Associatively Segmenting Instances and Semantics in Point Clouds

Xinlong Wang[1] Shu Liu[2] Xiaoyong Shen[2] Chunhua Shen[1] Jiaya Jia[2,3]
[1]The University of Adelaide [2]Youtu Lab, Tencent
[3]The Chinese University of Hong Kong
{wangxinlon, liushuhust, goodshenxy}@gmail.com
chunhua.shen@adelaide.edu.au leojia@cse.cuhk.edu.hk

Abstract

A 3D point cloud describes the real scene precisely and intuitively. To date how to segment diversified elements in such an informative 3D scene is rarely discussed. In this paper, we first introduce a simple and flexible framework to segment instances and semantics in point clouds simultaneously. Then, we propose two approaches which make the two tasks take advantage of each other, leading to a win-win situation. Specifically, we make instance segmentation benefit from semantic segmentation through learning semantic-aware point-level instance embedding. Meanwhile, semantic features of the points belonging to the same instance are fused together to make more accurate per-point semantic predictions. Our method largely outperforms the state-of-the-art method in 3D instance segmentation along with a significant improvement in 3D semantic segmentation. Code has been made available at: https://github.com/WXinlong/ASIS.

1. Introduction

Both instance segmentation and semantic segmentation aim to detect specific informative region represented by sets of smallest units in the scenes. For example, a point cloud can be parsed into groups of points, where each group corresponds to a class of stuff or an individual instance. The two tasks are related and both have wide applications in real scenarios, *e.g.*, autonomous driving and augmented reality. Though great progress has been made in recent years [10, 6, 21, 34, 16] for each single task, no prior method tackles these two tasks associatively.

In fact, instance segmentation and semantic segmentation conflict with each other in some respects. The former one distinguishes different instances of the same class clearly, while the latter one wants them to have the same label. However, the two tasks could cooperate with each other through seeking common grounds. Semantic segmentation distinguishes points of different classes, which is also one of the purposes of instance segmentation, as *points of different classes must belong to different instances.* Furthermore, instance segmentation assigns the same label to points belonging to the same instance, which is also consistent with semantic segmentation, as *points of the same instance must belong to the same category.* This observation makes one wonder how the two tasks could be associated together to lead to a win-win solution?

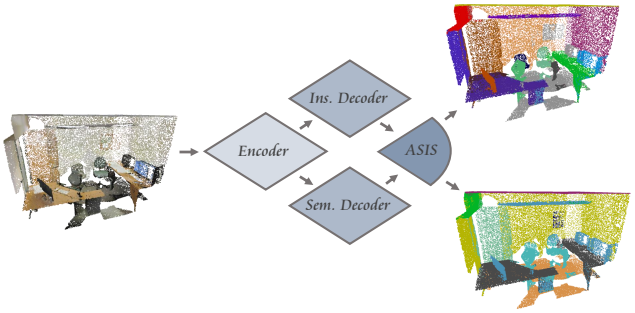

Figure 1: Instance segmentation and semantic segmentation results using ASIS. Our method takes raw point clouds as inputs and outputs instance labels and semantic labels for each point.

There may be two straightforward approaches. The first one is that, given the semantic labels, we could run instance segmentation independently on every semantic class to better distinguish individual instances. Thus, different class instances are separated simply but naively.

However, the instance segmentation would greatly depend on performance of semantic segmentation as incorrect semantic predictions would inevitably result in incorrect instance predictions. Otherwise, given the instance labels, one could classify each instance and assign the predicted class label to each point of this instance. Thus, the problem is transformed to an easier instance recognition problem. However, inaccurate instance predictions would deeply confuse the downstream object classifiers. Both these two approaches are in step-wise paradigm, which can be sub-optimal and inefficient. In this work, we integrate

the two tasks altogether into an end-to-end parallel training framework, which shares the same benefits in a soft and learnable fashion.

We first introduce a simple baseline to segment instances and semantics simultaneously. It is similar to the method in [6] for 2D images, but we tailor it for 3D point cloud. The network of the baseline has two parallel branches: one for per-point semantic predictions; the other outputs point-level instance embeddings, where the embeddings of points belonging to the same instance stay close while those of different instances are apart. Our baseline method can already achieve better performance than the recent state-of-the-art method, SGPN [35], as well as faster training and inference. Based on this flexible baseline, a novel technique is further proposed to associate instance segmentation and semantic segmentation closely together, termed ASIS (Associatively Segmenting Instances and Semantics).

With the proposed ASIS method, we are able to learn semantic-aware instance embeddings, where the embeddings of points belonging to different semantic classes are further separated automatically through feature fusion. As shown in Figure 2, the boundaries between different class points are clearer (chair & table, window & wall). Moreover, the semantic features of points belonging to the same instance are exploited and fused together to make more accurate per-point semantic predictions. The intuition behind it is that during semantic segmentation *a point being assigned to one of the categories is because the instance containing that point belongs to that category*. Thus, the two tasks can take advantage of each other to further boost their performance. Our method is demonstrated to be effective and general on different backbone networks, *e.g.*, the PointNet [26] and hierarchical architecture PointNet++ [28]. The method can also be used to tackle the panoptic segmentation [14] task, which unifies the semantic and instance segmentation. To summarize, our main contributions are as follows.

