# Supplementary Material for Decompose Novel into Known: Part Concept Learning For 3D Novel Class Discovery

**Anonymous Author(s)**
Affiliation
Address
email

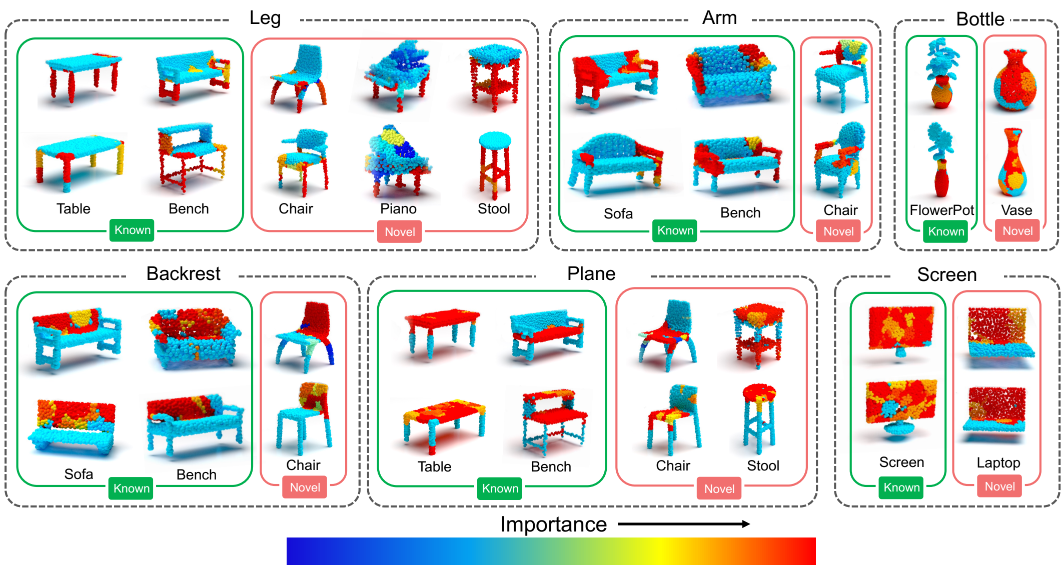



Figure I: Visualization of different part concepts and their corresponding activations on different shapes of known and novel classes. Red indicates higher activations and similarities.

## I In-depth Analysis of the Method

**Part Concept Visualization.** In order to reveal what part concepts can learn, we show concept activation maps between some part concepts and part features in Fig. I. We can see that part concepts can activate similar semantic parts of both known and novel shapes, such as legs, arms, bottles, backrests, planes, and screens. This confirms that part concepts can effectively generalize knowledge learned from known classes to novel classes and represent novel classes as the composition of part concepts. For example, a chair can be represented as {legs, arms, a backrest, and a plane}. We notice that part concepts can be robustly discovered and activated even if geometric structures are slightly different, e.g., arms of chairs, sofas, and benches.

**Feature Visualization of Novel Classes.** We show the feature distribution of novel classes learned by all compared methods on the high similarity task of ModelNet in Fig. II. DTC [2] mingles features of different novel classes together and impedes novel class recognition. The feature distributions of AutoNovel [3], NCL [9], UNO [1] and IIC [4] are less confusing but there is no clear category boundary for different novel classes. Kmeans+ can produce better intra-class consistency and can

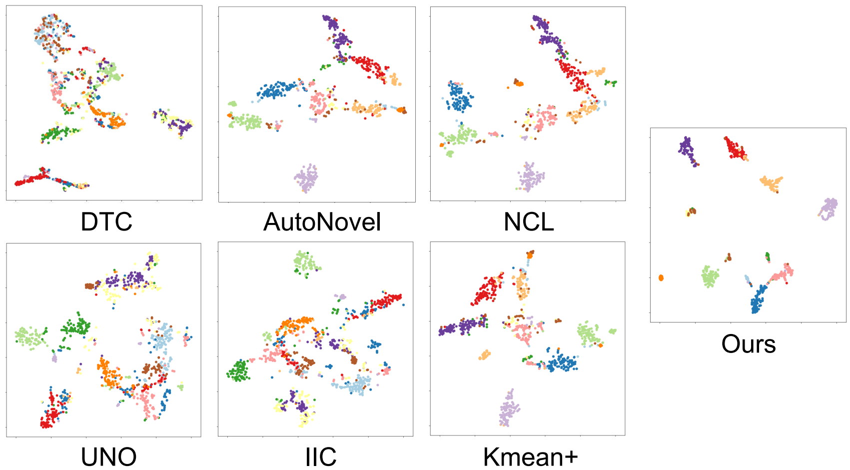
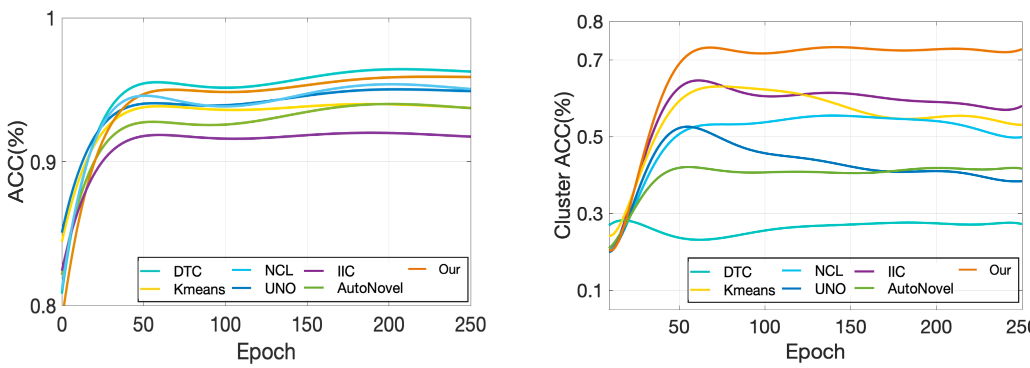



