PointLoRA: Low-Rank Adaptation with Token Selection for Point Cloud Learning

Supplementary Material

In this supplementary material, we further present the following descriptions and experiments to elaborate the results and conclusions addressed in the main paper.

- Section A: Detailed implementation specifications;
- Section B: Extended experimental results;
- Section C: Additional discussions for limitations, future work and broader impacts;
- Section D: License and consent for public resources.

A. Detailed Implementation Specifications

A.1. Point-MAE-based Fine-tuning

We leverage the Point-MAE [4] pretrained model to perform object classification experiments on real-world data (ScanObjectNN [6]) and synthetic data (ModelNet40 [7]). The training settings are described on the left of Tab. A4, following the pioneering work [4, 12]. All experiments are conducted on a single GeForce RTX 3090 GPU.

A.2. ReCon-based Fine-tuning

Similarly, more recent Recon [5] pre-trained model is used for few-shot learning experiments on ModelNet40 [7] and part segmentation on ShapeNetPart [8]. The training settings are detailed in the right half of Tab. A4, following the general configurations [4, 11], with all training conducted on a single GPU.

B. Extended Experimental Results

B.1. More Ablation Studies

Here we provide additional ablation experiments, adhering to the same settings described in the main paper. Specifically, we utilize the Point-MAE [4] pre-trained model and report fine-tuning results on the most challenging variant, PB-T50-RS, of ScanObjectNN [6].

Ablation on multi-scale token selection. We first conduct ablation experiments on the number of center points and neighboring points in the multi-scale token selection process. As shown in Tab. A1, these parameters influence the amount of information encoded in the tokens, which subsequently affects token selection and fine-tuning performance. For the two scales, we set these values at $(128, 32)$ and $(64, 64)$, respectively.

Ablation on the loss weight for mask learning. We also perform an ablation study on the loss weight λ of $\mathcal{L}_{\text{mask}}$, which balances the learning between Mask Predictor and downstream tasks. As illustrated in Tab. A2, the incorporation of mask loss $\mathcal{L}_{\text{mask}}$ improves the quality of the selected tokens, leading to improved classification accuracy. However, an excessively large λ for $\mathcal{L}_{\text{mask}}$ can disrupt the learning of the classification task, causing a slight performance drop. To achieve the best accuracy, we set λ to 0.004.

Table A1. Ablation study on the number of center points and neighbor points in multi-scale token selection.

Scale 1	Scale 2	PB-T50-RS
$(256, 16)$	$(64, 64)$	84.46
$(256, 16)$	$(128, 32)$	84.66
$(128, 16)$	$(64, 32)$	85.08
$(128, 48)$	$(64, 80)$	83.90
$(128, 32)$	$(64, 64)$	**85.53**

Table A2. Ablation on the loss weight for Mask Predictor learning.

Weight (λ)	0	0.002	0.004	0.006	0.008
PB-T50-RS	84.56	84.90	**85.53**	84.91	85.01

Table A3. Ablation study on the injected blocks for PointLoRA.

Blocks	TP	Ratio	PB_T50_RS
$1 \rightarrow 3$	0.61 M	2.72%	83.83
$1 \rightarrow 6$	0.66 M	2.96%	83.55
$1 \rightarrow 9$	0.72 M	3.19%	85.05
$4 \rightarrow 12$	0.72 M	3.19%	84.84
$8 \rightarrow 12$	0.64 M	2.88%	84.14
$1 \rightarrow 12$	0.77 M	3.43%	**85.53**

Ablation on the injected blocks for PointLoRA. Following DAPT [12], we also experimented with injecting the designed components into only a subset of point cloud transformer blocks ($L = 12$ in total) to further reduce the number of tunable parameters. As shown in Tab. A3, limiting injections to shallow or deeper blocks results in varying degrees of performance degradation. This could be attributed to the fact that different blocks in the pre-trained model capture critical information related to distinct aspects of the input point cloud. Consequently, we choose to integrate PointLoRA into the `qkv projection` and `FFN` layers of all blocks, leaving the investigation of block-specific configurations for future research.