- We propose a fast and efficient simple baseline for simultaneous instance segmentation and semantic segmentation on 3D point clouds.
- We propose a new framework, termed ASIS, to associate instance segmentation and semantic segmentation closely together. Specifically, two types of partnerships are proposed—semantics awareness for instance segmentation and instance fusion for semantic segmentation—to make these two tasks cooperate with each other.
- With the proposed ASIS, the model containing semantics-aware instance segmentation and instance-fused semantic segmentation are trained end-to-end, which outperforms state-of-the-art 3D instance segmentation methods on the S3DIS dataset [1] along with a significant improvement on the 3D semantic segmentation task. Furthermore, our experiments on the ShapeNet dataset [39] show that ASIS is also beneficial for the task of part segmentation.

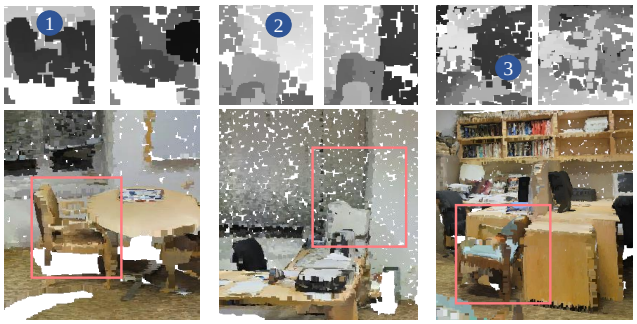

Figure 2: 1D embeddings of learned point-level instance embeddings. t-SNE [22] technique is used to visualize the learned instance embeddings for the points on *S3DIS test* data. Three close-up pairs are shown. Of each pair the left patch is from our baseline method, while the right one is from ASIS. Differences in color shade represent distances in instance embedding space.

2. Related Work

Instance Segmentation. 2D instance segmentation has attracted much research attention recently, leading to various top-performing methods. Inspired by the effectiveness of region-based CNN (R-CNN) [8] in object detection problem, [25, 4] learn to segment instances by proposing segment candidates. The mask proposals are further classified to obtain the final instance masks. Dai *et al*. [5] predict segment proposals based on bounding box proposals. He *et al*. [10] propose the simpler and flexible Mask R-CNN which predicts masks and class labels simultaneously. Different from the top-down detector-based approaches above, bottom-up methods learn to associate per-pixel predictions to object instances. Newell *et al*. [24] group pixels into instances using the learned associative embedding. Brabandere *et al*. [6] propose a discriminative loss function which enables to learn pixel-level instance embedding efficiently. Liu *et al*. [20] decompose the instance segmentation problem into a sequence of sub-grouping problems. However, 3D instance segmentation is rarely researched. Wang *et al*. [35] learn the similarity matrix of a point cloud to get instance proposals. In this work, we introduce a simple and flexible method that learns effective point-level instance embedding with the help of semantic features in 3D point clouds.

Semantic Segmentation. With the recent development of convolutional neural networks (CNNs) [15, 32], tremendous progress has been made in semantic segmentation. Approaches [18, 2, 19] based on fully convolutional networks (FCN) [21] dominate the semantic segmentation on

2D images. As for 3D segmentation, Huang *et al*. [11] propose 3D-FCNN which predict coarse voxel-level semantic label. PointNet [26] and following works [7, 38] use multilayer perceptron (MLP) to produce fine-grained point-level segmentation. Very recently, Landrieu *et al*. [16] introduce superpoint graph (SPG) to segment large-scale point clouds. In fact, few of previous works segment semantics taking advantages of the instance embedding, either in 2D images or 3D point clouds.

Deep Learning on Point Clouds. To take advantage of the strong representation capability of classic CNNs, a 3D point cloud is first projected into multiview rendering images in [33, 31, 27, 9], on which the well-designed CNNs for 2D images can be applied. But part of contextual information in point cloud is left behind during the projection process. Another popular representation for point cloud data is voxelized volumes. The works of [37, 23, 12, 30] convert point cloud data into regular volumetric occupancy grids, then train 3D CNNs or the varieties to perform voxel-level predictions. A drawback of volumetric representations is being both computationally and memory intensive, due to the sparsity of point clouds and the heavy computation of 3D convolutions. Therefore those methods are limited to deal with large-scale 3D scenes. To process raw point cloud directly, PointNet [26] is proposed to yield point-level predictions, achieving strong performance on 3D classification and segmentation tasks. The following works PointNet++ [28], RSNet [13], DGCNN [36] and PointCNN [17] further focus on exploring the local context and hierarchical learning architectures. In this work, we build a novel framework to associatively segment instances and semantics in point clouds, and demonstrate that it is effective and general on different backbone networks.

