge to explore the potential spatial association of the semantic information in point clouds.
- We propose an adaptive Water Filling Sampling (*WFS*) algorithm to balance category distribution in the point clouds, which is rarely considered but critical in 3D scene perception.
- Extensive experiments demonstrate that our *SASO* outperforms the state-of-the-art related methods on benchmark datasets in both semantic and instance segmentation criteria.

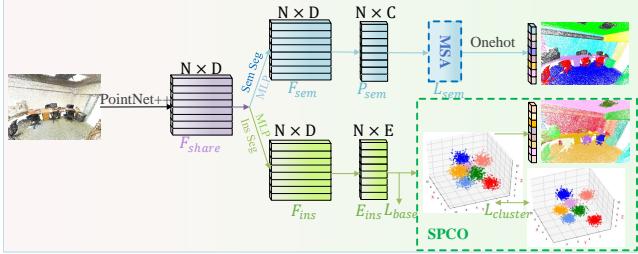


Fig. 1: An illustration of our joint learning framework. The input 3D point clouds are first encoded to F_{share} by PointNet++ [4], then the common feature will be decoded separately by semantic and instance segmentation branches. In semantic segmentation branch (blue), a *MSA* module based on statistics knowledge is proposed to explore the semantic association, we will expound it in Sec 3.2. For the instance segmentation (green), we proposed *SPCO* module to introduce clustering into the training process and focus on hard-distinguished points, which will be explained in Sec 3.3.

2 Related Works

This section reviews recent deep learning-based techniques applied to 3D point clouds. In recent years, a series of deep learning architectures have been proposed to perform the encoding and decoding for 3D point clouds or its derived representations, which are widely utilized in many 3D vision tasks such as semantic and instance segmentation, object part segmentation and object detection. We divide these methods into four categories based on the data representations. Further more, we will introduce recent 3D semantic and instance segmentation research progress based on above techniques.

2.1 Volumetric Methods

Due to 3D point clouds are irregular, the most simple but naive method is to voxelize the irregular point clouds to regular 3D grids so that 3D convolutions can be applied [1, 6, 23–32]. Specifically, Wu *et al.* [23] represented a geometric 3D shape as a probability distribution of binary variables on a 3D voxel grid, using a Convolutional Deep Belief Network. Maturana *et al.* [31] proposed an architecture to efficiently deal with large amounts of point cloud data by integrating a volumetric Occupancy Grid representation with a supervised 3D Convolutional Neural Network. Zhou *et al.* [25] removed the manual feature engineering for 3D point clouds and divided point clouds into equally spaced 3D voxels, then transformed a group of points within each voxel into a unified feature representation through a newly introduced voxel feature encoding layer. Wang *et al.* [1] designed a spatial dense extraction module to preserve the spatial resolution during the feature extraction procedure, alleviating the loss of detail caused by sub-sampling operations such as max-pooling. Although volumetric data representation is the most common and simplest form, there is an obvious drawback that cubic complexity of 3D convolutions leads to a dramatic increase in the memory consumption and computing resources. To tackle this issue, [24, 28] proposed octree representation to improve efficiency of network and reduce computing resources. In addition, [6, 30] proposed sparse convolutional operations to process spatially-sparse 3D point clouds and achieved impressive results. Although these methods try to alleviate the efficiency problem, they are much more complex than volumetric CNNs and can not fundamentally solve the memory consumption problem.

2.2 Multi-view Methods

Another common method for 3D point clouds are a multi-view representation. In recent years, Convolutional Neural Networks have been proved successful in a wide range of 2D visual tasks. To sufficiently take advantage of the strong extraction capability of classical CNNs, 3D point clouds are first projected into multiple pre-defined views, which are then processed by well-designed image-based CNNs to

extract features, such as [2, 26, 33–37]. Specifically, Guerry *et al.* [37] used 3D-coherent synthesis of scene observations and mixed them in a multi-view framework for 3D labeling. Su *et al.* [33] presented a novel CNN architecture that combines information from multiple views of a 3D shape into a single and compact shape descriptor offering even better recognition performance. Dai *et al.* [2] encoded the sparse 3D point clouds with a compact multi-view representation, including bird’s eye view and front view as well as RGB image to perform high-accuracy 3D object detection. You *et al.* [36] proposed PVNet to integrate both the point cloud and the multi-view data towards joint 3D shape recognition. Although the multi-view representation of point cloud data is reasonable, the project process from 3D to 2D will loss the full utilization of 3D geometric information.

2.3 Graph Convolution Methods

Graph structure is a native representation of irregular data, such as 3D point clouds, which offers a compact yet rich representation of contextual relationships between points of different object parts [19, 20, 38–41]. Specifically, Bruna *et al.* [38] proposed two constructions based on a hierarchical clustering of the domain and the spectrum of the graph Laplacian, to prove that for low-dimensional graphs, it is possible to learn convolutional layers with a number of parameters independent of the input size, resulting in efficient deep architectures. Wang *et al.* [39] operated spectral graph convolution on a local graph, combined with a novel graph pooling strategy to augment the relative layout of neighboring points as well as their features. Te *et al.* [40] treated features of points in a point cloud as signals on graph, and defined the convolution over graph by Chebyshev polynomial approximation leveraging on spectral graph theory. They also designed a graph-signal smoothness prior in the loss function to regularize the learning process. Although the graph convolutional methods have achieved significant performance, these methods constructed on Laplacian matrix, is computationally complex for Laplacian eigen-decomposition and has a large quantity of parameters to express the convolutional filters while lacks spatial localization.

2.4 Point clouds Methods

Point clouds are an intuitive, memory-efficient 3D representation which is well-suited for representing geometric details. How to apply deep learning techniques in point clouds directly, simply and efficiently is a critical problem. To address this challenge, Qi *et al.* [3] designed a novel type of neural network PointNet that directly consumes point clouds and well respects the permutation invariance of points in the input. More specifically, they solved the disorder problem of the point clouds through max pooling and maintained the rotation invariance through the spatial transformation network STN. The extracted features of each point are the combination of its own information and the global information. PointNet has been proved efficient in many applications ranging from object classification, part segmentation, object detection to scene semantic parsing. However, PointNet only relies on the max-pooling layer to learn global features and does not consider local relationships. Therefore, a series of works [4, 5, 8, 42, 43] were developed through investigations of the local context and hierarchical learning structures. Typically, Qi *et al.* [4] proposed PointNet++ based on their previous work PointNet, which utilizes pointnet as a local feature extraction module to operate hierarchical feature extraction like CNNs, and finally uses upsampling to generate the final high level features. Li *et al.* [43] proposed PointCNN which uses MLP to learn a transformation matrix to solve the disorder problem of point cloud, and then utilizes the introduced x-conv module to perform convolution on the transformed features. This method achieved similar performance as PointNet++.