B.2. Part Segmentation Visualization

We visualize the results of part segmentation obtained using the proposed approach, fine-tuned with the Recon [5] pre-trained model in ShapeNetPart [8]. As illustrated in Fig. A1 and Fig. A2, a total of eight representative categories are selected, with four viewpoints displayed for each category.

Table A4. Training settings for various downstream fine-tuning models and datasets used in our implementation.

Training Settings	Classification			Segmentation
	ScanObjectNN [6]	ModelNet40 [7]	ModelNet40 Few-shot [7]	ShapeNetPart [8]
Pre-trained Model	Point-MAE [4]	Point-MAE [4]	Recon [5]	Recon [5]
Optimizer	AdamW	AdamW	AdamW	AdamW
Learning rate	5×10^{-4}	5×10^{-4}	5×10^{-4}	2×10^{-4}
Weight decay	5×10^{-2}	5×10^{-2}	5×10^{-2}	5×10^{-2}
Learning rate scheduler	cosine	cosine	cosine	cosine
Training epochs	300	300	150	300
Warm-up epochs	10	10	10	10
Batch size	32	32	4	16
Drop path rate	0.3	0.1	0.3	0.1
Selected token number	32 & 8	32 & 8	32 & 8	32 & 8
Number of points	2048	1024	1024	2048
Number of point patches	128	64	64	128
Point patch size	32	32	32	32

Our method demonstrates promising segmentation performance across various categories while utilizing a minimal number of tunable parameters.

C. Additional Discussions

C.1. Explanatory Experiments and Discussions

Large model experiments and necessity for PEFT. The experiments in the main paper follow the *common settings*, validating our approach on small-scale models (22.1M) for fair comparison. This establishes a solid foundation for the extension to larger-scale models. We further fine-tune PointGPT-L [1, 3] (360.5M), the largest pre-trained model for object-level point clouds, using proposed PointLoRA on PB-T50-RS, As shown in Tab. A5, our method updates only 1.36% of parameters and outperforms full fine-tuning with significantly reduced storage space.

About the technical novelty of PointLoRA. First, we reveal the effectiveness of LoRA in point cloud and its connection to PointNet, which is *overlooked in previous research*. Second, adhering to the principle of *simplicity and effectiveness*, we design PointLoRA with multi-scale token selection that requires only minimal parameters to achieve SOTA performance. This simplicity and efficiency enable seamless extension to larger models and diverse scenarios.

Theoretical analysis of LoRA for point cloud. LoRA is well-suited for point clouds due to its alignment with the principles underlying point cloud architectures like PointNet. Both leverage efficient subspace representations: PointNet adopts *shared MLPs and pooling* to approximate *permutation-invariant* set functions, while LoRA reduces fine-tuning updates with *low-rank matrices*. This synergy allows LoRA to effectively adapt to the sparse, high-dimensional nature of point clouds to capture *global features* with minimal computational overhead.

Table A5. Explanatory experiments on large model with the proposed method.

Methods	Tunable Params.	Storage	PB-T50-RS
PointGPT-L (Full-FT)	360.5 M	4.0 GB	93.4
+PointLoRA	**4.9 M**	**< 60 MB**	**93.8**

C.2. Limitations and Future Work

While PointLoRA effectively reduces trainable parameters and achieves competitive performance across diverse tasks, it still has certain limitations. The effectiveness of fine-tuning heavily depends on the quality of pre-trained models, making it less adaptable to tasks involving domains significantly different from those used during pre-training. Additionally, the multi-scale token selection strategy is heuristically designed, and its performance may vary across various datasets and tasks. Furthermore, the scalability of our method to extremely large pre-trained models remains unexamined, partly due to the current absence of general large-scale models in 3D space. The variation in task-specific performance also highlights the need for more tailored solutions.

Future research could focus on developing more adaptive or learnable token selection mechanisms to enhance flexibility and robustness. Exploring task-conditioned fine-tuning strategies and hierarchical LoRA configurations may improve scalability and performance, particularly for larger models. Expanding the approach to handle multi-modal data, such as combining point clouds with images or text, presents another promising direction. Meanwhile, investigating domain-specific adaptation techniques could improve performance in scenarios with significant domain shifts from pre-training to downstream tasks.

