Supplementary Material for Invariant Training 2D-3D Joint Hard Samples for Few-Shot Point Cloud Recognition

Xuanyu Yi[1], **Jiajun Deng**[2], **Qianru Sun**[5], **Xian-Sheng Hua**[3],
Joo-Hwee Lim[4], **Hanwang Zhang**[1]
[1]Nanyang Technological University, [2]The University of Sydney
[3]Terminus Group, [4]Institute for Infocomm Research, [5]Singapore Management University
xuanyu001@e.ntu.edu.sg, jiajun.deng@sydney.edu.au, xshua@outlook.com,
joohwee@i2r.a-star.edu.sg, hanwangzhang@ntu.edu.sg,qianrusun@smu.edu.sg

The Appendix is organized as follows:

- **Section A:** provides more details about our training pipeline. Specifically, we detailed the implementation of 2D renderer, CLIP linear adapter as well as the modality fusion and invariant risk minimization (IRM).
- **Section B:** gives further discussion on joint hard sample from probability theory and Venn graph.
- **Section C:** shows more experiment results and ablation studies, *e.g.*, augmentations, OHEM strategy and parameter sensitivity analysis.

A. Implementation Details

As for 2D branch, We leverage the pretrained 2D knowledge for better point cloud analysis from two perspectives. 1) Beyond directly conducting CLIP visual encoder to projected depth maps in previous settings [2,23], with the guidance of the frozen pretrained weights, the inputs of this branch can extensively bridge the modality gap between regular ones in 2D pretrained datasets and those transformed from point clouds through differentiable renderer. 2) Since fine-tuning the whole CLIP visual backbone would easily result in over-fitting under few-shot settings, we followed the strategy of PointCLIP [23], freezing CLIP's visual and textual encoders and optimize the lightweight bottleneck adapter with the cross-entropy loss.

Differentiable Renderer. The renderer R, grounded in alpha compositing [17], is tasked with generating a rasterized object interpretation, utilizing the provided camera parameters. Learnable parameters $r = \{\rho, \theta, \phi\}$ are specifically harnessed to illustrate the camera's pose and position, with ρ representing the distance to the object rendered, θ embodying the azimuth, and ϕ denoting the elevation. We employ a differentiable renderer to optimize the generation of pseudo images for improved recognition, which parameter is established through the confidence correlation between ground-truth label-generated prompt and the zero-shot performance of CLIP on downstream training datasets. For ModelNet40, Toys4K, and ShapeNet-Core, we utilize a differentiable mesh renderer. We maintain a fixed light source directed towards the object's center, applying normal vectors for coloration, or default to white when these are not accessible. As for ScanObjectNN, we deploy a differentiable point cloud renderer with 2048 points. This serves as a lighter alternative to mesh rendering in instances where CAD is unavailable or the mesh contains a significant number of obstructive faces [3].

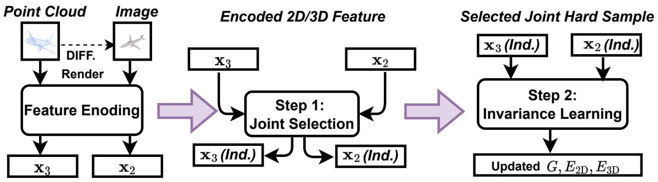

Figure 1. The simplified training framework.

Symbol	Definition	Symbol	Definition
$\mathbf{x}_2$ / $\mathbf{x}_3$	Encoded 2D/3D features	y	Class label
$\boldsymbol{z}_e$	Masked feature in each modality	r	Threshold parameter
G	Gate function	λ	IRM penalty weight
θ	Dummy classifier, calculating gradients	E_{2D}/E_{3D}	2D/3D Feature encoder

