Learning Regional Purity for Instance Segmentation on 3D Point Clouds

Shichao Dong[1,2], Guosheng Lin[*1,2], and Tzu-Yi Hung[3]

[1] S-lab, Nanyang Technological University, Singapore
[2] School of Computer Science and Engineering, Nanyang Technological University, Singapore
[3] Delta Research Center, Singapore
{scdong, gslin}@ntu.edu.sg, tzuyi.hung@deltaww.com

Abstract. 3D instance segmentation is a fundamental task for scene understanding, with a variety of applications in robotics and AR/VR. Many proposal-free methods have been proposed recently for this task, with remarkable results and high efficiency. However, these methods heavily rely on instance centroid regression and do not explicitly detect object boundaries, thus may mistakenly group nearby objects into the same clusters in some scenarios. In this paper, we define a novel concept of "regional purity" as the percentage of neighboring points belonging to the same instance within a fixed-radius 3D space. Intuitively, it indicates the likelihood of a point belonging to the boundary area. To evaluate the feasibility of predicting regional purity, we design a strategy to build a random scene toy dataset based on existing training data. Besides, using toy data is a "free" way of data augmentation on learning regional purity, which eliminates the burdens of additional real data. We propose Regional Purity Guided Network (RPGN), which has separate branches for predicting semantic class, regional purity, offset, and size. Predicted regional purity information is utilized to guide our clustering algorithm. Experimental results demonstrate that using regional purity can simultaneously prevent under-segmentation and over-segmentation problems during clustering.

Keywords: 3D Instance Segmentation, Point Cloud Representation Learning, Clustering Algorithm

1 Introduction

Semantic scene understanding is a crucial component for many real-world computer vision applications, such as indoor robots, autonomous driving, drones, AR/VR devices, etc. Although processing visual information for scene understanding is an instinctive ability for humans, it remains a fairly challenging task for robots. Many robotic applications cannot fully handle various situations due to the lack of semantic understanding of the target objects in the working environment. In recent times, with the rapid development of deep learning techniques, computer vision has achieved remarkable success in 2D image tasks. Different from 2D data captured from a conventional camera, 3D data are usually collected by Lidar sensor or RGB-D based 3D scanner. Point cloud

* Corresponding author: G.Lin (e-mail:gslin@ntu.edu.sg)

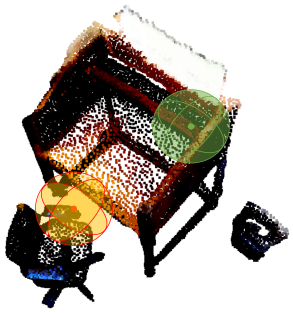

Fig. 1: Example of regional purity. Green point indicates high regional purity, since all its surrounding points belong to the same instance. Red point indicates low regional purity, since some of its neighboring points are from a different instance.

data involves a bunch of discrete points with their XYZ coordinates. Compared with images, 3D data retains the original geometric information and does not suffer from depth information loss during projection.

Existing approaches on 3D instance segmentation task can be classified into two categories, proposal-based methods [9, 17, 27, 39, 40] and proposal-free methods [8, 13, 18, 19, 21, 24, 26, 28, 36, 37]. Proposal-based methods perform object detection task first and predict a point-level mask for each proposed box. Whereas, proposal-free methods start solving the problem based on semantic segmentation result and discriminate points into clusters via post-processing steps [5]. Generally, proposal-free methods have relatively lower objectness since they do not perform computationally expensive object detection task.

Majority of the methods with good performance, including [13, 19, 21, 29, 30, 19] follows a same way of predicting instance centers and group nearby points into proposals or clusters. Although this has been proven to be very effective, there are some situations that are fairly challenging to predict accurate centroid from a single point. Specifically, points near object boundary or belonging to objects with distorted shapes are difficult to be predicted precisely. These points with inaccurate center prediction may potentially cause two nearby objects with the same semantic class to be wrongly grouped into the same cluster. Can we find an efficient way to tackle such problems without even performing object detection task? The answer is yes.

In this paper, we look at this problem from a different point of view and aim to find a more robust way to deal with these hard cases. Our work focus on approximating boundary area via predicting regional purity. On ground-truth, we explore the surrounding space for each point and calculate the percentage of the points with the same instance label. As shown in Figure 1, if most neighbor points belong to the same instance, this point is said to have high regional purity and vice versa. We build a random scene toy dataset and let our network learn to predict regional purity on it. Based on the predicted results, we can know which points are more likely to be in the bound-

ary area between objects and need to be cautious when grouping them. Comparing with bounding box detection, our approximated boundary areas own more adaptive shapes and are not constrained by rigid rectangular boxes. Meanwhile, our regional prediction is a direct output from one branch after the backbone network, thus does not bring much computational burden during processing.

To sum up, the key contributions of our work are following:

- We define a novel concept of regional purity, which encodes instance-aware contextual information of the surrounding region. Regional purity information can be employed to guide the clustering algorithm and provide good objectness for 3D instance segmentation.
- We propose a pretraining pipeline for learning regional purity and design rules to generate random toy scenes by extracting samples from existing training data.
- Our proposed method achieves state-of-the-art performance and the fastest processing speed among all the methods.

2 Related Work

To handle unstructured point cloud data [2], existing proposed feature learning methods can be classified into point-based methods [23, 31, 32, 35, 38, 43] and projection-based methods [10, 12, 16, 22] . Inspired by the success of convolution on images, projection-based methods transform original data into regular format and then implement with convolution operation. On the other hand, point-based methods directly work on irregular point cloud with different ways of feature extraction.