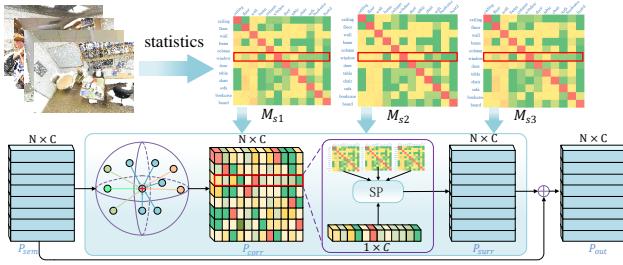


Fig. 2: An illustration of our *MSA* module. First, we create multi-scale semantic association map M_s by statistics with ball query upon all the training 3D scenes. For a point i in the semantic prediction result, we also generate a vector $P_{corr}(i)$ indicating the probabilities of different categories about surrounding points with ball query. Then we calculate the similarity between this vector and each line (category) in the M_s and normalize it as a probability vector, the detail of the calculation SP is formulated in equation 7. The final prediction for each point is the fusion of original predict probability and the fine-tuned probability, as formulated in equation 8.

2.5 3D semantic and instance segmentation

Recent advances in learning-based techniques have also led to various cutting-edge 3D semantic and instance segmentation approaches [1–8, 19, 20, 26, 40, 43–45]. Volumetric representation has been adapted by [1, 6] to transfer 3D point clouds to regular grids and operate CNNs to extract features. [19, 20, 40] utilized graph convolutional networks to model the relationships of 3D points which offers a compact yet rich representation of context. [2, 26] transferred 3D point clouds into multiple views to sufficiently take advantage of the strong extraction capability of classical CNNs. [3–5, 7] presented more efficient and flexible ways to utilize MLP directly upon point clouds and well respect the permutation invariance of points. [43–45] operated segmentation task by designing novel CNNs on point clouds, Huang *et al.* [8] and Ye *et al.* [9] proposed new approaches by slicing the point clouds and utilizing recurrent neural networks to exploit the inherent contextual features. 3D instance segmentation is a relatively new research area and attracts more and more attention [11–13]. Specifically, Lahoud *et al.* [12] proposed a network based on 3D voxel grids, which treats the instance segmentation task as multi-task learning problem. The network generates abstract feature embeddings for voxels and estimates instances' centers to learn instance information. Yang *et al.* [11] introduced a framework which simultaneously generates 3D bounding boxes and predicts the binary masks for the points within each box in one stage. Recently, Wang *et al.* [14] have opened up a framework by jointly operating semantic and instance segmentation in 3D point clouds. Inspired by the proposal mechanism in 2D FasterRcnn[46], they proposed similarity matrix indicating the similarity between each pair of points in embedded feature space to predict point grouping proposals, then the network will predict corresponding semantic class for each proposal to generate the final semantic-instance results. Although the similarity matrix is effective and natural to indicate the proposals, it will generate a large and inefficient matrix which suffers from the heavy computation and memory consumes. Some followed proposal methods [10, 18] were proposed to boost the performance of similar framework while still depended on two-stage procedure and the time-consuming non-maximum suppression algorithm. More recently, [15, 16] utilized clustering algorithm to divide points into different objects, which was demonstrated to be more effective and efficient than proposal methods. Nevertheless, they did not consider whether the convergence direction of the training process is coupled with the orientation of clustering algorithm. In addition, different points have various difficulties to be divided into distinct objects, which is rarely considered. In this work, we propose a framework which takes this critical problem into consideration and prove that it is significant and effective.

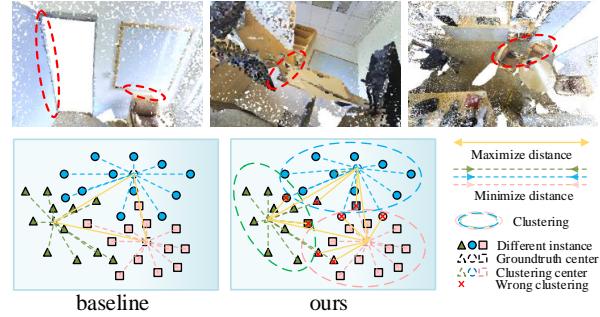


Fig. 3: An illustration of our *SPCO* module. As shown in the first line, the different points of one object have different difficulties to be distinguished, especially for the points of the joints among different objects. In terms of this problem, we introduce clustering into training procedure and saliently focus on the points that are harder to be distinguished in the clustering process.

3 Proposed Method

In this section, we first introduce the baseline framework of our network which jointly perform semantic and instance segmentation tasks. Then we give the details of our *MSA* module for semantic segmentation in Sec 3.2, as depicted in Figure 2. Next, we expound our *SPCO* module for instance segmentation in Sec 3.3, as shown in Figure 3. The whole framework of our method can be seen in Figure 1. Finally, the adaptive Water Filling Sampling (*WFS*) algorithm is explained in details in Sec 3.4.

3.1 Baseline Framework

As depicted in Figure 1, the network without *MSA* and replaced *SPCO* with normal clustering is the baseline framework. First, point clouds of size N are encoded into a high-dimensional feature matrix $F_{share} \in \mathbb{R}^{N \times D}$ by the encoder PointNet++ [4]. Next, two tasks separately decode F_{share} for their own missions. In the semantic segmentation branch, F_{share} is decoded into the semantic feature matrix $F_{sem} \in \mathbb{R}^{N \times D}$ and then outputs the semantic predictions $P_{sem} \in \mathbb{R}^{N \times C}$, where C is the semantic class number. The instance segmentation branch decodes F_{share} into the instance feature matrix $F_{ins} \in \mathbb{R}^{N \times D}$, which is utilized to predict the per-point instance embeddings $E_{ins} \in \mathbb{R}^{N \times E}$, where E denotes the length of the output embedding dimensions. These embeddings are used to calculate the distances among the points for instance clustering. During the training process, the semantic branch is supervised by cross entropy loss while the loss function for instance segmentation, inspired by [15], is formulated as follows:

$$\mathcal{L}_{base} = \mathcal{L}_{in} + \mathcal{L}_{out} + \lambda \mathcal{L}_{reg} \quad (1)$$

where the goal of \mathcal{L}_{in} is to pull the embeddings toward the mean embedding of the points in the instance, while \mathcal{L}_{out} guides the mean embedding of instances to repel each other. We denote \mathcal{L}_{reg} as a regularization term that bounds the embedding values. The three loss terms are denoted as:

$$\mathcal{L}_{in} = \frac{1}{I} \sum_{i=1}^I \frac{1}{N_i} \sum_{j=1}^{N_i} [\|\tau_i - f_j\|_1 - \zeta_v]^2_+ \quad (2)$$

$$\mathcal{L}_{out} = \frac{1}{I(I-1)} \sum_{i_a=1}^I \sum_{i_b=1, i_a \neq i_b}^I [2\zeta_d - \|\tau_{i_a} - \tau_{i_b}\|_1]^2_+ \quad (3)$$

$$\mathcal{L}_{reg} = \frac{1}{I} \sum_{i=1}^I \|\tau_i\|_1 \quad (4)$$

where I represents the number of ground-truth instances; N_i is the number of points in instance i ; τ_i denotes the mean embedding of

instance i ; f_j is an embedding of a point; ζ_v and ζ_d indicate margins for the variance and distance loss respectively; i_a and i_b represent different instances; $[x]_+ = \max(0, x)$ is the hinge function; and the l_1 distance is represented by $\|\cdot\|_1$.

For inference, we use mean-shift clustering [47] on the instance embeddings to obtain the final instance labels following [15]. The mode of the semantic labels for the points within the same instance is assigned as the predicted semantic class.

3.2 Multi-scale Semantic Association Module

In 3D semantic segmentation, for a point of an object, the categories of surrounding points are usually related to the category of the point itself, i.e., the spatial distribution of the semantic information has a strong association as the ensemble in Sec 1, which can be further exploited. Thus, based on the semantic context information, we propose our Multi-scale Semantic Association (*MSA*) Module, which can be seen in Figure 2.

As shown in Figure 2, on the one hand, we create multi-scale semantic association maps by statistics with ball query upon all the training 3D scenes, $M_s \in \mathbb{R}^{C \times C}$ means the map in scale s , C is the number of class. On the other hand, based on the decoded semantic output feature P_{sem} , we can also generate the probabilities P_{corr}^s of the categories from surrounding points with ball query in scale s . Then for each point i in P_{corr}^s , we calculate the distance between $P_{corr}^s(i)$ and each line in M_s , and transfer the result as a probability vector for this point, where the larger a bit is, the higher the probability for this point belonging to corresponding category is. Note that the *MSA* module will generate multiple probability vectors because of multiple scales and these probabilities are only come from surrounding points. At last, the original predicted probability vector is added by the multiple probability vectors to get the final prediction. The formula is described as equation (5)-(8)

$$O_{sem} = o(\text{argmax}(P_{sem})) \quad (5)$$

$$\mathcal{P}_{corr}^s(i) = \frac{1}{|B_i^s|} \sum_{\substack{j=1 \\ j \in B_i^s}}^{|B_i^s|} O_{sem}(j) \quad (6)$$

$$\mathcal{P}_{surr}^s(i) = \phi(1 - \sigma(\|\mathcal{P}_{corr}^s(i) - M_s\|^2)) \quad (7)$$

$$\mathcal{P}_{out} = \mathcal{P}_{sem} + \alpha_1 \mathcal{P}_{surr}^{s1} + \alpha_2 \mathcal{P}_{surr}^{s2} + \alpha_3 \mathcal{P}_{surr}^{s3} \quad (8)$$

where o means one hot operation, $|B_i^s|$ means the number of points in the ball query of point i in scale s , M_s means the semantic association map in scale s , σ means normalization and ϕ means softmax operation. Note that $\mathcal{P}_{corr}^s(i) \in \mathbb{R}^{1 \times C}$ and $M_s \in \mathbb{R}^{C \times C}$ can be operated with broadcast mechanism, and $\|\cdot\|^2$ is operated in axis 1. The final probability output is the sum of P_{sem} and \mathcal{P}_{corr} in different scales with different coefficients. In our experiment, we set s_1, s_2, s_3 equal to radius 0.2, 0.3, 0.5 m and $\alpha_1, \alpha_2, \alpha_3$ equal to 0.5, 0.3, 0.2 respectively.

3.3 Salient Point Clustering Optimization Module

As explained in the baseline framework, for instance segmentation, the goal of \mathcal{L}_{in} is to pull the embeddings toward the mean embedding of the points from the same object, while \mathcal{L}_{out} guides the mean embedding of instances to repel each other in the training process. In the inference time, mean shift clustering algorithm is utilized to distinguish points of different objects. However, the coupling between the convergence orientations in training and the clustering orientation in inference is not taken into consideration. In addition, the points from the same object have different difficulties in instance segmentation as the ensemble in Sec 1. Thus, in this paper, we propose a Salient Point Clustering Optimization (*SPCO*) module, which takes mean shift clustering algorithm into the training process and saliently focuses on the points that are harder to be distinguished in the clustering process. More specifically, as shown in Figure 3, mean shift clustering algorithm is operated in training process to simulate

the clustering procedure in inference. Then for the points clustered in one instance while are not belonging to this instance according to the ground truth, we generate an additional loss $\mathcal{L}_{cluster}$ to repel these embeddings away from the mean embedding of this instance. The loss $\mathcal{L}_{cluster}$ is formulated in equation (9), note that the ID of the clustered instance is decided by the mode of ID in the ground truth, and to converge on a reliable model, we add $\mathcal{L}_{cluster}$ into the training process from 10 epochs.

$$\mathcal{L}_{cluster} = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{1}{|W^i|} \sum_{\substack{j=1 \\ j \in W^i}}^{|W^i|} [2\zeta_d - \|E_j^i - \bar{E}^i\|_1]_+^2 \quad (9)$$

$$\mathcal{L}_{ins} = \mathcal{L}_{base} + \mathcal{L}_{cluster} \quad (10)$$

where N_c means the number of instances in clustering, $|W^i|$ means the number of wrong clustered points in instance i , E_j^i means the embedding of j th wrong clustered point in instance i and \bar{E}^i means the mean embedding of the correct clustered points in instance i . Equipped with our *SPCO* module, the network can simulate the clustering procedure in inference more realistically, and pay more attention to the points that are easy to be erroneously clustered, which is significant for improving the performance of instance segmentation.