C.3. Broader Impacts

The proposed approach facilitates parameter-efficient fine-tuning for pre-trained point cloud models, increasing acces-

sibility to advanced technologies in domains such as autonomous driving, robotics, and environmental monitoring. Its efficiency also contributes to reducing the environmental impact of deep learning by lowering energy consumption. However, the improved capabilities of point cloud modeling present risks, including potential misuse in privacy-invasive applications or the propagation of unintended biases in autonomous systems. To maximize its benefits while addressing these challenges, ethical deployment and responsible governance will be essential.

D. License and Consent Information

D.1. Public Datasets

We conducted all the experiments on the subsequent openly accessible datasets:

- ScanObjectNN [6][1] ..... MIT License
- ModelNet40 [7][2] ..... Other (specified in description)
- ShapeNetPart [8][3] ..... Other (specified in description)

D.2. Public Implementation

We compare and validate the effectiveness of the proposed method with the following publicly available pre-trained models and source codes:

- Point-MAE [4][4] ..... MIT License
- ReCon [5][5] ..... MIT License
- Point-BERT [9][6] ..... MIT License
- IDPT [10][7] ..... Other (specified in description)
- DAPT [12][8] ..... Apache License 2.0
- PPT [11][9] ..... MIT License
- LoRA [2][10] ..... MIT License

[1] https://hkust-vgd.github.io/scanobjectnn.
[2] https://modelnet.cs.princeton.edu.
[3] https://cs.stanford.edu/~ericyi/project_page/part_annotation.
[4] https://github.com/Pang-Yatian/Point-MAE.
[5] https://github.com/qizekun/ReCon.
[6] https://github.com/Julie-tang00/Point-BERT.
[7] https://github.com/zyh16143998882/ICCV23-IDPT.
[8] https://github.com/LMD0311/DAPT.
[9] https://github.com/zsc000722/PPT.
[10] https://github.com/microsoft/LoRA.