Table 1. Notation Table

Multi-view Feature Encoding. We simplify the inter-view adapter for PointCLIP to further encode the N-view image feature F_I with a proposed Multi-View adapter, which could capture the global and weighted view-wise feature simultaneously. With such simplification, we reduce the learnable parameters and avoid *post-search*. Specifically, given the N-view grid features $\mathbf{F}_v^D$, we first concatenate along the channel to obtain the global feature, then an aggregation function $A(\cdot)$ is calculated based on the pairwise

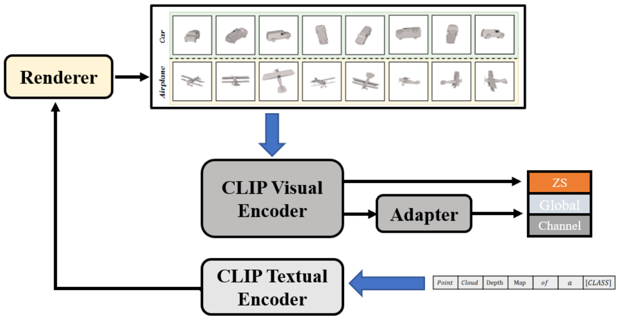

Figure 2. Detailed structure of 2D branch.

affinity matrix $T \in \mathbb{R}^{N\times N}$ with feature cosine similarity. By aggregating the view-level vectors via $A(\cdot)$, we integrates shape information by reweighted view-wise feature representation. Finally, the encoding process can be formulated as:

$$\mathcal{F}_{\text{Global}} = f_2\left(\text{ReLU}\left(f_1\left(\text{concat}\left(\{\mathbf{F}_v^D\}_{v=1}^N\right)\right)\right)\right) \tag{1}$$

$$\mathcal{F}_{\text{View}} = \text{ReLU}\left(A\left(\text{concat}\left(\{\mathbf{F}_v^D\}_{v=1}^N\right)\right)\right) \tag{2}$$

$$\mathcal{F}_I = (1-\delta)\mathcal{F}_{\text{Global}} + \delta\mathcal{F}_{\text{View}} \tag{3}$$

, where $A(\cdot)$ is the reweighting function, f_1 and f_2 are two-layer MLPs, and δ is mix-up combination coefficient.

Modality Fusion. As an *optional* enhancement, we introduce a bidirectional attention mechanism, bridging the inherent strengths of 2D and 3D modalities to birth an intermediate ***2.5D*** representation. This meticulously crafted modality proficiently harnesses localized nuances from both 2D images and 3D point clouds, imbibing the complementary details unique to each domain.

In detailed, our architecture is enriched with a bidirectional cross-attention layer [13]. In the forward direction, point cloud features emerge as the query tensor, with the image features serving the dual roles of key and value tensors. In stark contrast, the reverse direction sees the image features donning the mantle of the query, whilst the point cloud features settle as both the key and value tensors. This duality in approach guarantees a harmonious balance in the weighing of features, rooted in mutual affinities. The culmination is a fusion that harmoniously encapsulates the distinctiveness of both modalities:

$$Q_{\mathbf{x}_3} = \mathbf{x}_3 W_Q, \qquad K_{\mathbf{x}_2} = \mathbf{x}_2 W_K, \tag{4}$$

$$V_{\mathbf{x}_2} = \mathbf{x}_2 W_V, \qquad X_{\text{fused}} = \text{softmax}(Q_{\mathbf{x}_3} K_{\mathbf{x}_2}^T) V_{\mathbf{x}_2}, \tag{5}$$

$$Q_{\mathbf{x}_2} = \mathbf{x}_2 W_{Q'}, \qquad K_{\mathbf{x}_3} = \mathbf{x}_3 W_{K'}, \tag{6}$$

$$V_{\mathbf{x}_3} = \mathbf{x}_3 W_{V'}, \qquad X_{\text{fused-I}} = \text{softmax}(Q_{\mathbf{x}_2} K_{\mathbf{x}_3}^T) V_{\mathbf{x}_3}. \tag{7}$$

The subsequent melding of X_{fused} and $X_{\text{fused-I}}$ births an enriched feature set, $X_{\text{bi-fused}}$, echoing the attributes of both 2D images and 3D point clouds. Note that such fusion-based modality is only served as a optional regularization term for calculating modality-wise IRM loss. Therefore, there is no additional classification head, and thus not leveraged during inference for the ***2.5D*** modality environment.

