

Contrastive Boundary Learning for Point Cloud Segmentation

Liyao Tang¹, Yibing Zhan², Zhe Chen¹, Baosheng Yu¹, Dacheng Tao^{2,1}

¹ The University of Sydney, Australia ² JD Explore Academy, China

ltan9687@uni.sydney.edu.au, zhanyibing@jd.com, {zhe.chen1, baosheng.yu}@sydney.edu.au
dacheng.tao@gmail.com

Abstract

Point cloud segmentation is fundamental in understanding 3D environments. However, current 3D point cloud segmentation methods usually perform poorly on scene boundaries, which degenerates the overall segmentation performance. In this paper, we focus on the segmentation of scene boundaries. Accordingly, we first explore metrics to evaluate the segmentation performance on scene boundaries. To address the unsatisfactory performance on boundaries, we then propose a novel contrastive boundary learning (CBL) framework for point cloud segmentation. Specifically, the proposed CBL enhances feature discrimination between points across boundaries by contrasting their representations with the assistance of scene contexts at multiple scales. By applying CBL on three different baseline methods, we experimentally show that CBL consistently improves different baselines and assists them to achieve compelling performance on boundaries, as well as the overall performance, e.g. in mIoU. The experimental results demonstrate the effectiveness of our method and the importance of boundaries for 3D point cloud segmentation. Code and model will be made publicly available at <https://github.com/LiyaoTang/contrastBoundary>.

1. Introduction

3D point cloud semantic segmentation aims to assign semantic categories to each 3D data point, while robust 3D segmentation is very important for various applications [19, 64], including autonomous driving, unmanned aerial vehicles, and augmented reality.

However, despite that various point cloud segmentation methods have been developed, little attention has been put on boundaries in 3D point clouds. Accurate segmentation on scene boundaries can be of great importance. Firstly, a clean boundary estimation can be beneficial for overall segmentation performance. For example, in 2D image segmentation, accurate segmentation on boundary is the key to generate high-fidelity masks [8, 36, 69]. Secondly, compared

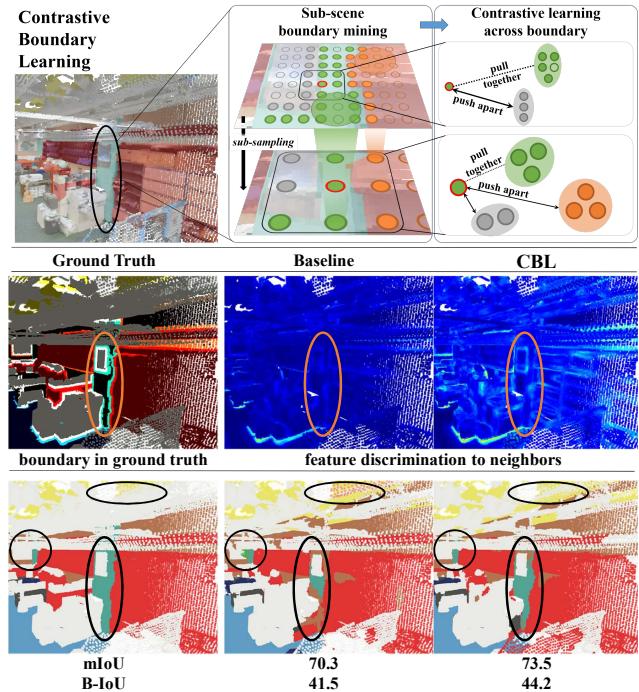


Figure 1. Contrastive Boundary Learning (top) discovers boundary from ground truth in each sub-sampled point cloud, i.e., sub-scene, through the sub-sampling procedure. By imposing contrastive optimization on boundary areas at multiple scales, CBL enhances the feature discrimination across boundaries (middle). Without an explicit boundary prediction, CBL improves boundary segmentation and achieves better scene segmentation results (bottom). The visualization is conducted on S3DIS testset Area 5.

to object categories that usually have a large portion of 3D points, such as buildings and trees, erroneous boundary segmentation could affect the recognition of object categories with much fewer points (e.g., pedestrians and pillars) to a greater extent. This can be particularly hazardous for applications like autonomous driving, e.g., crashing into curbs if boundaries are recognized inaccurately by a self-driving car.

Unfortunately, most previous 3D segmentation methods generally overlook the segmentation on scene boundaries. Though a few methods have considered boundaries, they still lack an explicit and comprehensive investigation to analyze the segmentation performance on boundary areas. They also perform unsatisfactorily on the overall segmentation performance.

Therefore, to deliver a more thorough study of the segmentation on boundaries, we first explore metrics to quantify the segmentation performance on scene boundaries. After revealing the unsatisfactory performance, we propose a novel Contrastive Boundary Learning (CBL) framework to help optimize the segmentation performance on boundaries particularly, which also consistently improves the overall performance for different baseline methods.

In particular, current popular segmentation metrics lack specific measurements on boundaries, making it difficult to reveal the boundary segmentation quality in existing methods. To make a clearer view on the performance on boundaries, we calculate the popular mean intersection-over-union (mIoU) for boundary areas and inner (non-boundary) areas separately. By comparing the performance on types of areas as well as the overall performance, the unsatisfactory performance on boundary areas can be directly revealed. Moreover, to describe the performance on boundaries more comprehensively, we consider the alignment between the boundary in the ground truth and the boundary in model segmentation results. Therefore, we introduce the popular boundary IoU [8] score (B-IoU) used in 2D instance segmentation for evaluation, which also gives a much lower score compared with the overall performance in mIoU.

After identifying the boundary segmentation difficulties, we further propose a novel contrastive boundary learning (CBL) framework to better align the boundaries of model predictions with ground-truth data's boundaries. As shown in Fig. 1, CBL optimizes a model on the feature representation of points in boundary areas, enhancing the feature discrimination across the scene boundaries. Furthermore, to make model better aware of the boundary areas at multiple semantic scales, we also develop a sub-scene boundary mining strategy, which leverages the sub-sampling procedure to discover boundary points in each sub-sampled point cloud, *i.e.*, sub-scene. Specifically, CBL operates across different sub-sampling stages and facilitates 3D segmentation methods to learn better feature representation around boundary areas.

Empirically, we experiment with three baselines across four datasets. We first present the unsatisfactory performance on boundary areas when using current point cloud segmentation methods and then show that CBL can assist baseline in achieving promising boundary and overall performance. For example, the proposed CBL helps RandLA-Net surpass current state-of-the-art methods on the Seman-

tic3D dataset and enables a basic ConvNet to achieve leading performance on the S3DIS dataset.

Our contributions are as follows:

- We explore the boundary problem in current 3D point cloud segmentation and quantify it with metrics that consider boundary area, *e.g.*, boundary IoU. The results reveal that current methods deliver much worse accuracy in boundary areas than their overall performance.
- We propose a novel Contrastive Boundary Learning (CBL) framework, which improves the feature representation by contrasting the point features across the scene boundaries. It thus improves the segmentation performance around boundary areas and subsequently the overall performance.
- We conduct extensive experiments and show that CBL can bring significant and consistent improvements on boundary area as well as overall performance across all baselines. These empirical results further demonstrate that CBL is effective for improving boundary segmentation performance, and accurate boundary segmentation is important for robust 3D segmentation.

2. Related work

Point cloud segmentation. Point cloud semantic segmentation aims to assign semantic labels to each 3D point. Recent deep learning methods have taken over traditional methods [33, 50] that use hand-crafted features, which can be roughly divided into projection-based and point-based methods.

Projection-based methods project 3D points to grid-like structure, either 2D image [7, 32, 43, 65] or 3D voxels [9, 22, 52, 57]. For the 2D image plane, we can make use of existing studies for 2D image processing. However, a complete 3D segmentation generally requires taking multiple viewpoints and re-projection [4, 35], which may result in surface occlusions. For 3D voxels, sparse convolutions [16, 17, 61] are proposed to alleviate the resource consumption in voxel construction, considering the large emptiness in 3D space. In general, the voxel resolution incurs the trade-off between losing detail and being resource-demanding [46]. Point-based network directly operates on 3D points, while a pioneering work in this direction is PointNet [45], which uses point-wise MLPs to process per-point feature. Following this success, recent works adopt an encoder-decoder paradigm [47]. Various local aggregation modules are proposed to examine the local context in point clouds, including 3D convolution [3, 37, 53, 62], attentional operations [18, 26, 73], and graph-based operation [34, 60]. To better process unstructured point cloud, sub-sampling [5, 11, 67, 68], up-sampling [48, 56], and post-processing modules [28, 41] are also considered to enhance

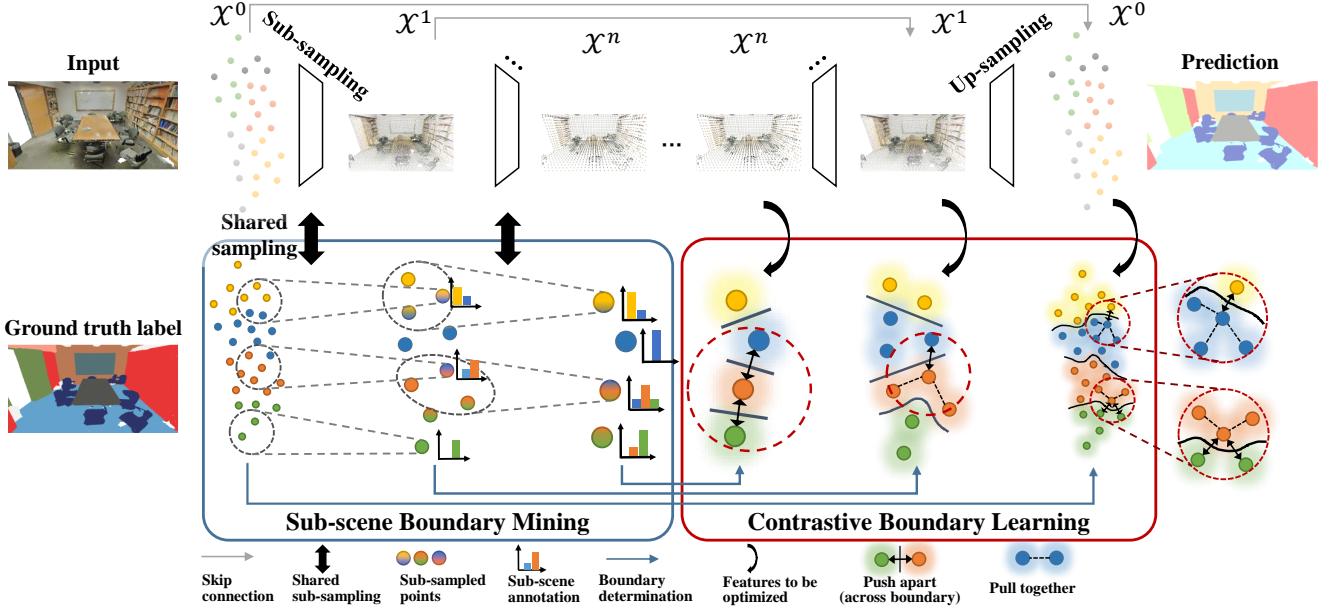


Figure 2. The detailed illustration of the Contrastive Boundary Learning.

point cloud representation. Despite these developments in different modules, the boundary in point cloud segmentation has rarely been explored.

Boundary in segmentation. Boundary problem has a long history in 2D image processing [8, 36, 42, 69], whereas only few works [15, 28] realize the significance of boundary in 3D point cloud segmentation. However, both works involve complex modules for explicit boundary prediction [15, 28] or local aggregation [28]. These operations largely increase the model complexity, yet yield limited performance gain for overall metric. Regarding segmentation performance on boundaries, they also only give qualitative results. In comparison, we present a contrastive learning framework that brings little overhead to the model and can improve upon various baselines with simple adaption. Additionally, we would like to note that, we for the first time, quantify the boundary quality with numeric metric, and demonstrate that boundary problem is indeed widely existing across current methods.