3.4 Water Filling Sampling algorithm

In indoor scenes, there exists some inherent structures. For example, the space is always surrounded by walls and floors. When we sample point clouds from indoor scenes, points of certain categories will occupy the main proportion, which will cause serious imbalance problem between these main categories and other normal categories, especially for tiny objects. In previous works of points segmentation task, this problem is rarely discussed. Therefore, in this paper, a Water Filling Sampling algorithm is proposed to solve the imbalance problem in indoor scenes, which is adaptive to different category distribution. Specifically, for a point cloud of a scene, we first cut it into blocks along X - Y plane and store corresponding semantic and instance labels for each point in the blocks. In addition, we define an accumulative vector $VB \in \mathbb{R}^{1 \times C}$ to store the block number for each category, and generate a list $SemB[i]$ to indicate which block contains points of category i . If the number of points in a block that belongs to category i is larger than a thresh t , the block index will be contained in $SemB[i]$ and $VB[i]$ will be added by 1. When we accomplished the cutting step, we can get the probabilities of block number for each category from VB . To keep the balance among the categories, we need to sample the same size of blocks from all the blocks with different probabilities. If the original probability of a category is high in the row data, the sample probability should be correspondingly low. To achieve this goal, we gradually add a small probability value δ to the category with the minimum sum of original probability and current sampling probability, until the sum of the total sampling probability values up to 1. The process is likely to fill water to the canyon consisting of original probabilities of all the categories, the details of the algorithm can be formulated as Algorithm 1. As for part segmentation datasets, such as ShapeNet, the algorithm becomes more concise because we can obtain $SemB$ and VB for each object directly and skip the cutting step. Note that because of the characteristic of part segmentation dataset, we perform *WFS* algorithm on super categories.

4 Experiments

In this part, we will compare our method with other SOTA methods in 3D point clouds semantic and instance segmentation tasks to demonstrate that our method is effective and robust on different kind of datasets, including large scale indoor 3D dataset and part segmentation 3D dataset.

Algorithm 1 Details of Water Filling Sampling algorithm (*WFS*)

Input: Training point clouds of all the scenes S with corresponding semantic-instance labels, and a series of parameters, including threshold t , number of points for each block N_p and number of categories N_c .

Output: All the balanced blocks Blk with corresponding semantic labels $SLab$ and instance labels $ILab$.

initialization: $SemB = [] * N_c, VB = [0] * N_c, SP = [0] * N_c, SPB = [], B = [], \Omega = 0, \delta = 0.0001$

```

1: for  $S_i$  in all the scenes  $S$  do
2:   Cut  $S^i$  into blocks  $B^i$  along  $X-Y$  plane.
3:   In each block, random sample  $N_p$  points with labels.
4:   for  $B_j^i$  in all the blocks  $B^i$  do
5:      $SLab_j^i \leftarrow$  Separate out corresponding labels.
6:     for  $c$  in range  $[0, N_c - 1]$  do
7:        $pc = \text{sum}(ISLab}_j^i == c)$ 
8:       if  $pc > t$  then
9:          $SemB[c] \leftarrow SemB[c]$  extended with  $[(i, j)]$ 
10:         $VB[c] += 1$ 
11:      end if
12:    end for
13:     $B \leftarrow B$  extended with  $B_j^i$ 
14:  end for
15: end for
16: Get the original probability  $OP = VB / \text{sum}(VB)$ 
17: while  $\Omega < 1$  do
18:    $idx = \text{argmin}(OP)$ 
19:    $OP[idx] += \delta$ 
20:    $SP[idx] += \delta$ 
21:    $\Omega += \delta$ 
22: end while
23: for  $c$  in range  $[0, N_c - 1]$  do
24:    $Sc = SP[c] * \text{length}(B)$ 
25:    $Bc \leftarrow$  Random sample  $Sc$  block indicates in  $SemB[c]$ 
26:    $SPB \leftarrow SPB$  extended with  $Bc$ 
27: end for
28:  $B \leftarrow B$  extended with  $B[SPB]$ 
29: Separate  $B$  into  $Blk, SLab$  and  $ILab$ 
30: Return  $Blk, SLab, ILab$ 

```

4.1 Datasets and Details

Datasets. Followed as [15], we conduct the experiments on two benchmark datasets: Stanford 3D Indoor Semantics Dataset (S3DIS) [48] and ShapeNet part segmentation Dataset [49]. The specific introduction of these datasets is as follows:

- S3DIS is a real 3D point cloud dataset generated by Matterport Scanners for indoor spaces, which contains 6 areas and 272 rooms. Each point contains 9 dimensions for the input feature including XYZ, RGB and normalized coordinates. For each point, an instance ID and a semantic category ID with 13 classes are annotated. Following [3], we split the rooms into $1\text{ m} \times 1\text{ m}$ overlapped blocks with stride 0.5 m along the $X-Y$ plane and sample 4096 points from each block.
- ShapeNet dataset is a synthetic scene mesh for part segmentation, which consists of 16881 shape models from 16 categories. Each object is annotated with 2 to 5 parts from 50 different sub-categories. We utilize the instance annotations generated by [14] as the ground-truth labels and we sample 2048 points for each shape during training followed as [3]. We split the dataset into training and validation followed [15] and 3-dimensional vector including XYZ is fed into our network as input.

Details. For instance segmentation, we trained SASO with $\lambda = 0.001$. We use five output embeddings following [15] and set α to 0.01. We select the Adam optimizer to optimize the network on a single GPU (Tesla P100) and set the momentum to 0.9 for the training process. During the inference process, we set the bandwidth to 0.6 for mean-shift clustering and apply the BlockMerging algorithm

Table 1 Semantic (green) and instance (red) segmentation results on S3DIS.