3. Our Method

3.1. A Simple Baseline

Here we introduce a simple yet effective framework. It is composed of a shared encoder and two parallel decoders. One of the decoders is for point-level semantic predictions, while the other one aims to handle the instance segmentation problem. Specifically, a point cloud of size N_p is first extracted and encoded into a feature matrix through the feature encoder (*e.g*., stacked PointNet layers). This shared feature matrix refers to the concatenation of local features and global features in PointNet architecture, or the output of the last set abstraction module for the PointNet++ architecture. The two parallel branches then fetch the feature matrix and proceed with their following predictions separately. The semantic segmentation branch decodes the shared feature matrix into $N_P \times N_F$ shaped semantic feature matrix F_{SEM}, and then outputs the semantic predictions P_{SEM} with shape of $N_P \times N_C$ where N_C is the number of semantic categories. The instance segmentation branch has the same architecture except the last output layer. The $N_P \times N_F$ instance feature matrix F_{INS} is used to predict per-point instance embedding E_{INS} with shape of $N_P \times N_E$ where N_E is the dimension of the embedding. The embeddings of a point cloud represent the the instance relationship between points in it: the points belonging to the same instance are close to each other in embedding space, while those points of different instances are apart.

At training time, the semantic segmentation branch is supervised by the classical cross entropy loss. As for the instance segmentation, the discriminative loss function for 2D image in [6] is adopted to supervise the instance embedding learning. We modify it and make it suitable for point clouds. The loss used in [6] is class-specific: the instance embeddings of different semantic class are learned separately, which means the semantic class should be given first. This step-wise paradigm is highly dependent on the quality of semantic prediction, as incorrect semantic prediction would inevitably result in incorrect instance recognition. Thus, we adopt the class-agnostic instance embedding learning strategy, where embeddings are in charge of distinguishing different instances and are blind to their categories. The loss function is formulated as follows:

$$L = L_{var} + L_{dist} + \alpha \cdot L_{reg}, \tag{1}$$

where L_{var} aims to pull embeddings towards the mean embedding of the instance, *i.e*. the instance center, L_{dist} make instances repel each other, and L_{reg} is a regularization term to keep the embedding values bounded. α is set to 0.001 in our experiments. Specifically, each term can be written as follows:

$$L_{var} = \frac{1}{I}\sum_{i=1}^{I}\frac{1}{N_i}\sum_{j=1}^{N_i}\left[\|\mu_i - e_j\|_1 - \delta_{\text{v}}\right]_+^2, \tag{2}$$

$$L_{dist} = \frac{1}{I(I-1)}\sum_{\substack{i_A=1\\ i_A\neq i_B}}^{I}\sum_{i_B=1}^{I}\left[2\delta_{\text{d}} - \|\mu_{i_A} - \mu_{i_B}\|_1\right]_+^2, \tag{3}$$

$$L_{reg} = \frac{1}{I}\sum_{i=1}^{I}\|\mu_i\|_1, \tag{4}$$

where I is the number of ground-truth instances; N_i is the number of points in instance i; μ_i is the mean embedding of instance i; $\|\cdot\|_1$ is the ℓ_1 distance; e_j is an embedding of a point; δ_{v} and δ_{d} are margins; $[x]_+ = \max(0, x)$ means the hinge.

During the test, final instance labels are obtained using mean-shift clustering [3] on instance embeddings. We assign the mode of the semantic labels of the points within the same instance as its final category. The pipeline is illustrated in Figure 3(a).

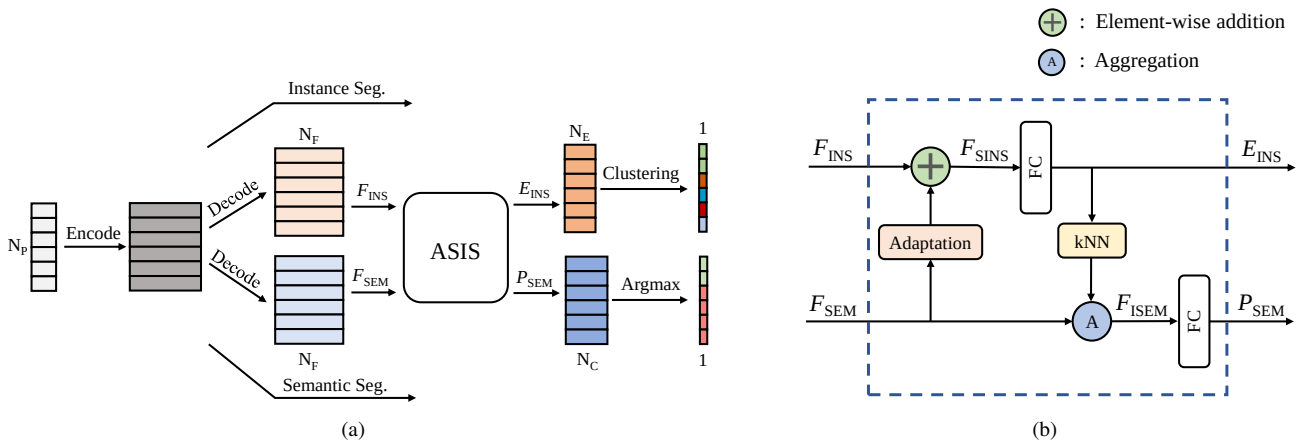

Figure 3: Illustration of our method for point cloud instance segmentation and semantic segmentation. (a) Full pipeline of the system. (b) Illustration of the ASIS module.