Contrastive learning. Contrastive learning [6, 13, 23, 31, 55, 66] has shown promising performance in representation learning for computer vision tasks, ranging from unsupervised settings to supervised settings. In recent works, contrastive learning has also been introduced into 2D segmentation [58, 59] as well as unsupervised representation learning in point cloud processing [25, 39, 63]. Especially, PointContrast [63] conducts point-wise contrastive learning to overcome geometric transformation, such as rigid transformation. P4Contrast [39] suggests a more flexible contrasting strategy to promote multi-modal fusion between geometric and RGB information. In contrast, in our work, we take

a supervised setting and demonstrate with CBL that contrastive learning is well-suited for improving segmentation quality on boundary areas. Additionally, unlike the above works that only use points at input point cloud, we utilize the sub-sampled point cloud to examine scene context at multiple scales.

3. Segmentation on Boundaries

Since most of the current works focus on the improvement of general metrics, such as mean intersection over union (mIoU), overall accuracy (OA), and mean average precision (mAP), the boundary quality in point cloud segmentation is usually overlooked. Unlike recent boundary-related works [15, 28] that give only qualitative results on boundaries, we are the first to quantify the quality of segmentation on boundaries. Particularly, we introduce a series of metrics for presentation, including mIoU@boundary, mIoU@inner and the boundary IoU (B-IoU) score from 2D instance segmentation tasks [8].

Based on ground-truths data, we consider a point as a boundary point if there exist points that have a different annotated label in its neighborhood. Similarly, for model predictions, we consider a point as a boundary point if there exist nearby points with a different predicted label. More formally, we note the point cloud as \mathcal{X} and the i -th point as x_i , whose local neighborhood is $\mathcal{N}_i = \mathcal{N}(x_i)$, corresponding ground truth label is l_i , and the model predicted label is p_i . We further note the set of boundary points in ground-truth as \mathcal{B}_l and those in predicted segmentation as \mathcal{B}_p , thus

we have:

$$\begin{aligned}\mathcal{B}_l &= \{x_i \in \mathcal{X} \mid \exists x_j \in \mathcal{N}_i, l_j \neq l_i\}, \\ \mathcal{B}_p &= \{x_i \in \mathcal{X} \mid \exists x_j \in \mathcal{N}_i, p_j \neq p_i\},\end{aligned}\quad (1)$$

where we set \mathcal{N}_i to be the radius neighborhood with a radius of 0.1 following the common practice [40, 53].

To examine the boundary segmentation results, an intuitive way is to calculate the mIoU within the boundary area, *i.e.*, mIoU@boundary. To further compare the model performance in boundary and non-boundary (inner) area, we further calculate the mIoU in the inner area, *i.e.* mIoU@inner. Given that mIoU is calculated on the whole point cloud \mathcal{X} as:

$$\text{mIoU}(\mathcal{X}) = \frac{1}{K} \sum_{k=1}^K \frac{\sum_{x_i \in \mathcal{X}} \mathbb{1}[p_i = k \wedge l_i = k]}{\sum_{x_j \in \mathcal{X}} \mathbb{1}[p_j = k \vee l_j = k]}, \quad (2)$$

where K is the total number of classes and $\mathbb{1}[\cdot]$ represents a boolean function that outputs 1 if the condition within $[\cdot]$ is true and 0 otherwise. We have the mIoU@boundary and mIoU@inner defined as:

$$\begin{aligned}\text{mIoU}@\text{boundary} &= \text{mIoU}(\mathcal{B}_l), \\ \text{mIoU}@@\text{inner} &= \text{mIoU}(\mathcal{X} - \mathcal{B}_l),\end{aligned}\quad (3)$$

where $\mathcal{X} - \mathcal{B}_l$ is the set of points in inner area.

However, the mIoU@boundary and mIoU@inner do not consider the false boundary in model predicted segmentation. Inspired by boundary IoU [8] for 2D instance segmentation, for better evaluation, we consider the alignment between boundary in segmentation predictions and boundary in ground truth data. It thus leads to the following B-IoU for evaluation:

$$\text{B-IoU} = \frac{|\mathcal{B}_l \cap \mathcal{B}_p|}{|\mathcal{B}_l \cup \mathcal{B}_p|}. \quad (4)$$

4. Method

In this section, we present our contrastive boundary learning (CBL) framework, shown in Fig. 2. It imposes contrastive learning to enhance the feature discrimination across boundaries. Then, to deeply augment the model performance on boundaries, we enable the CBL in sub-sampled point clouds, *i.e.*, sub-scene, through the sub-scene boundary mining.

Contrastive Boundary Learning. We follow the widely used InfoNCE loss [55] and its generalization [13, 20] to define the contrastive optimization goal on boundary points. In particular, for a boundary point $x_i \in \mathcal{B}_l$, we encourage learned representations more similar to its neighbor points from the same category and more distinguished from other

neighbor points from different categories, *i.e.*,

$$L_{CBL} = \frac{-1}{|\mathcal{B}_l|} \sum_{x_i \in \mathcal{B}_l} \log \frac{\sum_{x_j \in \mathcal{N}_i \wedge l_j = l_i} \exp(-d(f_i, f_j)/\tau)}{\sum_{x_k \in \mathcal{N}_i} \exp(-d(f_i, f_k)/\tau)}, \quad (5)$$

where f_i is the feature of x_i , $d(\cdot, \cdot)$ is a distance measurement and τ is the temperature in contrastive learning. The contrastive learning described by Eq. (5) focuses on boundary points only (the dashed circles in red in Fig. 2). First, we consider all the boundary points \mathcal{B}_l from ground-truth data as defined in Eq. (1). Then, for each point $x_i \in \mathcal{B}_l$, we restrict the sampling of its positive and negative points to be within its local neighborhood \mathcal{N}_i . With such strong spatial restriction, we obtain positive pairs for x_i as $\{x_j \in \mathcal{N}_i \wedge l_j = l_i\}$, and other neighboring points, *i.e.* $\{x_j \in \mathcal{N}_i \wedge l_j \neq l_i\}$, are negative pairs. Therefore, the contrastive learning enhances the feature discrimination across scene boundaries, which is important for improving segmentation on boundary areas.

Sub-scene Boundary Mining. To better explore scene boundaries, we examine the boundaries in sub-sampled point clouds at multiple scales, which enables the contrastive boundary learning on different sub-sampling stages of a backbone model. Collecting boundary points from the input point cloud is straightforward with the ground truth label. However, after sub-sampling, it is difficult to obtain a proper definition of boundary point set following Eq. (1), due to the undefined label for sub-sampled points [14]. Therefore, to enable CBL in sub-sampled point cloud, we propose the sub-scene boundary mining that determines the set of ground-truth boundary points in each sub-sampling stage. Specifically, we use superscripts to denote stage. At the sub-sampling stage n , we represent its sub-sampled point cloud as \mathcal{X}^n . For input point cloud, we have $\mathcal{X}^0 = \mathcal{X}$. When collecting a set of boundary points $\mathcal{B}_l^n \in \mathcal{X}^n$ in stage n , it is required to determine the label l_i^n of a sub-sampled point $x_i^n \in \mathcal{X}^n$, *i.e.*, the sub-scene annotation. As each sub-sampled point $x_i^n \in \mathcal{X}^n$ is aggregated from a group of points in its previous point cloud \mathcal{X}^{n-1} ; we thus utilize the sub-sampling procedure to determine the label iteratively. We take l_i^0 to be the one-hot label of ground truth label l_i for point $x_i^0 = x_i$, and have the following:

$$l_i^n = \text{AVG}(\{l_j^{n-1} | x_j^{n-1} \in \mathcal{N}^{n-1}(x_i^n)\}), \quad (6)$$

where $\mathcal{N}^{n-1}(x_i^n)$ denotes the local neighbors of x_i^n in previous stage (the dashed circles in grey in Fig. 2), *i.e.*, the group of points aggregated from \mathcal{X}^{n-1} to be represented by the single point $x_i^n \in \mathcal{X}^n$ after sub-sampling procedure, and AVG is the average-pooling.

With Eq. (6) and ground-truth labels, we can iteratively obtain the sub-scene annotation l_i^n as a distribution, whose

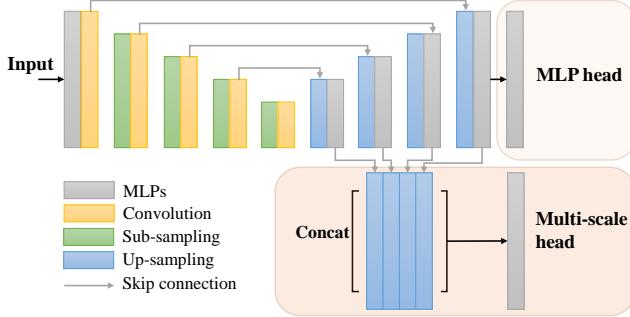


Figure 3. The architecture of the 3D ConvNet model, which follows the widely adopted encoder-decoder paradigm, with an optional multi-scale prediction head. More details are provided in the appendix.

k -th location describes the proportion of k -th class in its corresponding group of points in the input point cloud. To determine the set of boundary points in sub-sampled point cloud \mathcal{X}^n , we simply take $\arg \max l_i^n$ to allow the evaluation of boundary point in Eq. (1)¹, and use the feature of sub-sampled point for the contrastive boundary optimization in Eq. (5). Finally, with sub-scene boundary mining, we have CBL applied at all stages and the final loss is

$$L = L_{\text{cross entropy}} + \lambda \sum_n L_{\text{CBL}}^n, \quad (7)$$

where L_{CBL}^n is the CBL loss at stage n and λ is the loss weight.

5. Implementation Details and Baselines

As 3D ConvNet has been a popular backbone model for point cloud processing, to present a generalized implementation, we illustrate with a ConvNet baseline (Fig. 3) as a case study for applying CBL in point cloud processing. Following [2, 24], we build the ConvNet with convolution in 3D continuous space:

$$f_i = (h \circ g)(x_i) = \sum_{x_j \in \mathcal{N}_i} g(x_i - x_j)h(x_j), \quad (8)$$

where \circ denotes convolution operator and the continuous kernel $g(\cdot)$ is approximated by one-layer MLP and set $h(x_j) = f_j$ to simply use the feature of point x_j . We note that the 3D convolution in Eq. (8) is purely based on spatial location between the center point and its neighbors, compared to other advanced local aggregation modules that utilize the local context [26, 73].

To better utilize the boundary features optimized by CBL at multiple scales, we use a multi-scale head for prediction, which simply concatenates the point feature from each

¹We choose $\arg \max$ for its simplicity and non-parametric nature. We provide more analysis on this choice in the appendix.

| methods | overall | mIoU | | B-IoU |
|--------------------|--------------|--------------|--------------|--------------|
| | | @boundary | @inner | |
| pointnet [45] | 41.1 | 30.2 | 53.4 | 35.6 |
| KPConv [53] | 67.3 | 50.5 | 71.1 | 58.9 |
| JSE-Net [28]* | 67.7 | 50.5 | 71.4 | 60.9 |
| RandLA-Net [26] | 62.6 | 44.1 | 65.8 | 45.4 |
| CloserLook3D [40] | 66.9 | 50.0 | 70.7 | 59.2 |
| ConvNet | 67.4 | 50.1 | 71.2 | 59.6 |
| RandLA-Net + CBL | 65.3 +2.7 | 47.4 +3.3 | 67.2 +1.4 | 49.9 +4.5 |
| CloserLook3D + CBL | 67.5 +0.6 | 50.6 +0.6 | 71.0 +0.3 | 60.4 +1.2 |
| ConvNet + CBL | 69.4 +2.0 | 52.6 +2.5 | 73.1 +1.9 | 61.5 +1.9 |

Table 1. The results are obtained on the S3DIS datasets testset Area 5, following the instruction of the officially released code of each method. Method with * also consider boundaries.

sub-sampled point cloud into the last output layer. As we would show in the ablation study (Sec. 6.3), such concatenation across multiple scales fails without the CBL. Note that CBL can be married to any other multi-stage backbone. Specifically, we also apply the CBL to two other popular baselines: the RandLA-Net [26] and CloserLook3D [40], to demonstrate the generalizability. RandLA-Net leverages random sampling and attentive local aggregation to handle the large-scale scene with fast processing; CloserLook3D proposes a parameter-free PosPool module that largely reduces model parameters and resources consumption, while achieving comparable performance against other methods with parametric aggregation module, such as KPConv [53]. Together with the ConvNet baseline, our experiments cover the backbone with most of the typical local aggregation methods for point cloud, ranging from convolution, attentional operation, to parameter-free operation. For training, we follow the setup of baseline and set the loss weight $\lambda = 0.1$. More details will be provided in the appendix.