Dataset	Method	mCov	mWCov	mPrec	mRec	mAcc	mIoU	oAcc
Area5	SGPN [14]	32.7	35.5	36.0	28.7			
	JSIS3D [16]	32.6	35.6	39.7	29.1	59.2	51.8	86.9
	3D-BoNet [11]	41.5	44.6	57.6	40.2	59.2	51.8	86.9
	ASIS [15]	44.6	47.8	55.3	42.4	60.9	53.4	86.9
	OURS	49.0	51.9	59.5	45.9	63.5	55.5	87.5
6-Fold CV	SGPN [14]	37.9	40.8	38.2	31.2			
	JSIS3D [16]	37.3	41.0	49.5	33.4	59.8	48.5	79.9
	3D-BoNet [11]	48.4	52.4	65.6	47.6	69.3	59.4	86.3
	ASIS [15]	51.2	55.1	63.6	47.5	70.1	59.3	86.2
	OURS	54.5	58.3	64.2	50.8	72.8	61.1	87.0

Table 2 Ablation study on the S3DIS dataset in Area5.

SPCO	MSA	WFS	mWCov	mPrec	mAcc	mIoU
×	×	×	47.1	51.9	59.7	52.0
✓	✗	✗	50.3	56.0	61.6	53.6
✗	✓	✗	47.1	51.9	61.4	53.2
✗	✗	✓	49.8	55.3	61.2	53.3
✓	✓	✗	50.3	56.0	62.7	54.5
✓	✓	✓	51.9	59.5	63.5	55.5

[14] to merge instances from different blocks.

Evaluation. Following [15], we evaluate the experimental results in the following metrics. For semantic segmentation, we calculate the overall accuracy ($oAcc$), mean accuracy ($mAcc$) and mean IoU ($mIoU$) across all the semantic classes along with the detailed scores of the per-class IoU . To evaluate the performance of instance segmentation, we use the coverage (Cov) and weighted coverage ($WCov$) [50–52]. Cov is the average instance-wise IoU of the prediction matched with ground truth, and $WCov$ is the Cov score after being weighted by the size of ground truth. For the predicted regions P and the ground-truth regions G , Cov and $WCov$ are defined as:

$$\text{Cov}(\mathcal{G}, \mathcal{P}) = \sum_{i=1}^{|\mathcal{G}|} \frac{1}{|\mathcal{G}|} \max_j \text{IoU}(r_i^{\mathcal{G}}, r_j^{\mathcal{P}}) \quad (11)$$

$$\text{WCov}(\mathcal{G}, \mathcal{P}) = \sum_{i=1}^{|\mathcal{G}|} w_i \max_j \text{IoU}(r_i^{\mathcal{G}}, r_j^{\mathcal{P}}) \quad (12)$$

$$w_i = \frac{|r_i^{\mathcal{G}}|}{\sum_k |r_k^{\mathcal{G}}|} \quad (13)$$

where $|r_i^{\mathcal{G}}|$ is the number of points in ground-truth region i . We also measure the classical metrics of mean precision ($mPrec$) and mean recall ($mRec$) with an IoU threshold of 0.5.

4.2 S3DIS Evaluation

We conduct the experiments on the S3DIS dataset with the backbone networks PointNet++. We train the network for 50 epochs with a batch size of 12, the initial learning rate is set to 0.001 and divided by 2 every 300 k iterations.

Quantitative Results. For classical Area5 validation scenes, the quantitative results of SASO in instance and semantic segmentation tasks are shown in Table 1. As we can see, SASO achieves 51.9 $mWCov$ and 59.5 $mPrec$, which dramatically outperforms the state-of-the-art method 3D-BoNet [11] by 7.3 in $mWCov$ and 1.9 in $mPrec$. As for semantic segmentation, our method significantly improves the $mAcc$ and $mIoU$ by 2.6 and 2.1 respectively, compared with advanced ASIS [15]. For a more comprehensive comparison, we evaluate our method with 6 fold cross validation on S3DIS dataset. As shown in the table, our method achieves 58.3 $mWCov$ and 72.8 $mAcc$, which significantly outperforms the state-of-the-art methods by a large margin. The stable improvement in both semantic and instance segmentation demonstrates the effectiveness of our method. For a more detailed comparison with our