3.2. Mutual Aid

As depicted in Figure 3(b), benefiting from the simple and flexible framework described above, we are able to build upon it the novel ASIS module and achieve semantic-aware instance segmentation and instance-fused semantic segmentation.

Semantic-aware Instance Segmentation. Semantic features of a point cloud construct a new and high-level feature space, where points are naturally positioned according to their categories. In that space, points of the same semantic class lie close together while different classes are separated. We abstract the semantic awareness (SA) from semantic features and integrate it into the instance features, producing semantic-aware instance features. Firstly, the semantic feature matrix F_{SEM} is adapted to instance feature space as F'_{SEM} through a point independent fully connected layer (FC) with batch normalization and ReLU activation function. F'_{SEM} has the same shape with F_{SEM}. Then, We add the adapted semantic feature matrix F'_{SEM} to instance feature matrix F_{INS} element-wise, producing semantic-aware instance feature matrix F_{SINS} . The procedure can be formulated as:

$$F_{\text{SINS}} = F_{\text{INS}} + FC(F_{\text{SEM}}). \tag{5}$$

In this soft and learnable way, points belonging to different category instances are further repelled in instance feature space, whereas same category instances are rarely affected. The feature matrix F_{SINS} is used to generate final instance embeddings.

Instance-fused Semantic Segmentation. Given the instance embeddings, we use K nearest neighbor (kNN) search to find a fixed number of neighboring points for each point (including itself) in instance embedding space. To make sure the K sampled points belonging to the same instance, we filter the outliers according to the margin δ_v used in Equation 2. As described in Section 3.1, the hinged loss term L_{var} supervises the instance embedding learning through drawing each point embedding close to the the mean embedding within a distance of δ_v. The output of the kNN search is an index matrix with shape of $N_P \times K$. According to the index matrix, the semantic features (F_{SEM}) of those points are grouped to a $N_P \times K \times N_F$ shaped feature tensor, which is groups of semantic feature matrix where each group corresponds to a local region in instance embedding space neighboring its centroid point. Inspired by the effectiveness of channel-wise max aggregation in [26, 36, 38], semantic features of each group are fused together through a channel-wise max aggregation operation, as the refined semantic feature of the centroid point. The instance fusion (IF) can be formulated as below. For the $N_P \times N_F$ shaped semantic feature matrix $F_{\text{SEM}} = \{x_1, ..., x_{N_P}\} \subseteq \mathbb{R}^{N_F}$, instance-fused semantic features are calculated as:

$$x'_i = \text{Max}(x_{i1}, x_{i2}, ..., x_{ik}), \tag{6}$$

where $\{x_{i1}, ..., x_{ik}\}$ represent the semantic features of K neighboring points centered point i in instance embedding space, and Max is an element-wise maximum operator which takes K vectors as input and output a new vector. After the instance fusion, the output is a $N_P \times N_F$ feature matrix F_{ISEM}, the final semantic features to be fed into the last semantic classifier.

4. Experiments

4.1. Experiment Settings

Datasets. We carry out experiments on two public datasets: Stanford 3D Indoor Semantics Dataset (S3DIS) [1] and ShapeNet [39]. S3DIS contains 3D scans from Matterport Scanners in 6 areas that in total have 272 rooms. Each point in the scene point cloud is associated with an instance label and one of the semantic labels from 13 categories. Besides the large real scene benchmark S3DIS, we also evaluate our methods on the ShapeNet part dataset. This dataset contains $16,881$ 3D shapes from 16 categories. Each point sampled from the shapes is assigned with one of the 50 different parts. The instance annotations from [35] are used as the instance ground-truth labels.

Evaluation Metrics. Our experiments involved S3DIS are conducted following the same k-fold cross validation with micro-averaging as in [26]. We also report the performance on the fifth fold following [34], as Area 5 is not present in other folds. For evaluation of semantic segmentation, overall accuracy (oAcc), mean accuracy (mAcc) and mean IoU (mIoU) across all the categories are calculated along with the detailed scores of per class IoU. As for instance segmentation, (weighted) coverage (Cov, WCov) [29, 20, 40] are adopted. Cov is the average instance-wise IoU of prediction matched with ground-truth. The score is further weighted by the size of the ground-truth instances to get WCov. For ground-truth regions $\mathcal{G}$ and predicted regions $\mathcal{O}$, these values are defined as

$$\text{Cov}(\mathcal{G}, \mathcal{O}) = \sum_{i=1}^{|\mathcal{G}|} \frac{1}{|\mathcal{G}|} \max_j \text{IoU}(r_i^G, r_j^O), \tag{7}$$

$$\text{WCov}(\mathcal{G}, \mathcal{O}) = \sum_{i=1}^{|\mathcal{G}|} w_i \max_j \text{IoU}(r_i^G, r_j^O), \tag{8}$$

$$w_i = \frac{|r_i^G|}{\sum_k |r_k^G|}, \tag{9}$$

where $|r_i^G|$ is the number of points in ground-truth region i. Besides, the classical metrics mean precision (mPrec) and mean recall (mRec) with IoU threshold 0.5 are also reported.

