A. Overview

In this supplementary material, we first provide more additional experiments to further verify the superiority of our model in Section B. Besides, we show the our network architecture details in Section C.

B. Additional Experiment

B.1. Part Segmentation

Due to space limitation, we illustrate the part segmentation experiments using man-made synthetic dataset ShapeNet [15], which contains 16,881 shapes from 16 classes and 50 parts. We use the data provided by [11] and adopt the same training and test strategy, i.e., randomly pick 2048 points as the input and concatenate the one-hot encoding of the object label to the last layer.

The quantitative comparisons with the state-of-the-art point-based methods are summarized in Tab. 5. Note that we only compare with methods use 2048 points. When compared with the state-of-the-arts, PointASNL achieves comparable result, which is only slightly lower than RS-CNN [9] using different sampling and voting strategy (as the same reason for classification task).

B.2. Selection of Adaptive Sampling

Two variable conditions, i.e., the sampling strategy for initial sampled points and deformation method, are investigated for this issue. Tab. 1 summarizes the results. For the initial sampling points, we chose two strategies, i.e., FPS and random sampling (RS). Also for local coordinate points and feature updates, we compare the effects of using the weight learning by group feature (GF) and simple average of all neighbors’ coordinates and features. Note that the number of neighbors is set to be equal for a fair comparison.

As Tab. 1 shows, if we just use RS sample the initial points and then average their coordinates and features (model A), we will get very low accuracy of 87.9%. However, if we use FPS instead of RS (model B), it can increase to 91.5%. Furthermore, model C and D illustrate the weight learning using group features can largely increase the inference ability of our model. However, if we use RS as sampling strategy, it will cause some accuracy loss while we add the group features learning. This shows that AS module can only finely adjust the distribution of the sampled point cloud instead of creating the missing information.

B.3. Visualization of L-NL Module

We further demonstrate the local-global learning of PointASNL in Fig. 2. In the first layer of the network, PNL can find global points that have similar characteristics with sampled points (e.g., edge and normal vectors). In
the second layer, these global highly responsive points have the same semantics information with sampled points, even when sampled points are at the junction of the two different semantics. This is why global features can help sampled points to better aggregate local features.

B.4. Visualization of Adaptive Sampling

When the input point cloud has a lot of noise, adaptive sampling has the ability to ensure the distribution of the sample point manifold. We give some examples of comparative visualization in Fig. 6 to prove the robustness of the AS module. As can be seen from Fig. 6, AS module can effectively reduce noise in the sample points and maintain the shape of the sampled manifold.

B.5. Further Improvement of PointASNL

The result in manuscript only conducts a fair comparison (same model structure and training strategy) against appealing recent methods under the same setting of PointNet++ [11]. However, our PointASNL can still achieve further improvement if we use other data pre-processing or deeper structure.

As shown in Tab. 2, our PointASNL can still improve its performance if we use grid sampling pre-processing, more input points and deeper structure. As for the structure of deeper PointASNL, we add an additional point local cell at the end of each layer. Furthermore, by conducting ensemble learning with model from different training epochs, we can finally achieve 66.6% on ScanNet benchmark.

B.6. Concrete Results

In this section, we give our detailed results on the S3DIS (Tab.6 and Tab.7) and SemanticKITTI (Tab.8) dataset as a benchmark for future work. ScanNet [1] is an online benchmark, the class scores can be found on its website. Furthermore, we provide more visualization results to illustrate the performance of our model in complicated scenes.

C. Network Architectures

C.1. Layer Setting

For each encoder layer, it can be written as the following form: Abstraction(npoint, nsample, as_neighbor, mlp), where npoint is the number of sampled points of layer. nsample and as_neighbor are number of group neighbors in point local cell and AS module, and they share the same k-NN query. mlp is a list for MLP construction in our layers and used in both PL and PNL. Tab. 3 shows the configuration of PointASNL on both classification and segmenttaion tasks.

C.2. Loss Function

Like other previous works, we use cross entropy (CE) loss in classification and part segmentation, and consider the number of each category as weights in semantic segmentation. Furthermore, in order to avoid the sampled points being too close to each other in some local areas after the AS module transformation, we also use Repulsion Loss [16] to restrict the deformation of sampled point clouds. In particular, we only use this loss in the first layer since it has the highest point density. The Repulsion loss does not bring any performance improvement, but the training procedure is significantly accelerated.

Altogether, we train the PointASNL in an end-to-end
manner by minimizing the following joint loss function:

\[
L(\theta) = L_{CE} + \alpha L_{Rep} + \beta ||\theta||^2,
\]

\[
L_{Rep} = \sum_{i=0}^N \sum_{i' \in N(x_i)} w(||x_{i'} - x_i||),
\]

(1)

where \(\theta\) indicates the parameters in our network, \(\alpha = 0.01\) balances the CE loss and Repulsion loss, and \(\beta\) denotes the multiplier of the weight decay. For Repulsion loss, it penalizes the sampled point \(x_i\) only when it is too close to its neighboring points \(x_{i'} \in N(x_i)\). \(w(r) = e^{-r/h^2}\) is a fast-decaying weight function and \(N\) is the number of sampled points.