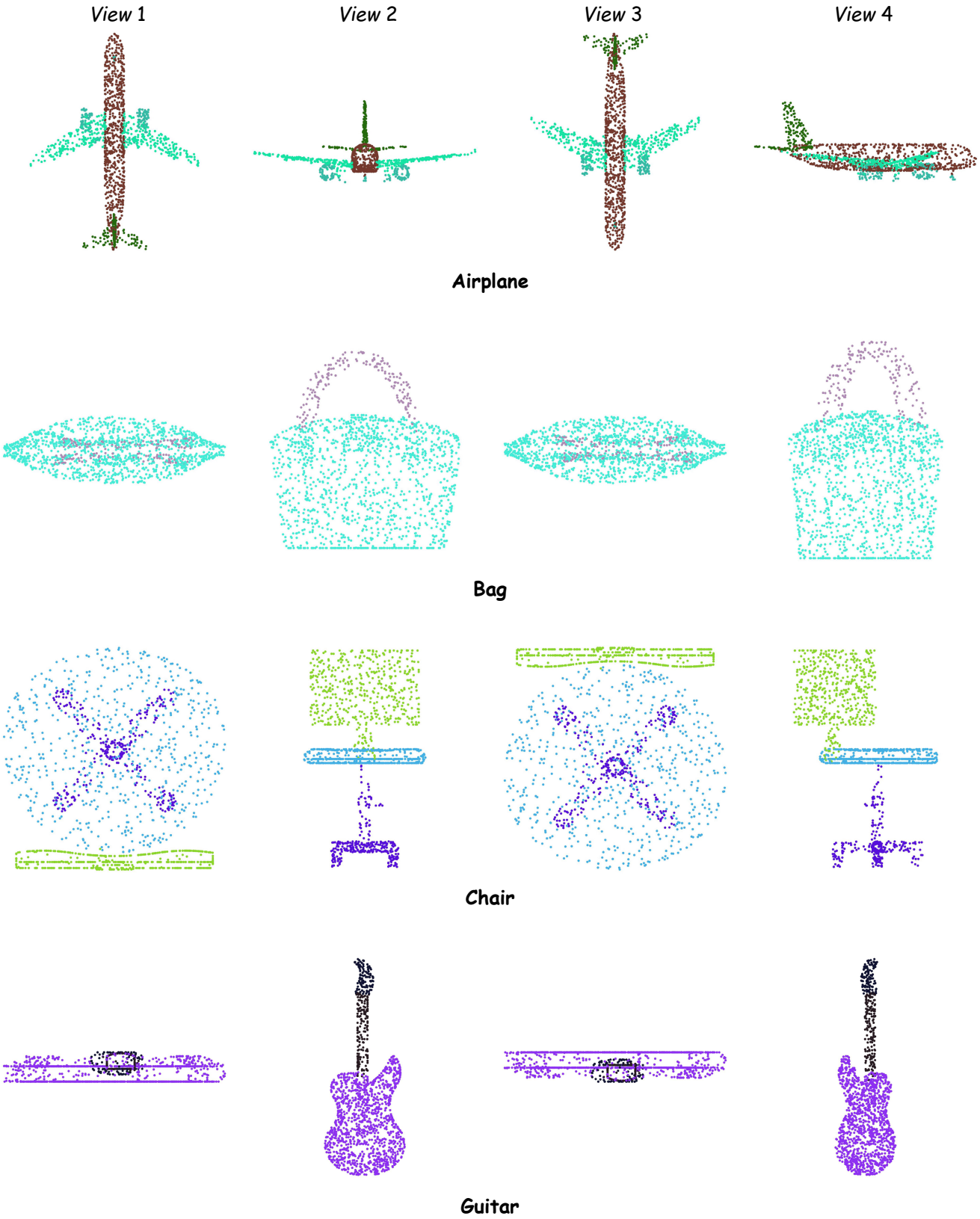

Figure A1. Visualization results for part segmentation on ShapeNetPart [8]. We present projected prediction images from PointLoRA across four different viewpoints, including “Airplane”, “Bag”, “Chair” and “Guitar”.

