Supplementary Material for No Time to Train: Empowering Non-Parametric Networks for Few-shot 3D Scene Segmentation

Xiangyang Zhu[*1,3], Renrui Zhang[*†‡2,3], Bowei He[1], Ziyu Guo[2,3], Jiaming Liu[4]
Han Xiao[3], Chaoyou Fu[5], Hao Dong[4], Peng Gao[3]

[*] Equal contribution † Project leader ‡ Corresponding author

[1]City University of Hong Kong [2]The Chinese University of Hong Kong
[3]Shanghai AI Lab [4]Peking University [5]Tencent Youtu Lab
{xiangyzhu6-c, boweihe2-c}@my.cityu.edu.hk
{zhangrenrui, gaopeng}@pjlab.org.cn

In this material, we first investigate the related literature about this work. Then, we provide detailed descriptions of the experimental setup and hyperparameters, along with additional framework details. At last, we present more ablation studies and further analysis.

1. Related Work

Point Cloud Semantic Segmentation aims to assign each point with the correct category label within a predefined label space. The early PointNet [16], PointNet++ [17] establish the basic framework for learning-based 3D analysis. The follow-up works [3, 6, 22, 25–28] further improve the 3D representation and segmentation performance. Despite this, these methods are data-hungry and require extensive additional labeled data to be fine-tuned for unseen classes. To address this issue, a series of 3D few-shot learning methods have been proposed [8, 11, 13, 15, 21, 23, 24]. AttMPTI [29] extracts multiple prototypes from support-set features and predicts query labels in a transductive style. 2CBR [31] proposes cross-class rectification to alleviate the query-support domain gap. PAP-FZ3D [7] jointly trains few-shot and zero-shot semantic segmentation tasks. All of these methods adopt the meta-learning strategy including both pre-training and episodic training stages. In this work, we propose more efficient solutions for 3D few-shot semantic segmentation. We first devise a non-parametric encoder to discard the time-consuming pre-training stage. Based on this, a parameter-free model, Seg-NN, and a parametric variant, Seg-PN, are proposed, which achieve competitive performance with minimal resources and simplify the traditional meta-learning pipelines.

Positional Encoding (PE) projects a location vector into a high-dimensional embedding that can preserve spatial information and, at the same time, be learning-friendly for downstream algorithms [12]. Transformer [20] first utilizes PE to indicate the one-dimensional location of parallel input entries in a sequence, which is composed of trigonometric functions. Such trigonometric PE can encode both absolute and relative positions, and each of its dimensions corresponds to a predefined frequency and phase, which has also been employed for learning high-frequency functions [19] and improving 3D rendering in NeRF [14]. Some Transformers incorporate Gaussian random frequencies [10], and Mip-NeRF [2] reduces aliasing artifacts in rendering by suppressing high frequencies. In 3D domains, RobustPPE [30] adopts Gaussian random features for robust 3D classification. Point-NN [27] is the first non-parametric model for shape classification which leverages basic trigonometric PEs to encode point coordinates for shape classification. In this work, we extend Point-NN to scene segmentation and utilize trigonometric PE to encode positional and color information, where manually designed filters are used for scene-level geometry encoding.

2. Experimental Setup

Dataset Split S3DIS [1] consists of 272 room point clouds from three different buildings with distinct architectural styles and appearances. We exclude the background clutter class and focus on 12 explicit semantic classes. ScanNet [4] comprises 1,513 point cloud scans from 707 indoor scenes, with 20 explicit semantic categories provided for segmentation. Tab. 1 lists the class names in the S_0 and S_1 splits of S3DIS and ScanNet datasets.

	S_0	S_1
S3DIS	beam, board, bookcase, ceiling, chair, column	door, floor, sofa, table, wall, window
ScanNet	bathtub, bed, bookshelf, cabinet, chair, counter, curtain, desk, door, floor	other furniture, picture, refrigerator, show curtain, sink, sofa, table, toilet, wall, window

Table 1. **Seen and Unseen Classes Split** for S3DIS and ScanNet. We follow [29] to evenly assign categories to S_0 and S_1 splits.

Hyperparameters The Seg-NN encoder is frozen in all experiments. In the Seg-NN encoder, we sample 16 neighbor points with k-NN to build the neighborhood of the center point for both manipulation and upsampling layers. For Seg-PN, we first use a fully connected layer to refine the features extracted by the non-parametric encoder and then feed the refined features to the QUEST module. The fully connected layer consists of 2 linear projection operations and each linear projection is followed by a batch normalization [9] and a rectified linear activation [5] function. The detailed structure of the fully connected layer is: '(BN+ReLU) + (Linear+BN+ReLU) + (Linear+BN+ReLU)', where 'BN', 'ReLU', and 'Linear' represent batch normalization, rectified linear activation, and linear projection, respectively. We set the kernel size and stride of the local maximum pooling to 32 in the QUEST module. Since each point cloud contains $M = 2048$ points, the local maximum pooling operation outputs $M' = 64$ statistics for each point cloud.