Training and Inference Details. For the S3DIS dataset, each point is represented by a 9-dim feature vector (XYZ, RGB and normalized coordinates as to the room). During training, we follow the procedure in [26] and split the rooms into $1m \times 1m$ overlapped blocks on the ground plane, each containing 4096 points. For the instance segmentation branch, we train the network with $\sigma_v = 0.5$, $\sigma_d = 1.5$, and 5 output embedding dimensions. For the kNN search in instance fusion, K is set to 30. We train the network for 50 epochs and 100 epochs for PointNet and PointNet++ respectively, with batch size 24, base learning rate set to 0.001 and divided by 2 every $300k$ iterations. The Adam solver is adopted to optimize the network on a single GPU. Momentum is set to 0.9. At test time, bandwidth is set to 0.6 for mean-shift clustering. BlockMerging algorithm [35] is used to merge instances from different blocks. For ShapeNet dataset, each shape is represented by a point cloud with 2048 points, as in [26]. Each point is represented by a 3-dim vector (XYZ).

Backbone	Method	mCov	mWCov	mPrec	mRec
Test on Area 5					
PN	SGPN [35]	32.7	35.5	36.0	28.7
	ASIS (*vanilla*)	38.0	40.6	42.3	34.9
	ASIS	**40.4**	**43.3**	**44.5**	**37.4**
PN++	ASIS (*vanilla*)	42.6	45.7	53.4	40.6
	ASIS	**44.6**	**47.8**	**55.3**	**42.4**
Test on 6-fold CV					
PN	SGPN [35]	37.9	40.8	38.2	31.2
	ASIS (*vanilla*)	43.0	46.3	50.6	39.2
	ASIS	**44.7**	**48.2**	**53.2**	**40.7**
PN++	ASIS (*vanilla*)	49.6	53.4	62.7	45.8
	ASIS	**51.2**	**55.1**	**63.6**	**47.5**

Table 1: Instance segmentation results on S3DIS dataset.

Backbone	Method	mAcc	mIoU	oAcc
Test on Area 5				
PN	PN (*RePr*)	52.1	43.4	83.5
	ASIS (*vanilla*)	52.9	44.7	83.7
	ASIS	**55.7**	**46.4**	**84.5**
PN++	ASIS (*vanilla*)	58.3	50.8	86.7
	ASIS	**60.9**	**53.4**	**86.9**
Test on 6-fold CV				
PN	PN [26]	-	47.7	78.6
	PN (*RePr*)	60.3	48.9	80.3
	ASIS (*vanilla*)	60.7	49.5	80.4
	ASIS	**62.3**	**51.1**	**81.7**
PN++	ASIS (*vanilla*)	69.0	58.2	85.9
	ASIS	**70.1**	**59.3**	**86.2**

Table 2: Semantic segmentation results on S3DIS dataset.

4.2. S3DIS Results

We conduct experiments on S3DIS dataset using PointNet and PointNet++ (single-scale grouping) as our backbone networks. If no extra notes, our main analyses are based on PointNet.

4.2.1 Baseline Method

We report instance segmentation results of our baseline method in Table 1. Based on PointNet backbone, our method achieves 46.3 mWCov when evaluate by 6-fold cross validation, which shows an absolute 5.5-point improvement over the state-of-the-art method SGPN[1]. The superiority is consistent across the four evaluation metrics. Semantic segmentation results are shown in Table 2. The mIoU of training without instance segmentation branch is 48.9, which can be regarded as the result of pure backbone PointNet. Equipped with instance segmentation training, our semantic segmentation baseline result achieves 49.5 mIoU, which is slightly better. It indicates that the supervision of instance segmentation helps learn more general shared feature representation. As for the training time, SGPN needs $16 \sim 17$ hours (excluding pre-training) to converge, while it only takes $4 \sim 5$ hours for our method to train from scratch, both on a single GPU. More computation time comparisons can be referred to Table 5. Our baseline method is demonstrated to be effective and efficient.

4.2.2 ASIS

Semantic Segmentation. In Table 1, we report the results of ASIS on instance segmentation task. ASIS yields 48.2 mWCov, which outperforms our baseline by 1.9-point. In terms of another metric mean precision, a larger 2.6-point gain is observed. When evaluated on Area 5, the improvements are more significant: 2.7 mWCov and 2.2 mPrec. Through visualizations in Figure 4, our baseline method tends to group two nearby different class instances together into one instance (*e.g.*, board & wall). With ASIS, they are well distinguished as semantic awareness helps repel them in instance embedding space. Per class performance changes are in accordance with our observations. Shown in Table 4, ASIS yields 5.0 WCov and 2.4 WCov gains on class "board" and class "wall" on instance segmentation.