Similarity Group Proposal Network (SGPN) [36] is a pioneering work that directly tackles instance segmentation task with deep learning technique on 3D point cloud data. It learns point feature using PointNet++ [32] backbone and merges group proposals from similarity matrix for instance segmentation. Submanifold sparse convolution [12] has been proven to be a very effective backbone network for 3D semantic segmentation, which transforms sparse point cloud into voxels and performs convolution only on non-empty voxels. Liu et al. [26] proposed MASC, a U-net architecture with submanifold sparse convolution [12]. It predicts semantic scores for every voxel and the affinity between neighbouring voxels at different scales. Wang et al. [37] introduced Associatively Segmenting Instance and Semantics(ASIS), which has two separate branches for semantic segmentation and instance segmentation that can mutually support each other. JSIS3D [28] uses multi-value Conditional Random Field (CRF) for joint optimization. MTML [21] introduced directional loss and discriminative loss for feature embedding into 3D instance segmentation. Since then, center based methods dominate this area. PointGroup [19] finds the void space between objects and leverage dual set of proposals to boost performance. Occuseg [13] introduces occupancy signal to guide graph-based clustering algorithm. Overall, these proposal-free methods do not require region proposal network which makes them less computational expensive. Proposal-based methods including GSPN[36], 3D-SIS[17] and 3D-BoNet[39] explicitly detect object boundaries and perform binary mask prediction on top of the detection result. The recent 3D-MPA [9] makes dense center predictions on all points and aggregate the features between proposals via graph convolutional network for mask prediction.

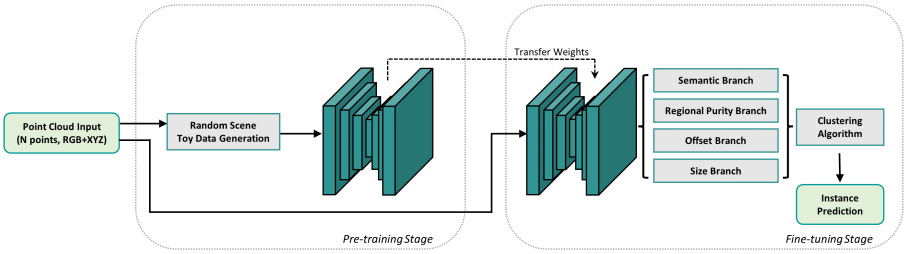

Fig. 2: Pipeline of the proposed pretraining scheme for learning regional purity. First, we extract samples from training set of point cloud data and use our montage assembly method to generate random scenes. Then, we pretrain our backbone network on toy dataset and fine tune on real dataset. Our network has four branches for point feature prediction and a clustering algorithm to output final result.

3 Method

In this work, the predominant objective is to evaluate if our defined regional purity signal can be well learned and predicted. Therefore we start by building a toy dataset with clean data. Following that, the next question is to show how those useful information can be utilized.

As shown in Fig. 2, we proposed a pipeline of the training scheme, consisting of pre-training stage and fine-tuning stage. Our random scene toy dataset serves two main purposes: 1) to evaluate the feasibility of predicting regional purity; and 2) to introduce "free" additional data for data augmentation.

3.1 Random Scene Toy Dataset

For the network to learn regional purity in a generalized way, we need a large amount of data that contains different combined cases of nearby objects. However, scenes in public dataset usually only have limited high-quality data for learning regional purity. Moreover, many objects are not close enough to each other, which makes the unbalance problem between high purity class and low purity class even worse. Excessive background points are not helpful for learning regional purity but inevitably lead to additional computational costs. To tackle these issues, we design a novel strategy of building a toy dataset by sample extraction and montage assembly. The created toy dataset contains foreground points only and keeps objects highly compacted.

Sample Extraction As the preparation step, we select and crop those points belonging to a particular instance from the training set. Since cropped point clouds can be at different positions in the original coordinate system, its coordinates need to be normalized by shifting the origin of the coordinate system to its mean center. To make the object on the floor, we then shift all points upwards until no negative Z values exist. The semantic labels are kept but instance labels will be reassigned in the next steps.

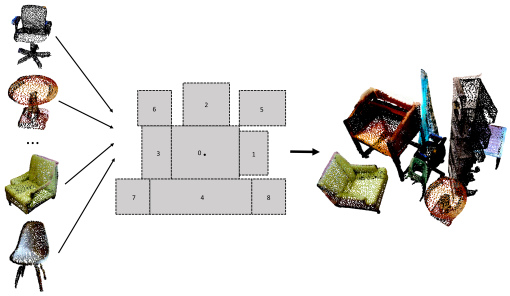

Fig. 3: Illustration of random toy scene generation. Extracted toy samples are flexibly joined into new scenes via a template-based montage assembly method.

Montage Assembly To create more cases of the boundary area, we use a template-based assembling method. Specifically, nine samples from extraction stages will be randomly selected and stitched into a toy scene. The template is designed on the bird-view of objects. Instance samples are added to the template sequentially. Oversized objects can cause the next object to be shifted aside. Afterward, we shift all surrounding objects towards the center object by a Gaussian random distance. These hard cases can benefit network training. All objects are augmented with random rotation. The whole scene is also randomly adjusted and one object is randomly dropped out, to prevent any potential possibility of over-fitting. In summary, our assembly method creates random scene toy dataset with high coverage and few overlaps, meanwhile keeping the data size consistent.