Training Details The proposed Seg-NN and Seg-PN are implemented using PyTorch. Seg-PN is trained on a GForce A6000 GPU. The meta-training is performed directly on C_{train} split, using AdamW optimizer ($\beta_1 = 0.9, \beta_2 = 0.999$) to update the QUEST module of Seg-PN. The initial learning rate is set to 0.001 and halved every 7,000 iterations. In episodic training, each batch contains 1 episode, which includes a support set and a query set. The support set randomly selects N-way-K-shot point clouds and the query set randomly selects N unseen samples.

3. Additional Ablation Study

We conduct additional ablation experiments to reveal the roles of different detailed designs. By default, we still conduct experiments under 2-way-1-shot settings on the S_0 split of S3DIS dataset and use mIoU (%) as criteria to evaluate the results of both Seg-NN and Seg-PN.

3.1. Ablation for Seg-NN

In this section, we mainly investigate different hyperparameters and designs of PEs, embedding manipulation layers, and upsampling layers.

mIoU	**Coordinate**			**+ Color**		
	S_0	S_1	Avg	S_0	S_1	Avg
Seg-NN	48.81	49.04	48.93	**49.45**	**49.60**	**49.53**
Seg-PN	**67.47**	66.78	**67.13**	64.84	**67.98**	66.41

Table 2. **Ablation for Position and Color Information** under 2-way-1-shot settings on both S_0 and S_1 splits of S3DIS. We report Seg-NN and Seg-PN's results (%).

θ	10	20	30	40	60	80	100
Seg-NN	49.12	49.38	**49.45**	47.87	44.76	43.54	40.16
Seg-PN	61.85	62.34	**64.84**	63.08	62.56	63.68	64.05

Table 3. **Ablation for Parameter θ in PEs**.

d	5	8	10	15	20	24	30
Seg-NN	44.24	46.37	47.65	48.90	**49.45**	48.68	48.53
Seg-PN	63.37	63.49	**64.84**	63.76	63.35	63.45	63.81

Table 4. **Ablation on the Dimensionality of PEs**.

Role of Position and Color Information In Tab. 2, we exhibit more results to investigate the role of the position and color information. For Seg-NN, we observe that both position and color information are helpful for the segmentation, indicating that our model is capable of encoding geometries and integrating two types of information. However, in Seg-PN, employing colors hinders the prediction, which suggests that color information is not crucial for few-shot tasks and may lead to overfitting. This aligns with the observation of [18], which randomly abandons color information during training to reduce overfitting.

Hyperparameters of PEs **1) Parameter θ in PE.** In the initial PE, we utilize d log-linear spaced frequencies $\mathbf{u} = [u_1, ..., u_d]$ to project positions and colors into high-dimensional encodings, where $u_i = \theta^{i/d}$ with a base number θ. In Tab. 3, we explore the impact of θ on Seg-NN and Seg-PN, where $\theta \in \{10, 20, 30, 40, 60, 80, 100\}$. From the table, we observe that Seg-NN is more sensitive to θ, while Seg-PN exhibits higher tolerance. In addition, Seg-NN prefers low θ values, and a larger θ will cause significant performance degradation. **2) Dimensionality of PEs.** We then examine the effect of the dimensionality of PEs. We sample d frequencies to construct the PE. Tab. 4 presents the results with different frequency numbers d. We explore $d \in \{5, 8, 10, 15, 20, 24, 30\}$ and the corresponding dimensionality of PEs are $6d \in \{30, 48, 60, 90, 120, 144, 180\}$. We observe that $d = 20$ and 10 are the best choices for Seg-NN and Seg-PN, respectively. This suggests that reducing the dimension of PEs has the potential to impair the performance of Seg-NN, while Seg-PN can effectively learn shape

Frequency Distribution			Seg-NN	Seg-PN
Gaussian	Laplace	Uniform		
✓			**49.45**	**64.84**
	✓		45.21	63.21
		✓	49.35	62.49

Table 5. **Ablation Study for Frequency Distribution** in embedding manipulation layers.

