ULIP: Learning Unified Representation of Language, Image and Point Cloud for 3D Understanding

Le Xue[1] *, Mingfei Gao[1] , Chen Xing[1] , Roberto Martín-Martín[1,2] , Jiajun Wu[3] , Caiming Xiong[1]
Ran Xu[1] , Juan Carlos Niebles[1] and Silvio Savarese[1]

[1] Salesforce Research, Palo Alto, USA
[2] UT Austin, Texas, USA [3] Stanford University, Stanford, USA

Project Website

Abstract

The understanding capabilities of current state-of-the-art 3D models are limited by datasets with a small number of annotated data and a pre-defined set of categories. In its 2D counterpart, recent advances have shown that similar problems can be significantly alleviated by employing knowledge from other modalities, such as language. Inspired by this, leveraging multimodal information for 3D modality could be promising to improve 3D understanding under the restricted data regime, but this line of research is not well studied. Therefore, we introduce ULIP to learn a unified representation of image, text, and 3D point cloud by pre-training with object triplets from the three modalities. To overcome the shortage of training triplets, ULIP leverages a pre-trained vision-language model that has already learned a common visual and textual space by training with massive image-text pairs. Then, ULIP learns a 3D representation space aligned with the common image-text space, using a small number of automatically synthesized triplets. ULIP is agnostic to 3D backbone networks and can easily be integrated into any 3D architecture. Experiments show that ULIP effectively improves the performance of multiple recent 3D backbones by simply pre-training them on ShapeNet55 using our framework, achieving state-of-the-art performance in both standard 3D classification and zero-shot 3D classification on ModelNet40 and ScanObjectNN. ULIP also improves the performance of PointMLP by around 3% in 3D classification on ScanObjectNN, and outperforms PointCLIP by 28.8% on top-1 accuracy for zero-shot 3D classification on ModelNet40. Our code and pre-trained models will be released here.

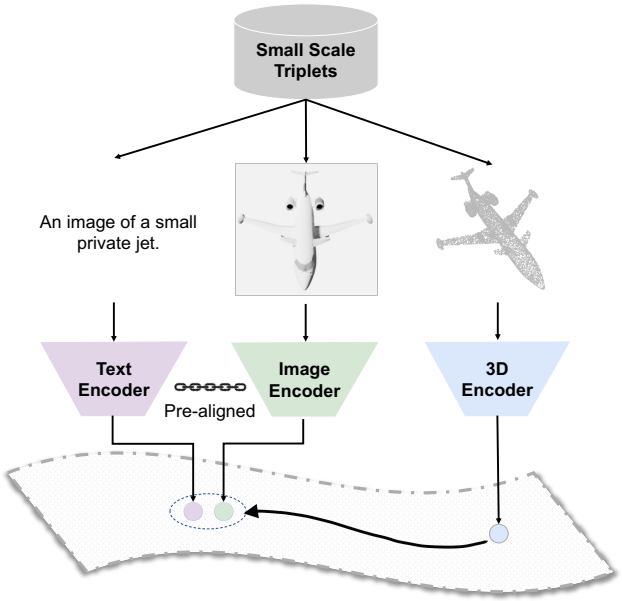

Figure 1. Illustration of ULIP. ULIP improves 3D understanding by aligning features from image, text and point cloud in the same space. To reduce the demand of 3D data, ULIP leverages image and text encoders that are pre-trained with large-scale image-text pairs, and aligns 3D representation to the pre-aligned image-text feature space using a small scale of training triplets.

1. Introduction

3D visual understanding research [10, 14, 15, 22, 23, 51] is drawing significant attention in recent years due to the increasing demand of real-world applications such as augmented/virtual reality [1, 25, 29, 44], autonomous driving [21, 55] and robotics [2, 46]. However, compared to its 2D counterpart, 3D visual understanding research is still limited by datasets with a small number of samples and a small set of pre-determined categories [43, 49]. For exam-

* Contact: lxue@salesforce.com

ple, ShapeNet55 [3], one of the largest publicly available 3D datasets, only contains around 52.5k samples of 3D objects with 55 category labels. That is in contrast to the 2D domain, where ImageNet [5] contains millions of images that cover thousands of categories. This scale limit of 3D data, caused by the high cost of 3D data collection and annotation [3, 9, 49, 56], has been hindering the generalization of 3D recognition models and their real-world applications.

To tackle the shortage of annotated data, existing work in other domains shows that employing knowledge from different modalities can significantly help the concept understanding in the original modality [37, 52]. Among such work, CLIP [37] pioneered alignment between 2D visual and textual features by pre-training on large-scale image-text pairs. It improves state-of-the-art visual concept recognition and enables zero-shot classification of unseen objects. However, multimodal learning that involves 3D modality, and whether it can help 3D recognition tasks are still not well studied.

In this paper, we propose Learning **U**nified Representation of **L**anguage, **I**mage and **P**oint Cloud (ULIP). An illustration of our framework is shown in Figure 1. Obtaining a unified representation space of all three modalities requires large-scale triplets of image, text, and point cloud as training data. However, such triplets remain hard to collect compared to the large-scale image-text pairs available. To circumvent the lack of triplet data, we take advantage of a vision-language model pre-trained on massive image-text pairs, and align the feature space of a 3D point cloud encoder to the pre-aligned vision/language feature space. When training the 3D encoder for space alignments, we use a small number of automatically synthesized triplets from ShapeNet55 [3] without requiring manual annotations. Making use of a pre-trained vision-language model lets us leverage the abundant semantics captured in the image-text feature space for 3D understanding. Our framework uses CLIP as the vision and language model because of its excellent generalization performance. During pre-training, we keep the CLIP model frozen and train the 3D encoder by aligning the 3D feature of an object with its corresponding textual and 2D visual features from CLIP using contrastive learning. The pre-trained 3D backbone model can be further fine-tuned for different downstream tasks.