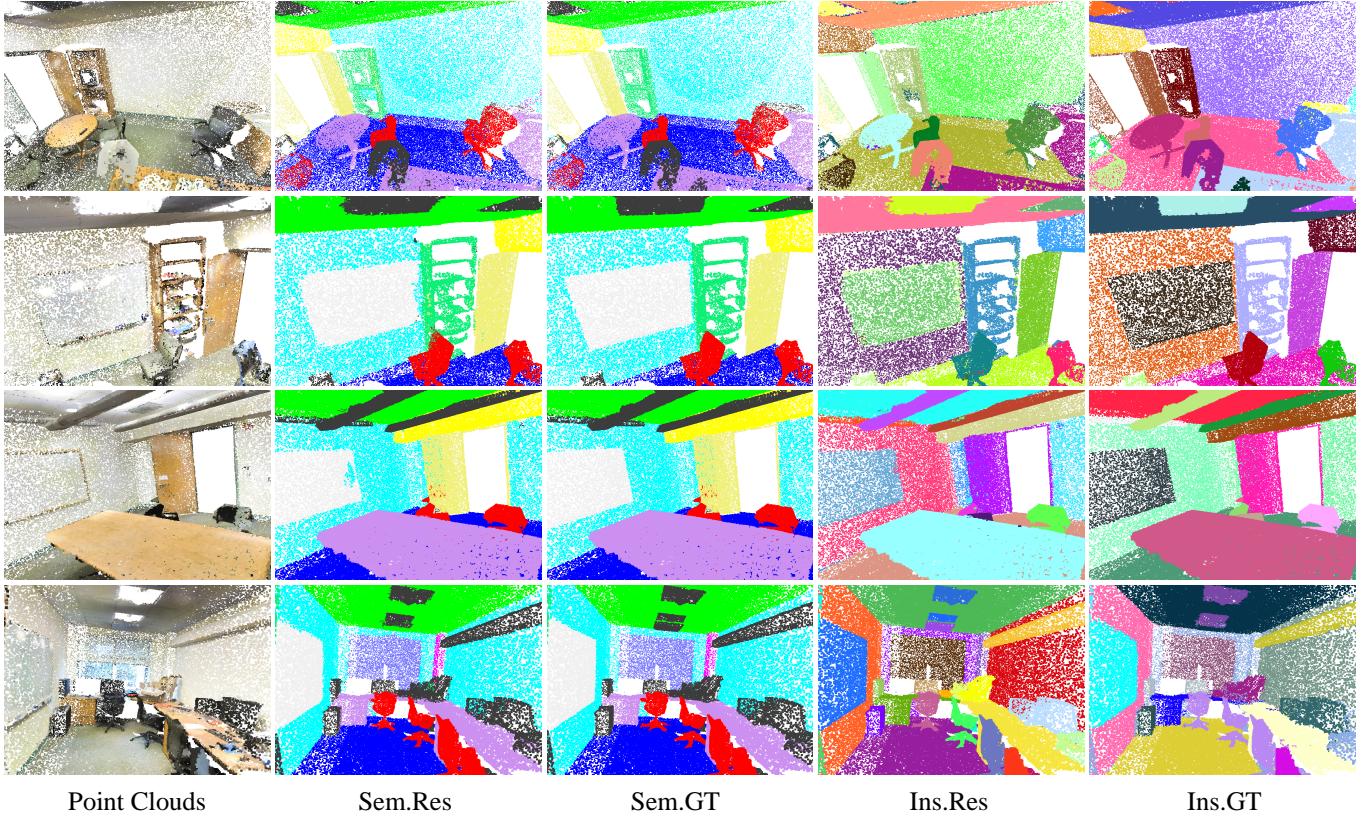


Fig. 4: Qualitative results of our method on the S3DIS dataset. For semantic results, each color refers to a particular category and for instance results, different colors represent different objects.

Table 3 Per class results on the S3DIS dataset.

Metrics	Method	mean	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
Wcov	BASE	47.1	89.7	88.7	68.3	0.0	3.4	60.9	5.0	51.8	67.6	23.9	53.6	50.3	49.5
	ASIS* [15]	47.6	89.0	89.2	72.4	0.0	8.8	58.1	4.7	52.4	76.6	46.3	50.1	64.4	45.5
	OURS	51.9	89.0	87.3	73.1	0.0	9.1	60.1	13.3	54.3	69.8	48.7	55.0	68.1	46.6
Sem IoU	BASE	52.0	92.8	97.8	74.8	0.0	7.9	51.9	16.1	72.3	77.9	35.4	56.1	42.5	50.8
	ASIS* [15]	53.4	92.4	98.4	76.7	0.0	15.6	49.5	21.4	72.3	78.7	38.0	55.9	45.8	49.7
	OURS	55.5	92.5	97.7	77.2	0.0	11.7	50.8	29.0	74.2	80.3	41.3	60.0	56.6	50.8

Table 4 Comparisons of computation time, GPU memory and performance.

Method	Train		Test		mPrec
	time (m)	memory (MB)	time (m)	memory (MB)	
SGPN [14]	59.3	7549	209.5	420	36.0
ASIS [15]	64.7	4275	54.2	1235	55.3
OURS	75.0	1203	40.4	373	59.5

baseline framework and ASIS [15], Table 3 shows the results for specific categories in both instance and semantic segmentation based on Area5 scene in S3DIS. Note that for a fair comparison, we reproduce the result of ASIS [15] with PointNet++ backbone using the author’s code to get the per class results.

Qualitative Results. To intuitively present our results, we visualize the predict results and annotations on point clouds, as shown in Figure 4. For instance segmentation, different colors represent different instances. For semantic segmentation, each color refers to a particular category. It is obvious that our method has a great performance, especially at the boundaries of different objects.

Ablation Study. The ablation study results are shown in Table 2. Equipped with different modules of our method upon the baseline framework, we can find that with our *SPCO* module, we obtain 3.2 gains in *mWCov* and 4.1 gains in *mPrec*. It is interesting



Fig. 5: Qualitative results for semantic and instance segmentation on ShapeNet dataset.

that the semantic segmentation results are also improved with this module, we think this is because the semantic and instance segmentation tasks share the shallow features, the improvement in the

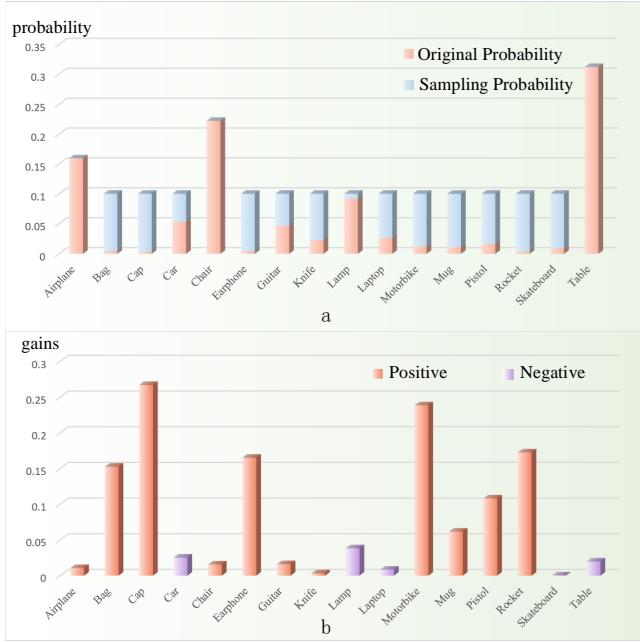


Fig. 6: The sampling probability and the corresponding improvement for different categories upon ShapeNet dataset.

(a). The orange color means the original frequency for different categories in the training dataset, the blue color represents the sampling probabilities for different categories.

(b). The orange color means positive boost while the purple color represents negative influence. Note that for an intuitive visualization, the value are multiplied by 5.

instance segmentation branch can be beneficial to semantic segmentation branch. When we add *MSA* module to the baseline, we can find that the semantic segmentation results are improved with 1.7 in *mAcc* and 1.2 in *mIoU*. With the *WFS* algorithm added to the baseline framework, we obtain 3.4 gains in *mPrec* and 1.3 gains in *mIoU*, which means the balance among different categories is critical to both two tasks. Finally, compared with the baseline framework, our full method has a dramatic improvement in both two tasks, including 7.6 *mPrec* gains in instance segmentation task and 3.5 *mIoU* gains in the semantic segmentation task.

Consumption of memory and time. Table 4 shows a comparison of the memory cost and computation time. For a fair comparison, we conducted the experiments in the same environment, including the same GPU (GTX 1080), batch size (4) and data (Area5 including 68 rooms). Note that all the time units are minutes, and all the memory units are MB. In the training process, the result is the time and memory required for one epoch. As we can see, our method needs relatively more time for training because we introduce clustering into training process, while costs little memory because of the brief but efficient architecture. In the inference process, the results show the resource consumption for Area5. Our approach takes only 373 MB and needs 40.4 minutes while acquires better performance, which is significantly faster and more efficient than the state-of-the-art methods.