6. Experiments

We first present the boundary problem with experiments. We then evaluate the benefits of the proposed CBL on multiple large-scale point cloud segmentation datasets, including in-door scenes (S3DIS [1], ScanNet [10]) and out-door scenes (Semantic3D [21], NPM3D [49]).

6.1. The Boundary Problem in Experiment

We experimentally compare the score given by mIoU, mIoU@boundary, mIoU@inner as well as the B-IoU. As shown in Tab. 1, for recent 3D point cloud segmentation methods, the mIoU@boundary is much lower than the mIoU@inner. With the overall performance sitting between these two scores, it suggests that it is the boundary area that degrades the overall segmentation performance.

| methods | mIoU | OA | mACC | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|-------------------|-------------|-------------|-------------|-------------|-------------|-------------|------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| PointNet [45] | 41.1 | - | 49.0 | 88.8 | 97.3 | 69.8 | 0.1 | 3.9 | 46.3 | 10.8 | 59.0 | 52.6 | 5.9 | 40.3 | 26.4 | 33.2 |
| SegCloud [52] | 48.9 | - | 57.4 | 90.1 | 96.1 | 69.9 | 0.0 | 18.4 | 38.4 | 23.1 | 70.4 | 75.9 | 40.9 | 58.4 | 13.0 | 41.6 |
| PointCNN [37] | 57.3 | 85.9 | 63.9 | 92.3 | 98.2 | 79.4 | 0.0 | 17.6 | 22.8 | 62.1 | 74.4 | 80.6 | 31.7 | 66.7 | 62.1 | 56.7 |
| SPGraph [34] | 58.0 | 86.4 | 66.5 | 89.4 | 96.9 | 78.1 | 0.0 | 42.8 | 48.9 | 61.6 | 84.7 | 75.4 | 69.8 | 52.6 | 2.1 | 52.2 |
| PCT [18] | 61.3 | - | 67.7 | 92.5 | 98.4 | 80.6 | 0.0 | 19.4 | 61.6 | 48.0 | 76.6 | 85.2 | 46.2 | 67.7 | 67.9 | 52.3 |
| HPEIN [30] | 61.9 | 87.2 | 68.3 | 91.5 | 98.2 | 81.4 | 0.0 | 23.3 | 65.3 | 40.0 | 75.5 | 87.7 | 58.5 | 67.8 | 65.6 | 49.4 |
| MinkowskiNet [9] | 65.4 | - | 71.7 | 91.8 | 98.7 | 86.2 | 0.0 | 34.1 | 48.9 | 62.4 | 81.6 | 89.8 | 47.2 | 74.9 | 74.4 | 58.6 |
| KPConv [53] | 67.1 | - | 72.8 | 92.8 | 97.3 | 82.4 | 0.0 | 23.9 | 58.0 | 69.0 | 81.5 | 91.0 | 75.4 | 75.3 | 66.7 | 58.9 |
| JSENet [28]* | 67.7 | - | - | 93.8 | 97.0 | 83.0 | 0.0 | 23.2 | 61.3 | 71.6 | 89.9 | 79.8 | 75.6 | 72.3 | 72.7 | 60.4 |
| CGA-Net [41] | 68.6 | - | - | 94.5 | 98.3 | 83.0 | 0.0 | 25.3 | 59.6 | 71.0 | 92.2 | 82.6 | 76.4 | 77.7 | 69.5 | 61.5 |
| RandLA-Net [26] | 62.4 | 87.2 | 71.4 | 91.1 | 95.6 | 80.2 | 0.0 | 24.7 | 62.3 | 47.7 | 76.2 | 83.7 | 60.2 | 71.1 | 65.7 | 53.8 |
| + CBL | 65.3 | 87.5 | 74.5 | 92.2 | 97.7 | 81.0 | 0.0 | 36.8 | 61.0 | 39.4 | 78.1 | 88.1 | 81.4 | 71.5 | 68.7 | 52.6 |
| CloserLook3D [40] | 66.9 | 90.0 | 72.1 | 94.8 | 98.4 | 82.5 | 0.0 | 25.5 | 51.3 | 70.9 | 92.1 | 81.9 | 76.7 | 70.1 | 64.5 | 61.2 |
| + CBL | 67.5 | 90.2 | 72.7 | 94.9 | 98.4 | 83.1 | 0.0 | 27.3 | 55.0 | 71.2 | 91.9 | 82.9 | 75.9 | 71.3 | 63.5 | 60.4 |
| ConvNet | 67.4 | 90.1 | 72.9 | 94.1 | 98.1 | 83.1 | 0.0 | 24.9 | 53.5 | 73.0 | 91.7 | 82.3 | 76.5 | 72.3 | 66.9 | 60.8 |
| + CBL | 69.4 | 90.6 | 75.2 | 93.9 | 98.4 | 84.2 | 0.0 | 37.0 | 57.7 | 71.9 | 91.7 | 81.8 | 77.8 | 75.6 | 69.1 | 62.9 |

Table 2. Quantitative results on S3DIS Area 5 dataset [1], showing the mean IoU (mIoU) overall accuracy (OA) and the mean accuracy (mACC). The red denotes improvement over baseline and the bold or bold denotes the best performance. Method with * also consider boundaries in their design.

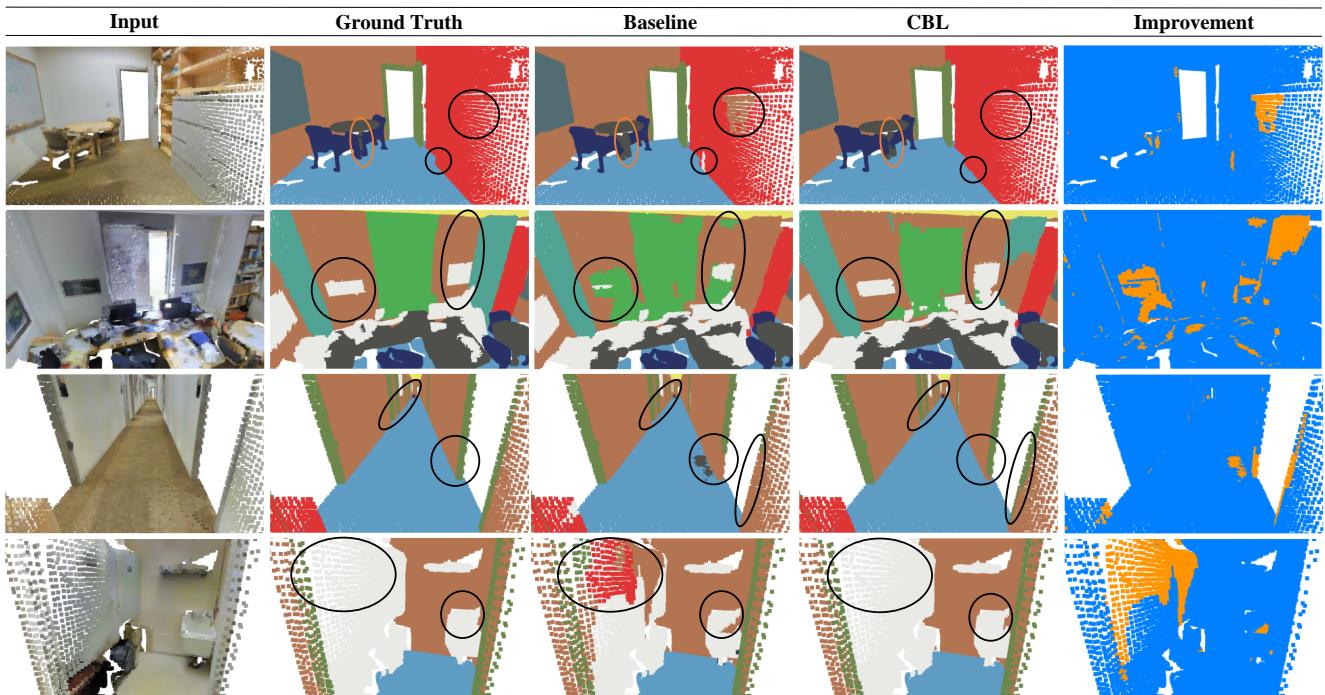


Figure 4. We compare the results of ConvNet baseline with CBL on several different scenes and show that the improvements are from boundaries. In offices (top 2), CBL can effectively improve the results on boundary areas, especially in a cluttered one (2nd row). In the last two rows (hallway and others), CBL avoids unnecessary boundaries, and repairs the missing boundary between walls and doors/objects at the right place. The visualization is done on S3DIS testset Area 5.

Similarly, B-IoU also agrees with the mIoU@boundary by giving a score that is far lagged behind the general performance of mIoU score. Hence, such observation indicates the unsatisfied segmentation quality on boundary areas. While with the proposed CBL, the improvement on both mIoU@boundary and B-IoU is larger than the improvement on overall mIoU as well as the mIoU@inner, across all three baselines. Due to the limited space, we

provide more thorough studies in presenting the boundary problem in the appendix.

6.2. Performance Comparison

S3DIS Indoor Scene Segmentation. S3DIS [1] is a challenging point cloud dataset of indoor scenes. It contains 3D RGB point clouds of 6 indoor areas covering 272 rooms. Each point is annotated with one of the 13 semantic cat-

| methods | mIoU | OA | mACC | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|-----------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| PointNet [45] | 47.6 | 78.6 | 66.2 | 88.0 | 88.7 | 69.3 | 42.4 | 23.1 | 47.5 | 51.6 | 54.1 | 42.0 | 9.6 | 38.2 | 29.4 | 35.2 |
| RSNet [29] | 56.5 | - | 66.5 | 92.5 | 92.8 | 78.6 | 32.8 | 34.4 | 51.6 | 68.1 | 59.7 | 60.1 | 16.4 | 50.2 | 44.9 | 52.0 |
| SPG [34] | 62.1 | 86.4 | 73.0 | 89.9 | 95.1 | 76.4 | 62.8 | 47.1 | 55.3 | 68.4 | 73.5 | 69.2 | 63.2 | 45.9 | 8.7 | 52.9 |
| PointCNN [37] | 65.4 | 88.1 | 75.6 | 94.8 | 97.3 | 75.8 | 63.3 | 51.7 | 58.4 | 57.2 | 71.6 | 69.1 | 39.1 | 61.2 | 52.2 | 58.6 |
| PointWeb [72] | 66.7 | 87.3 | 76.2 | 93.5 | 94.2 | 80.8 | 52.4 | 41.3 | 64.9 | 68.1 | 71.4 | 67.1 | 50.3 | 62.7 | 62.2 | 58.5 |
| ShellNet [71] | 66.8 | 87.1 | - | 90.2 | 93.6 | 79.9 | 60.4 | 44.1 | 64.9 | 52.9 | 71.6 | 84.7 | 53.8 | 64.6 | 48.6 | 59.4 |
| RandLA-Net [26] | 70.0 | 88.0 | 82.0 | 93.1 | 96.1 | 80.6 | 62.4 | 48.0 | 64.4 | 69.4 | 69.4 | 76.4 | 60.0 | 64.2 | 65.9 | 60.1 |
| KPConv [53] | 70.6 | - | 79.1 | 93.6 | 92.4 | 83.1 | 63.9 | 54.3 | 66.1 | 76.6 | 57.8 | 64.0 | 69.3 | 74.9 | 61.3 | 60.3 |
| SCF-Net [12] | 71.6 | 88.4 | 82.7 | 93.3 | 96.4 | 80.9 | 64.9 | 47.4 | 64.5 | 70.1 | 71.4 | 81.6 | 67.2 | 64.4 | 67.5 | 60.9 |
| BAAF [48] | 72.2 | 88.9 | 83.1 | 93.3 | 96.8 | 81.6 | 61.9 | 49.5 | 65.4 | 73.3 | 72.0 | 83.7 | 67.5 | 64.3 | 67.0 | 62.4 |
| ConvNet | 69.7 | 88.6 | 76.8 | 93.8 | 91.9 | 84.2 | 46.3 | 52.1 | 66.7 | 78.5 | 75.2 | 72.8 | 70.1 | 71.7 | 57.1 | 61.3 |
| + CBL | 73.1 | 89.6 | 79.4 | 94.1 | 94.2 | 85.5 | 50.4 | 58.8 | 70.3 | 78.3 | 75.7 | 75.0 | 71.8 | 74.0 | 60.0 | 62.4 |