Advanced IRM. In the manuscript, we introduced the modality-wise IRM to capture the common feature for better alignment. For its practical implementation [1], we transitioned to REx [4], an optimized version especially adept under co-variate shifts. The MM(Min-Max)-REx is given by:

$$\begin{aligned}\text{MM-REx}(\theta) &= \max_{\substack{\sum_{e=1}^m \lambda_e = 1\\ \lambda_e \geq \lambda_{\text{min}}}} \sum_{e=1}^{m} \lambda_e \mathcal{L}_e(\theta) \\ &= (1 - m\lambda_{\text{min}}) \max_e \mathcal{L}_e(\theta) + \lambda_{\text{min}} \sum_{e=1}^{m} \mathcal{L}_e(\theta).\end{aligned} \tag{8}$$

With goals akin to IRM, REx ensures invariance across environments in a more efficient and stable manner. We also leveraged the V-REx variant with additional 2.5D modality:

$$\text{RV-REx}(\theta) = \beta \text{Var}(\{\mathcal{L}_1(\theta), ..., \mathcal{L}_m(\theta)\}) + \sum_{e=1}^{m} \mathcal{L}_e(\theta). \tag{9}$$

Here, β regulates between reducing average risk and ensuring risk consistency. Specifically, with $\beta = 0$ it aligns with ERM, while a higher β emphasizes risk equalization. Specifically, our modality-wise IRM differs from traditional ones as we utilize contrastive objectives $\mathcal{L}_e$, which show high efficiency for learning discriminative features. After filtering out conflicting features, we finally regularize $E_{3\text{D}}$, $E_{2\text{D}}$ in the collaborative feature space $G(\mathbf{x}_e)$, aligning them with the cross-modality NT-Xent loss $\mathcal{L}_{align}$.

Object Retrieval. we leverage LFDA reduction to project and fuse the encoded feature (w/o the last layer for 3D branch) as the signature to describe a shape, which is further compared through Kd-Tree searching. Figure 3 shows some qualitative retrieval examples.

B. Discussion On Joint Hard Sample

Probability Theory Perspective. We still refer to the Bayesian Decomposition introduced in the manuscript:

$$p(y = c \mid z_c, z_d) = p(y = c \mid z_c) \cdot \frac{\overbrace{p\left(z_d \mid y = c, z_c\right)}^{\text{modality bias}}}{p\left(z_d \mid z_c\right)}, \tag{10}$$

Since the previous modality bias in [12, 20] is revealed in the same model, it hurts the OOD generalization and decreases the performance under distribution shifts. However,

[1] We discards the dummy classifier w and calculate Min-Max or variance of risks as the penalty term of IRM.

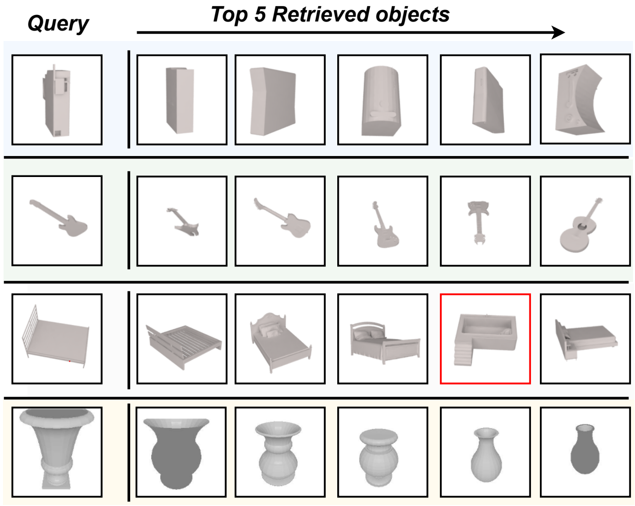

Figure 3. Qualitative Examples for 3D Shape Retrieval on ModelNet40: (*left*): Query objects from the test set. (*right*): Top 5 matches +for each query, with mistakes highlighted in red.