Variance	0.5	1	2	5	10	20
Seg-NN	48.37	**49.45**	49.43	49.02	47.53	46.13
Seg-PN	63.87	64.84	64.86	**65.72**	64.29	63.10

Table 6. **Ablation for the Variance of the Gaussian Distribution** in frequencies sampling.

representations from relatively lower-dimensional embeddings.

Embedding Manipulation **1) Different Distributions of Sampled Frequencies** In embedding manipulation layers, we sample frequencies $\mathbf{v}$ to generate the projection weights. In Tab. 5, we compare the effects of different frequency distributions. Totally three types of distribution are compared, Gaussian, Laplace, and uniform distributions. By comparison, we observe that the best performance is achieved when the sampled frequencies follow a Gaussian distribution. The reason behind this may be that both Laplace and uniform distribution contain more mid- and high-frequency information, thereby introducing excessive noises and redundancies into shape representation. **2) Variance of Gaussian Distribution.** In Tab. 6, we investigate the impact of the variance of the Gaussian distribution used for frequency sampling. A larger variance indicates more middle or high frequencies are exploited in feature extraction. We find that Seg-PN can learn useful information from higher frequencies to enhance performance, while Seg-NN benefits more from low frequencies.

Scaling Factor γ in the Segmentation Head In the non-parametric segmentation head, we use $\varphi(x) = \exp(-\gamma(1-x))$ as an activation function, where γ is a scaling factor. In this part, we explore the effect of different values of γ. Tab. 7 presents the results of Seg-NN with different γs. We experiment with $\gamma \in \{100, 300, 500, 700, 1000, 1200, 1500\}$ and observe that $\gamma \leq 500$ guarantees more accurate prediction and $\gamma > 1000$ causes a rapid performance drop.

3.2. Ablation for Seg-PN

Pooling Operation in the QUEST Module In the QUEST module, we use local maximum pooling to obtain M' statistics for each point cloud. We mainly explore two hyperparameters: the kernel size and stride of the local maximum pooling, the values of which are set to be the same. Tab. 8 presents the effect of different kernel sizes. We observe that the best performance is achieved when the kernel size is 24, though the experiments in the main paper are conducted with a kernel size of 32.

γ	100	300	500	700	1000	1200	1500
Seg-NN	50.25	50.13	**50.66**	50.17	49.45	44.21	31.32

Table 7. **Ablation for Scaling Factor γ in the Segmentation Head** of Seg-NN.

Kernel Size	8	16	24	32	40	48
Seg-PN	59.67	64.87	**66.06**	64.84	64.75	63.90

Table 8. **Ablation for the Kernel Size and Stride** of the local maximum pooling operation in the QUEST module.

Source	**S3DIS**		**ScanNet**	
Target	S3DIS	ScanNet	S3DIS	ScanNet
Seg-NN	59.4	43.9	59.4	43.9
Seg-PN	**67.6**	**64.6**	**63.3**	**67.0**
DGCNN	56.6	44.8	49.4	42.7
AttMPTI	61.6	46.3	49.7	54.0
2CBR	63.5	49.6	54.9	52.3
PAP3D	65.4	52.3	57.0	64.5

Table 9. **Transferability among datasets.** We report the results under 2-way-5-shot settings on the S_0 split.

3.3. More Analysis

Reduction of 'seen'/'unseen' domain gap. **1)** We have shown that Seg-NN can reduce the 'seen'/'unseen' domain gap in the main paper's Fig. 2 (a). We further extend the experiments in Fig. 2, where the mIoU difference between 'seen' and 'unseen' classes by our Seg-NN and Seg-PN is much smaller (DGCNN's 38% vs our 3.1% and 12.0% on average). **2)** We also show the t-SNE visualization in Fig. 4, where the 3D features by Seg-NN are more discriminative among 'unseen' classes than DGCNN. This indicates the 'seen'/'unseen' semantic gap can be significantly alleviated by our encoder. The DGCNN in this experiment is trained on seen classes and tested on both seen and unseen classes.