Table 5 Semantic segmentation results on ShapeNet datasets.

Method	mIoU
PointNet++ [4]	84.3
ASIS [15]	85.0
SGPN[14]	85.8
SpiderCNN [53]	85.3
SSCN [6]	86.0
PointConv [45]	85.7
BASE	83.5
OURS	86.4

4.3 ShapeNet Evaluation

We also validate our method on part segmentation dataset ShapeNet, the semantic annotations are publicly available while the instance segmentation annotations are the generated results as [14]. Because of the deficiency of ground truth for instance annotations, we only provide the qualitative results for instance segmentation in Figure 5 as [15]. Four lines from top to bottom in Figure 5 mean semantic segmentation results, semantic annotations, instance segmentation results and instance annotations respectively. As we can see, different parts in the same object are well grouped into individual instances, especially the boundaries of different parts. The semantic segmentation results are exhibited in Table 5. Our approach obviously boosts the result upon baseline framework by 2.9 *mIoU* and outperforms the state-of-the-art method ASIS [15], PointConv [45] and SSCN [6]. These results reveal that our proposed method also has the capability to improve the part segmentation performance.

To prove the effectiveness of our *WFS* algorithm intuitively, we show the sampling probability and the corresponding improvement for different categories, as depicted in Figure 6. In the upper graph (a), the orange color means the original frequency of different categories in the training dataset, the blue color represents the sampling probabilities for different categories. We can find that the distribution of different categories is more balanced with our *WFS* algorithm. The second graph (b) shows the improvement for different categories, the orange color means positive boost while the purple color represents negative influence. For the categories with low frequency existing in the raw data, the corresponding improvements are obvious, while for the categories with high frequency, the results are rarely influenced. It demonstrates that our *WFS* algorithm is effective and critical for alleviating the imbalance problem.

5 Conclusion

In this paper, we propose a novel framework which jointly performs semantic and instance segmentation. For the instance segmentation task, a module named *SPCO* is proposed to introduce clustering into the training process and saliently focus on the points that are harder to be distinguished in the clustering process. For the semantic segmentation branch, we introduce *MSA* module based on the statistic knowledge to exploit the potential association of spatial semantic distribution. In addition, we propose a Water Filling Sampling algorithm to address the imbalance problem of category distribution. Qualitative and quantitative experiment results on challenging benchmark datasets demonstrate the effectiveness and robustness of our method.

Acknowledgment

* This project was supported by National Natural Science Foundation of China (No.61806189) and Shanghai Municipal Science and Technology Major Project (Grant No. 2018SHZDZX01, ZHANGJIANG LAB).

6 References

- 1 Wang, Z., Lu, F.: 'Voxsegnet: Volumetric cnns for semantic part segmentation of 3d shapes', *IEEE transactions on visualization and computer graphics*, 2019,
- 2 Dai, A., Nießner, M.: '3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation'. In: Proceedings of the European Conference on Computer Vision (ECCV). (, 2018. pp. 452–468
- 3 Qi, C.R., Su, H., Mo, K., Guibas, L.J. '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
- 4 Qi, C.R., Yi, L., Su, H., Guibas, L.J. 'Pointnet++: Deep hierarchical feature learning on point sets in a metric space'. In: Advances in neural information processing systems. (, 2017. pp. 5099–5108
- 5 Engelmann, F., Kontogianni, T., Hermans, A., Leibe, B. 'Exploring spatial context for 3d semantic segmentation of point clouds'. In: Proceedings of the IEEE International Conference on Computer Vision. (, 2017. pp. 716–724
- 6 Graham, B., Engelcke, M., van der Maaten, L. '3d semantic segmentation with submanifold sparse convolutional networks'. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 9224–9232

