Take-A-Photo: 3D-to-2D Generative Pre-training of Point Cloud Models
Supplemental Material

Ziyi Wang[1,2*] Xumin Yu[1,2*] Yongming Rao[1,2] Jie Zhou[1,2] Jiwen Lu[1,2†]
[1]Department of Automation, Tsinghua University [2]BNRist
{wziyi22, yuxm20}@mails.tsinghua.edu.cn;
raoyongming95@gmail.com; {lujiwen, jzhou}@tsinghua.edu.cn

A. Ablation Studies

In this section, we conduct more ablation studies on hyperparameter choices of the proposed 3D-to-2D generative pre-training, discussing more thoroughly the insights into architectural design and objective function design. We implement PointMLP [3] as the 3D backbone model and conduct these ablation experiments on the hardest PB-T50-RS variant of the ScanObjectNN [6] dataset. We report the classification accuracy of the fine-tuning results.

A.1. Cross-Attention Hyperparameters

In Table 1a, we display the results of ablation studies on the number of layers and feature channels of the cross-attention layers in our proposed Photograph module. From the quantitative results, we can conclude that 2 layers with 128 channels is the best hyperparameter group for cross-attention layers. When we implement a shallow layer setting (2 layers in line 1 and 4 layers in line 2), lower feature channels (128 dims) achieves better performance. On the contrary, when we implement a deeper layer setting (6 layers in line 3 and 8 layers in line 4), relatively higher feature channels (256 dims) is the best choice. Additionally, if we use 1024 dims as the feature channels in cross-attention layers, which is the same as the channels of output features from the 3D backbone model, the pre-training stage totally collapses and the fine-tuning results are much lower than models of 128 dims and 256 dims, no matter how much layers are implemented. This result indicates that a bottleneck design in our proposed photograph module is essential for the successful pre-training of the proposed 3D-to-2D generation.

The overall trend is that a lightweight architectural design of the cross-attention layers is better than a heavy module design. This may be because we completely drop the photograph module and only keep the 3D backbone in the fine-tuning stage. Therefore, a lightweight photograph module in the pre-training stage will encourage the 3D backbone to exploit more representation ability and avoid information loss in the fine-tuning stage to the best extent. On the contrary, if we implement a heavy photograph module with deep cross-attention layers and high feature dimensions, the photograph module will dominate the generation process and the importance of the 3D backbone will be neglected. What's worse, in the fine-tuning stage, the rich geometry information in the heavy photograph module is totally dropped out and no longer helpful for downstream tasks.

Table 1: **Ablation Studies on Hyperparameters.** We implement PoinMLP [3] as the 3D backbone model and conduct experiments on the hardest PB-T50-RS variant of ScanObjectNN [6] dataset.

(a) Cross-Attention Hyperparameters.

LayerNum \ Channels	128 Dims	256 Dims	1024 Dims
2 Layers	**89.1**	87.9	86.3
4 Layers	88.7	88.0	85.9
6 Layers	88.3	88.5	85.3
8 Layers	87.7	88.1	85.8

(b) Loss Weight Hyperparameters.

Model	G_1	G_2	G_3	G_4	G_5	G_6	H_1	H_2
w^{fg}	2	5	10	20	30	50	0	20
w^{bg}	1	1	1	1	1	1	0	1
w^{feat}	0	0	0	0	0	0	2	2
Acc. (%)	87.1	87.2	88.0	**88.5**	88.0	86.8	86.3	87.8

A.2. Objective Function

In this subsection, we discuss the objective function design of our proposed 3D-to-2D generative pre-training. In our main paper, we implement pixel-level supervision with MSE loss between generative view images I_{gen} and ground truth images I_{gt}:

$$\mathcal{L}_{\text{pix}}(I_{\text{gen}}, I_{\text{gt}}) = w^{\text{fg}}\mathcal{D}(I^{\text{fg}}_{\text{gen}}, I^{\text{fg}}_{\text{gt}}) + w^{\text{bg}}\mathcal{D}(I^{\text{bg}}_{\text{gen}}, I^{\text{bg}}_{\text{gt}}) \quad (1)$$

where fg denotes foreground region, bg denotes background region and $\mathcal{D}$ is the MSE distance. However, in 2D genera-

Figure 1: Visualization of the outputs from the 3D-to-2D generative pre-training. The first line shows the generated view images from the model. The second line shows the ground truth images for reference.