Table 3. Quantitative results on S3DIS [1] with 6-fold cross validation. The red denotes improvement over baseline and the bold or bold denotes the best performance.

| methods | mIoU (%) | OA (%) | man-made. | natural. | high veg. | low veg. | buildings | hard scape | scanning art. | cars |
|-----------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|---------------|-------------|
| SnapNet [4] | 59.1 | 88.6 | 82.0 | 77.3 | 79.7 | 22.9 | 91.1 | 18.4 | 37.3 | 64.4 |
| SEGCloud [52] | 61.3 | 88.1 | 83.9 | 66.0 | 86.0 | 40.5 | 91.1 | 30.9 | 27.5 | 64.3 |
| SPG [34] | 73.2 | 94.0 | 97.4 | 92.6 | 87.9 | 44.0 | 83.2 | 31.0 | 63.5 | 76.2 |
| RGNet [54] | 74.7 | 94.5 | 97.5 | 93.0 | 88.1 | 48.1 | 94.6 | 36.2 | 72.0 | 68.0 |
| KPConv [53] | 74.6 | 92.9 | 90.9 | 82.2 | 84.2 | 47.9 | 94.9 | 40.0 | 77.3 | 79.7 |
| RFCR [14] | 77.8 | 94.3 | 94.2 | 89.1 | 85.7 | 54.4 | 95.0 | 43.8 | 76.2 | 83.7 |
| SCF-Net [12] | 77.6 | 94.7 | 97.1 | 91.8 | 86.3 | 51.2 | 95.3 | 50.5 | 67.9 | 80.7 |
| ConvNet | 72.8 | 92.6 | 92.2 | 79.9 | 84.4 | 41.3 | 95.2 | 41.2 | 62.6 | 85.6 |
| + CBL | 75.0 | 94.0 | 96.2 | 90.1 | 84.0 | 47.5 | 94.7 | 36.0 | 64.8 | 86.3 |
| RandLA-Net [26] | 77.4 | 94.8 | 95.6 | 91.4 | 86.6 | 51.5 | 95.7 | 51.5 | 69.8 | 76.8 |
| + CBL | 78.4 | 95.0 | 95.3 | 91.3 | 87.9 | 55.6 | 96.3 | 56.2 | 65.9 | 78.2 |

Table 4. Quantitative results on Semantic3D reduced-8 benchmark [21]. The metrics shown the mean IoU (mIoU) and overall accuracy (OA) obtained from benchmark site with only the recent published works included. The red denotes improvement over baseline and the bold or bold denotes the best performance.

egories, e.g., ceiling, floor, clutter. As shown in Tab. 2, our methods consistently improve across all three baselines, showing to be effective with different local aggregation modules. Notably, the improvements are much more significant in classes, such as column (+13 compared to ConvNet baseline), than in other classes with large areas, such as wall and ceiling. Such observation shows our effectiveness on boundary areas; and with the consistent improvement across different classes, it also suggests that the CBL is NOT trading off between scenes of major and minor classes, but is indeed separating them more clearly. With the benefit of a cleaner boundary, the ConvNet finally achieves a leading performance of 69.4 in mIoU.

We further demonstrate qualitatively in Fig. 4 that, the CBL effectively improves the overall performance by improving segmentation on boundary areas. Compared with JSENet [28] that also considers boundaries, we demonstrate our superiority by obtaining a much larger relative improvement to our baselines than that made by JSENet on its baseline, i.e., KPConv [53], especially in classes that boundaries are important, e.g., column, window, sofa, bookcase and clutter, as well as the overall performance. To avoid overfitting on S3DIS Area 5, we further conduct the 6-fold cross-validation, with the result reported in Tab. 3. A large

improvement is also shown in column (+9.5), and consistent improvement is made across all classes except one (-0.2). Therefore, the proposed CBL can be indeed regarded as a general and effective method, achieving 73.1 in mIoU with a common ConvNet baseline.

Semantic3D Outdoor Scene Segmentation. In addition to improvement on S3DIS [1], we demonstrate the generalizability across different types of scenes by evaluating CBL on point cloud collected at the outdoor environment, the Semantic3D [21] dataset. It is a large-scale dataset comprising over 4 billion points and provides 15 large point clouds for training, with each point annotated to one of the 8 classes, e.g., cars, buildings. We use the reduced-8 benchmark and present the quantitative results in Tab. 4. We evaluate with both ConvNet and RandLA-Net [26] as baselines and observe consistent improvements. Especially, RandLA-Net has achieved state-of-the-art performance on multiple outdoor datasets and the improvement made on it can better demonstrate the effectiveness of our CBL. Notably, significant improvement is made in the high vegetation and low vegetation class, which are two classes that confuse most of the other methods. It is because the high/low vegetation usually co-exists at a near spatial distance and has a similar appearance, e.g., trees surrounded by bushes/grass,

| methods | modality | mIoU (%) |
|--------------------|------------|--------------|
| DCM-Net [51] | 3D + Mesh | 65.8 |
| VMNet [27] | | 74.6 |
| SparseConvNet [16] | | 72.5 |
| MinkowskiNet [9] | | 73.6 |
| O-CNN [57] | 3D (voxel) | 76.2 |
| OccuSeg [22] | | 76.4 |
| Mix3D [44] | | 78.1 |
| BA-GEM [15]* | | 63.5 |
| PointConv [62] | | 66.6 |
| PointASNL [67] | | 66.6 |
| KP-Conv [53] | 3D (point) | 68.4 |
| FusionNet [70] | | 68.8 |
| JSENNet [28]* | | 69.9 |
| RFCR [14] | | 70.2 |
| ConvNet + CBL | 3D (point) | 69.1 70.5 |

Table 5. Quantitative results on ScanNet [10] benchmark. Performance is taken from the official benchmark site by the time of submission. Methods with * also consider boundaries.

| methods | mIoU (%) |
|-------------------|--------------|
| HDGCN [38] | 68.3 |
| ConvPoint [2] | 75.9 |
| RandLANet [26] | 78.5 |
| KP-Conv [53] | 82.0 |
| FKACconv [3] | 82.7 |
| PyramidPoint [56] | 82.9 |
| ConvNet + CBL | 76.2 78.6 |

Table 6. Quantitative results on Paris-Lille-3D of NPM3D [49] benchmark, results obtained from online benchmark site by the time of submission.

which makes the separation of these two scenes challenging. The large improvement in both of these two classes demonstrates the effective improvement on scene boundaries. Lastly, with CBL, RandLA-Net obtains a leading performance of 78.4 in mIoU.

Further experiments on NPM3D and ScanNet. To further demonstrate the generalization of the proposed CBL, we report on another two popular dataset, the ScanNet [10] (indoor scene) and NPM3D [49] (outdoor scene). As shown in Tab. 5 and Tab. 6, our method achieves reasonable results and consistent improvement over the baseline. It thus shows that CBL is robust to different baselines, datasets, and types of scenes. Detailed results are available in the appendix.

6.3. Ablation Studies

We conduct ablation studies on the ScanNet validation set to evaluate the effectiveness of different components in the proposed CBL scheme.

| | CBL @input | CBL @sub-scenes | mIoU(%) | OA(%) |
|------------------------------|---------------|--------------------|--------------------|--------------------|
| ConvNet | ✓ | | 69.71 - | 88.97 - |
| | ✓ | ✓ | 70.05 +0.34 | 89.01 +0.04 |
| | | | 70.98 +1.27 | 89.31 +0.34 |
| ConvNet (multiscale head) | ✓ | ✓ | 69.83 +0.12 | 88.88 -0.09 |
| | | | 71.33 +1.62 | 89.40 +0.43 |

Table 7. Results on validation set of ScanNet [10]. The CBL @input refers to only conduct contrastive boundary learning on the input point cloud (with point feature extracted from last upsampled stage), and @sub-scene refers to the CBL with sub-scene boundary mining. The red indicates relative improvement.

The Effectiveness of CBL. As shown in Tab. 7, the direct application of CBL on the input point cloud (without sub-scene boundary mining) can improve the performance, which demonstrates that boundary areas are worth more attention. By introducing sub-scene boundary mining, a more significant improvement is gained, as boundaries at multiple scales are identified and optimized in the CBL.

The Effect of Multi-scale Head. Comparing the ConvNet baseline with and without the multi-scale head, we find that a direct application of multi-scale head can even hurt the performance (-0.09 in OA). It shows that a direct concatenation across multiple scales can not bring much benefit. In contrast, with multi-scale head, ConvNet with CBL is further boosted to gain a larger improvement in both mIoU and OA. It shows that the main improvement is originated from the more discriminative features learned by CBL at different sub-sampled point clouds.

7. Conclusion

In this paper, we comprehensively analyze the segmentation performance on scene boundaries for the current point cloud segmentation methods. We show that the current segmentation accuracy on boundaries is unsatisfactory and quantitatively present the boundary problem with metrics, including mIoU@boundary and B-IoU. We further propose Contrastive Boundary Learning (CBL) to explicitly optimize the feature on boundaries and improve the model performance on boundaries. The leading performance and consistent improvement across various baselines and datasets demonstrate the effectiveness of CBL and the importance of scene boundaries in 3D point cloud segmentation.

Limitation and future work. One of our limitation is that we mainly concentrate on the scene boundaries while ignoring the broad inner areas. Therefore, in the future, we would like to further explore the role of boundary in point cloud segmentation and its relation with inner areas.

Acknowledgement. Dr Baosheng Yu and Mr Liyao Tang are supported by ARC FL-170100117, and Dr Zhe Chen is supported by ARC IH-180100002.