Regional Purity Label For automatically generating ground-truth regional purity labels, we use k-d tree which organizes points in a space partitioning data structure. This allows fast retrieval of neighboring points in 3D space. Given a seed point q, we search and find all points within a fixed radius r. Its receptive space can be expressed as:

$$\mathcal{N}(q, r) = \{p \in \mathcal{P} \mid \|p - q\| < r\}, \tag{1}$$

where r is the radius of the receptive space and p is taken from the set of points $\mathcal{P}$ of the entire point cloud scene.

Here, we define the following rules for regional purity label generation. We consider $id1$ as high purity, $id2$ as low purity and $id0$ as medium purity.

$$\text{regional purity label} = \begin{cases} 1 & \frac{\mathcal{M}}{\mathcal{N}} \geq \theta_1 \\ 2 & \frac{\mathcal{M}}{\mathcal{N}} \leq \theta_2 \\ 0 & otherwise \end{cases}, \tag{2}$$

where $\mathcal{M}$ is the number of points with the same instance label as the seed point, $\mathcal{N}$ is the number of total found points. We empirically set θ_1 to be 95% and θ_2 to be 80%.

3.2 Network Architecture

Our network uses a shared U-net backbone and several branches for joint task learning. Proposed clustering algorithm considers predicted information and directly outputs instance segmentation result.

3.3 Multi-task Learning

As a necessary preliminary step, the major role of backbone network is to extract the contextual and geometric information from input data. Subsequently, we apply linear transformation for different branches to predict semantic labels, regional purity labels, offsets and size labels. The training of our network is supervised by following the joint loss function,

$$L_{joint} = L_{sem} + L_{purity} + L_{offset} + L_{size}. \tag{3}$$

Semantic Segmentation Branch Based on point feature vectors, semantic score can be predicted for N classes. The training process is supervised by a conventional cross entropy loss [11] L_{sem}.

Regional Purity Branch As mentioned in the previous section, we assigned regional purity labels to be either 0, 1, or 2 on ground-truth. Normal data distribution may include much more label id 1 than id 2. To deal with the class imbalance issue, we propose a joint loss with three terms for regional purity,

$$L_{purity} = L_{CE} + L_{dice} + 0.1 * L_{dist}. \tag{4}$$

For learning regional purity, Cross-Entropy loss with softmax activation is applied for three equally weighted categories,

$$L_{CE} = -\frac{1}{N}\sum_{i=1}^{N} H_{CE}(y_i, c_i). \tag{5}$$

where N is the number of points, i is the index of point, c_i is the one-hot-encoding of the ground-truth regional purity label of point i and H_{CE} is the cross-entropy function.

Accuracy in class imbalance tasks can be misleading sometimes. $F1$ score, as known as Dice coefficient [34, 3], is a more reliable measure. It represents the harmonic mean of precision and recall.

$$Dice = \frac{2 \mid A \cap B \mid}{\mid A \mid + \mid B \mid} = \frac{2TP}{2TP + FP + FN}. \tag{6}$$

This metric is directly adapted in dice loss function [4, 20, 33]. Here, we add different weight coefficients for FP (false positives) and FN (false negatives). For predicting low purity label, a false positive prediction will surfer more punishment than a false negative prediction. In other words, we think precision is more important than recall

in this task. Comparing with predicting nothing, wrong prediction hurts performance more. This loss term is formulated as:

$$L_{dice} = 1 - \frac{1 + p\hat{p}}{1 + p\hat{p} + \alpha p(1-\hat{p}) + \beta(1-p)\hat{p}}. \tag{7}$$

where α is the coefficient for FP (false positives) and β is the coefficient for FN (false negatives). The sum of these two factors must be 1. If both are set to 0.5, it is just same as regular Dice loss. A value of 1 is added at both numerator and denominator of the fraction to smooth the loss.

A distance map [15] is derived on ground-truth by searching the distance to the nearest low purity point for each point. The purpose of using distance penalty term is to treat false positive point differently. For example, a point wrongly predicted as low purity point but just nearby other positive points is much tolerable, but predicting a low purity point at the center of an object is beyond reasonable limits. Using distance penalty term, it guides the network to focus towards the target area at boundary regions. This term is defined as:

$$L_{dist} = \frac{1}{M}\sum_{i=1}^{M}(1 + \Phi) \odot L_{CE}, \tag{8}$$

Here, M is the number of predicted low purity points, Φ is the distance map created, i is the index of Φ. This is an additional term to the Cross-Entropy loss.

Offset Branch Following previous work [19], we use L1 loss to regress object centroid for all points on instances. Offset labels are three dimensional vectors which generated on ground-truth.

$$L_{o\_reg} = \frac{1}{\sum_i m_i}\sum_i ||o_i - (\hat{c}_i - p_i)|| \cdot m_i, \tag{9}$$

where m is a binary mask to filter out background points.

Directly regressing instance center is a challenging task. Here, an additional direction loss term is introduced to guide the network based on cosine similarity.

$$L_{o\_dir} = 1 - \frac{1}{\sum_i m_i}\sum_i \frac{o_i}{||o_i||_2} \cdot \frac{\hat{c}_i - p_i}{||\hat{c}_i - p_i||_2} \cdot m_i, \tag{10}$$

Note that we add a constant of 1 to avoid negative loss value, since the range of cosine similarity is between -1 and 1. The combined offset loss can be written as

$$L_{offset} = L_{o\_reg} + L_{o\_dir}. \tag{11}$$

Size Branch For the network to learn instance-level contextual feature, we also introduce a size branch for auxiliary purposes. Based on the length of diagonal of bird view 2D bounding box, instance size is classified into six categories and the fixed interval between classes is 0.4m.