Instance Segmentation. Table 2 reports the results of ASIS on semantic segmentation task. ASIS improves the mIoU by 1.6-point. We obseve more significant improvements of 2.8 mAcc and 1.7 mIoU when evaluating on Area 5. In Figure 5 we show some comparison examples on semantic segmentation. ASIS performs better on complicated categories (*e.g.*, bookcase) and is aware of instance integrity (*e.g.*, table, window) as instance fusion aggregates points belonging to the same instance to produce more accurate predictions. Table 4 shows that ASIS outperforms the baseline by 3.5 IoU and 2.2 IoU on class "table" and class "bookcase", which are in line with our analysis.

[1]We reproduced the results of SGPN using the code at github, published by the authors.

Method	+IF	+SA	mIoU	mWCov
Baseline			49.5	46.3
	✓		50.0	47.0
		✓	49.8	47.4
	✓	✓	**51.1**	**48.2**

Table 3: Ablation study on the S3DIS dataset. IF refers to instance fusion; SA refers to semantic awareness.

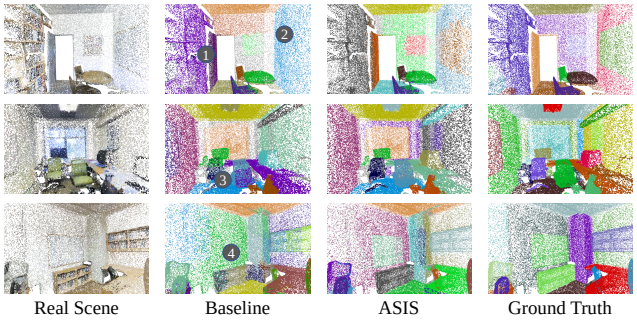

Figure 4: Comparison of our baseline method and ASIS on instance segmentation. Different colors represent different instances.

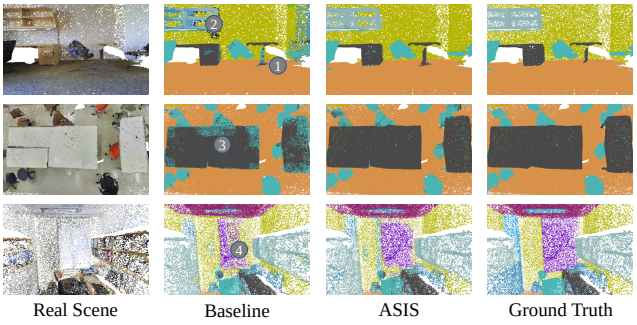

Figure 5: Comparison of our baseline method and ASIS on semantic segmentation.

Stronger Backbone. Both the two tasks benefit largely from our novel method. When adopt the stronger architecture PointNet++ as our backbone network, we observe consistent improvements: 2.1 mWCov and 2.6 mIoU gains on Area 5; 1.7 mWCov and 1.1 mIoU gains for 6-fold cross validation. The results on PointNet++ indicate that our ASIS is a general framework and can be built upon different backbone networks.

4.2.3 Analysis

Ablative Analysis. Equipped with only instance fusion for semantic segmentation, our method achieves 50.0 mIoU and 47.0 mWCov. Compared to the baseline, there is a 0.5-point gain on mIoU. Furthermore, better semantic predictions assign more correct class labels to instances, improv-

(a) Input (b) Ins. (c) Ins. GT (d) Sem. (e) Sem. GT

Figure 6: Qualitative results of ASIS on the S3DIS test fold.

ing the instance segmentation performance. When adopt semantic awareness alone, we achieve an improvement of 1.1 mWCov (from 46.3 to 47.4). The improvement of one task also helps the other one, as better shared features are learned. Applying instance fusion and semantic awareness together, the performance boost is larger than using only one of them. On the basis of instance fusion, semantic awareness could bring additional 1.1 mIoU and 1.2 mWCov gains. The semantic awareness strengthens the instance segmentation, as well as improving the semantic segmentation. It is because that the improved instance embedding predictions could amplify the improvements brought by instance fusion, thus leading to a further 1.1 mIoU gain. The similar results can also be observed when add instance fusion on semantic awareness. To conclude, the two components not only perform their own duty well, but also enlarge the function of the other one.

Category-based Analysis. We show how the performance of each category changes in Figure 7. Interestingly the categories being helped by ASIS module are different for instance segmentation and semantic segmentation. On instance segmentation, our ASIS module largely helps the categories in which instances often surround with instances of other classes (*e.g.*, beam, board and window). For example, board is hung on the wall. The board is easily being ignored during instance segmentation, as the body of the board has similar color and shape with the wall. Our semantic awareness in ASIS module shows great superiority on these cases: 5.0 WCov and 2.4 WCov improvements on class "board" and class "wall". Some visualization examples of the comparison are illustrated in Figure 4. On semantic segmentation, ASIS module significantly boosts the performance of the categories in which instances have complicated shapes (*e.g.*, table, chair and bookcase), because they benefit much from instance fusion.