References

- [1] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 5, 6, 7, 14, 15
- [2] Alexandre Bouchnak. Convpoint: Continuous convolutions for point cloud processing. *Computers & Graphics*, 88:24 – 34, 2020. 5, 8, 14
- [3] Alexandre Bouchnak, Gilles Puy, and Renaud Marlet. Fkaconv: Feature-kernel alignment for point cloud convolution, 2020. 2, 8, 14
- [4] A. Bouchnak, B. Le Saux, and N. Audebert. Unstructured point cloud semantic labeling using deep segmentation networks. In *Proceedings of the Workshop on 3D Object Retrieval, 3Dor ’17*, page 17–24, Goslar, DEU, 2017. Eurographics Association. 2, 7
- [5] Chen Chen, Zhe Chen, Jing Zhang, and Dacheng Tao. Sasa: Semantics-augmented set abstraction for point-based 3d object detection. In *AAAI*, 2022. 2
- [6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 3
- [7] Zhe Chen, Jing Zhang, and Dacheng Tao. Progressive lidar adaptation for road detection. *IEEE/CAA Journal of Automatica Sinica*, 6:693–702, 05 2019. 2
- [8] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C. Berg, and Alexander Kirillov. Boundary IoU: Improving object-centric image segmentation evaluation. In *CVPR*, 2021. 1, 2, 3, 4
- [9] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019. 2, 6, 8, 15
- [10] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 5, 8, 12, 15
- [11] Oren Dovrat, Itai Lang, and Shai Avidan. Learning to sample. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. 2
- [12] Siqi Fan, Qilei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14504–14513, June 2021. 7
- [13] Nicholas Frosst, Nicolas Papernot, and Geoffrey E. Hinton. Analyzing and improving representations with the soft nearest neighbor loss. In *ICML*, 2019. 3, 4
- [14] Jingyu Gong, Jiachen Xu, Xin Tan, Haichuan Song, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Omni-supervised point cloud segmentation via gradual receptive field component reasoning. *CoRR*, abs/2105.10203, 2021. 4, 7, 8, 15
- [15] Jingyu Gong, Jiachen Xu, Xin Tan, Jie Zhou, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Boundary-aware geometric encoding for semantic segmentation of point clouds, 2021. 3, 8, 15
- [16] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. 2, 8, 15
- [17] Benjamin Graham and Laurens van der Maaten. Submanifold sparse convolutional networks. *ArXiv*, abs/1706.01307, 2017. 2
- [18] Meng-Hao Guo, Junxiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R. Martin, and Shi-Min Hu. PCT: point cloud transformer. *CoRR*, abs/2012.09688, 2020. 2, 6
- [19] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1
- [20] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. *Journal of Machine Learning Research - Proceedings Track*, 9:297–304, 01 2010. 4
- [21] Timo Hackel, N. Savinov, L. Ladicky, Jan D. Wegner, K. Schindler, and M. Pollefeys. SEMANTIC3D.NET: A new large-scale point cloud classification benchmark. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume IV-1-W1, pages 91–98, 2017. 5, 7
- [22] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Ocuseg: Occupancy-aware 3d instance segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2937–2946, 2020. 2, 8, 15
- [23] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019. 3
- [24] P. Hermosilla, T. Ritschel, P-P Vazquez, A. Vinacua, and T. Ropinski. Monte carlo convolution for learning on non-uniformly sampled point clouds. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2018)*, 37(6), 2018. 5
- [25] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. *CoRR*, abs/2012.09165, 2020. 3
- [26] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. *CoRR*, abs/1911.11236, 2019. 2, 5, 6, 7, 8, 12, 13, 14
- [27] Zeyu Hu, Xuyang Bai, Jiaxiang Shang, Runze Zhang, Jiayu Dong, Xin Wang, Guangyuan Sun, Hongbo Fu, and Chiew-Lan Tai. Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation. *CoRR*, abs/2107.13824, 2021. 8, 15
- [28] Zeyu Hu, Mingmin Zhen, Xuyang Bai, Hongbo Fu, and Chiew-Lan Tai. Jsenet: Joint semantic segmentation and edge detection network for 3d point clouds. *CoRR*, abs/2007.06888, 2020. 2, 3, 5, 6, 7, 8, 15
- [29] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds.

- 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2626–2635, 2018. 7
- [30] L. Jiang, H. Zhao, S. Liu, X. Shen, C. Fu, and J. Jia. Hierarchical point-edge interaction network for point cloud semantic segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10432–10440, 2019. 6
- [31] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020. 3
- [32] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David A. Ross, Brian Brewington, Thomas A. Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3d semantic segmentation. In *ECCV*, 2020. 2
- [33] Loïc Landrieu, Hugo Raguet, Bruno Vallet, Clément Mallet, and Martin Weinmann. A structured regularization framework for spatially smoothing semantic labelings of 3d point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 132:102–118, 2017. 2
- [34] L. Landrieu and M. Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4558–4567, 2018. 2, 6, 7
- [35] Felix Järemo Lawin, Martin Danelljan, Patrik Tostberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Deep projective 3d semantic segmentation. *Lecture Notes in Computer Science*, page 95–107, 2017. 2
- [36] Hong Joo Lee, Jung Uk Kim, Sangmin Lee, Hak Gu Kim, and Yong Man Ro. Structure boundary preserving segmentation for medical image with ambiguous boundary. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 3
- [37] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhua Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 2, 6, 7
- [38] Zhidong Liang, Ming Yang, Liuyuan Deng, Chunxiang Wang, and Bing Wang. Hierarchical depthwise graph convolutional neural network for 3d semantic segmentation of point clouds. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8152–8158, 2019. 8, 14
- [39] Yunze Liu, Li Yi, Shanghang Zhang, Qingnan Fan, Thomas A. Funkhouser, and Hao Dong. P4contrast: Contrastive learning with pairs of point-pixel pairs for RGB-D scene understanding. *CoRR*, abs/2012.13089, 2020. 3
- [40] Ze Liu, Han Hu, Yue Cao, Zheng Zhang, and Xin Tong. A closer look at local aggregation operators in point cloud analysis. *ECCV*, 2020. 4, 5, 6, 12, 13
- [41] Tao Lu, Limin Wang, and Gangshan Wu. Cga-net: Category guided aggregation for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11693–11702, June 2021. 2, 6
- [42] Dmitri Marin, Zijian He, Peter Vajda, Priyam Chatterjee, Sam Tsai, Fei Yang, and Yuri Boykov. Efficient segmentation: Learning downsampling near semantic boundaries. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 3
- [43] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4213–4220, 2019. 2
- [44] Alexey Nekrasov, Jonas Schult, Or Litany, Bastian Leibe, and Francis Engelmann. Mix3d: Out-of-context data augmentation for 3d scenes, 2021. 8, 15
- [45] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593, 2016. 2, 5, 6, 7
- [46] Charles R. Qi, Hao Su, Matthias NieBner, Angela Dai, Mengyuan Yan, and Leonidas J. Guibas. Volumetric and multi-view cnns for object classification on 3d data. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2016. 2
- [47] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017. 2
- [48] Shi Qiu, Saeed Anwar, and Nick Barnes. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. *CoRR*, abs/2103.07074, 2021. 2, 7
- [49] Xavier Roynard, Jean-Emmanuel Deschaud, and François Goulette. Paris-lille-3d: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *The International Journal of Robotics Research*, 37(6):545–557, 2018. 5, 8, 14
- [50] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009. 2
- [51] Jonas Schult, Francis Engelmann, Theodora Kontogianni, and Bastian Leibe. Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. *CoRR*, abs/2004.01002, 2020. 8, 15
- [52] Lyne Tchapmi, Christopher Choy, Iro Armeni, Jun Young Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3d point clouds. *2017 International Conference on 3D Vision (3DV)*, Oct 2017. 2, 6, 7
- [53] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J. Guibas. Kpconv: Flexible and deformable convolution for point clouds. *CoRR*, abs/1904.08889, 2019. 2, 4, 5, 6, 7, 8, 13, 14, 15
- [54] G. Truong, S. Z. Gilani, S. M. S. Islam, and D. Suter. Fast point cloud registration using semantic segmentation. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8, 2019. 7
- [55] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018. 3, 4
- [56] Nina Varney, Vijayan K. Asari, and Quinn Graehling. Pyramid point: A multi-level focusing network for revisiting feature layers, 2020. 2, 8, 14

- [57] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis. *ACM Transactions on Graphics (SIGGRAPH)*, 36(4), 2017. [2](#), [8](#), [15](#)
- [58] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. *CoRR*, abs/2101.11939, 2021. [3](#)
- [59] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training, 2020. [3](#)
- [60] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph CNN for learning on point clouds. *CoRR*, abs/1801.07829, 2018. [2](#)
- [61] Zongji Wang and Feng Lu. Voxsegnet: Volumetric cnns for semantic part segmentation of 3d shapes. *IEEE transactions on visualization and computer graphics*, 2018. [2](#)
- [62] Wenxuan Wu, Zhongang Qi, and Fuxin Li. Pointconv: Deep convolutional networks on 3d point clouds. *CoRR*, abs/1811.07246, 2018. [2](#), [8](#), [15](#)
- [63] Saining Xie, Jiatao Gu, Demi Guo, Charles R. Qi, Leonidas J. Guibas, and Or Litany. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. *CoRR*, abs/2007.10985, 2020. [3](#)
- [64] Y. Xie, J. Tian, and X. X. Zhu. Linking points with labels in 3d: A review of point cloud semantic segmentation. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):38–59, 2020. [1](#)
- [65] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeeze-seg3: Spatially-adaptive convolution for efficient point-cloud segmentation. *ArXiv*, abs/2004.01803, 2020. [2](#)
- [66] Yufei Xu, Qiming Zhang, Jing Zhang, and Dacheng Tao. Regioncl: Can simple region swapping contribute to contrastive learning?, 2021. [3](#)
- [67] Xu Yan, Chaoda Zheng, Zhen Li, Sheng Wang, and Shuguang Cui. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. *CoRR*, abs/2003.00492, 2020. [2](#), [8](#), [15](#)
- [68] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. Modeling point clouds with self-attention and gumbel subset sampling. *CoRR*, abs/1904.03375, 2019. [2](#)
- [69] Jianlong Yuan, Zelu Deng, Shu Wang, and Zhenbo Lui. Multi receptive field network for semantic segmentation. *CoRR*, abs/2011.08577, 2020. [1](#), [3](#)
- [70] Feihu Zhang, Jin Fang, Benjamin W. Wah, and Philip H. S. Torr. Deep fusionnet for point cloud semantic segmentation. In *ECCV*, 2020. [8](#), [15](#)
- [71] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct 2019. [7](#)
- [72] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia. Pointweb: Enhancing local neighborhood features for point cloud processing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. [7](#)
- [73] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer, 2021. [2](#), [5](#), [14](#), [15](#)

A. Introduction.

In this supplementary material, we provide more details regarding baseline architecture (Appendix B), the boundary problem Appendix C, visualization results (Appendix D), the training setup (Appendix E), the effect of temperature (Appendix F), the effect of design regarding sub-scene annotation (Appendix G), and experiment results (Appendix H).

Especially, CBL achieves a new stat-of-the-art on S3DIS with the newly released transformer model (Tab. 14).

B. Architecture of ConvNet Baseline

We show the specific architecture of our ConvNet baseline in Fig. 5. With a consistent notation, \mathcal{X}^n is the point cloud in sub-sampling stage n , f_i is the feature of point x_i , and $N^n = |\mathcal{X}^n|$ with $N = N^0$. We use the multi-scale head on all baselines when adapting the CBL.

C. Further Analysis on Boundary Problem

We further account for the type of areas and class-specific analysis for better exploring the boundary problem. Specifically, we provide per-class IoU score that is separately calculated on boundary area \mathcal{B}_l and inner area $\mathcal{X} - \mathcal{B}_l$.

As shown in Tab. 9, we evaluate for all three baselines with and without the proposed CBL. We notice that, large improvements are made on small objects, *e.g.* column, which aligns with the observation in Tab. 2 in main paper. We would like to add that, despite that CBL focuses only on boundaries, improvements are also made on inner area. We hypothesize the reason might be that the false boundary in model predicted segmentation is restrained, as features in inner area implicitly becomes more similar when the features across boundaries are optimized to be more distinctive by the CBL.

Moreover, for all three baselines, the improvement on boundary area is much more than that made on inner area, which is summarized in Tab. 8.

Therefore, with metrics separately calculated on boundary and inner area, we clearly see that the improvement brought by CBL is mainly from the boundary areas. Such observation further emphasizes the importance of clear scene boundaries in point cloud segmentation task.