3.4 Regional Purity Guided Clustering Algorithm

In this section, we employ predicted regional purity to guide a standard breath-first search algorithm. Figure 4 shows a typical grouping process towards some ungrouped points (in grey). In (a), a new cluster starts from a random initial seed point (in blue). The seed point will search and find target points in space. If target points satisfy its criteria, they are grouped into the cluster and becoming the seed points in next grouping iteration.

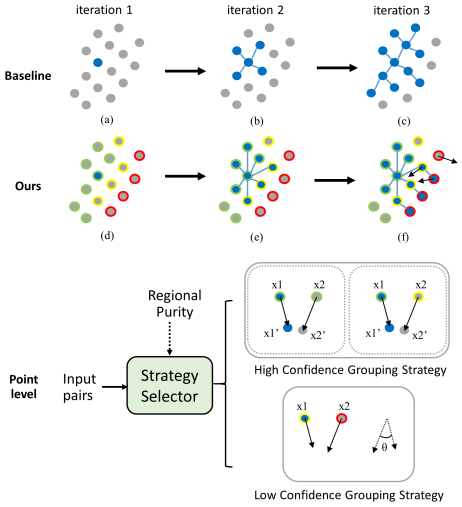

Fig. 4: Illustration of clustering algorithm grouping strategy

In our method, points are not treated equally. On point-level, before grouping of each pair, we check the regional purity prediction of the seed point and target point (represented by the outline color of points in $(d)(e)(f)$). Based on that, strategy selector makes decision to go for high confidence grouping strategy or low confidence grouping strategy, other cases will block and skip. The core idea is to better group the inner part of objects with high purity while isolating instances by utilizing low purity points.

High purity pairs are more likely at inner part of objects and their offset prediction is relatively reliable. Thus we use high confidence strategy by shifting their coordinates to their predicted object centers. Real point cloud data are often holey and inconsistent, which can potentially cause an over-segmentation problem. To reduce the impact, we also make the grouping criteria more tolerable by introducing additional radius Δr_1.

Low confidence grouping strategy is defined towards low purity points. At boundary area, offset prediction is often not reliable, if we use shift coordinates may cause two clusters to be mistakenly merged. To make the grouping more robust, we use the cosine similarity of the direction between their offset vectors as an additional criteria. Here only medium purity points can be used as seed points, because they are geometrically closer to low purity points and comparing the offset vector directions between nearby points has more reference value.

Algorithm 1 Clustering algorithm. N is the number of points. M is the number of clusters found by the algorithm.

Input: clustering radius r;
clustering additional radius Δr_1 and Δr_2;
cluster point number threshold N_θ;
cosine similarity threshold ϕ;
coordinates $\mathbf{X} = \{x_1, x_2, ..., x_N\} \in \mathbb{R}^{N\times 3}$
offset $\mathbf{D} = \{d_1, d_2, ..., d_N\} \in \mathbb{R}^{N\times 3}$; and
semantic labels $\mathbf{S} = \{s_1, ..., s_N\} \in \mathbb{R}^N$.
regional purity labels $\mathbf{P} = \{p_1, ..., p_N\} \in \mathbb{R}^N$.
Output: clusters $\mathbf{C} = \{C_1, ..., C_M\}$.

initialize an array v (visited) of length N with all zeros
initialize an empty cluster set $\mathbf{C}$
for $i = 1$ to N **do**
  if s_i is a background class **then**
    $v_i = 1$
for $i = 1$ to N **do**
  if $v_i == 0$ **then**
    if $p_i == 1$ **then**
      initialize an empty queue Q
      initialize an empty cluster C
      $v_i = 1$; Q.enqueue(i); add i to C
    while Q is not empty **do**
      $k = Q$.dequeue()
      for $j \in [1, N]$ **do**
        if $s_j == s_k$ and $v_j == 0$ **then**
          ▶high confidence grouping strategy
          if $p_i == 1$ and $p_j != 2$ **then**
            $r\prime \leftarrow r + \Delta r_1$
            $x_j\prime \leftarrow x_j + d_j$
            $x_k\prime \leftarrow x_k + d_k$
            if $||x_j\prime - x_k\prime||_2 < r\prime$ **then**
              $v_j = 1$
              Q.enqueue(j);add j to C
          ▶low confidence grouping strategy
          if $p_i == 0$ and $p_j == 2$ **then**
            $r\prime = r + \Delta r_2$
            if $||x_j\prime - x_k\prime||_2 < r\prime$ **then**
              $cos\theta(d_j, d_k) = \frac{d_j}{\|d_j\|} * \frac{d_k}{\|d_k\|}$
              if $cos\theta < \phi$ **then**
                $v_j = 1$
                Q.enqueue(j);add j to C
      if number of points in $C > N_\theta$ **then**
        add C to $\mathbf{C}$
return $\mathbf{C}$

Our algorithm only takes one set of points as input. Since each point can only be visited once, there will be no overlapping clusters. Thus non-maximum-suppression (NMS) [14] is not needed as a post-processing step.

4 Experiment

In this section, we evaluate our method on created toy dataset and public dataset of ScanNet v2 [7] and S3DIS [1] to show the effectiveness of our approach.

4.1 Evaluation on Random Scene Toy Dataset

Toy scenes are randomly generated by our introduced assembly method and split into training set and validation set at a ratio of 4:1. The network takes coordinates with RGB color information of points as input and predicts regional purity at point-level.

Radius	0.2m		0.3m	
Purity label	high	low	high	low
Precision	97.4	88.5	95.1	89.2
Recall	99.0	86.4	98.7	88.2
$F1$ score	98.2	87.4	96.9	88.7
IoU	96.5	77.7	93.9	79.6

Table 1: Evaluation of regional purity prediction at different scales on toy dataset validation set.