Figure II: Learned embeddings visualized by 2D t-SNE. We use the ModelNet high-similarity task for the experiment. Each color denotes a different novel class.




(a) Known class

(b) Novel class

Figure III: (a) The accuracy curves of known classes during training. (b) The clustering accuracy curves of novel classes during training.

separate most novel classes. A possible reason is that pseudo labels by K-means clustering on the unlabeled training set can reflect the overall distribution to some extent. However, some shapes are incorrectly grouped into the wrong class due to the poor discrimination of learned features. By exploring the part concept compositions of different novel classes, our method can push away the embedding regions of novel classes to a greater extent compared to other methods and achieve much tighter intra-class representations.

**Train Curves.** As described in Sec. 3.1, the learned features of compared methods are gradually biased to known classes during the training process and lose generalizability to novel classes. To further confirm our analysis, we demonstrate the converging curves of known class accuracy and novel class cluster accuracy during the training of seven methods on the high similarity task of ModelNet in Fig III. As shown in Fig. IIIa, the known class accuracy of all methods increases quickly in the first 30 epochs and grows slowly after 100 epochs. However, in Fig. IIIb, with the increase in the known class accuracy after 50 epochs, the clustering accuracies for novel classes of UNO, IIC, and Kmeans+ start to decrease. This indicates the known class accuracy competes with the novel class clustering accuracy, and features beneficial to known class recognition instead turn out to harm the generalization to novel classes. The clustering accuracy of our method stays stable after 60 epochs. Meanwhile, the trend of the cluster accuracy is consistent with that of the known class accuracy suggesting that learned features on known classes encourage the recognition of novel ones. This good generalization is attributed to the fact that part concepts construct a shared embedding space between known and novel classes.

# II Ablation Study

To further investigate our proposed model components, we conduct additional ablation experiments in Tab. I. In ②, ③ and ④, both Poincareé distance and $L_{sd}$ are proposed to encourage sharper part concept activation. Each of these two modules can be used individually to obtain performance gains (+3.7% vs +8.5%, +5.1% vs +7.4%), and they can be used together to achieve an even bigger improvement (+9.5% vs +15.2%). Without Poincareé distance and $L_{sd}$, using $L_{sc}$ and $L_{cd}$ alone can only achieve minor improvements as shown in ⑤ and ⑥. This is because that part concepts are less diverse and tend to encode the global feature of known classes that are hard to be generalized to novel classes. Since novel shapes may contain new part concepts that are not present in known classes, we concatenate the local part features $Z_l$ with the part composite features $Z_p$. As shown in ⑧, this design can obtain better results on low-similarity novel classes (+0.7% vs + 1.8%).

Table I: Ablation experiments. PCB means part concept bank. LGA, Poincaré, Concat and PRE are local geometric aggregation, Poincaré distance, concatenate the local part features $Z_l$ and position relation encoder as shown in Fig. 2.

| | LGA | PCB | Poincaré | $L_{sd}$ | $L_{sc}$ | $L_{cd}$ | Concat | PRE | High | Low |
|---|---|---|---|---|---|---|---|---|---|---|
| ① | ✓ | ✓ | | | | | ✓ | | 58.4 | 48.7 |
| ② | ✓ | ✓ | ✓ | | | | ✓ | | 62.1 | 57.2 |
| ③ | ✓ | ✓ | | ✓ | | | ✓ | | 63.5 | 56.1 |
| ④ | ✓ | ✓ | ✓ | ✓ | | | ✓ | | 67.9 | 63.9 |
| ⑤ | ✓ | ✓ | | | ✓ | | ✓ | | 60.1 | 51.5 |
| ⑥ | ✓ | ✓ | | | | ✓ | ✓ | | 59.8 | 49.3 |
| ⑦ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | 70.6 | 63.3 |
| ⑧ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | 71.3 | 65.1 |
| ⑨ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | **73.2** | **66.4** |