D. More Visualizations

We provide more qualitative results as a support for the improvement made by CBL on boundaries. The visualization results include various scenes, including rooms (Fig. 7), cluttered space (Fig. 8), hallways (Fig. 9), and offices (Fig. 10). For each scene, we further attempt to visualize the features discrimination between center points and their corresponding neighbors and the results are presented

| baselines (+ CBL) | mIoU | | OA | | mACC | |
|-------------------|----------|-------|----------|-------|----------|-------|
| | boundary | inner | boundary | inner | boundary | inner |
| RandLA-Net [26] | +3.3 | +1.4 | +4.1 | -0.3 | +3.4 | +2.4 |
| CloserLook3D [40] | +0.6 | +0.2 | +0.1 | +0.2 | +0.7 | +0.4 |
| ConvNet | +2.5 | +2.0 | +1.0 | +0.7 | +3.2 | +2.8 |

Table 8. The improvement brought by CBL on different baselines and types of area (boundary / inner area).

in the every second row. Specifically, we calculate the normalized feature distance between the point feature f_i and features of its neighboring points $\{f_j | x_j \in \mathcal{N}_i\}$. We then take the mean distance for visualization.

According to the presented figures, it shows that the CBL significantly enhances the feature distances around the scene boundaries and improves the baseline to obtain a more detailed and cleaner boundary in prediction for different type of scenes. The visualization is done on S3DIS testset Area 5.

E. Training Setup in Details

For the RandLA-Net [26] and CloserLook3D [40] baselines, we follow their instructions of released code for training and evaluation, which are [here](#) (RandLA-Net) and [here](#) (CloserLook3D), respectively. Especially, in CloserLook3D [40], there are two non-parametric module, we use the one with sin/cos spatial embedding.

For the ConvNet baseline, we use the SGD optimizer to train for 600 epoch, with a weight decay of 0.001. We set the initial learning rate to 0.01 and use a momentum of 0.98 with a decay rate of $0.1^{1/200}$. It roughly takes 24 hours to train on 4 Nividia v100 GPUs, and we does not observe obvious increase in training time after applying the CBL.

F. Effect of Temperature in CBL

We conduct empirical study on ScanNet [10] validation set to analyze the effect of temperature τ in the CBL (Eq. (5)). We use the ConvNet baseline and train for 600 epoch on training set. As shown in Tab. 10, we find that the proper temperature for CBL is within $(0.5, 2)$, and we set the temperature to $\tau = 1$ by default.

G. Effect of Design of Sub-scene annotation

While the sub-scene annotation is a distribution, we only use the simple arg max when evaluating the boundary points. Therefore, it raises two particular question: 1) is it necessary to maintain the distribution? 2) is there any better way in utilizing the sub-scene annotation than the arg max?

In this section, we explore other alternatives and answer to this two questions with a particular focus of how they affect the model performance on boundaries.

Necessities of maintaining distribution. There are two main reasons to leverage the average pooling on labels and maintain the distribution. First, current methods may not

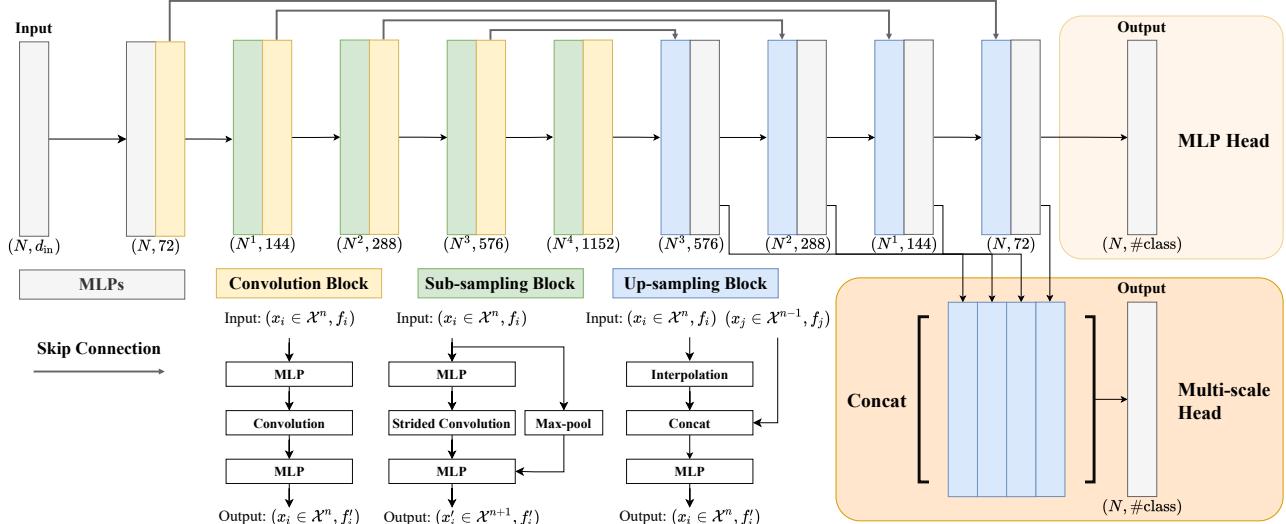


Figure 5. The detail architecture of ConvNet baseline.

| methods | mIoU | OA | mACC | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|-------------------|-------------|-------------|-------------|-------------|-------------|-------------|------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| RandLA-Net [26] | 44.1 | 67.1 | 59.1 | 65.5 | 69.4 | 52.2 | 0.0 | 21.4 | 28.6 | 55.0 | 55.0 | 56.0 | 41.1 | 41.2 | 45.8 | 42.1 |
| + CBL | 47.4 | 71.2 | 62.5 | 78.2 | 85.9 | 56.0 | 0.0 | 30.3 | 25.7 | 42.6 | 58.4 | 60.9 | 50.0 | 42.5 | 52.2 | 44.2 |
| CloserLook3D [40] | 50.0 | 76.6 | 58.5 | 80.7 | 88.6 | 63.9 | 0.0 | 21.1 | 15.6 | 57.5 | 73.3 | 64.7 | 52.2 | 43.1 | 37.2 | 52.6 |
| + CBL | 50.6 | 76.7 | 59.2 | 80.9 | 88.6 | 64.6 | 0.0 | 26.5 | 15.6 | 55.9 | 73.0 | 65.0 | 50.4 | 47.6 | 38.4 | 51.2 |
| ConvNet | 50.1 | 76.5 | 58.3 | 80.4 | 88.3 | 63.5 | 0.0 | 26.5 | 15.2 | 58.3 | 72.1 | 63.4 | 52.3 | 40.8 | 38.7 | 52.2 |
| + CBL | 52.6 | 77.5 | 61.5 | 80.5 | 88.8 | 65.7 | 0.0 | 32.5 | 20.9 | 61.8 | 71.7 | 62.4 | 52.5 | 46.7 | 47.4 | 52.5 |

(a) The full metrics calculated on boundary points from ground truth (*i.e.*, \mathcal{B}_l) only.

| methods | mIoU | OA | mACC | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|-------------------|-------------|-------------|-------------|-------------|-------------|-------------|------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| RandLA-Net [26] | 65.8 | 89.6 | 73.0 | 93.3 | 98.6 | 84.6 | 0.0 | 25.9 | 65.7 | 46.5 | 81.1 | 88.9 | 65.4 | 75.5 | 71.9 | 58.2 |
| + CBL | 67.2 | 89.3 | 75.4 | 93.0 | 99.1 | 84.6 | 0.0 | 37.3 | 64.1 | 39.4 | 82.7 | 91.5 | 79.3 | 75.9 | 73.9 | 56.0 |
| CloserLook3D [40] | 70.7 | 92.2 | 75.2 | 96.4 | 99.9 | 86.5 | 0.0 | 25.9 | 55.1 | 76.5 | 95.9 | 87.1 | 81.9 | 75.1 | 72.5 | 66.2 |
| + CBL | 70.9 | 92.4 | 75.6 | 96.5 | 99.9 | 86.9 | 0.0 | 27.0 | 59.3 | 78.1 | 95.7 | 87.7 | 80.8 | 75.4 | 69.4 | 65.6 |
| ConvNet | 71.2 | 92.1 | 75.5 | 95.0 | 99.8 | 85.9 | 0.0 | 34.6 | 56.0 | 82.7 | 95.4 | 87.4 | 81.3 | 73.8 | 68.4 | 65.7 |
| + CBL | 73.2 | 92.8 | 78.3 | 95.3 | 99.9 | 88.0 | 0.0 | 38.4 | 62.2 | 76.4 | 95.9 | 87.5 | 82.7 | 81.2 | 75.2 | 68.6 |

(b) The full metrics calculated on inner points from ground truth (*i.e.*, $\mathcal{X} - \mathcal{B}_p$) only.

Table 9. The improvement CBL brought on baselines, separately calculated in boundary area (a) and inner area (b). The red denotes improvement is made on baseline.

| temperature | mIoU | OA | mACC |
|-------------|-------|-------|-------|
| 0.3 | 70.67 | 89.16 | 77.91 |
| 0.5 | 70.98 | 89.31 | 78.27 |
| 1 | 71.33 | 89.40 | 78.69 |
| 2 | 70.73 | 89.10 | 77.98 |
| 10 | 70.03 | 88.97 | 77.58 |

Table 10. The effect of temperature on CBL.

preserve the original input points after sub-sampling, *e.g.* grid sub-sampling in KPConv [53]. Therefore, the original label of a sub-sampled point is not presented and the sub-scene annotation is thus demanded. Although we may use the label of the nearest point for approximation, Tab. 12 shows that CBL (nearest) is sub-optimal. Second, despite that we only use the “argmax” result of the sub-scene annotation, maintaining distribution still preserves more information

than just maintaining “argmax” result. As “argmax” discards the minor classes during sampling, such elimination of minority may further accumulate through more sub-sampling stages and leads to imprecise boundary, as depicted in Fig. 6. Experimentally, in Tab. 12, though CBL (argmax) improves boundary (B-IoU), it compromises overall performance.

Better treatment than Argmax. While “argmax” is straight forward, it introduces the problem of ”label-flipping” when the distribution of sub-scene annotation is close to a uniform distribution, *i.e.*, when the number of points of different classes are roughly the same.

To avoid this, we leverage the KL divergence as a measure of the semantic distance among sub-scene annotations. We then threshold on the KL-distance to determine if two sub-scene annotations belong to the same semantic class

| | mIoU (%) | Ground | Building | Pole | Bollard | Trash can | Barrier | Pedestrian | Car | Natural |
|-------------------|-------------|--------|-------------|-------------|-------------|-------------|-------------|------------|-------------|---------|
| HDGCN [38] | 68.3 | 99.4 | 93.0 | 67.7 | 75.7 | 25.7 | 44.7 | 37.1 | 81.9 | 89.6 |
| ConvPoint [2] | 75.9 | 99.5 | 95.1 | 71.6 | 88.7 | 46.7 | 52.9 | 53.5 | 89.4 | 85.4 |
| RandLANet [26] | 78.5 | 99.5 | 97.0 | 71.0 | 86.7 | 50.5 | 65.5 | 49.1 | 95.3 | 91.7 |
| KP-Conv [53] | 82.0 | 99.5 | 94.0 | 71.3 | 83.1 | 78.7 | 47.7 | 78.2 | 94.4 | 91.4 |
| FKACConv [3] | 82.7 | 99.6 | 98.1 | 77.2 | 91.1 | 64.7 | 66.5 | 58.1 | 95.6 | 93.9 |
| PyramidPoint [56] | 82.9 | 99.6 | 97.1 | 74.6 | 84.3 | 56.0 | 65.9 | 79.1 | 95.1 | 93.9 |
| ConvNet | 76.2 | 99.5 | 96.3 | 68.5 | 67.4 | 41.4 | 41.5 | 80.6 | 96.3 | 94.1 |
| + CBL | 78.6 | 99.5 | 96.7 | 72.1 | 72.6 | 46.2 | 60.4 | 70.1 | 97.2 | 93.2 |

Table 11. Quantitative results on Paris-Lille-3D of NPM3D [49] benchmark, results obtained from online benchmark site by the time of submission. The red denotes the improvement made on baseline.