the modality bias in the proposed 2D-3D ensemble module is multi-modal and come from different networks, *i.e.*, the second term in Eq. (10) is only variant across models. In fact, for a specific modality, some descriptive feature z_d is even beneficial. For example, when $p_{2D}\left(z_d \mid y=c, z_c\right)$ is significantly larger than $p_{2D}\left(z_d \mid y \neq c, z_c\right)$ for 2D images from class $y=c$, such an modality-specific z_d is good for 2D classification even if it's not hold in 3D point clouds, *e.g.*, fine-grained texture in 2D images disappear in 3D representation. Therefore, considering ensemble, instead of removing all the modality bias as [12, 20] did, we need to find (reweight) joint hard samples and only removing the conflict while keeping some beneficial z_d for each modalities.

A Venn Diagram Perspective. We maintain the assumption that, under few-shot fine-tuning, the primary improvement from the ensemble paradigm arises from the reduction of conflicting predictions, rather than from enhancements of a specific modality [2]. Figure 5 illustrates that the essence of an effective 2D-3D ensemble lies in minimizing the high confidence assigned to incorrect labels. In this pursuit, our invariant training strategy is anchored on ensuring the consistency of different (2D-3D) representations, serving to curtail biases stemming from conflicting modalities. An alternative method involves calibrating [5] the 2D/3D logits within each modality, balancing between confidence ***avg.*** and accuracy ***avg.***. However, this technique demands a robust validation set, which is a rarity in the current 3D standard datasets. Looking ahead, we intend to juxtapose

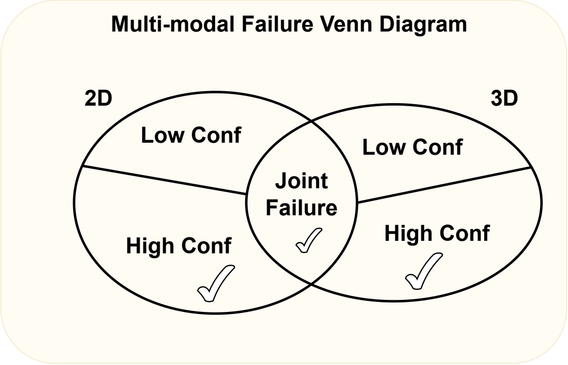

Figure 4. The diagram of multi-modal failure Venn, where **Conf** denotes model's confidence on certain category for a test sample.

and adapt our invariant learning approach with the logit calibration model, especially within the context of the latest large-scale 3D datasets [18, 22].

C. More Experiment

C.1. Additional Results on Many-shot ModelNet40

In the main manuscript, we followed [11] and presented its performance for comparison. It is interesting to discuss the results under the regular settings as [2], thus we compared INVJOINT with previous state-of-the-art methods under the full training ModelNet40. Note that in this setting, we use point cloud rendering instead of mesh rendering for data pre-processing. From Table 5, our method shows comparable results with state-of-the-art methods on sufficient data. This further supports that our proposed framework can adapt to many-shot settings.

Table 2. Full Training on ModelNet40 with regular setting.

ID	Pretrain	Methods	Full-shot
1	N/A	PointNet++ [9]	90.7
2		PointMLP [6]	**94.1**
3		PointNeXt [10]	94.0
4		CurveNet [7]	93.9
5	3D	Dgcnn-ocCo [14]	93.0
6		Ponit-BERT [21]	93.2
7		Point-MAE [8]	93.8
8		CrossPoint [1]	91.2
9	2D	P2P [16]	94.0
10		PonitCLIP [23]	90.9
11		INVJoint	93.9

C.2. Additional Results on Data Efficient Learning

We follow [19], evaluating INVJOINT under limited data scenario. From Table 3, we could find that INVJOINT

[2] In other words, empirical findings suggest that joint 2D-3D training, without resorting to either contrastive or distillation methods, does not enhance the performance of a particular modality when compared to individual training in **few-shot settings**.