- 7 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. pp. 4548–4557
- 8 Huang, Q., Wang, W., Neumann, U. ‘Recurrent slice networks for 3d segmentation of point clouds’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 2626–2635
- 9 Ye, X., Li, J., Huang, H., Du, L., Zhang, X.: ‘3D Recurrent Neural Networks with Context Fusion for Point Cloud Semantic Segmentation: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII’. (, 2018)
- 10 Yi, L., Zhao, W., Wang, H., Sung, M., Guibas, L.J. ‘Gspn: Generative shape proposal network for 3d instance segmentation in point cloud’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 3947–3956
- 11 Yang, B., Wang, J., Clark, R., Hu, Q., Wang, S., Markham, A., et al.: ‘Learning object bounding boxes for 3d instance segmentation on point clouds’, *arXiv preprint arXiv:190601140*, 2019.
- 12 Lahoud, J., Ghanem, B., Pollefeys, M., Oswald, M.R.: ‘3d instance segmentation via multi-task metric learning’, *arXiv preprint arXiv:190608650*, 2019.
- 13 Liu, C., Furukawa, Y.: ‘Masc: Multi-scale affinity with sparse convolution for 3d instance segmentation’, *arXiv preprint arXiv:190204478*, 2019.
- 14 Wang, W., Yu, R., Huang, Q., Neumann, U. ‘Sgpn: Similarity group proposal network for 3d point cloud instance segmentation’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 2569–2578
- 15 Wang, X., Liu, S., Shen, X., Shen, C., Jia, J. ‘Associatively segmenting instances and semantics in point clouds’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 4096–4105
- 16 Pham, Q.H., Nguyen, T., Hua, B.S., Roig, G., Yeung, S.K. ‘Jsis3d: Joint semantic-instance segmentation of 3d point clouds with multi-task pointwise networks and multi-value conditional random fields’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 8827–8836
- 17 Liang, Z., Yang, M., Wang, C.: ‘3d graph embedding learning with a structure-aware loss function for point cloud semantic instance segmentation’, *arXiv preprint arXiv:190205247*, 2019.
- 18 Hou, J., Dai, A., Nießner, M. ‘3d-sis: 3d semantic instance segmentation of rgbd scans’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 4421–4430
- 19 Landrieu, L., Simonovsky, M. ‘Large-scale point cloud semantic segmentation with superpoint graphs’. (, 2018.
- 20 Wang, L., Huang, Y., Hou, Y., Zhang, S., Shan, J. ‘Graph attention convolution for point cloud semantic segmentation’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 10296–10305
- 21 Elich, C., Engelmann, F., Schult, J., Kontogianni, T., Leibe, B.: ‘3d-bevis: Birds-eye-view instance segmentation’, *arXiv preprint arXiv:190402199*, 2019.
- 22 Lafferty, J., McCallum, A., Pereira, F.C.: ‘Conditional random fields: Probabilistic models for segmenting and labeling sequence data’, 2001.
- 23 Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., et al. ‘3d shapenets: A deep representation for volumetric shapes’. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (, 2015. pp. 1912–1920
- 24 Wang, P.S., Liu, Y., Guo, Y.X., Sun, C.Y., Tong, X.: ‘O-cnn: Octree-based convolutional neural networks for 3d shape analysis’, *ACM Transactions on Graphics (TOG)*, 2017, **36**, (4), pp. 72
- 25 Zhou, Y., Tuzel, O. ‘Voxelenet: End-to-end learning for point cloud based 3d object detection’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 4490–4499
- 26 Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J. ‘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
- 27 Klokov, R., Lempitsky, V. ‘Escape from cells: Deep kd-networks for the recognition of 3d point cloud models’. In: Proceedings of the IEEE International Conference on Computer Vision. (, 2017. pp. 863–872
- 28 Riegler, G., OsmanUlusoy, A., Geiger, A. ‘Octnet: Learning deep 3d representations at high resolutions’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2017. pp. 3577–3586
- 29 Lei, H., Akhtar, N., Mian, A. ‘Octree guided cnn with spherical kernels for 3d point clouds’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 9631–9640
- 30 Ren, M., Pokrovsky, A., Yang, B., Urtasun, R. ‘Sbnet: Sparse blocks network for fast inference’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 8711–8720
- 31 Maturana, D., Scherer, S. ‘Voxnet: A 3d convolutional neural network for real-time object recognition’. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). (IEEE, 2015. pp. 922–928
- 32 Huang, J., You, S. ‘Point cloud labeling using 3d convolutional neural network’. In: 2016 23rd International Conference on Pattern Recognition (ICPR). (IEEE, 2016. pp. 2670–2675
- 33 Su, H., Maji, S., Kalogerakis, E., Learned.Miller, E. ‘Multi-view convolutional neural networks for 3d shape recognition’. In: Proceedings of the IEEE international conference on computer vision. (, 2015. pp. 945–953
- 34 Shi, B., Bai, S., Zhou, Z., Bai, X.: ‘Deeppano: Deep panoramic representation for 3-d shape recognition’, *IEEE Signal Processing Letters*, 2015, **22**, (12), pp. 2339–2343
- 35 Roveri, R., Rahmann, L., Oztireli, C., Gross, M. ‘A network architecture for point cloud classification via automatic depth images generation’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 4176–4184
- 36 You, H., Feng, Y., Ji, R., Gao, Y. ‘Pvnet: A joint convolutional network of point cloud and multi-view for 3d shape recognition’. In: 2018 ACM Multimedia Conference on Multimedia Conference. (ACM, 2018. pp. 1310–1318
- 37 Guerry, J., Boulch, A., Le.Saux, B., Moras, J., Plyer, A., Filliat, D. ‘Snapsnet-r: Consistent 3d multi-view semantic labeling for robotics’. In: Proceedings of the IEEE International Conference on Computer Vision. (, 2017. pp. 669–678
- 38 Bruna, J., Zaremba, W., Szlam, A., LeCun, Y.: ‘Spectral networks and locally connected networks on graphs’, *arXiv preprint arXiv:13126203*, 2013,
- 39 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. pp. 52–66
- 40 Te, G., Hu, W., Zheng, A., Guo, Z. ‘Rgcnn: Regularized graph cnn for point cloud segmentation’. In: 2018 ACM Multimedia Conference on Multimedia Conference. (ACM, 2018. pp. 746–754
- 41 Simonovsky, M., Komodakis, N. ‘Dynamic edge-conditioned filters in convolutional neural networks on graphs’. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (, 2017. pp. 3693–3702
- 42 Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: ‘Dynamic graph cnn for learning on point clouds’, *ACM Transactions on Graphics (TOG)*, 2019, **38**, (5), pp. 146
- 43 Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B. ‘Pointcnn: Convolution on x-transformed points’. In: Advances in Neural Information Processing Systems. (, 2018. pp. 820–830
- 44 Hua, B.S., Tran, M.K., Yeung, S.K. ‘Pointwise convolutional neural networks’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2018. pp. 984–993
- 45 Wu, W., Qi, Z., Fuixin, L. ‘Pointconv: Deep convolutional networks on 3d point clouds’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2019. pp. 9621–9630
- 46 Ren, S., He, K., Girshick, R., Sun, J. ‘Faster r-cnn: Towards real-time object detection with region proposal networks’. In: Advances in neural information processing systems. (, 2015. pp. 91–99
- 47 Comaniciu, D., Meer, P.: ‘Mean shift: A robust approach toward feature space analysis’, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2002, , (5), pp. 603–619
- 48 Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., et al. ‘3d semantic parsing of large-scale indoor spaces’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2016. pp. 1534–1543
- 49 Yi, L., Kim, V.G., Ceylan, D., Shen, I., Yan, M., Su, H., et al.: ‘A scalable active framework for region annotation in 3d shape collections’, *ACM Transactions on Graphics (TOG)*, 2016, **35**, (6), pp. 210
- 50 Ren, M., Zemel, R.S. ‘End-to-end instance segmentation with recurrent attention’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2017. pp. 6656–6664
- 51 Liu, S., Jia, J., Fidler, S., Urtasun, R. ‘Sgn: Sequential grouping networks for instance segmentation’. In: Proceedings of the IEEE International Conference on Computer Vision. (, 2017. pp. 3496–3504
- 52 Zhuo, W., Salzmann, M., He, X., Liu, M. ‘Indoor scene parsing with instance segmentation, semantic labeling and support relationship inference’. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (, 2017. pp. 5429–5437
- 53 Xu, Y., Fan, T., Xu, M., Zeng, L., Qiao, Y. ‘Spidercnn: Deep learning on point sets with parameterized convolutional filters’. In: Proceedings of the European Conference on Computer Vision (ECCV). (, 2018. pp. 87–102