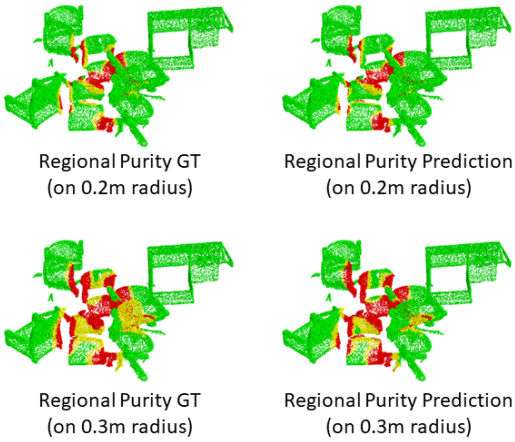

Fig. 5: Visualization of regional purity prediction at different scales on toy dataset validation set.

We evaluate on two different scales of regional purity on 0.2m and 0.3m searching radius, with same 95% and 80% criteria for label generation. The results in Table 1 and Figure 5 show that our network is able to learn the contextual information of defined regional purity. For regional purity, green color represents high purity with $id1$, red color represents low purity with $id2$, yellow color represents medium purity with $id0$.

4.2 Evaluation on ScanNet Dataset

To demonstrate the effectiveness of our approach, we conduct experiments on ScanNet dataset [7]. It is a popular point cloud dataset containing 1513 real-world indoor scenes. The 3D meshed data are annotated with point-level semantic label and instance label.

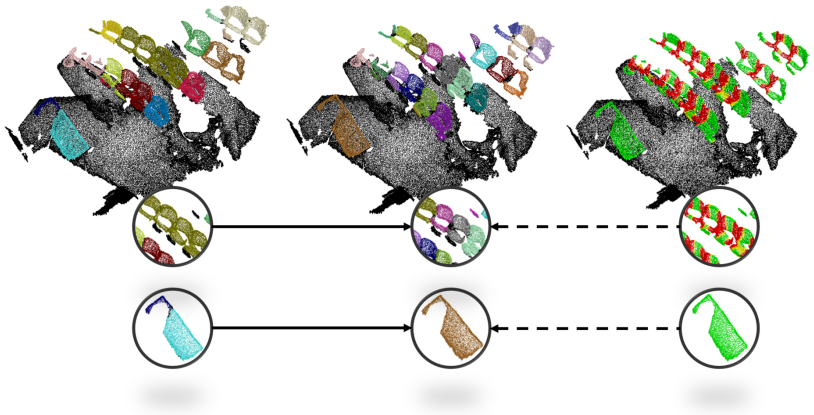

Fig. 6: Result Analysis on ScanNet validation set. From left to right: (1) instance prediction by baseline algorithm without regional purity (2) instance prediction by regional purity guided clustering algorithm (3) regional purity prediction (red color area represents low regional purity, green color area represents high regional purity). This shows that using regional purity information can simultaneously resolve under-segmentation and over-segmentation problem.

Implementation Details We use Adam solver for optimization with an initial learning rate of 0.001. At pretraining stage, the network takes 10k randomly generated toy scenes as input. The backbone network are initially frozen when transferring to real dataset. After the last two linear layers are well trained, we unfreeze whole network and continue to train it until convergence. The training takes 4-5 days on a single GPU.

mAP@0.25	bath	bed	bkshf	cab	chair	cntr	curt	desk	door	ofurn	pic	fridg	showr	sink	sofa	tabl	toil	wind	**avg**
SGPN [36]	90.3	8.1	0.8	23.3	17.5	28.0	10.6	15.0	20.3	17.5	48.0	21.8	14.3	54.2	40.4	15.3	39.3	4.9	26.1
3D-BEVIS [8]	66.7	68.7	41.9	13.7	58.7	18.8	23.5	35.9	21.1	9.3	8.0	31.1	57.1	38.2	75.4	30.0	87.4	35.7	40.1
R-PointNet	50.0	65.5	66.1	66.3	76.5	43.2	21.4	61.2	58.4	49.9	20.4	28.6	42.9	65.5	65.0	53.9	95	49.9	54.4
3D-SIS [17]	100	77.3	61.4	50.3	69.1	20.0	41.2	49.8	54.6	31.1	10.3	60.0	85.7	38.2	79.9	44.5	93.8	37.1	55.8
MASC [26]	71.1	80.2	54.0	75.7	77.7	2.9	57.7	58.8	52.1	60.0	43.6	53.4	69.7	61.6	83.8	52.6	98	53.4	61.5
3D-BoNet [39]	100	88.7	83.6	58.7	64.3	55.0	62.0	72.4	52.2	50.1	24.3	51.2	100	75.1	80.7	66.1	90.9	61.2	68.7
PanopticFusion [27]	100	85.2	65.5	61.6	78.8	33.4	76.3	77.1	45.7	55.5	65.2	51.8	85.7	76.5	73.2	63.1	94.4	57.7	69.3
SSEN [42]	100	92.6	78.1	66.1	84.5	59.6	52.9	76.4	65.3	48.9	46.1	50.0	85.9	76.5	87.2	76.1	100	57.7	72.4
MTML [21]	100	99.2	77.9	60.9	74.6	30.8	**86.7**	60.1	60.7	53.9	51.9	55.0	100	82.4	86.9	72.9	100	61.6	73.1
3D-MPA [9]	100	93.3	78.5	**79.4**	83.1	27.9	58.8	69.5	61.6	55.9	55.6	65.0	100	80.9	87.5	69.6	100	60.8	73.7
OccuSeg [13]	100	92.3	78.5	74.5	86.7	55.7	57.8	72.9	67.0	64.4	48.8	57.7	100	79.4	83.0	62.0	**100**	55.0	74.2
PE [41]	100	90.0	86.0	72.8	86.9	40.0	85.7	77.4	56.8	70.1	60.2	64.6	93.3	84.3	**89.0**	69.1	99.7	70.9	77.6
PointGroup [19]	100	90.0	79.8	71.5	86.3	49.3	70.6	**89.5**	56.9	**70.1**	57.6	63.9	100	88	85.1	71.9	99.7	70.9	77.8
SSTNet [25]	100	84	**88.8**	71.7	83.5	**71.7**	68.4	62.7	**72.4**	65.2	**72.7**	60	100	**91.2**	82.2	75.7	100	69.1	78.9
HAIS [6]	100	**99.4**	82	75.9	85.5	55.4	88.2	82.7	61.5	67.6	63.8	64.6	100	**91.2**	79.7	**76.7**	99.4	**72.6**	80.3
RPGN (Ours)	**100**	99.2	78.9	72.3	**89.1**	65.0	81	83.2	66.5	69.9	65.8	**70.0**	**100**	88.1	83.2	77.4	99.7	61.3	**80.6**