ULIP has three major advantages. First, ULIP can substantially improve the recognition ability of 3D backbone models. Second, ULIP is agnostic to the architecture of 3D models; therefore, we can easily plug in any 3D backbones and improve them with ULIP. Third, aligning three modalities in the same feature space can potentially enable more cross-domain downstream tasks, including zero-shot 3D classification and image-to-3D retrieval.

We quantitatively evaluate ULIP on two fundamental 3D tasks: standard 3D classification and zero-shot 3D classification. We experiment with recent 3D networks including PointNet++ [34], PointMLP [27] and PointBERT [56]. Experimental results show that ULIP achieves state-of-the-art (SOTA) performance for both standard 3D classification and zero-shot 3D classification on ModelNet40 and ScanObjectNN. Specifically, ULIP surpasses PointMLP by around 3% in standard 3D classification on ScanObjectNN [43]. ULIP also outperforms PointCLIP [57] (the previous SOTA) by around 28.8% top-1 accuracy in zero-shot 3D classification on ModelNet40. Moreover, we showcase the potential of applying ULIP on the image to point cloud retrieval task. Qualitative evaluation demonstrate our promising potential for cross-modal applications.

2. Related Work

Multi-modal Representation Learning. Most existing multimodal approaches are about image and text modalities. Among these methods, one line of research focuses on learning interaction between image regions and caption words [4, 16, 17, 19, 26, 41] using transformer-based architectures. These methods show great predictive capability while being costly to train. The other line of research, such as CLIP [37], uses image and text encoders to output a single image/text representation for each image-text pair, and then aligns the representations from both modalities. This simple architecture makes training with massive noisy web data efficient, facilitating its zero-shot generalization capability.

The success of CLIP has promoted many image-text related research directions, including text-based image manipulation [32], open vocabulary object detection [8, 11] and language grounding [18]. The most related method to our work is PointCLIP [57]. It first converts the 3D point cloud into a set of depth maps and then leverages CLIP directly for zero-shot 3D classification. Unlike PointCLIP, which targets reshaping the task of point cloud and text matching to image and text alignment, our method learns a unified representation among image, text, and point cloud that substantially improves 3D understanding.

3D Point Cloud Understanding. There are mainly two streams of research lines for point cloud modeling. One is projecting a point cloud into 3D voxels [28, 40] and then using 2D/3D convolutions for feature extraction. PointNet [33] explores ingesting 3D point clouds directly. It extracts permutation-invariant feature from the point cloud that significantly impacts point-based 3D networks. PointNet++ [34] proposes a hierarchical neural network that extracts local features with increasing contextual scales. Recently, PointMLP [27] proposes a pure residual MLP network and achieves competitive results without integrating sophisticated local geometrical extractors. Moreover, self-supervised learning for 3D point clouds has also shown promising performance in 3D understanding field. Point-

BERT [56] adopts mask language modeling from BERT [6] to the 3D field, where it tokenizes 3D patches using an external model, randomly masks out 3D tokens, and predicts them back during pre-training. A more recent work, Point-MAE [31], directly operates the point cloud by masking out 3D patches and predicting them back using L2 loss. Our method is orthogonal to the above 3D encoders. Their performance on 3D recognition can be potentially improved by ULIP with no/minor modification.

3. Learning Unified Representation of Language, Image and Point Cloud

ULIP learns a unified representation space of images, texts and 3D point clouds via pre-training on triplets from these three modalities. In this section, we first introduce how we create such triplets for pre-training. Then, we present our pre-training framework.

3.1. Creating Training Triplets for ULIP

We build our dataset of triplets from ShapeNet55 [3], which is one of the most extensive public 3D CAD datasets. ShapeNet55 is the publicly-available subset of ShapeNet. It contains around 52.5K CAD models, each of which is associated with metadata that textually describes the semantic information of the CAD model. For each CAD model i in the dataset, we create a triplet $T_i : (I_i, S_i, P_i)$ of image I_i, text description S_i and point cloud P_i. ULIP will then use these triplets for pre-training.

Point Cloud Generation. We directly use the generated point cloud of each CAD model in ShapeNet55. We uniformly sample N_p points from the original point cloud. During pre-training, standard data augmentation techniques of 3D point clouds are performed, including random point drop, random scaling point cloud, shift point cloud and rotate perturbation. Then a 3D encoder takes the augmented point cloud P_i as input and outputs its 3D representation $\mathbf{h}_i^P$ via

$$\mathbf{h}_i^P = f_P(P_i), \tag{1}$$

where $f_P(\cdot)$ represents the 3D backbone encoder.

Multi-view Image Rendering. ShapeNet55 CAD models do not come with images. To obtain images that semantically align well with each CAD model, we synthesize multi-view images of each CAD model by placing virtual cameras around each object and rendering the corresponding RGB images and depth maps from each viewpoint.[1] Specifically, we render an RGB image with a depth map for every 12 degrees. Therefore, we get 30 RGB images and 30 depth maps for each object, 60 image candidates in total. During each iteration of pre-training, we randomly select one image or depth map from each CAD model's 60 renderred candidates as I_i and take I_i as input of the image encoder $f_I(\cdot)$ to extract the image feature $\mathbf{h}_i^I$,

$$\mathbf{h}_i^I = f_I(I_i). \tag{2}$$

Text Generation. We leverage the metadata that comes with each CAD model as the corresponding text description. The metadata includes a synset of taxonomy as a textual description of each CAD model. For each word in the metadata, we adopt simple prompts to construct meaningful sentences that will be utilized during pre-training. We follow prior works [8, 11] that use 63 prompts such as "a picture of [WORD]" in image-text pre-training tasks and additionally add a dedicated prompt "a point cloud model of [WORD]" to accommodate the 3D modality. In each training iteration, we randomly choose a word from the metadata and apply the 64 templates on the word to build a set of text descriptions, $\mathbf{S}_i$. Then we input $\mathbf{S}_i$ into our text encoder $f_S(\cdot)$ and get a set of representations, respectively. Finally, we conduct average pooling over the set of outputs as the text-domain representation $\mathbf{h}_i^S$ of object i,