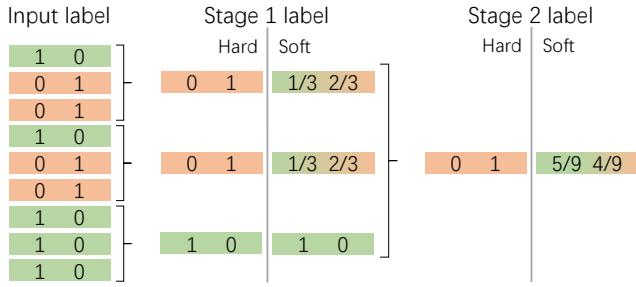


Figure 6. With every 3 points being sub-sampled into 1 in each stage, tracking distribution (soft label) describes original input faithfully, but hard label fails due to accumulated errors.

| methods | mIoU | | | B-IoU |
|-------------------------|---------|-----------|--------|-------|
| | overall | @boundary | @inner | |
| ConvNet | 67.4 | 50.1 | 71.2 | 59.6 |
| ConvNet + CBL | 69.4 | 52.6 | 73.1 | 61.5 |
| ConvNet + CBL (nearest) | 68.3 | 52.1 | 71.8 | 60.9 |
| ConvNet + CBL (argmax) | 66.8 | 50.6 | 70.4 | 60.6 |
| ConvNet + CBL (kl) | 69.5 | 52.5 | 73.2 | 62.0 |

Table 12. Same setting as in Tab. 1 in main paper.

or not, which further enables us to determine the boundary points in sub-sampled point cloud. Specifically, we set the threshold to 0.5 and CBL (kl) can bring a small improvement on overall performance, and a slightly larger boost on boundary performance, as in Tab. 12. Yet, as “thresholding KL distance” introduces extra hyper-parameters and complexity, we opt for “argmax” for simplicity in the main paper.

Summary. Therefore, we summarize the reason for designing the sub-scene annotation as a distribution as it can preserve much more information and can be extended to a more robust boundary determination using KL-distance.

H. Further Experiments

Results on ScanNet and NPM3D datasets. We provide the detail results on ScanNet in Tab. 13; and the detail results on NPM3D in Tab. 11.

CBL with Transformer. We use the open-source code

base ([here](#)) to re-produce the performance of newly released point Transformer [73] on S3DIS [1] Area 5 dataset.

In Tab. 14, the same consistent improvement is made on classes such as column. CBL with better boundaries further boosts the overall performance to 71.0 in mIoU, achieving a new state-of-the-art performance.

| Method | mIoU | bathhtub | bed | books. | cabinet | chair | counter | curtain | desk | door | floor | other | pic | fridge | shower | sink | sofa | table | toilet | wall | wndw |
|----------------------|------|----------|------|--------|---------|-------|---------|---------|------|------|-------|-------|------|--------|--------|------|------|-------|--------|------|------|
| DCM-Net [51] | 65.8 | 77.8 | 70.2 | 80.6 | 61.9 | 81.3 | 46.8 | 69.3 | 49.4 | 52.4 | 94.1 | 44.9 | 29.8 | 51.0 | 82.1 | 67.5 | 72.7 | 56.8 | 82.6 | 80.3 | 63.7 |
| VMNet [27] | 74.6 | 87.0 | 83.8 | 85.8 | 72.9 | 85.0 | 50.1 | 87.4 | 58.7 | 65.8 | 95.6 | 56.4 | 29.9 | 76.5 | 90.0 | 71.6 | 81.2 | 63.1 | 93.9 | 85.8 | 70.9 |
| SparseConvNet [16] | 72.5 | 64.7 | 82.1 | 84.6 | 72.1 | 86.9 | 53.3 | 75.4 | 60.3 | 61.4 | 95.5 | 57.2 | 32.5 | 71.0 | 87.0 | 72.4 | 82.3 | 62.8 | 93.4 | 86.5 | 68.3 |
| MinkowskiNet [9] | 73.6 | 85.9 | 81.8 | 83.2 | 70.9 | 84.0 | 52.1 | 85.3 | 66.0 | 64.3 | 95.1 | 54.4 | 28.6 | 73.1 | 89.3 | 67.5 | 77.2 | 68.3 | 87.4 | 85.2 | 72.7 |
| O-CNN [57] | 76.4 | 75.8 | 79.6 | 83.9 | 74.6 | 90.7 | 56.2 | 85.0 | 68.0 | 67.2 | 97.8 | 61.0 | 33.5 | 77.7 | 81.9 | 84.7 | 83.0 | 69.1 | 97.2 | 88.5 | 72.7 |
| OccuSeg [22] | 76.2 | 92.4 | 82.3 | 84.4 | 77.0 | 85.2 | 57.7 | 84.7 | 71.1 | 64.0 | 95.8 | 59.2 | 21.7 | 76.2 | 88.8 | 75.8 | 81.3 | 72.6 | 93.2 | 86.8 | 74.4 |
| Mix3D [44] | 78.1 | 96.4 | 85.5 | 84.3 | 78.1 | 85.8 | 57.5 | 83.1 | 68.5 | 71.4 | 97.9 | 59.4 | 31.0 | 80.1 | 89.2 | 84.1 | 81.9 | 72.3 | 94.0 | 88.7 | 72.5 |
| BA-GEM [15]* | 63.5 | | | | | | | | | | | | | | | | | | | | |
| PointConv [62] | 66.6 | 78.1 | 75.9 | 69.9 | 64.4 | 82.2 | 47.5 | 77.9 | 56.4 | 50.4 | 95.3 | 42.8 | 20.3 | 58.6 | 75.4 | 66.1 | 75.3 | 58.8 | 90.2 | 81.3 | 64.2 |
| PointASNL [67] | 66.6 | 70.3 | 78.1 | 75.1 | 65.5 | 83.0 | 47.1 | 76.9 | 47.4 | 53.7 | 95.1 | 47.5 | 27.9 | 63.5 | 69.8 | 67.5 | 75.1 | 55.3 | 81.6 | 80.6 | 70.3 |
| KP-Conv [53] | 68.4 | 84.7 | 75.8 | 78.4 | 64.7 | 81.4 | 47.3 | 77.2 | 60.5 | 59.4 | 93.5 | 45.0 | 18.1 | 58.7 | 80.5 | 69.0 | 78.5 | 61.4 | 88.2 | 81.9 | 63.2 |
| FusionNet [70] | 68.8 | 70.4 | 74.1 | 75.4 | 65.6 | 82.9 | 50.1 | 74.1 | 60.9 | 54.8 | 95.0 | 52.2 | 37.1 | 63.3 | 75.6 | 71.5 | 77.1 | 62.3 | 86.1 | 81.4 | 65.8 |
| JSENNet [28] | 69.9 | 88.1 | 76.2 | 82.1 | 66.7 | 80.0 | 52.2 | 79.2 | 61.3 | 60.7 | 93.5 | 49.2 | 20.5 | 57.6 | 85.3 | 69.1 | 75.8 | 65.2 | 87.2 | 82.8 | 64.9 |
| RFCR [14] | 70.2 | 88.9 | 74.5 | 81.3 | 67.2 | 81.8 | 49.3 | 81.5 | 62.3 | 61.0 | 94.7 | 47.0 | 24.9 | 59.4 | 84.8 | 70.5 | 77.9 | 64.6 | 89.2 | 82.3 | 61.1 |
| ConvNet + CBL | 70.5 | 76.9 | 77.5 | 80.9 | 68.7 | 82.0 | 43.9 | 81.2 | 66.1 | 59.1 | 94.5 | 51.5 | 17.1 | 63.3 | 85.6 | 72.0 | 79.6 | 66.8 | 88.9 | 84.7 | 68.9 |

Table 13. Quantitative results on ScanNet [10] benchmark, results obtained from online benchmark site by the time of submission. We group method by the 3D representation type, which is respectively, from top to down, 3D + mesh, 3D voxel and 3D point, and we also use 3D point. The empty line denotes no record of detailed performance found. The method with * also considers boundary.

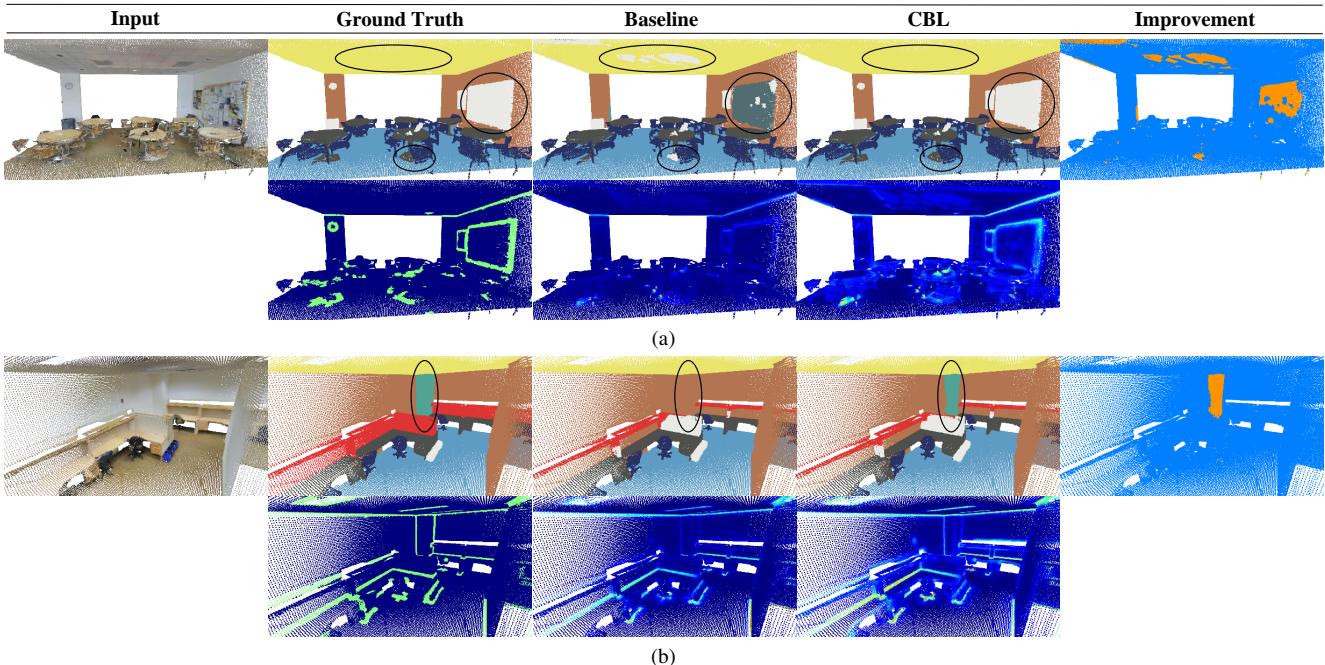


Figure 7. Large rooms. We compare the results of ConvNet baseline with CBL. On the every second row, we visualize the boundary points calculated from the ground truth label, and the feature discrimination among neighboring points for each model. The improvement on the first row and the enhanced feature discrimination on the second row show that CBL improves the features across boundaries to obtain a better segmentation quality on boundary areas. The visualization is done on S3DIS testset Area 5.

| methods | mIoU | OA | mACC | ceiling | floor | wall | beam | column | window | door | table | chair | sofa | bookcase | board | clutter |
|------------------------|--------------|--------------|--------------|---------|-------|--------------|------|--------------|--------------|--------------|-------------|-------|-------|----------|--------------|---------|
| pt trans [73]* | 70.4 | 90.8 | 76.5 | 94.0 | 98.5 | 86.3 | 0.0 | 38.0 | 63.4 | 74.3 | 89.1 | 82.4 | 74.3 | 80.2 | 76.0 | 59.3 |
| pt trans [73] + CBL | 70.0 | 90.5 | 76.5 | 95.2 | 98.6 | 85.1 | 0.0 | 36.7 | 62.5 | 75.9 | 81.5 | 91.0 | 75.1 | 71.9 | 76.4 | 60.2 |
| | 71.0* | 90.9* | 77.5* | 94.3* | 98.3 | 87.4* | 0.0 | 42.1* | 64.0* | 78.5* | 82.5 | 88.9* | 75.1* | 71.1 | 81.3* | 59.6* |

Table 14. Quantitative results on S3DIS Area 5 dataset [1], showing the mean IoU (mIoU), overall accuracy (OA), mean accuracy (mACC), and per-class IoU scores. We include both performance reported in original paper (with *, the first row) and the re-produced performance (without *, the second row). We use red to denote improvement over the re-produced point transformer, and * to denote the improvement over the performance reported in original paper.