Transferability among different datasets. In addition to the domain gap between the support set and query set, a natural extension is to investigate the transferability of our non-parametric model across different data domains. In Tab. 9, we present the transferring performance between S3DIS [1] and ScanNet [4] datasets. We train models on the source dataset and then utilize the target dataset to evaluate the

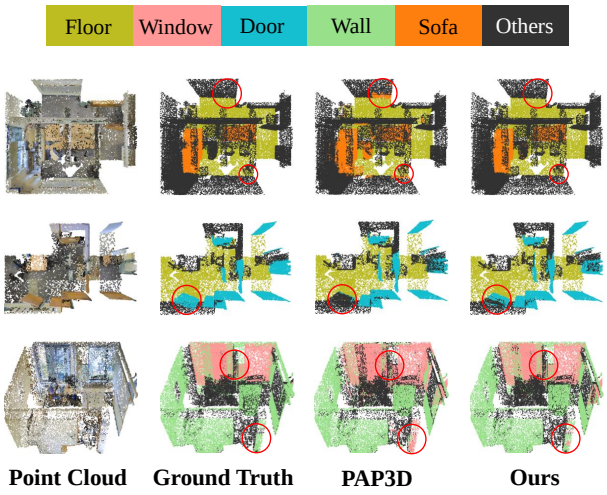

Figure 1. **Visualization of Results** on S3DIS dataset. We compare Seg-PN's results with the SOTA PAP3D model.

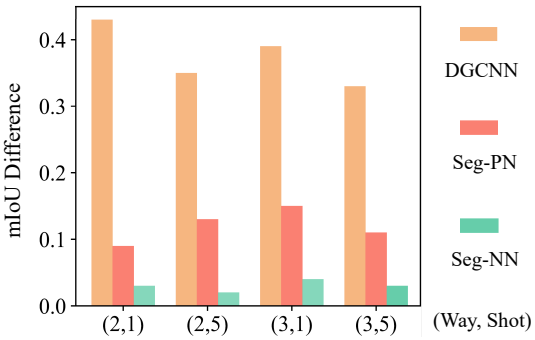

Figure 2. **'Seen'/'unseen' Performance Gap.** We compare the performance difference of segmentation on the S3DIS dataset, where DGCNN shows a large performance difference between seen and unseen classes.

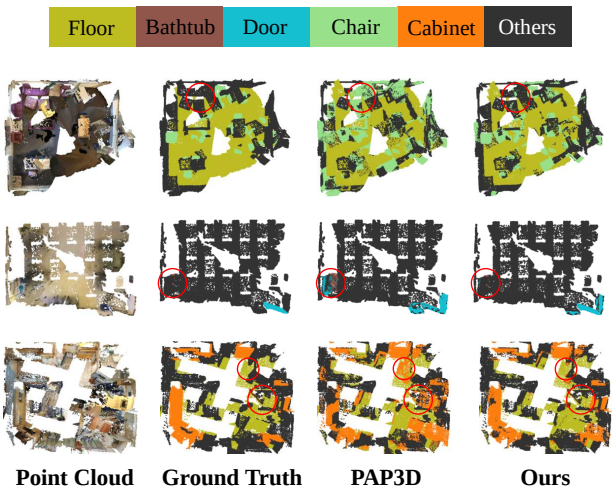

Figure 3. **Visualization of Results** on ScanNet dataset. We compare Seg-PN's results with the SOTA PAP3D model.

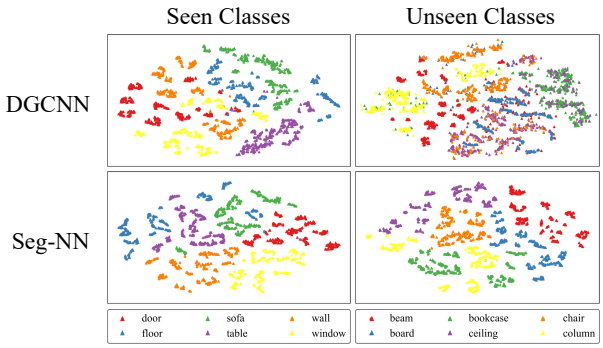

Figure 4. **t-SNE Visualization of Features** on S3DIS, which suggests the non-parametric Seg-NN can extract discriminative embeddings for both seen and unseen classes.

model. As shown in the table, *even trained on S3DIS, our Seg-PN can attain the best ScanNet performance* compared to all existing methods. The results demonstrate our superior cross-dataset generalization capacity.

4. Visualization

We present several qualitative results of 2-way-1-shot tasks in Fig. 1 and Fig. 3. Seg-PN achieves better segmentation than the existing SOTA, PAP3D [7], which demonstrates the effectiveness of Seg-PN. It is worth noting that due to sparse sampling in certain regions of the rooms, some ScanNet rooms may appear incomplete, as shown in Fig. 3. All rooms are presented in a top-down view.