$$\mathbf{h}_i^S = Avg(f_S(\mathbf{S}_i)). \tag{3}$$

3.2. Aligning Representations of Three Modalities

With the created triplets of image, text, and point cloud, ULIP conducts pre-training to align representations of all three modalities into the same feature space. Specifically, we take advantage of pre-trained vision-language models, i.e., CLIP, and train a 3D encoder by aligning the 3D feature with the features of image and text encoders ($f_I(\cdot)$ and $f_S(\cdot)$) of CLIP. By doing so, we hope that the abundant semantics already captured and aligned by CLIP's encoders can be employed for better 3D understanding. The resulting unified feature space enables numerous cross-modal applications among these three modalities and potentially improves the 3D recognition performance of the underlying 3D backbone encoder $f_P(\cdot)$.

Cross-modal Contrastive Learning. As shown in Figure 2, for an object i, features $\mathbf{h}_i^I$, $\mathbf{h}_i^S$ and $\mathbf{h}_i^P$ are extracted from image, text, and 3D point cloud encoders. Then contrastive loss among each pair of modalities is computed as follows,

$$\begin{aligned} L_{(M1,M2)} = \sum_{(i,j)} & -\frac{1}{2} log \frac{exp(\mathbf{h}_i^{M_1} \mathbf{h}_j^{M_2})}{\sum_k exp(\mathbf{h}_i^{M_1} \mathbf{h}_k^{M_2})} \\ & -\frac{1}{2} log \frac{exp(\mathbf{h}_i^{M_1} \mathbf{h}_j^{M_2})}{\sum_k exp(\mathbf{h}_k^{M_1} \mathbf{h}_j^{M_2})}, \end{aligned} \tag{4}$$

where M_1 and M_2 represent two modalities and (i, j) indicates a positive pair in each training batch.

Finally, we minimize $L_{(M_1,M_2)}$ for all modality pairs with different coefficients,

$$L_{final} = \alpha L_{(I,S)} + \beta L_{(I,P)} + \theta L_{(P,S)}. \tag{5}$$

[1] We utilize the following repository with their default settings in practice.
https://github.com/panmari/stanford-ShapeNet55-renderer/

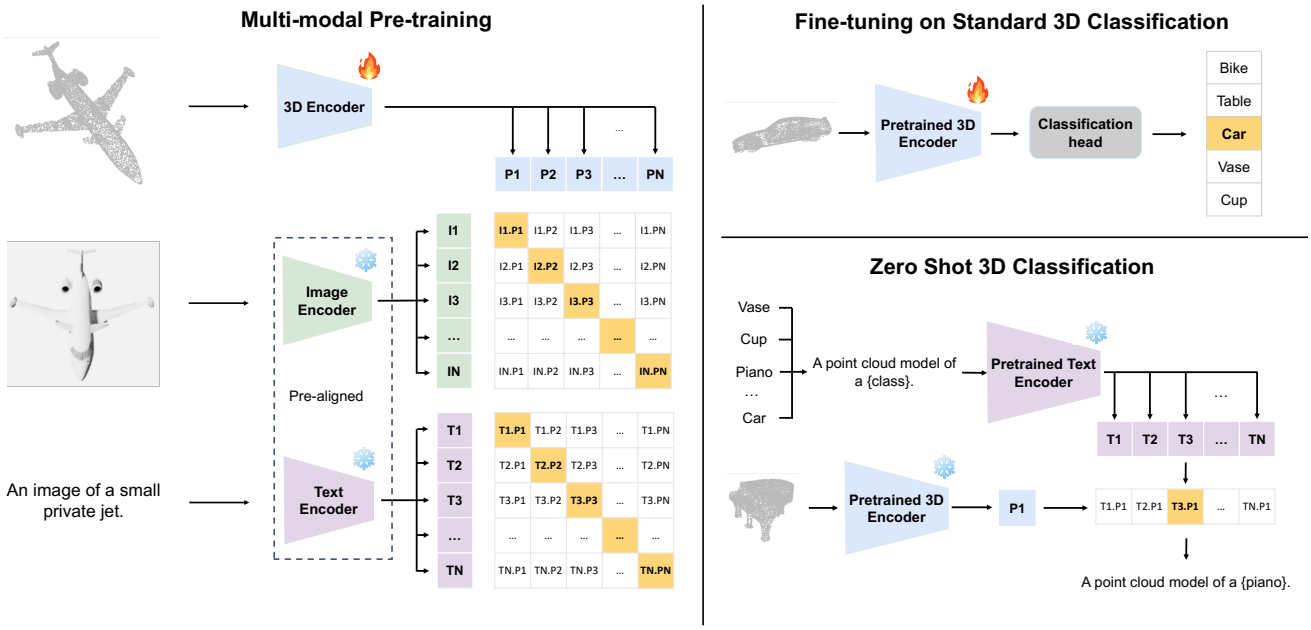

Figure 2. Illustration of our method. The inputs of multimodal pre-training (**Left**) are a batch of objects represented as triplets (image, text, point cloud). Image and text features are extracted from a pre-trained (frozen) vision and language model such as CLIP, and 3D features are extracted from a 3D encoder. Contrastive losses are applied to align the 3D feature of an object to its image and text features during pre-training. The pre-trained 3D encoders are further fine-tuned in downstream tasks, including standard 3D classification (**Top Right**) and zero-shot 3D classification (**Bottom Right**).