Table 2: 3D instance segmentation results on ScanNet v2 [7] on 18 classes.

The results of 3D instance segmentation on ScanNet [7] are presented in Table 2, Table 3 and Figure 8, which show our predicted regional purity information can be leveraged to improve the performance of instance segmentation.

mAP@0.5	cab	bed	chair	sofa	tabl	door	wind	bkshf	pic	cntr	desk	curt	fridg	showr	toil	sink	bath	ofurn	**avg**
SegCluster	10.4	11.9	15.5	12.8	12.4	10.1	10.1	10.3	0.0	11.7	10.4	11.4	0.0	13.9	17.2	11.5	14.2	10.5	10.8
MRCNN	11.2	10.6	10.6	11.4	10.8	10.3	0.0	0.0	11.1	10.1	0.0	10.0	12.8	0.0	18.9	13.1	11.8	11.6	9.1
SGPN [36]	10.1	16.4	20.2	20.7	14.7	11.1	11.1	0.0	0.0	10.0	10.3	12.8	0.0	0.0	48.7	16.5	0.0	0.0	11.3
3D-SIS [17]	19.7	37.7	40.5	31.9	15.9	18.1	0.0	11.0	0.0	0.0	10.5	11.1	18.5	24.0	45.8	15.8	23.5	12.9	18.7
MTML [21]	14.5	54.0	79.2	48.8	42.7	32.4	32.7	21.9	10.9	0.8	14.2	39.9	42.1	64.3	96.5	36.4	70.8	21.5	40.2
PointGroup [19]	48.1	69.6	87.7	**71.5**	62.9	42.0	46.2	54.9	37.7	22.4	41.6	44.9	37.2	64.4	98.3	**61.1**	80.5	53.0	56.9
3D-MPA [9]	51.9	72.2	83.8	66.8	63.0	43.0	44.5	58.4	38.8	**31.1**	43.2	47.7	61.4	80.6	99.2	50.6	**87.1**	40.3	59.1
RPGN (Ours)	50.9	76.6	92.1	62.6	70.6	47.2	52.1	59.8	41.7	17.6	45.7	**51.9**	63.3	**91.5**	**100**	42.7	**87.1**	61.4	61.9
RPGN† (Ours)	**57.3**	**75**	**92.6**	63.6	**71.9**	**49.8**	**56.4**	**62.6**	**46.6**	22.1	**54.8**	51.2	**65.3**	90.0	**100**	48.0	83.9	**64.3**	**64.2**

Table 3: 3D instance segmentation results on ScanNet v2 [7] validation set with on 18 classes.† represents using refined semantic prediction via label smoothing.

Discussion In Figure 6, we compare the instance segmentation results before and after adding the regional purity prediction into the algorithm. In case 1, four chairs are wrongly grouped into one cluster. By using our well predicted regional purity information, the clustering algorithm can successfully separate them into different clusters. In case 2, the table is predicted as two instances due to the inaccuracy in offset prediction. Since all points on the table have high purity label, we give more tolerance for grouping them into one piece.

4.3 Evaluation on S3DIS Dataset

To study the generalizability of our pretrained model, we also evaluate our proposed RPGN model on S3DIS dataset [1]. The dataset has 272 scenes under six large-scale indoor areas. Different from ScanNet [7], all 13 classes including background are annotated as instances and require prediction. Following previous methods, we use the mean precision (mPre) and mean recall (mRec) with an IoU threshold of 0.5 as evaluation metric. We report results on both Area 5 and 6-fold cross validation over six areas in Table 4. Using pretrained model reduces overfitting on small dataset and dramatically boost the performance.

	Area 5		**6-fold**	
	mPrec	mRec	mPrec	mRec
ASIS[37]	55.3	42.4	63.6	47.5
PointGroup[19]	61.9	62.1	69.6	69.2
OccuSeg[13]	-	-	72.8	60.3
SSTNet[25]	65	64.2	73.5	**73.4**
HAIS[6]	**71.1**	**65.0**	73.2	69.4
RPGN (Ours)	64.0	63.0	**84.5**	70.5

Table 4: 3D instance segmentation results on S3DIS dataset [1]

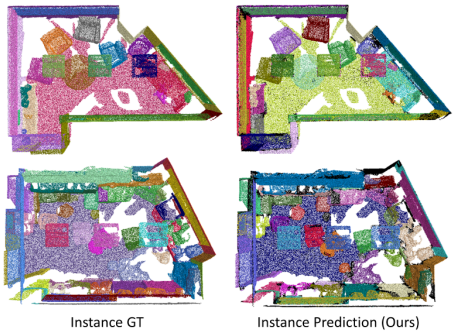

Fig. 7: Instance Segmentation Result on S3DIS dataset [1].