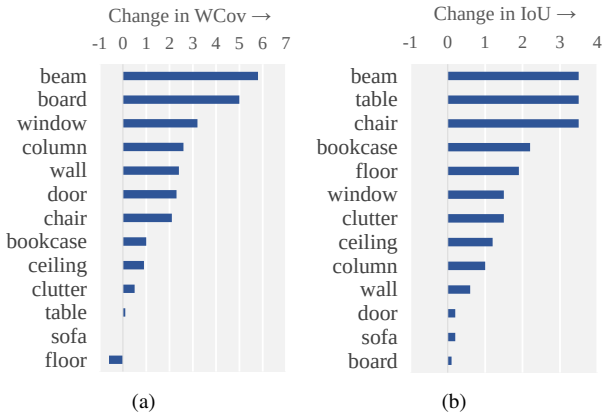

Figure 7: Per class performance changes. (a) Changes of instance segmentation performance compared to our baseline method. (b) Changes of semantic segmentation performance compared to our baseline method.

4.2.4 Qualitative Results

Figure 6 shows some visualization examples of ASIS. For instance segmentation, different colors represent different

	mean	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
WCov	46.3	79.2	**77.0**	63.7	47.6	6.6	55.6	47.5	50.5	57.3	**9.9**	31.3	33.7	41.5
	48.2	**80.1**	76.4	**66.1**	**53.4**	**9.2**	**58.8**	**49.8**	**50.6**	**59.4**	**9.9**	**32.3**	**38.7**	**42.0**
Sem IoU	49.5	90.1	87.8	69.2	42.3	26.0	50.4	54.9	57.5	45.8	8.9	38.0	33.4	39.2
	51.1	**91.3**	**89.7**	**69.8**	**45.8**	**27.0**	**51.9**	**55.1**	**61.0**	**49.3**	**9.1**	**40.2**	**33.5**	**40.7**

Table 4: Per class results on the S3DIS dataset.

Method	Inference Time (ms)			mWCov
	Overall	Network	Grouping	
SGPN	726	18	708	35.5
ASIS (*vanilla*)	212	11	201	41.4
ASIS	205	20	185	43.6
ASIS (*vanilla*.PN++)	150	35	115	45.7
ASIS (PN++)	179	54	125	47.8

Table 5: Comparisons of computation speed and performance. Inference time is estimated and averaged on Area 5, which is the time to process a point cloud with size 4096×9. The instance segmentation results on Area 5 are reported.

instances, while the color itself does not mean anything. Either same class instances or different class instances are distinguished properly. For example, the points of the tables and the surrounding chairs are grouped into distinct instances. As for semantic segmentation, specific color refers to particular class (*e.g.*, yellow for wall, purple for window). We also show some failure cases in Figure 6. In the scenes of the second and third row, two nearby chairs are mistakenly segmented together as a single instance. Though our method does not draw the point embeddings of the same class instances close, we yet do not contribute on better distinguishing this kind of cases. We leave it to future works to explore better solutions.

4.2.5 Computation Time

In Table 5, we report computation time measured on a single Tesla P40 GPU. The inference procedure can be divided in to two steps: the network inference, and point grouping which groups points into individual instances. For SGPN, the grouping step refers to their GroupMerge algorithm. In our ASIS, it is the mean-shift clustering. We achieve comparable speed with SGPN on network inference, while our grouping step is much faster. Overall, it takes 205ms for ASIS to process an input point cloud with size 4096×9 and output the final labels, which is $3.5\times$ faster than SGPN.

4.3. ShapeNet Results

We conduct experiments on ShapeNet dataset using instance segmentation annotations generated by [35], which are not "real" ground truths. Following [35], only qualitative results of part instance segmentation are provided. As shown in Figure 8, tires of the car and legs of the chair are

(a) Ins. (b) Ins. GT (c) Sem. (d) Sem. GT

Figure 8: Qualitative results of ASIS on ShapeNet test split. (a) Instance segmentation results of ASIS. (b) Generated ground truth for instance segmentation. (c) Semantic segmentation results of ASIS. (d) Semantic segmentation ground truth.

well grouped into individual instances. Semantic segmentation results are reported in Table 6. Using PointNet as backbone, we achieve a 0.6-point improvement. Based on PointNet++, ASIS outperforms the baseline by 0.7 mIoU. These results demonstrate that our method is also beneficial for part segmentation problem.

Method	mIoU
PointNet [26]	83.7
PointNet (*RePr*)	83.4
PointNet++ [28]*	84.3
ASIS (PN)	84.0
ASIS (PN++)	85.0

Table 6: Semantic segmentation results on ShapeNet datasets. *RePr* is our reproduced PointNet. PointNet++* denotes the PointNet++ trained by us without extra normal information.