During pre-training, we find that if we update CLIP's image and text encoders, catastrophic forgetting will emerge due to our limited data size. This will lead to a significant performance drop when applying ULIP to downstream tasks. Therefore we freeze the weights of $f_S(\cdot)$ and $f_I(\cdot)$ during the entire pre-training and only update $f_P(\cdot)$ with L_{final}.

4. Experiments

To demonstrate the benefits of pre-training 3D backbone networks using ULIP, we conduct experiments on two 3D tasks: a standard 3D classification task that involves a single modality and a zero-shot 3D classification task that involves multimodal inputs. In this section, we first present experimental settings, including our experimenting 3D backbones, downstream datasets, and implementation details. Then we present the quantitative results of standard 3D classification and zero-shot 3D classification, respectively. Lastly, we include analyses of our model and show results on cross-modal retrieval.

4.1. 3D Backbone Networks

We experiment with the following 3D backbone networks under our framework.

PointNet++ [34] is an advanced version of PointNet [33]. It uses a hierarchical structure to better capture the local geometry of the point cloud, and becomes the cornerstone of many point cloud applications.

PointBERT [56] utilizes a transformer architecture for point cloud feature extraction. It improves its recognition ability by conducting self-supervised pre-training on ShapeNet55.

PointMLP [27] is the SOTA method on standard 3D classification task. It uses a residual MLP network with a lightweight geometric affine module to better capture local geometric features.

4.2. Downstream Datasets

We use the following two datasets for both standard and zero-shot 3D classification.

ModelNet40 is a synthetic dataset of 3D CAD models. It contains 9,843 training samples and 2,468 testing samples, covering 40 categories.[2]

ScanObjectNN is a dataset of scanned 3D objects from the real world. It contains 2,902 objects that are categorized into 15 categories. It has three variants: *OBJ_ONLY* includes ground truth segmented objects extracted from the scene meshes datasets; *OBJ_BJ* has objects attached with background noises and *Hardest* introduces perturbations such as translation, rotation, and scaling to the dataset [43].[3]

[2]For each CAD model, we utilized preprocessed point cloud from [27].
[3]We used the variants provided by [56] in our experiments.

Model	Overall Acc	Class-mean Acc
PointNet [33]	68.2	63.4
PointNet++ [34]	77.9	75.4
DGCNN [47]	78.1	73.6
MVTN [13]	82.8	–
PointBERT [56]	83.1	–
RepSurf-U [38]	84.6	–
PointMAE [31]	85.2	–
RepSurf-U (2x) [38]	86.0	–
PointBERT [56]	83.1	–
PointBERT + ULIP	**86.4** (↑ 3.3)	–
PointMLP [27]	85.7	84.4
PointMLP + ULIP	**88.8** (↑ 3.1)	**87.8** (↑ 3.4)
PointMLP †	86.5	85.1
PointMLP †+ ULIP	**89.4** (↑ 2.9)	**88.5** (↑ 3.4)

Table 1. 3D classification results on ScanObjectNN. ULIP significantly improves our baselines. Our best result outperforms SOTA largely by around 3% on Overall Acc. †indicates a model uses 2K sampled points and all others use 1K sampled points.

4.3. Implementation Details

Pre-training. For the 3D input, we uniformly sample $N_p = 1024$, 2048, or 8192 points for accommodating the requirements of different backbones. The inputs of image and text modalities are generated as described in Section 3.1. During pre-training, we utilize an advanced version of CLIP, namely SLIP [30], that shows superior performance as our image-text encoders. As mentioned in Section 3.2, we freeze the image and text encoders and only update the 3D encoder's parameters during pre-training. ULIP is trained for 250 epochs. We use 64 as the batch size, 10^{-3} as the learning rate, and AdamW as the optimizer.

Standard 3D Classification. On ModelNet40, we use the learning rate as 0.00015 and fine-tune our model for 200 epochs, with the batch size as 24 for PointNet++. For PointMLP, we set the learning rate as 0.1 and fine-tune the model for 300 epochs, with the batch size as 32.

On ScanObjectNN, we use the learning rate of 0.03 and finetune for 350 epochs with batch size 32 for PointMLP. For PointBERT, we use the learning rate of 0.0002 and fine-tune for 300 epochs with batch size 32.

Zero-Shot 3D Classification. Following [57], zero-shot 3D classification is conducted by measuring distances between the 3D features of an object and the text features of category candidates. The category that introduces the smallest distance is selected as the predicted category, as shown in Figure 2. We use our pre-trained models as they are when performing zero-shot classification. There is no finetuning stage involved. We keep using the same prompt strategy as it is during pre-training when constructing text features for each category candidate in this task.

All of our experiments are conducted using PyTorch. Pre-training and finetuning experiments use 8 and 1 A100 GPUs, respectively.

4.4. Standard 3D Classification

We demonstrate the effectiveness of ULIP by improving different 3D classification baselines. We follow the original settings of the baselines in our experiments. When applying ULIP, the only difference is that we pre-train the 3D networks under our framework before finetuning them with the labeled point cloud. Since the structure of the 3D backbone is unchanged, our framework does not introduce extra latency during inference time. For all experiments, we follow the community practice of using OA (Overall Accuracy) and mAcc (Class Average Accuracy) as our evaluation metrics.

Model	Overall Acc	Class-mean Acc
PointNet [33]	89.2	86.0
PointCNN [20]	92.2	-
SpiderCNN [54]	92.4	-
PointConv [48]	92.5	-
Point Transformer [58]	92.8	-
KPConv [42]	92.9	-
DGCNN [45]	92.9	90.2
PCT [12]	93.2	-
RS-CNN* [24]	93.6	-
GDANet [53]	93.8	-
GBNet [36]	93.8	91.0
MTVN [13]	93.8	92.0
RPNet [39]	94.1	-
CurveNet [50]	94.2	-
PointNet++(ssg)	90.7	-
PointNet++(ssg) + ULIP	**93.4** (↑ 2.7)	91.2
PointBERT	93.2	-
PointBERT + ULIP	**94.1** (↑ 0.9)	-
PointMLP	94.1	91.3
PointMLP + ULIP	**94.3** (↑ 0.2)	**92.3** (↑ 1.0)
PointMLP*	94.5	91.4
PointMLP* + ULIP	**94.7** (↑ 0.2)	**92.4** (↑ 1.0)

Table 2. Standard 3D classification results on ModelNet40. ULIP significantly improves our baselines. Our best number achieves new SOTA. * means a voting technique is applied to the method to boost performance.