5 Ablation Study

To evaluate the effectiveness of each component in proposed method, we conduct ablation studies on the validation set of ScanNet dataset in Table 5.

	High Purity	Low Purity	Offset	Direction	mAP	mAP@0.5
baseline	×	×	×	×	0.284	0.508
(a)	✓	×	×	×	0.294	0.518
(b)	✓	✓	×	×	0.322	0.545
(c)	✓	✓	✓	×	0.352	0.572
(d)	✓	✓	✓	✓	0.359	0.582

Table 5: Ablation results of instance segmentation task for clustering algorithm on ScanNet v2 [7] validation set

Ablation on Clustering Algorithm To analysis our proposed clustering algorithm, we use mentioned baseline algorithm and step-by-step add our components onto it.

In step (a), we utilize high purity points by allowing an additional Δr_1 radius when grouping other high purity points. We argue that regions with high purity prediction should be safer to group other nearby points. By bringing additional tolerance, high purity points can help to step over the gaps inside objects.

In step (b), we define low purity points can only be grouped by medium purity points within $(r + \Delta r_2)$ radius on original coordinates and cannot group any other points. This constrains the grouping direction to be regional purity guided, only from high to low. Allowing inverse direction grouping can potentially cause different instances to be connected. Our soft barrier formed by low purity points can help to prevent such cases.

In step (c), predicted offset feature is used for high purity points to be better grouped. Note that we only shift high purity points to their predicted instance center. We argue that points with low purity labels can hardly predict accurate offset to their center, since they are more likely on the boundary. Therefore, grouping low purity points only considers the original coordinates.

In step (d), we add an additional condition to compare the angle of offset vector between seed point and target point. The grouping is only proceeded if their cosine similarity is above 0.8. Even though we have low confidence in their predicted instance centers to be precise, rough directions still have value for assigning low purity points into the right clusters.

Pretraining vs Training from Scratch We compare two training strategies for the network to predict regional purity. The proposed pre-training strategy brings an improvement of 3.8% mAP on ScanNet v2 [7] validation set.

Dual Set vs Single Set The previous work [19] uses two sets of points for proposal generation and filter out duplicated cases by a scoring network and Non-Maximum Suppression (NMS). In this work, we make rational use of predicted information. Our clustering algorithm can directly generate high-quality proposals and get rid of NMS.

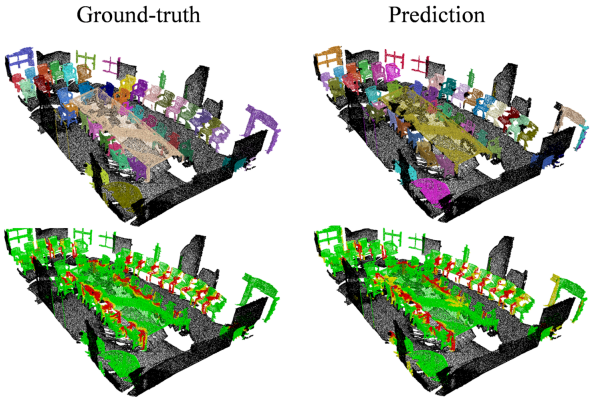

Fig. 8: Qualitative Results of Instance Segmentation and Regional Purity Prediction on ScanNet v2 [7] validation set.

Runtime Analysis In Table 6, we compare the processing time on full validation set of ScanNet (312 scenes) with other methods according to [13, 39]. In general, the inference time of our network for a single scene with 20k points is around 0.3 seconds.

	Total Processing Time
SGPN [36]	49433
ASIS [37]	56757
3D-SIS [17]	38841
GSPN [40]	3963
3D-BoNet [39]	2871
OccuSeg [13]	594
PointGroup [19]	141
HAIS [6]	128
RPGN(Ours)	**89**

Table 6: Total processing time (in seconds) on the validation set of ScanNet v2 [7]

6 Conclusion

In this paper, we have presented our defined regional purity concept and its learning strategy with the random scene toy dataset generation and pretraining scheme. Predicted regional purity can be used to guide the clustering process for 3D instance segmentation. Without performing object detection tasks, we use regional purity area to approximate object boundaries in a more flexible and robust form.

Acknowledgements This research is supported under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). This research is also supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG-RP-2018-003), the Ministry of Education, Singapore, under its Academic Research Fund Tier 2 (MOE-T2EP20220-0007) and Tier 1 (RG95/20).