The Repulsion loss also ensures that each sample point itself has a larger weight in the AS module in a relatively constant density, which makes them cannot move too far.

C.3. Model Speeds

Tab. 4 shows the statistics of our models on different datasets. Since our L-NL module only uses sampled points as query points instead of the whole point cloud, the AS module and NL cell can be both efficient and effective with the bottleneck structures (only around 30% extra time).

### References


[3] Li Jiang, Hengshuang Zhao, Shi Liu, Xiaoyong Shen, Chi-Wing Fu, and Jiaya Jia. Hierarchical point-edge interaction network for point cloud semantic segmentation. 2019. 4


---

**Table 4. The running time on ModelNet40 and ScanNet datasets.**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>process</th>
<th>input</th>
<th>time (s/sample)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ModelNet40</td>
<td>Training</td>
<td>1024 pnt</td>
<td>0.00046</td>
</tr>
<tr>
<td>ScanNet</td>
<td>Training</td>
<td>8192 pnt</td>
<td>0.17611</td>
</tr>
<tr>
<td>ModelNet40</td>
<td>Inference</td>
<td>1024 pnt</td>
<td>0.00024</td>
</tr>
<tr>
<td>ScanNet</td>
<td>Inference</td>
<td>8192 pnt</td>
<td>0.11363</td>
</tr>
</tbody>
</table>
Table 5. Part segmentation performance with part-averaged IoU on ShapeNetPart.

<table>
<thead>
<tr>
<th>Method</th>
<th>pIoU</th>
<th>areo</th>
<th>bag</th>
<th>cap</th>
<th>car</th>
<th>chair</th>
<th>ear</th>
<th>guitar</th>
<th>knife</th>
<th>lamp</th>
<th>laptop</th>
<th>motor</th>
<th>mug</th>
<th>pistol</th>
<th>rocket</th>
<th>skate</th>
<th>phone</th>
<th>board</th>
</tr>
</thead>
<tbody>
<tr>
<td>#shapes</td>
<td>2690</td>
<td>76</td>
<td>55</td>
<td>898</td>
<td>3758</td>
<td>69</td>
<td>787</td>
<td>392</td>
<td>1547</td>
<td>451</td>
<td>202</td>
<td>184</td>
<td>286</td>
<td>66</td>
<td>152</td>
<td>5271</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PointNet [10]</td>
<td>83.7</td>
<td>83.4</td>
<td>78.7</td>
<td>82.5</td>
<td>74.9</td>
<td>89.6</td>
<td>73.0</td>
<td>91.5</td>
<td>85.9</td>
<td>80.8</td>
<td>95.3</td>
<td>65.2</td>
<td>93.0</td>
<td>81.2</td>
<td>57.9</td>
<td>72.8</td>
<td>80.6</td>
<td></td>
</tr>
<tr>
<td>SO-Net [6]</td>
<td>84.9</td>
<td>82.8</td>
<td>77.8</td>
<td>88.0</td>
<td>77.3</td>
<td>90.6</td>
<td>73.5</td>
<td>90.7</td>
<td>83.9</td>
<td>82.8</td>
<td>94.8</td>
<td>69.1</td>
<td>94.2</td>
<td>80.9</td>
<td>53.1</td>
<td>72.9</td>
<td>83.0</td>
<td></td>
</tr>
<tr>
<td>PointNet++ [11]</td>
<td>85.1</td>
<td>82.4</td>
<td>79.0</td>
<td>87.7</td>
<td>77.3</td>
<td>90.8</td>
<td>71.8</td>
<td>91.0</td>
<td>85.9</td>
<td>83.7</td>
<td>95.3</td>
<td>71.6</td>
<td>94.1</td>
<td>81.3</td>
<td>58.7</td>
<td>76.4</td>
<td>82.6</td>
<td></td>
</tr>
<tr>
<td>DGCNN [14]</td>
<td>85.1</td>
<td>84.2</td>
<td>83.7</td>
<td>84.4</td>
<td>77.1</td>
<td>90.9</td>
<td>78.5</td>
<td>91.5</td>
<td>87.3</td>
<td>82.9</td>
<td>96.0</td>
<td>67.8</td>
<td>93.3</td>
<td>82.6</td>
<td>59.7</td>
<td>75.5</td>
<td>82.0</td>
<td></td>
</tr>
<tr>
<td>P2Sequence [8]</td>
<td>85.2</td>
<td>82.6</td>
<td>81.8</td>
<td>87.5</td>
<td>77.3</td>
<td>90.8</td>
<td>77.1</td>
<td>91.1</td>
<td>86.9</td>
<td>83.9</td>
<td>95.7</td>
<td>70.8</td>
<td>94.6</td>
<td>79.3</td>
<td>58.1</td>
<td>75.2</td>
<td>82.8</td>
<td></td>
</tr>
<tr>
<td>PointCNN [7]</td>
<td>86.1</td>
<td>84.1</td>
<td>86.5</td>
<td>80.8</td>
<td>90.6</td>
<td>79.7</td>
<td>92.3</td>
<td>88.4</td>
<td>85.3</td>
<td>96.1</td>
<td>77.2</td>
<td>95.2</td>
<td>84.2</td>
<td>80.0</td>
<td>83.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RS-CNN [9]</td>
<td>86.2</td>
<td>83.5</td>
<td>84.8</td>
<td>88.8</td>
<td>79.6</td>
<td>91.2</td>
<td>81.1</td>
<td>91.6</td>
<td>88.4</td>
<td>86.0</td>
<td>96.0</td>
<td>73.7</td>
<td>94.1</td>
<td>83.4</td>
<td>60.5</td>
<td>77.7</td>
<td>83.6</td>
<td></td>
</tr>
<tr>
<td>PointASNL</td>
<td>86.1</td>
<td>84.1</td>
<td>84.7</td>
<td>87.9</td>
<td>79.7</td>
<td>92.2</td>
<td>73.7</td>
<td>91.0</td>
<td>87.2</td>
<td>84.2</td>
<td>95.8</td>
<td>74.4</td>
<td>95.2</td>
<td>81.0</td>
<td>63.0</td>
<td>76.3</td>
<td>83.2</td>
<td></td>
</tr>
</tbody>
</table>