Table 3. Data efficient learning on ModelNet40.

Data percentage	w/ PointCMT [19]	InvJoint
2 %	75.2	79.4 (+ 4.2)
5%	83.5	85.4 (+ 1.9)
10 %	87.9	89.7 (+ 1.8)
20 %	89.3	91.3 (+ 2.0)

Table 4. Different Augmentation Strategies.

ID	Augmentation Strategy
A	**Random translation** & **Random Scaling**
B	**Jittering** & **Random Rotation Along Y-axis**
C	A & **RandomInputDropout** & **Random Rotation**
D	B & **RotatePerturbation**
E	A & B & **RandomInputDropout**

consistently retains robust performance and out-performs PointCMT in all cases.

C.3. Additional Results on Augmentation Strategies

As illustrated in [2], auxiliary factors like different evaluation schemes, data augmentation strategies, and loss functions, which are independent of the model architecture, make a large difference in point cloud recognition performance. In order to test the robustness of InvJoint, we conduct different type of augmentations strategies with DGCNN [15] as our 3D branch encoder, and report the results of individual 3D branch as well as the joint prediction of InvJoint. In detail, Table 4 summarize the compared 5 types of augmentation strategies. We could find from Figure 5 that though the choice of augmentation strategy greatly influences the performance of 3D branch in few-shot settings, the joint prediction maintains encouraging and stable enhancement thanks to the collaborative joint training.

C.4. Parameter Sensitivity Analysis

The following sensitivity analyses were conducted in ModelNet40 with 16-shot settings. (1) We observed the optimal λ in Eq.5. The Top-1 Accuracy is *87.32 / 87.90 / 88.94 / 86.23 %* (λ = 0.1 / 1 / 5 / 10). (2) Keeping λ= 5, the Top-1 Accuracy is *85.47 / 88.94 / 88.65 / 84.10 %* with an added weight ratios (α = 0.1 / 1 / 5 / 10) between $\mathcal{L}_{CE}$ and $\mathcal{L}_{\text{align}}$.

C.5. Additional Results on HEM methods in Step 1

The GMM [3] module is only used for selecting hard samples in each modality and is ***NOT** our technical contribution*. Therefore, we further replace it with other loss discrimination and OHEM (online hard example mining) methods for ablation. Specifically in Table 5, if we replace our GMM module with Beta Mixture Model (BMM) in ModelNet40 with different few-shot settings, InvJoint can still achieve comparative results and largely outperform other methods in different few-shot settings.

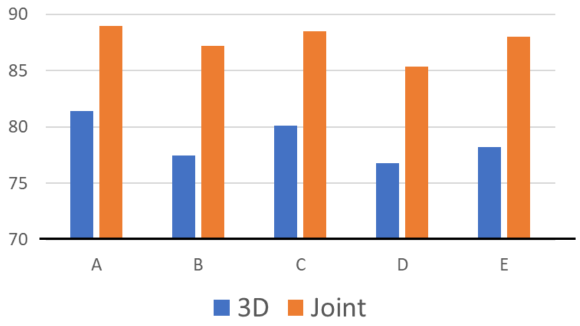

Figure 5. Performances with different augmentation strategy on 16-shot ModelNet40. Accuracy of 3D branch (*Blue*) and InvJoint (*Orange*) are reported.

Table 5. Few-shot performance on ModelNet40 with different OHEM methods.