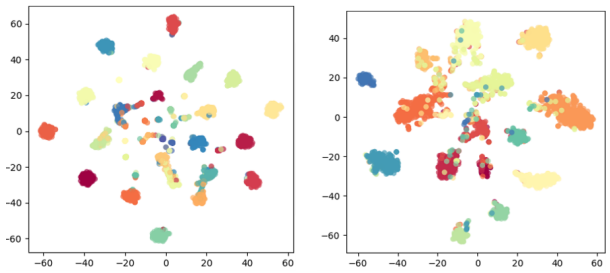
tion, perceptual loss [2] is of equal importance with pixel-wise loss. While pixel-wise MSE loss focuses on low-level similarities, perceptual loss measures high-level semantic differences between feature representations of the images computed by the pre-trained loss network. Technically, perceptual loss makes use of a loss network ϕ pre-trained for image classification, which is typically a 16-layer VGG [5] network pre-trained on the ImageNet [4] dataset. If we denote $\phi_j(x)$ as the feature map with size $c_j \times h_j \times w_j$ of the jth layer of the network ϕ, then the perceptual loss is defined as the Euclidean distance:

$$\mathcal{L}_{\text{feat}}(I_{\text{gen}}, I_{\text{gt}}) = \frac{1}{N} \sum_j \frac{1}{c_j h_j w_j} \|\phi_j(I_{\text{gen}}) - \phi_j(I_{\text{gt}})\|_2^2 \quad (2)$$

where N is the number of total layers of the VGG network and $1 \leq j \leq N$. If we combine the pixel-wise loss $\mathcal{L}_{\text{pix}}$ with the perceptual loss $\mathcal{L}_{\text{feat}}$, then the final objective function of the proposed 3D-to-2D generation is:

$$\mathcal{L} = \mathcal{L}_{\text{pix}} + w^{\text{feat}} \mathcal{L}_{\text{feat}} \quad (3)$$

In Table 1b, we conduct ablations on loss weight of foreground pixel-wise loss w^{fg}, background pixel-wise loss w^{bg} and perceptual loss w^{feat}. In Model G_1 to G_6, we only implement pixel-wise loss. In Model H_1, we only implement perceptual loss. In Model H_2, we combine pixel-wise loss with perceptual loss. From the ablation results, we can conclude that $w^{\text{fg}} : w^{\text{bg}} = 20 : 1$ is the best hyperparameter choice for pixel-wise loss. However, the perceptual loss is not effective when we compare Model G_4, Model H_1 and Model H_2. This is mainly due to the reason that the rendered view image of synthetic ShapeNet [1] dataset is out of the distribution of the realistic ImageNet [4] that the loss model ϕ is pre-trained on. Therefore, the high-level semantic representation ability of ϕ on view images is relatively poor and cannot guide the optimization of the 3D-to-2D generation process. If the rendered images are more realistic with colors and background, then the perceptual loss is expected to help 3D-to-2D generative pre-training.



(a) ModelNet40. (b) ScanObjectNN.

Figure 2: Visualization of feature distributions in t-SNE representations. Best view in colors.

B. Visualization Results

B.1. Generated View Images

Figure 1 displays more visualization results of our generated view images from the 3D-to-2D generative pre-training process. We take ShapeNet [1] as the pre-training dataset and implement PointMLP [3] as the 3D backbone model. The first line shows the generated results from our model while the second line shows ground truth images for reference. The visualization results convey that our 3D-to-2D generative pre-training can successfully predict the shape and colors of the objects from specific projection views. There are also some unsatisfactory cases in the last three columns, where there are some vague details in our generated images. This is mainly due to the large downsample ratio ($\times 32$) in our model design.

B.2. Feature Distributions

Figure 2 shows feature distributions of ModelNet40 [7] and ScanObjectNN [6] datasets in t-SNE visualization. We choose PointMLP [3] as the 3D backbone and pre-train on ShapeNet [1] dataset. We can conclude that with our proposed 3D-to-2D pre-training, the 3D backbone model can extract discriminative features after fine-tuning on downstream classification datasets.

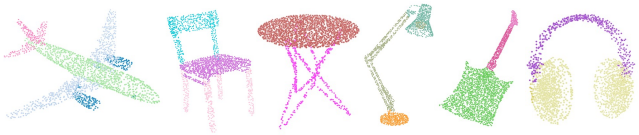


Figure 3: Illustration of part segmentation results.

B.3. Part Segmentation Visualizations

Figure 3 presents visualizations of part segmentation results on samples from the ShapeNetPart dataset. Each part is represented by a distinct color for clarity. These qualitative results serve as compelling visual evidence and provide a vivid illustration of the efficacy of our fine-tune model in achieving accurate part segmentation.

References

[1] 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

[2] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*. Springer, 2016. 2

[3] 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. In *ICLR*, 2022. 1, 2

[4] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2

[5] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[6] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Duc Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *ICCV*, 2019. 1, 2

[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 *CVPR*, 2015. 2