Experimental Results. We present the standard 3D classification performances of our baselines and our methods on ScanObjectNN in Table 7. As shown, the performances of

Model	ALL		Medium		Hard	
	top-1	top5	top-1	top-5	top-1	top-5
PointCLIP	20.2	–	10.4	–	8.3	–
PointNet++(ssg) + ULIP	55.7	75.7	35.6	64.4	33.7	55.8
PointNet++(msg) + ULIP	58.4	78.2	36.9	67.2	33.9	59.6
PointMLP + ULIP	61.5	80.7	43.2	72.0	36.3	65.0
PointBERT + ULIP	**60.4** (↑ 40.2)	84.0	**40.4** (↑ 30.0)	72.1	**37.1** (↑ 28.8)	66.3

Table 3. Zero-shot 3D classification on ModelNet40. ULIP-based methods outperform the previous SOTA (PointCLIP) by a very large margin in different evaluation sets.

Model	ALL	
	top-1	top-5
PointCLIP	15.4	–
PointMLP + ULIP	44.6	82.3
PointNet++(ssg) + ULIP	45.6	73.8
PointBERT + ULIP	48.5	79.9
PointNet++(msg) + ULIP	**49.9** (↑ 34.5)	78.8

Table 4. Zero-shot 3D classification on ScanObjectNN. ULIP-based methods outperform the previous SOTA (PointCLIP) by a very large margin (at least 29.2% on top-1 accuracy).

our baselines are significantly improved by ULIP. Specifically, our framework improves PointBERT and PointMLP significantly by around 3%. When we apply ULIP on the strongest backbone, PointMLP, ULIP+PointMLP† achieves the new SOTA performance, and outperforms previous SOTA, RepSurf-U(2×), by 3.4% Overall Accuracy.

In Table 2, we show the standard 3D classification results on ModelNet40. Unlike ScanObjectNN, which contains scans of real objects, ModelNet40 is a synthetic dataset thus it is easier for classification. The Overall Accuracy of recent methods is already saturated around 94% on this dataset. Even in such a scenario, from Table 2 we can see that ULIP is still able to improve the Overall Accuracy of all of our baselines. Among them, ULIP+PointMLP* achieves a new SOTA. For the class-mean accuracy metric, we also observe decent performance improvement when using ULIP and achieves a new SOTA as well.

4.5. Zero-Shot 3D Classification

By aligning the 3D representation with text and image representations, ULIP also enables the 3D backbone networks to conduct tasks involving multiple modalities. We evaluate zero-shot 3D classification in this section.

PointCLIP is the first work and the current SOTA for zero-shot 3D classification. It conducts zero-shot 3D classification by first converting a 3D point cloud into 6 orthogonal depth maps, then using CLIP's image encoder to get ensembled depth map features, and finally using CLIP to match text and depth map features for zero-shot classification. We use it as our major baseline and follow its evaluation protocol in this task. For all experiments, we report top-1 and top-5 OA (Overall Accuracy).

Model	Modality	ALL	
		top-1	top-5
PointNet++ ssg + ULIP	P+T	33.4	73.0
PointNet++ ssg + ULIP	P+I	35.3	72.8
PointNet++ ssg + ULIP	P+I+T	**45.6**	**73.8**
PointNet++ msg + ULIP	P+T	39.2	70.4
PointNet++ msg + ULIP	P+I	34.9	71.3
PointNet++ msg + ULIP	P+I+T	**49.9**	**78.8**
PointMLP + ULIP	P+T	41.3	76.1
PointMLP + ULIP	P+I	33.6	74.7
PointMLP + ULIP	P+I+T	**44.6**	**82.3**
PointBERT + ULIP	P+T	31.0	69.0
PointBERT + ULIP	P+I	36.3	71.3
PointBERT + ULIP	P+I+T	**48.5**	**79.9**

Table 5. Analysis of aligning three vs. two modalities on zero-shot 3D classification on ScanObjectNN. Results show that aligning representations of three modalities always produces better results than two modalities.

Evaluation Sets. To perform a fair comparison with PointCLIP, we evaluate zero-shot 3D classification on the entire test sets of both ModelNet40 and ScanObjectNN. We refer to this set as *ALL*.

Besides, we notice that there are some common classes between our pre-train dataset, ShapeNet55, and ModelNet40. Evaluations on these common classes might introduce an unfair comparison of zero-shot performance. To deal with this issue, we create two more sets in ModelNet40, referred to as *Medium* and *Hard* for evaluation.
Medium set: We remove the ModelNet40 categories whose exact category names exist in our pre-training category list.
Hard set: In the "Medium" category list, there are still some category names that are synonyms to the pre-training cat-