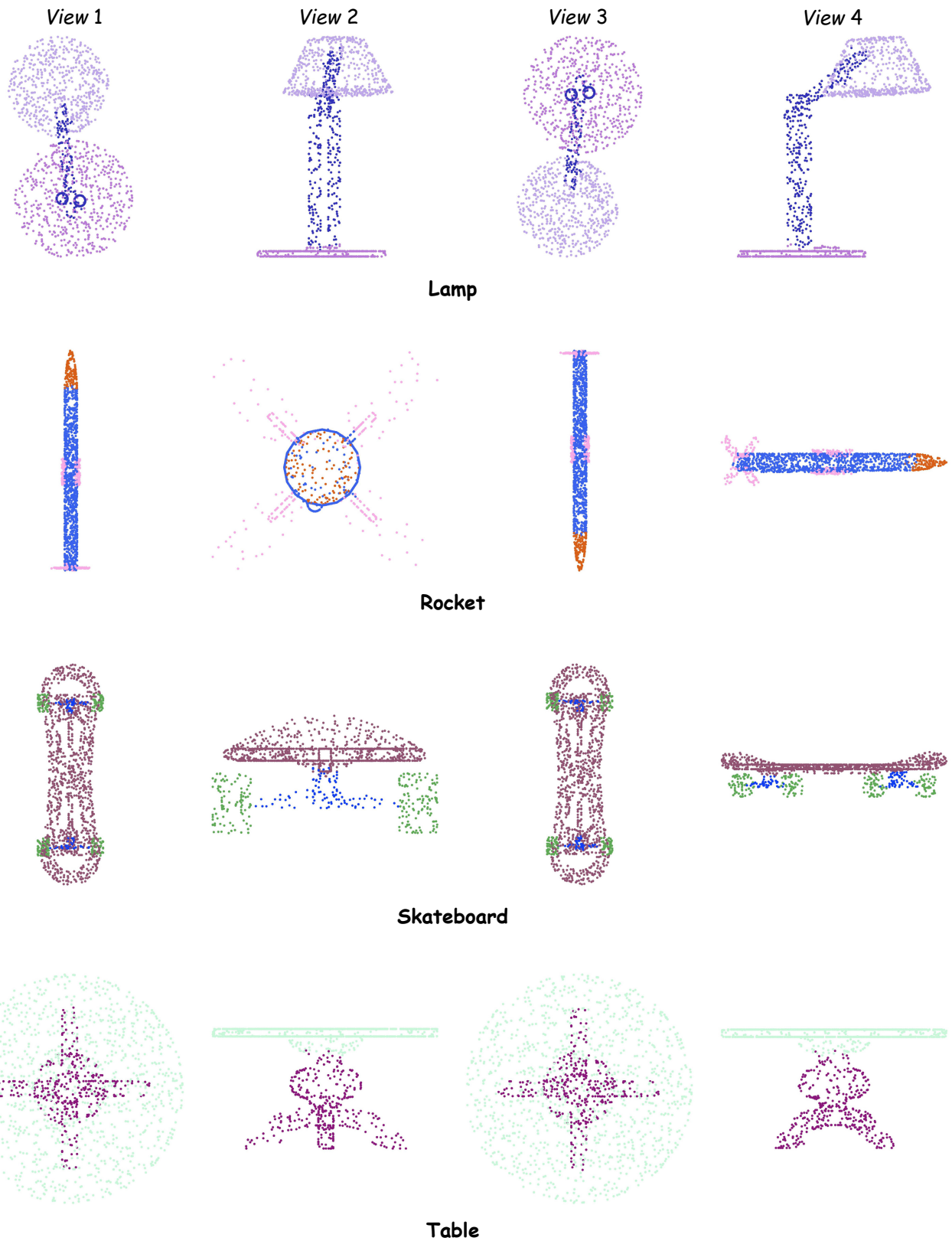

Figure A2. Visualization results for part segmentation on ShapeNetPart [8]. Projected prediction images from **PointLoRA** are shown across four different viewpoints, including the categories “Lamp”, “Rocket”, “Skateboard” and “Table”.

References

[1] Guangyan Chen, Meiling Wang, Yi Yang, Kai Yu, Li Yuan, and Yufeng Yue. Pointgpt: Auto-regressively generative pre-training from point clouds. In *Advances in Neural Information Processing Systems*, 2024. 2

[2] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021. 3

[3] Dingkang Liang, Tianrui Feng, Xin Zhou, Yumeng Zhang, Zhikang Zou, and Xiang Bai. Parameter-efficient fine-tuning in spectral domain for point cloud learning. *arXiv preprint arXiv:2410.08114*, 2024. 2

[4] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In *European Conference on Computer Vision*, 2022. 1, 2, 3

[5] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*. PMLR, 2023. 1, 2, 3

[6] 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 *IEEE/CVF International Conference on Computer Vision*, 2019. 1, 2, 3

[7] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015. 1, 2, 3

[8] Li Yi, Vladimir G Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. A scalable active framework for region annotation in 3d shape collections. *ACM Transactions on Graphics*, 35(6):1–12, 2016. 1, 2, 3, 4, 5

[9] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 3

[10] Yaohua Zha, Jinpeng Wang, Tao Dai, Bin Chen, Zhi Wang, and Shu-Tao Xia. Instance-aware dynamic prompt tuning for pre-trained point cloud models. In *IEEE/CVF International Conference on Computer Vision*, 2023. 3

[11] Shaochen Zhang, Zekun Qi, Runpei Dong, Xiuxiu Bai, and Xing Wei. Positional prompt tuning for efficient 3d representation learning. *arXiv preprint arXiv:2408.11567*, 2024. 1, 3

[12] Xin Zhou, Dingkang Liang, Wei Xu, Xingkui Zhu, Yihan Xu, Zhikang Zou, and Xiang Bai. Dynamic adapter meets prompt tuning: Parameter-efficient transfer learning for point cloud analysis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14707–14717, 2024. 1, 3