5. Conclusion

In this paper, a novel segmentation framework, namely ASIS, is proposed for associating instance segmentation and semantic segmentation on point clouds. The relationships between the two tasks are explicitly explored and directly guide our method design. Our experiments on S3DIS

dataset and ShapeNet part dataset demonstrate the effectiveness and efficiency of ASIS. We expect wide application of the proposed method in 3D instance segmentation and 3D semantic segmentation, as well as hoping the novel design provides insights to future works on segmentation tasks, *e.g.*, panoptic segmentation, and beyond.

References

[1] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 2, 4

[2] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv: Comp. Res. Repository*, 2017. 2

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002. 3

[4] J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In *Proc. Eur. Conf. Comp. Vis.*, 2016. 2

[5] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 2

[6] B. De Brabandere, D. Neven, and L. Van Gool. Semantic instance segmentation with a discriminative loss function. *arXiv: Comp. Res. Repository*, 2017. 1, 2, 3

[7] E. Francis, K. Theodora, H. Alexander, and L. Bastian. Exploring spatial context for 3d semantic segmentation of point clouds. In *Proc. 3DRMS Workshop of Int. Conf. Computer Vision*, 2017. 3

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2014. 2

[9] J. Guerry, A. Boulch, B. Le Saux, J. Moras, A. Plyer, and D. Filliat. Snapnet-r: Consistent 3d multi-view semantic labeling for robotics. In *Proc. Workshop of Int. Conf. Computer Vision*, 2017. 3

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2017. 1, 2

[11] J. Huang and S. You. Point cloud labeling using 3d convolutional neural network. In *Proc. Int. Conf. Patt. Recogn.*, 2016. 2

[12] J. Huang and S. You. Point cloud labeling using 3d convolutional neural network. In *Proc. Int. Conf. Patt. Recogn.*, 2016. 3

[13] Q. Huang, W. Wang, and U. Neumann. Recurrent slice networks for 3d segmentation of point clouds. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2018. 3

[14] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic segmentation. *arXiv: Comp. Res. Repository*, 2018. 2

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Advances in Neural Inf. Process. Syst.*, 2012. 2

[16] L. Landrieu and M. Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2018. 1, 3

[17] Y. Li, R. Bu, M. Sun, and B. Chen. Pointcnn. *arXiv: Comp. Res. Repository*, 2018. 3

[18] G. Lin, A. Milan, C. Shen, and I. D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2

[19] G. Lin, C. Shen, A. van den Hengel, and I. Reid. Exploring context with deep structured models for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017. 2

[20] S. Liu, J. Jia, S. Fidler, and R. Urtasun. Sequential grouping networks for instance segmentation. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2017. 2, 5

[21] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 1, 2

[22] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 2008. 2

[23] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots & Systems*, 2015. 3

[24] A. Newell, Z. Huang, and J. Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Proc. Advances in Neural Inf. Process. Syst.* 2017. 2

[25] P. O. Pinheiro, R. Collobert, and P. Dollr. Learning to segment object candidates. In *Proc. Advances in Neural Inf. Process. Syst.*, 2015. 2

[26] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 2, 3, 4, 5, 8

[27] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2016. 3

[28] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proc. Advances in Neural Inf. Process. Syst.*, 2017. 2, 3, 8

[29] M. Ren and R. S. Zemel. End-to-end instance segmentation with recurrent attention. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 5

[30] G. Riegler, A. O. Ulusoy, and A. Geiger. Octnet: Learning deep 3d representations at high resolutions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 3

[31] B. Shi, S. Bai, Z. Zhou, and X. Bai. DeepPano: Deep panoramic representation for 3-d shape recognition. *IEEE Signal Processing Letters*, 2015. 3

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv: Comp. Res. Repository*, 2014. 2

[33] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2015. 3

[34] L. Tchapmi, C. Choy, I. Armeni, J. Gwak, and S. Savarese. Segcloud: Semantic segmentation of 3d point clouds. In *Proc. Int. Conf. 3D Vision*, 2017. 1, 5

[35] W. Wang, R. Yu, Q. Huang, and U. Neumann. SGPN: Similarity group proposal network for 3d point cloud instance segmentation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2018. 2, 5, 8

[36] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon. Dynamic graph cnn for learning on point clouds. *arXiv: Comp. Res. Repository*, 2018. 3, 4

[37] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2015. 3

[38] X. Ye, J. Li, H. Huang, L. Du, and X. Zhang. 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In *Proc. Eur. Conf. Comp. Vis.*, 2018. 3, 4

[39] L. Yi, V. G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, A. Sheffer, and L. Guibas. A scalable active framework for region annotation in 3d shape collections. *SIGGRAPH Asia*, 2016. 2, 5

[40] W. Zhuo, M. Salzmann, X. He, and M. Liu. Indoor scene parsing with instance segmentation, semantic labeling and support relationship inference. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2017. 5