Model	Modalitiy aligned	ALL		Medium		Hard	
		top1	top5	top1	top5	top1	top5
PointNet++ ssg + ULIP	P+T	44.9	70.3	17.2	55.0	20.3	50.1
PointNet++ ssg + ULIP	P+I	35.3	67.2	33.6	62.4	30.1	55.1
PointNet++ ssg + ULIP	P+I+T	**55.7**	**75.7**	**35.6**	**64.4**	**33.7**	**55.8**
PointNet++ msg + ULIP	P+T	48.0	63.8	17.8	42.3	21.3	40.7
PointNet++ msg + ULIP	P+I	36.4	64.4	34.7	59.0	31.0	52.0
PointNet++ msg + ULIP	P+I+T	**58.4**	**78.2**	**36.9**	**67.2**	**33.9**	**59.6**
PointMLP + ULIP	P+T	52.2	73.0	23.3	60.8	18.1	52.2
PointMLP + ULIP	P+I	34.6	64.3	31.3	61.7	27.0	53.7
PointMLP + ULIP	P+I+T	**61.5**	**80.7**	**43.2**	**72.0**	**36.3**	**65.0**
PointBERT + ULIP	P+T	44.7	66.0	19.4	49.3	14.7	39.3
PointBERT + ULIP	P+I	35.5	66.9	35.0	64.4	34.1	59.1
PointBERT + ULIP	P+I+T	**60.4**	**84.0**	**40.4**	**72.1**	**37.1**	**66.3**

Table 6. Analysis of aligning three vs. two modalities on zero-shot 3D classification on ModelNet40. Results show that aligning representations of three modalities always produces better results than two modalities.

egories, such as 'cup' vs. 'mug' and 'chair' vs. 'stool.' Therefore, for the "Hard" ModelNet40 category list, we remove the categories from the "Medium" list with semantically similar counterparts in pre-training categories.

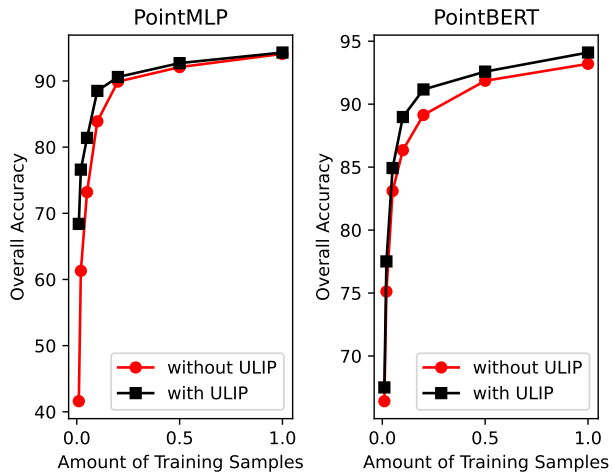

Figure 3. Data efficiency comparison. The X axis indicates the percentage of samples used for training and Y axis denotes the overall accuracy. Both PointMLP and PointBERT are significantly improved when pre-training with ULIP.

Experimental Results. We present the zero-shot 3D classification results on ModelNet40 in Table 3 and the results on ScanObjectNN in Table 4. We can see that all ULIP-based methods significantly outperform our major baseline, PointCLIP, by a large margin in every evaluation set. Specifically, on the *Hard set*, our best performing method, ULIP + PointBERT, outperforms PointCLIP by around 29% in top-1 accuracy. It also indicates that the superior performance of ULIP-based methods is not caused by pre-training the model on exact/similar categories as the target categories. Instead, it suggests that aligning the representations of different modalities can benefit the recognition of rare categories in general. Results in Table 4 demonstrate that ULIP-based methods consistently surpass PointCLIP on real scanned objects. Furthermore, all of the 3D backbones outperform the SOTA zero-shot method, PointCLIP, by ∼30% with the help of our ULIP framework.

4.6. Analyses

Align Representations, Three Modalities or Two? As described in Eq. 5, ULIP by default aligns the 3D representation with both the text and image representations during pre-training. We wonder to what extent ULIP will still work if we align the 3D representation to only the text feature or image feature alone. In this section, we conduct an ablation study for ULIP by aligning two rather than three modalities in zero-shot settings. Results are shown in Table 5 and Table 6 for ScanObjectNN and ModelNet40 datasets, respectively. As we can see in both tables, aligning the 3D modality with both text and image modalities consistently achieves the best performance compared to aligning with either image or text modality in every scenario with each baseline.

Data Efficiency. Model pre-training could potentially reduce the demand for labeled data during fine-tuning in downstream tasks. We validate the data efficiency of ULIP by comparing it with baselines under a varying number of fine-tuning samples. The comparison results are shown in Figure 3. As shown in Figure 3 (left), PointMLP's per-