# III Different Backbones

We inspect how different network backbones impact the performance of 3D NCD. The comparisons are conducted on three popular 3D backbones including PointNet++ [6], DGCNN [8], and PointNeXt [7]. Among them, the PointNeXt achieves state-of-the-art performance for 3D classification and segmentation. We replace the backbone of compared methods with the above network architectures and evaluate their performance on the ModelNet high similarity task. The results are shown in Tab. II. the known class accuracy of the same 3D NCD method on different backbones does not vary too much (within $1\%$). On the other hand, the backbone has a significant influence on the novel-class performance for all the compared methods. For example, the weaker backbone DGCNN can achieve better novel class accuracy than the other two backbones. This is because DGCNN uses graphs to represent the local features of the point cloud, thus increasing the feature generalization on novel classes. Our method can surpass all the baselines by a very large margin (at least $+8.9\%$) on the novel-class accuracy even using a weak backbone (PointNet), while attaining comparable good performance on the known-class recognition. This superiority is primarily attributed to the part concept-based features that are more generalizable to novel classes than other backbones.

# IV Implementation Details

In all experiments, we adopt PointNet [5] as the backbone for point-wise feature learning and sample $1024$ points from the original point clouds as inputs. We use scaling, rotation, jittering, and translation as data augmentation to generate two augmented point clouds for both labeled and unlabeled inputs respectively during training. The network is optimized with the Adam optimizer with an initial learning rate of $0.001$. In our experiments, we first train the network for 30 epochs with labeled dataset $D^l$ to learn the part concept bank, then we train all networks for another $220$ epochs with both labeled and unlabeled datasets. We specify both the feature vector size $D$ and the dimension of the part concepts to $256$. The overall method is implemented in PyTorch, and we conduct all

Table II: The accuracy results of different backbones on the similarity split 3D NCD task. *High, Low* denote the novel classes have *High, Low* semantic similarity with known classes.

| | PointNeXt | | | PointNet++ | | | DGCNN | | |
|---|---|---|---|---|---|---|---|---|---|
| | High | Low | Known | High | Low | Known | High | Low | Known |
| DTC [2] | 43.8 | 32.6 | 93.9 | 50.9 | 44.4 | 94.3 | 51.3 | 38.0 | 94.6 |
| AutoNovel [3] | 51.6 | 46.5 | <u>96.0</u> | <u>59.0</u> | 43.4 | 96.7 | 49.0 | 40.9 | 96.7 |
| NCL [9] | 59.9 | 46.5 | 95.9 | 51.3 | 37.6 | **97.0** | <u>64.3</u> | 44.9 | 95.1 |
| UNO [1] | 61.6 | 32.9 | 95.8 | 53.0 | 43.2 | 93.0 | 62.9 | 45.6 | 96.7 |
| IIC [4] | <u>62.3</u> | 49.7 | 95.3 | 57.6 | <u>46.0</u> | 94.6 | 64.0 | <u>51.8</u> | **96.8** |
| Kmeans+ | 61.2 | <u>50.9</u> | **96.4** | 55.2 | 40.7 | 96.2 | 64.0 | 45.5 | 92.3 |
| Ours (PointNet) | **73.2** | **66.4** | 95.2 | **73.2** | **66.4** | 95.2 | **73.2** | **66.4** | 95.2 |
| Improvement | +10.9 | +15.5 | -1.2 | +14.2 | +20.4 | -1.8 | +8.9 | +14.6 | -1.6 |

experiments on a computer with an NVIDIA Tesla V100 GPU. Our method takes 1.26M parameters, 14.7G memory cost and 0.43s train time per batch measured on ModelNet. Readers are encouraged to refer to the released code for detailed implementation.

## References


[1] Enrico Fini, Enver Sangineto, Stéphane Lathuilière, Zhun Zhong, Moin Nabi, and Elisa Ricci. A unified objective for novel class discovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9284–9292, 2021.

[2] Kai Han, Andrea Vedaldi, and Andrew Zisserman. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8401–8409, 2019.

[3] Kai Han, Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Andrea Vedaldi, and Andrew Zisserman. Autonovel: Automatically discovering and learning novel visual categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6767–6781, 2021.

[4] Wenbin Li, Zhichen Fan, Jing Huo, and Yang Gao. Modeling inter-class and intra-class constraints in novel class discovery. *arXiv preprint arXiv:2210.03591*, 2022.

[5] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[6] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[7] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[8] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019.

[9] Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10867–10875, 2021.