References

1. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (2016)
2. Bello, S.A., Yu, S., Wang, C.: Review: deep learning on 3d point clouds (2020)
3. Bertels, J., Eelbode, T., Berman, M., Vandermeulen, D., Maes, F., Bisschops, R., Blaschko, M.B.: Optimizing the dice score and jaccard index for medical image segmentation: Theory and practice. Medical Image Computing and Computer Assisted Intervention – MICCAI 2019 p. 92–100 (2019). https://doi.org/10.1007/978-3-030-32245-8˙11, http://dx.doi.org/10.1007/978-3-030-32245-8_11
4. Bokhovkin, A., Burnaev, E.: Boundary loss for remote sensing imagery semantic segmentation (2019)
5. Brabandere, B.D., Neven, D., Gool, L.V.: Semantic instance segmentation with a discriminative loss function (2017)
6. Chen, S., Fang, J., Zhang, Q., Liu, W., Wang, X.: Hierarchical aggregation for 3d instance segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 15467–15476 (October 2021)
7. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE (2017)
8. Elich, C., Engelmann, F., Kontogianni, T., Leibe, B.: 3d bird's-eye-view instance segmentation. Pattern Recognition p. 48–61 (2019). https://doi.org/10.1007/978-3-030-33676-9˙4, http://dx.doi.org/10.1007/978-3-030-33676-9_4
9. Engelmann, F., Bokeloh, M., Fathi, A., Leibe, B., Nießner, M.: 3d-mpa: Multi proposal aggregation for 3d semantic instance segmentation (2020)
10. Gojcic, Z., Zhou, C., Wegner, J.D., Guibas, L.J., Birdal, T.: Learning multiview 3d point cloud registration. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
11. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org
12. Graham, B., van der Maaten, L.: Submanifold sparse convolutional networks. CoRR **abs/1706.01307** (2017)
13. Han, L., Zheng, T., Xu, L., Fang, L.: Occuseg: Occupancy-aware 3d instance segmentation (2020)
14. Hosang, J., Benenson, R., Schiele, B.: Learning non-maximum suppression (2017)
15. Jadon, S.: A survey of loss functions for semantic segmentation (2020)
16. Jaritz, M., Gu, J.Y., Su, H.: Multi-view pointnet for 3d scene understanding. ArXiv **abs/1909.13603** (2019)
17. Ji, H., Dai, A., Nießner, M.: 3d-sis: 3d semantic instance segmentation of rgb-d scans. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE (2019)
18. Jiang, H., Yan, F., Cai, J., Zheng, J., Xiao, J.: End-to-end 3d point cloud instance segmentation without detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
19. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: Pointgroup: Dual-set point grouping for 3d instance segmentation (2020)
20. Kervadec, H., Bouchtiba, J., Desrosiers, C., Granger, E., Dolz, J., Ben Ayed, I.: Boundary loss for highly unbalanced segmentation. Proceedings of Machine Learning Research, vol. 102, pp. 285–296. PMLR, London, United Kingdom (08–10 Jul 2019), http://proceedings.mlr.press/v102/kervadec19a.html

21. Lahoud, J., Ghanem, B., Pollefeys, M., Oswald, M.R.: 3d instance segmentation via multi-task metric learning (2019)
22. Li, L., Zhu, S., Fu, H., Tan, P., Tai, C.L.: End-to-end learning local multi-view descriptors for 3d point clouds. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
23. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B.: Pointcnn: Convolution on x-transformed points. In: NeurIPS, pp. 820–830. Curran Associates, Inc. (2018)
24. Liang, Z., Yang, M., Wang, C.: 3d graph embedding learning with a structure-aware loss function for point cloud semantic instance segmentation (2019)
25. Liang, Z., Li, Z., Xu, S., Tan, M., Jia, K.: Instance segmentation in 3d scenes using semantic superpoint tree networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2783–2792 (October 2021)
26. Liu, C., Furukawa, Y.: MASC: multi-scale affinity with sparse convolution for 3d instance segmentation. CoRR (2019)
27. Narita, G., Seno, T., Ishikawa, T., Kaji, Y.: Panopticfusion: Online volumetric semantic mapping at the level of stuff and things (2019)
28. Pham, Q.H., Nguyen, T., Hua, B.S., Roig, G., Yeung, S.K.: Jsis3d: Joint semantic-instance segmentation of 3d point clouds with multi-task pointwise networks and multi-value conditional random fields. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
29. Qi, C.R., Chen, X., Litany, O., Guibas, L.J.: Imvotenet: Boosting 3d object detection in point clouds with image votes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
30. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep hough voting for 3d object detection in point clouds. In: Proceedings of the IEEE International Conference on Computer Vision (2019)
31. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 77–85 (2016)
32. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NIPS (2017)
33. Salehi, S.S.M., Erdogmus, D., Gholipour, A.: Tversky loss function for image segmentation using 3d fully convolutional deep networks (2017)
34. Shamir, R.R., Duchin, Y., Kim, J., Sapiro, G., Harel, N.: Continuous dice coefficient: a method for evaluating probabilistic segmentations (2019)
35. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. ArXiv **abs/1904.08889** (2019)
36. Wang, W., Yu, R., Huang, Q., Neumann, U.: Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In: CVPR (2018)
37. Wang, X., Liu, S., Shen, X., Shen, C., Jia, J.: Associatively segmenting instances and semantics in point clouds. In: CVPR (2019)
38. Wu, W., Qi, Z., Fuxin, L.: Pointconv: Deep convolutional networks on 3d point clouds. arXiv preprint arXiv:1811.07246 (2018)
39. Yang, B., Wang, J., Clark, R., Hu, Q., Wang, S., Markham, A., Trigoni, N.: Learning object bounding boxes for 3d instance segmentation on point clouds (2019)
40. Yi, L., Zhao, W., Wang, H., Sung, M., Guibas, L.: Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. arXiv preprint arXiv:1812.03320 (2018)
41. Zhang, B., Wonka, P.: Point cloud instance segmentation using probabilistic embeddings (2019)
42. Zhang, D., Chun, J., Cha, S.K., Kim, Y.M.: Spatial semantic embedding network: Fast 3d instance segmentation with deep metric learning (2020)

43. Zhao, H., Jiang, L., Fu, C.W., Jia, J.: PointWeb: Enhancing local neighborhood features for point cloud processing. In: CVPR (2019)