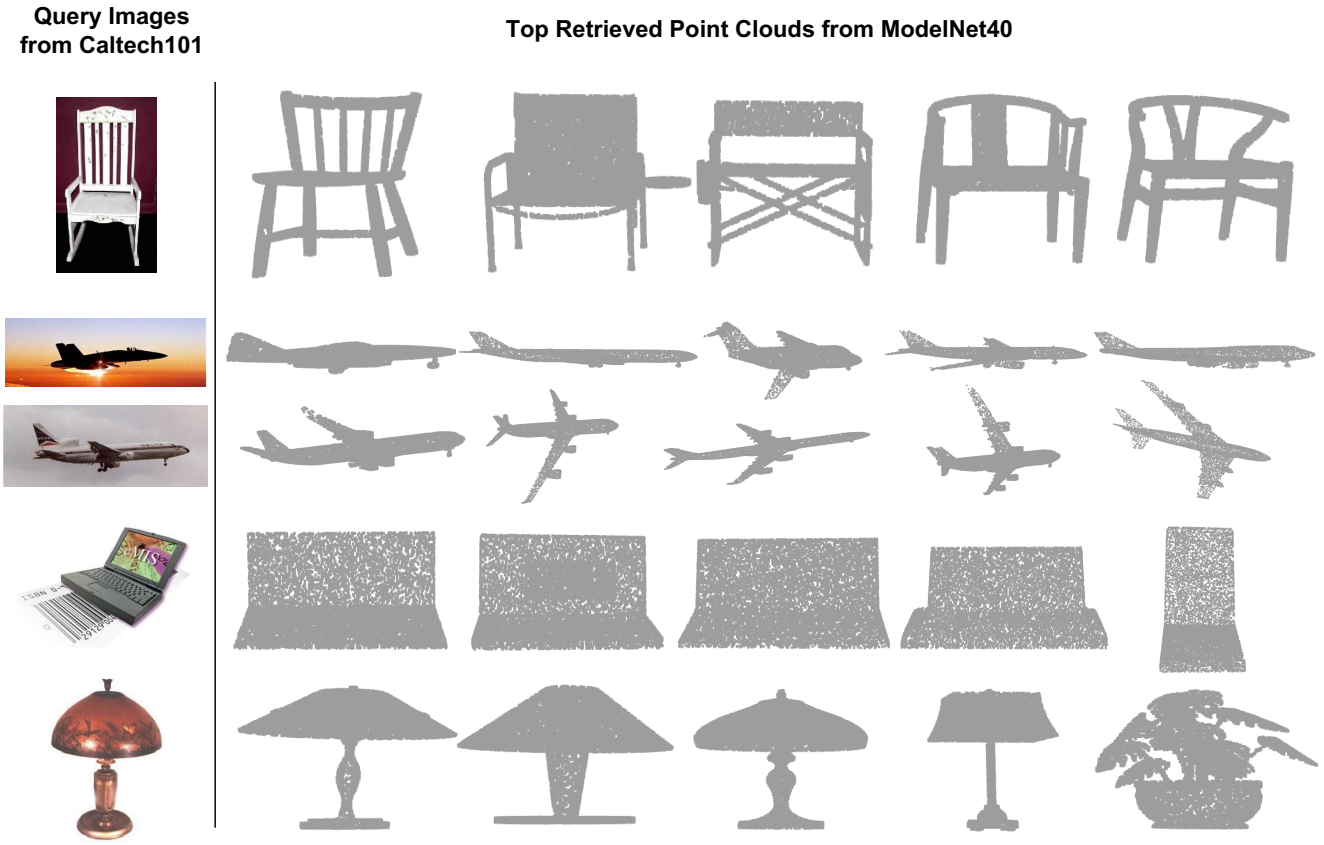

Figure 4. Qualitative results of real image to point cloud retrieval. Query images are from Caltech101, and point clouds are from ModelNet40. We show the top-5 retrieved point cloud models, ranked in order. The results demonstrate the retrieval capability of our model.

formance is largely improved in the low data regime when pre-trained under the ULIP framework. When we compare PointBERT with PointMLP baselines (two red lines in the two sub-figures), we observe that PointBERT performs better than PointMLP when using less than 20% training data. This is because of that the PointBERT model itself is pre-trained on ShapeNet55. Although both ULIP and PointBERT are pre-trained on ShapeNet55, ULIP still improves PointBERT by a clear margin, as shown in Figure 3 (right).

4.7. Cross-Modal Retrieval

As mentioned in Section 1, one of the benefits of ULIP is that it enables more cross-modal downstream tasks. Here, we qualitatively show the potential of using ULIP to conduct real image to point cloud retrieval.

We use our pre-trained ULIP with PointBERT as the 3D encoder directly. We conduct a small-scale experiment with real images from Caltech101 [7] and use the images to retrieve 3D point clouds from around 2.5k samples over 40 categories in ModelNet40. In Figure 4, we show the top-5 retrieved 3D point cloud models (ranked in order) using image examples from categories of *chair*, *airplane*, *laptop* and *lamp*. The results show encouraging signs that our pre-trained model has learned meaningful features across image and 3D point cloud modalities. Surprisingly, the top-1 retrieved 3D models have the closest appearance to the query images compared to other retrieved 3D models. For example, when we use images from different aircraft types (fight and airliner) for retrieval (2nd and 3rd rows), the retrieved top-1 point clouds maintain the subtle difference of the query images.

5. Conclusions

We propose ULIP, a pre-training framework that aligns multiple modalities of image, text, and point cloud in the same feature space. We take advantage of the pre-trained text and image encoders and improve different 3D encoders using our framework. Experiments results show that ULIP can effectively improve representations of 3D backbones. Our method achieves state-of-the-art performance in both zero-shot and standard 3D classification tasks, and our qualitative results show that ULIP has promising potential for cross-modal retrieval applications.

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016. 1

[2] Cesar Cadena, Anthony R Dick, and Ian D Reid. Multi-modal auto-encoders as joint estimators for robotics scene understanding. In *Robotics: Science and systems*, volume 5, 2016. 1

[3] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 3

[4] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020. 2

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[7] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004. 8

[8] Mingfei Gao, Chen Xing, Juan Carlos Niebles, Junnan Li, Ran Xu, Wenhao Liu, and Caiming Xiong. Open vocabulary object detection with pseudo bounding-box labels. *arXiv preprint arXiv:2111.09452*, 2021. 2, 3

[9] Ankit Goyal, Hei Law, Bowei Liu, Alejandro Newell, and Jia Deng. Revisiting point cloud shape classification with a simple and effective baseline. In *International Conference on Machine Learning*, pages 3809–3820. PMLR, 2021. 2

[10] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9224–9232, 2018. 1

[11] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 2, 3

[12] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021. 5

[13] Abdullah Hamdi, Silvio Giancola, and Bernard Ghanem. Mvtn: Multi-view transformation network for 3d shape recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2021. 5

[14] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11108–11117, 2020. 1

[15] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4558–4567, 2018. 1

[16] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 2

[17] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[18] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 2

[19] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 2

[20] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on X-transformed points. *arXiv preprint arXiv:1801.07791*, 2018. 5

[21] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17182–17191, 2022. 1

[22] Zhichao Li, Feng Wang, and Naiyan Wang. Lidar r-cnn: An efficient and universal 3d object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7546–7555, 2021. 1