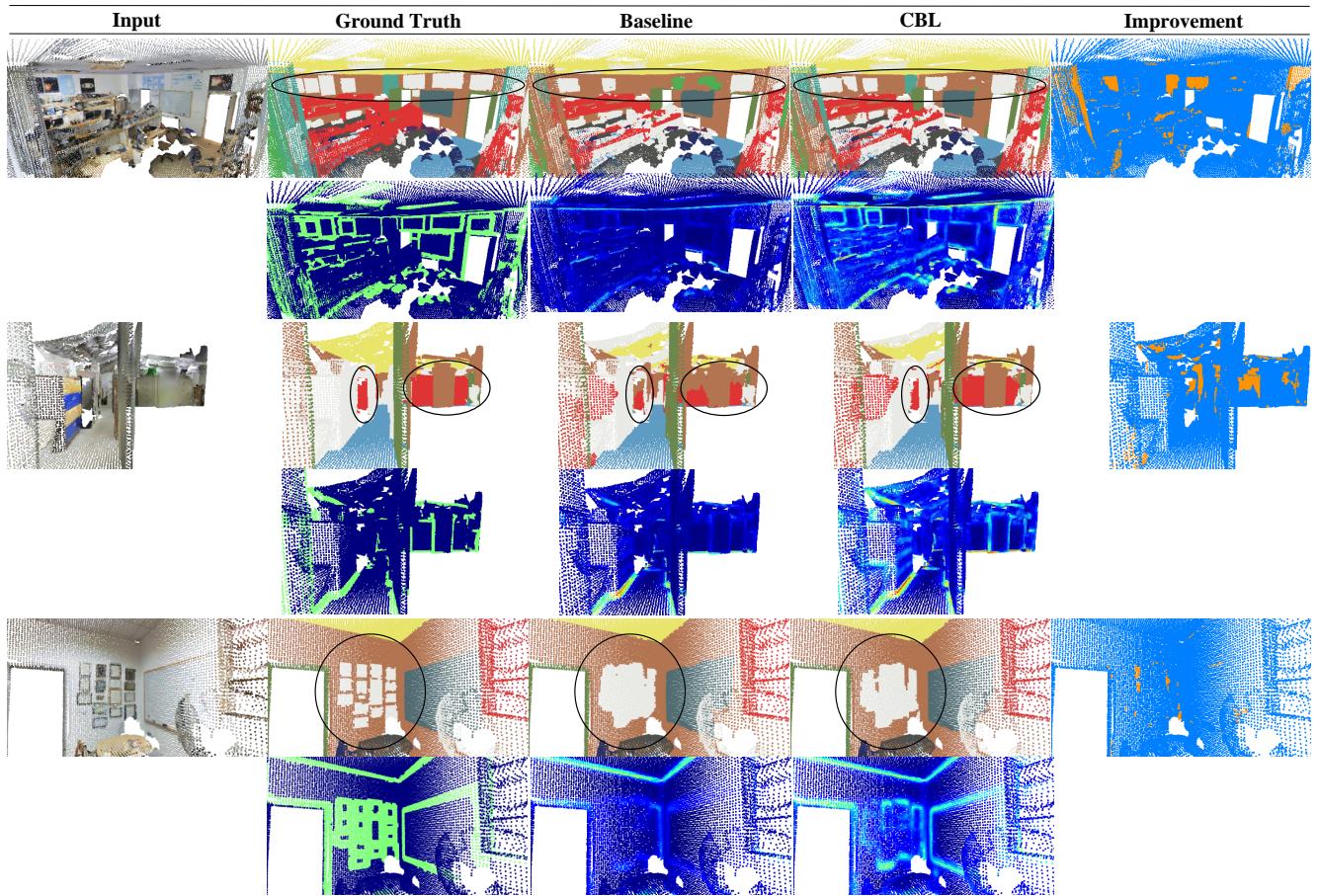


Figure 8. Cluttered space. Same as above (Fig. 7).

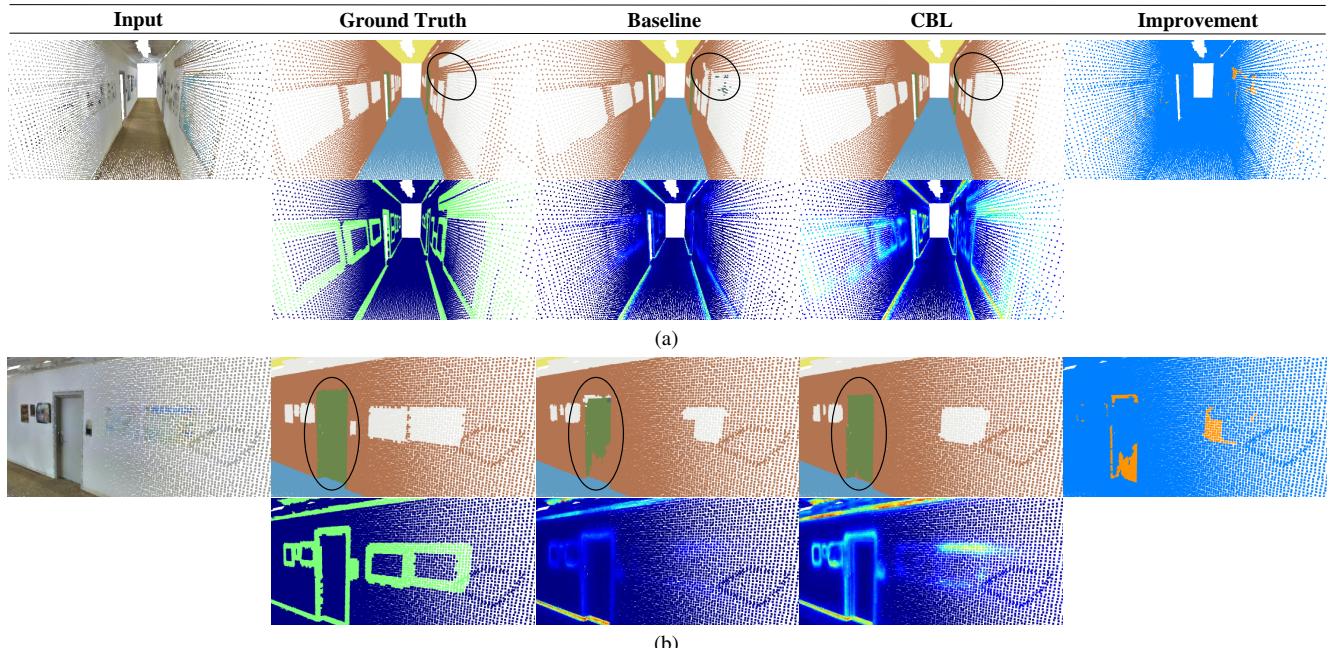


Figure 9. Hallways. Same as above (Fig. 7).

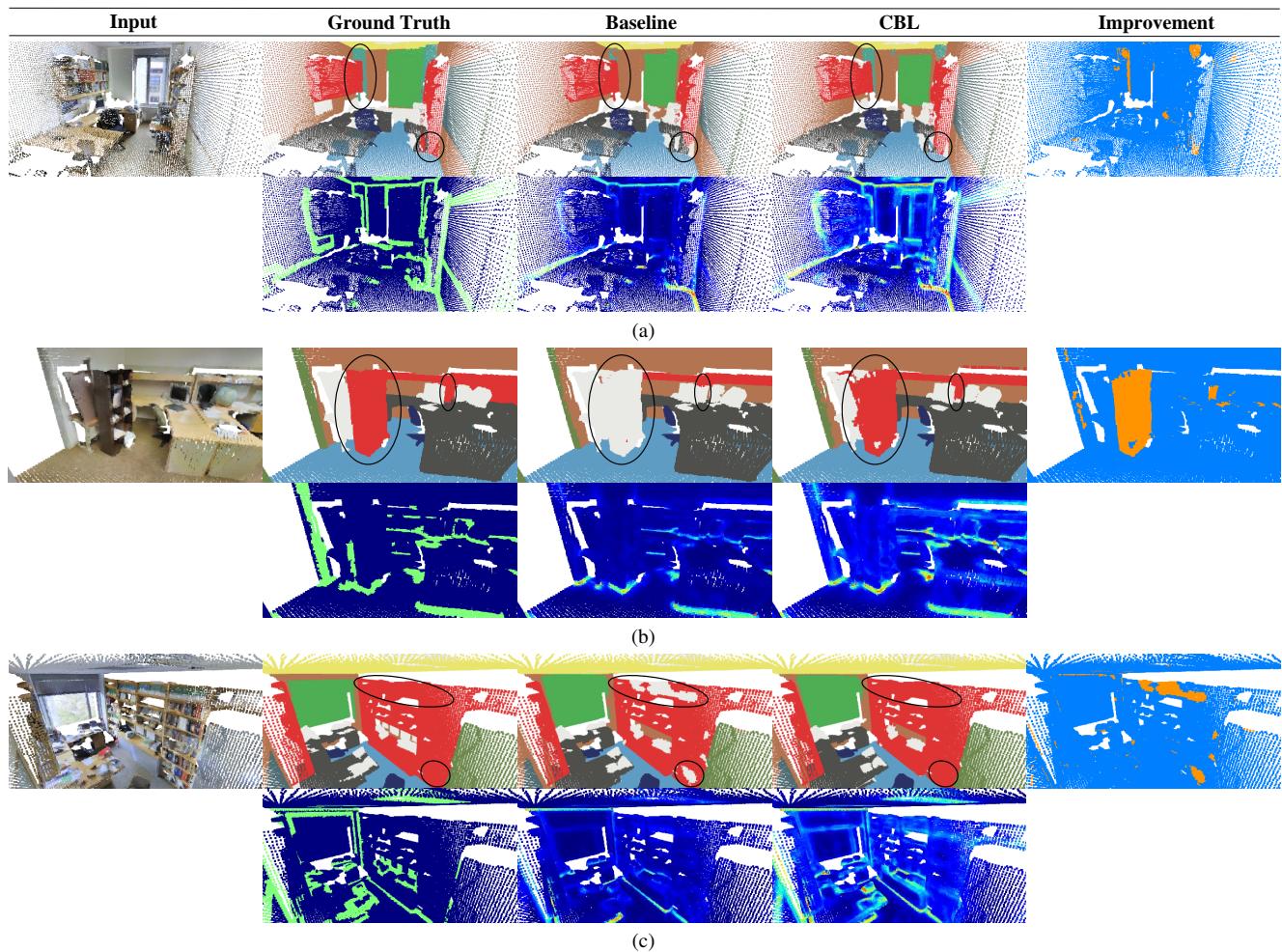


Figure 10. Offices. Same as above (Fig. 7).