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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1543, 2016. 1, 3

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *IEEE International Conference on Computer Vision*, pages 5855–5864, 2021. 1

[3] Anthony Chen, Kevin Zhang, Renrui Zhang, Zihan Wang, Yuheng Lu, Yandong Guo, and Shanghang Zhang. Pimae: Point cloud and image interactive masked autoencoders for 3d object detection. *CVPR 2023*, 2023. 1

[4] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017. 1, 3

[5] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Inter-*

national Conference on Artificial Intelligence and Statistics, pages 315–323, 2011. 2

[6] Ziyu Guo, Renrui Zhang, Longtian Qiu, Xianzhi Li, and Pheng Ann Heng. Joint-mae: 2d-3d joint masked autoencoders for 3d point cloud pre-training. *IJCAI 2023*, 2023. 1

[7] Shuting He, Xudong Jiang, Wei Jiang, and Henghui Ding. Prototype adaption and projection for few-and zero-shot 3d point cloud semantic segmentation. *IEEE Transactions on Image Processing*, 2023. 1, 4

[8] Dingchang Hu, Siang Chen, Huazhong Yang, and Guijin Wang. Query-guided support prototypes for few-shot 3d indoor segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 1

[9] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015. 2

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 1

[11] Minghua Liu, Yinhao Zhu, Hong Cai, Shizhong Han, Zhan Ling, Fatih Porikli, and Hao Su. Partslip: Low-shot part segmentation for 3d point clouds via pretrained image-language models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21736–21746, 2023. 1

[12] Gengchen Mai, Krzysztof Janowicz, Yingjie Hu, Song Gao, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. A review of location encoding for GeoAI: methods and applications. *International Journal of Geographical Information Science*, 36(4): 639–673, 2022. 1

[13] Yongqiang Mao, Zonghao Guo, LU Xiaonan, Zhiqiang Yuan, and Haowen Guo. Bidirectional feature globalization for few-shot semantic segmentation of 3d point cloud scenes. In *International Conference on 3D Vision*, pages 505–514, 2022. 1

[14] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1

[15] Zhenhua Ning, Zhuotao Tian, Guangming Lu, and Wenjie Pei. Boosting few-shot 3d point cloud segmentation via query-guided enhancement. In *Proceedings of the ACM International Conference on Multimedia*, pages 1895–1904, 2023. 1

[16] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 1

[17] 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. 1

[18] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. In *Advances in Neural Information Processing Systems*, 2022. 2

[19] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 1

[20] 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. 1

[21] Jiahui Wang, Haiyue Zhu, Haoren Guo, Abdullah Al Mamun, Cheng Xiang, and Tong Heng Lee. Few-shot point cloud semantic segmentation via contrastive self-supervision and multi-resolution attention. In *IEEE International Conference on Robotics and Automation*, pages 2811–2817, 2023. 1

[22] 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*, 38(5):1–12, 2019. 1

[23] Yating Xu, Conghui Hu, Na Zhao, and Gim Hee Lee. Generalized few-shot point cloud segmentation via geometric words. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 21506–21515, 2023. 1

[24] Canyu Zhang, Zhenyao Wu, Xinyi Wu, Ziyu Zhao, and Song Wang. Few-shot 3d point cloud semantic segmentation via stratified class-specific attention based transformer network. *arXiv preprint arXiv:2303.15654*, 2023. 1

[25] Renrui Zhang, Ziyu Guo, Peng Gao, Rongyao Fang, Bin Zhao, Dong Wang, Yu Qiao, and Hongsheng Li. Point-m2ae: Multi-scale masked autoencoders for hierarchical point cloud pre-training. *NeurIPS 2022*, 2022. 1

[26] Renrui Zhang, Liuhui Wang, Yu Qiao, Peng Gao, and Hongsheng Li. Learning 3d representations from 2d pre-trained models via image-to-point masked autoencoders. *CVPR 2023*, 2023.

[27] Renrui Zhang, Liuhui Wang, Yali Wang, Peng Gao, Hongsheng Li, and Jianbo Shi. Parameter is not all you need: Starting from non-parametric networks for 3d point cloud analysis. *arXiv preprint arXiv:2303.08134*, 2023. 1

[28] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *IEEE International Conference on Computer Vision*, pages 16259–16268, 2021. 1

[29] Na Zhao, Tat-Seng Chua, and Gim Hee Lee. Few-shot 3d point cloud semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8873–8882, 2021. 1, 2

[30] Jianqiao Zheng, Xueqian Li, Sameera Ramasinghe, and Simon Lucey. Robust point cloud processing through positional embedding, 2023. 1

[31] Guanyu Zhu, Yong Zhou, Rui Yao, and Hancheng Zhu. Cross-class bias rectification for point cloud few-shot segmentation. *IEEE Transactions on Multimedia*, 2023. 1