Method	OHEM	1-shot	2-shot	4-shot	8-shot	16-shot
PointCLIP [23]	-	52.96	66.73	74.47	80.96	85.45
Crosspoint [1]	-	48.24	59.95	64.25	75.75	79.70
InvJoint	GMM	68.85	70.24	**78.95**	**82.85**	**88.94**
InvJoint	BMM	**70.12**	**71.30**	76.08	82.63	87.15

Table 6. The enhancement ability of 2D branch in InvJoint.

Methods	Before Fuse	After Fuse	Gain
PointNet++ [9]	80.2	85.1	4.9
CrossPoint [1]	79.7	83.2	3.5
DGCNN [15]	81.4	89.0	7.6
CurveNet [7]	81.8	87.3	5.5

C.6. Additional Results on 3D Backbones

To verify the complementarity and the coordination of 3D and 2D, we further aggregate the fine-tuned 16-shot 2D branch on ModelNet40 with different 16-shot 3D backbones, including PointNet++ [9], CrossPoint [1], DGCNN [15], CurveNet [7]. Table 6 illustrates that the enhancement ability of 2D branch in InvJoint with the alleviation of modality conflict.

[3] GMM-based loss discrimination has been widely adopted to identify outliers in de-noising and hard example mining because of its efficiency and high compatibility.

References

[1] Mohamed Afham, Isuru Dissanayake, Dinithi Dissanayake, Amaya Dharmasiri, Kanchana Thilakarathna, and Ranga Ro-

drigo. Crosspoint: Self-supervised cross-modal contrastive learning for 3d point cloud understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9902–9912, 2022. 3, 4

[2] 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. 1, 3, 4

[3] 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. 1

[4] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021. 2

[5] Ananya Kumar, Tengyu Ma, Percy Liang, and Aditi Raghunathan. Calibrated ensembles can mitigate accuracy tradeoffs under distribution shift. In *Uncertainty in Artificial Intelligence*, pages 1041–1051. PMLR, 2022. 3

[6] 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. 3

[7] AAM Muzahid, Wanggen Wan, Ferdous Sohel, Lianyao Wu, and Li Hou. Curvenet: Curvature-based multitask learning deep networks for 3d object recognition. *IEEE/CAA Journal of Automatica Sinica*, 8(6):1177–1187, 2020. 3, 4

[8] 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

[9] 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. 3, 4

[10] 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. 3

[11] Jiachen Sun, Qingzhao Zhang, Bhavya Kailkhura, Zhiding Yu, Chaowei Xiao, and Z Morley Mao. Benchmarking robustness of 3d point cloud recognition against common corruptions. *arXiv preprint arXiv:2201.12296*, 2022. 3

[12] Kaihua Tang, Mingyuan Tao, Jiaxin Qi, Zhenguang Liu, and Hanwang Zhang. Invariant feature learning for generalized long-tailed classification. In *ECCV*, 2022. 2, 3

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[14] Hanchen Wang, Qi Liu, Xiangyu Yue, Joan Lasenby, and Matt J Kusner. Unsupervised point cloud pre-training via occlusion completion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9782–9792, 2021. 3

[15] 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. 4

[16] Ziyi Wang, Xumin Yu, Yongming Rao, Jie Zhou, and Jiwen Lu. P2p: Tuning pre-trained image models for point cloud analysis with point-to-pixel prompting. *arXiv preprint arXiv:2208.02812*, 2022. 3

[17] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7467–7477, 2020. 1

[18] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 803–814, 2023. 3

[19] Xu Yan, Heshen Zhan, Chaoda Zheng, Jiantao Gao, Ruimao Zhang, Shuguang Cui, and Zhen Li. Let images give you more: Point cloud cross-modal training for shape analysis. *arXiv preprint arXiv:2210.04208*, 2022. 3, 4

[20] Xuanyu Yi, Kaihua Tang, Xian-Sheng Hua, Joo-Hwee Lim, and Hanwang Zhang. Identifying hard noise in long-tailed sample distribution. In *ECCV*, 2022. 2, 3

[21] 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. 3

[22] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9150–9161, 2023. 3

[23] 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. 1, 3, 4