Table 6. Semantic segmentation results on S3DIS dataset evaluated on Area 5.

<table>
<thead>
<tr>
<th>Method</th>
<th>OA</th>
<th>mAcc</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>ceiling</td>
<td>49.0</td>
<td>41.1</td>
<td>88.8</td>
</tr>
<tr>
<td>floor</td>
<td>63.9</td>
<td>57.3</td>
<td>92.3</td>
</tr>
<tr>
<td>wall</td>
<td>92.0</td>
<td>98.2</td>
<td>98.4</td>
</tr>
<tr>
<td>beam</td>
<td>21.1</td>
<td>59.7</td>
<td>34.8</td>
</tr>
<tr>
<td>column</td>
<td>65.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>window</td>
<td>46.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>door</td>
<td>69.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>table</td>
<td>52.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chair</td>
<td>64.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sofa</td>
<td>69.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bookcase</td>
<td>87.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>board</td>
<td>55.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>clutter</td>
<td>67.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7. Semantic segmentation results on the S3DIS dataset with 6-fold cross validation.

<table>
<thead>
<tr>
<th>Method</th>
<th>OA</th>
<th>mAcc</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>ceiling</td>
<td>66.2</td>
<td>47.6</td>
<td>88.0</td>
</tr>
<tr>
<td>floor</td>
<td>66.5</td>
<td>56.5</td>
<td>92.5</td>
</tr>
<tr>
<td>wall</td>
<td>92.4</td>
<td>96.4</td>
<td>79.2</td>
</tr>
<tr>
<td>beam</td>
<td>34.4</td>
<td>51.6</td>
<td>68.1</td>
</tr>
<tr>
<td>column</td>
<td>65.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>window</td>
<td>59.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>door</td>
<td>65.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>table</td>
<td>66.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>chair</td>
<td>65.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sofa</td>
<td>71.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bookcase</td>
<td>52.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>board</td>
<td>41.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>clutter</td>
<td>50.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8. Semantic segmentation results on the SemanticKITTI.

<table>
<thead>
<tr>
<th>Method</th>
<th>mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>road</td>
<td>61.6</td>
</tr>
<tr>
<td>sidewalk</td>
<td>35.7</td>
</tr>
<tr>
<td>parking</td>
<td>15.8</td>
</tr>
<tr>
<td>other-ground</td>
<td>41.4</td>
</tr>
<tr>
<td>building</td>
<td>46.3</td>
</tr>
<tr>
<td>car</td>
<td>0.1</td>
</tr>
<tr>
<td>truck</td>
<td>1.3</td>
</tr>
<tr>
<td>bicycle</td>
<td>0.3</td>
</tr>
<tr>
<td>motorcycle</td>
<td>0.8</td>
</tr>
<tr>
<td>other-vehicle</td>
<td>31.0</td>
</tr>
<tr>
<td>vegetation</td>
<td>4.6</td>
</tr>
<tr>
<td>trunk</td>
<td>17.6</td>
</tr>
<tr>
<td>terrain</td>
<td>0.2</td>
</tr>
<tr>
<td>person</td>
<td>0.2</td>
</tr>
<tr>
<td>bicyclist</td>
<td>12.9</td>
</tr>
<tr>
<td>motorcyclist</td>
<td>2.4</td>
</tr>
<tr>
<td>fence</td>
<td>3.7</td>
</tr>
<tr>
<td>pole</td>
<td>15.9</td>
</tr>
<tr>
<td>traffic sign</td>
<td>0.8</td>
</tr>
</tbody>
</table>

PointASNL | 46.8 | 1.8  | 0.0  | 25.1 | 29.2 | 84.1 | 52.2 | 70.6 | 34.2 | 57.6 | 43.9 | 57.8 | 36.9 |
Figure 3. More examples on S3DIS datasets.

Figure 4. More examples on ScanNet datasets.
Figure 5. More examples on SemanticKITTI datasets.

Figure 6. Visualized results of AS module. (a) Sampled points via farthest point sampling (FPS). (b) Sampled points adjusted by AS module.