[23] Yongcheng Liu, Bin Fan, Gaofeng Meng, Jiwen Lu, Shiming Xiang, and Chunhong Pan. Densepoint: Learning densely contextual representation for efficient point cloud processing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5239–5248, 2019. 1

[24] Yongcheng Liu, Bin Fan, Shiming Xiang, and Chunhong Pan. Relation-shape convolutional neural network for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8895–8904, 2019. 5

[25] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *Proceed-*

ings of the IEEE/CVF International Conference on Computer Vision, pages 2949–2958, 2021. 1

[26] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 2

[27] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *arXiv preprint arXiv:2202.07123*, 2022. 2, 4, 5

[28] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 922–928. IEEE, 2015. 2

[29] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2917, 2021. 1

[30] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *European Conference on Computer Vision*, pages 529–544. Springer, 2022. 5

[31] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. *arXiv preprint arXiv:2203.06604*, 2022. 3, 5

[32] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 2

[33] 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. 2, 4, 5

[34] 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. 2, 4, 5

[35] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Abed Al Kader Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *arXiv preprint arXiv:2206.04670*, 2022. 12

[36] Shi Qiu, Saeed Anwar, and Nick Barnes. Geometric back-projection network for point cloud classification. *IEEE Transactions on Multimedia*, 24:1943–1955, 2021. 5

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[38] Haoxi Ran, Jun Liu, and Chengjie Wang. Surface representation for point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18942–18952, 2022. 5

[39] Haoxi Ran, Wei Zhuo, Jun Liu, and Li Lu. Learning inner-group relations on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15477–15487, 2021. 5

[40] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10529–10538, 2020. 2

[41] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 2

[42] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 5

[43] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1, 2, 4

[44] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022. 1

[45] 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. 5

[46] Christian Wojek, Stefan Walk, Stefan Roth, and Bernt Schiele. Monocular 3d scene understanding with explicit occlusion reasoning. In *CVPR 2011*, pages 1993–2000. IEEE, 2011. 1

[47] Bo Wu, Yang Liu, Bo Lang, and Lei Huang. Dgcnn: Disordered graph convolutional neural network based on the gaussian mixture model. *Neurocomputing*, 321:346–356, 2018. 5

[48] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, 2019. 5

[49] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 1, 2

[50] Tiange Xiang, Chaoyi Zhang, Yang Song, Jianhui Yu, and Weidong Cai. Walk in the cloud: Learning curves for point clouds shape analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 915–924, 2021. 5

[51] Yu Xiang, Wonhui Kim, Wei Chen, Jingwei Ji, Christopher Choy, Hao Su, Roozbeh Mottaghi, Leonidas Guibas, and Silvio Savarese. Objectnet3d: A large scale database for 3d ob-

ject recognition. In *European Conference Computer Vision (ECCV)*, 2016. 1

[52] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O Pinheiro. Adaptive cross-modal few-shot learning. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[53] Mutian Xu, Junhao Zhang, Zhipeng Zhou, Mingye Xu, Xiaojuan Qi, and Yu Qiao. Learning geometry-disentangled representation for complementary understanding of 3d object point cloud. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3056–3064, 2021. 5

[54] Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 87–102, 2018. 5

[55] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 1

[56] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19313–19322, 2022. 2, 3, 4, 5

[57] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8552–8562, 2022. 2, 5

[58] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021. 5

A. Appendix

A.1. PointNeXt Backbone Experiments

PointNeXt [35] is a concurrent work which proposes a lightweight backbone based on PointNet++ and in particularly it gives promising results on the ScanObjectNN benchmark. In order to demonstrate the effectiveness of our ULIP on this most recent backbone, we pre-train PointNeXt using ULIP, and use the pre-trained weights to finetune on the ScanObjectNN dataset.

As shown in Table 7, ULIP significantly improves PointNeXt in both Overall Accuracy and Class-mean Accuracy.

Model	Overall Acc	Class-mean Acc
PointNeXt* [35]	87.4	85.8
PointNeXt + ULIP	**89.2** (↑ 1.8)	**88.0** (↑ 2.2)
PointNeXt †*	87.5	85.9
PointNeXt †+ ULIP	**89.7** (↑ 2.2)	**88.6** (↑ 2.7)

Table 7. 3D classification results on ScanObjectNN for PointNeXt. †indicates a model uses 2K sampled points and all others use 1K sampled points. * indicates it's reproduced result.

A.2. Details of Evaluation Sets in Zero Shot Classification

When evaluating zeroshot classification, we notice that there are some common classes between our pre-train dataset, ShapeNet55, and ModelNet40. Evaluations on these common classes might introduce an unfair comparison of zeroshot performance. Therefore, we introduced three different validation sets for evaluating our models and our baselines on ModelNet40.

All Set: Includes all the categories in ModelNet40 as shown in Table 8.

airplane	bathtub	bed	bench	bookshelf
bottle	bowl	car	chair	cone
cup	curtain	desk	door	dresser
flower_pot	glass_box	guitar	keyboard	lamp
laptop	mantel	monitor	night_stand	person
piano	plant	radio	range_hood	sink
sofa	stairs	stool	table	tent
toilet	tv_stand	vase	wardrobe	xbox

Table 8. ModelNet40 All Set.

Medium Set: We remove categories whose exact category names exist in our pre-training dataset. The resulting categories in this set is shown in Table 9.

cone	cup	curtain	door	dresser
glass_box	mantel	monitor	night_stand	person
plant	radio	range_hood	sink	stairs
stool	tent	toilet	tv_stand	vase
wardrobe	xbox			

Table 9. ModelNet40 Medium Set.

Hard Set: We remove both extract category names and their synonyms in our pre-training dataset. The final *Hard Set* is shown in Table 10

cone	curtain	door	dresser	glass_box
mantel	night_stand	person	plant	radio
range_hood	sink	stairs	tent	toilet
tv_stand	xbox			

Table 10. ModelNet40 Hard Set.