Supplementary Material for
WalkFormer: Point Cloud Completion via Guided Walks

Mohang Zhang[1,2] Yushi Li[1*] Rong Chen[3] Yushan Pan[1]
Jia Wang[1] Yunzhe Wang[4] Rong Xiang[5]

[1]School of Advanced Technology, Xi'an Jiaotong-Liverpool University
[2]University of Liverpool [3]Dalian Maritime University
[4]Suzhou University of Science and Technology [5]The Hong Kong Polytechnic University

A. Implementation Details

A.1. Training Details

We implemented the proposed WalkFormer by PyTorch. We set the number of neighbours K in K-NN to 20 for Point Selector and $K = 8$ for Route Transformer. Adam [1] optimizer is used with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and set the initial learning rate to 10^{-3}. We train the model end-to-end on a single NVIDIA RTX 3090 GPU. For the Completion3D dataset, the input and output number of points are both 2048. We set the batch size to 48 and train the model for 200 epochs with the continuous learning rate decay of 0.8 for every 20 epochs. For the PCN dataset, our model takes 2048 points as input and completes a point cloud with 2048 points eight times, resulting in a final output of 16384 points. We set the batch size to 32 and train the model for 300 epochs with the continuous learning rate decay of 0.9 for every 30 epochs.

A.2. Encoder Details

We stack several set abstraction and feature propagation layers [2] with point transformer blocks [8] to extract features. We use farthest point sampling (FPS) to hierarchically down-sample the point cloud in each layer of the set abstraction. We list the detailed encoder architecture in Figure 1, where the number of input and output points are both $N = 2048$, the feature dimension is $C = 3$ and 128, respectively.

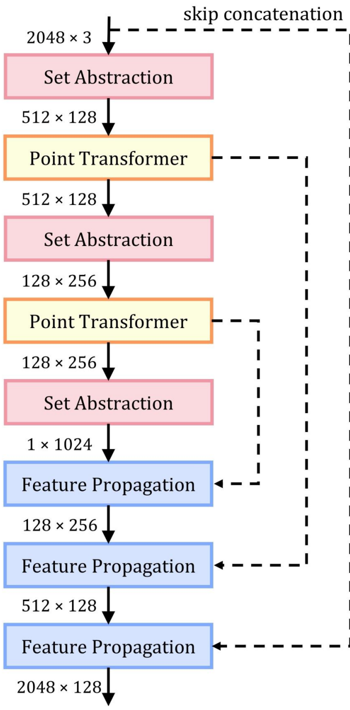

Figure 1. Detailed architecture of the Encoder.

B. More Experiment Results

B.1. Searching Radius

We provide more ablation experiments on the Completion3D dataset. The searching radius [3] determines the maximum distance that points can be moved during each deformation step. Table 1 evaluates the effectiveness of different searching radii, indicating that the baseline ($[1.0, 1.0, 0.1, 0.1, 0.01]$) surpasses other variations when employing this fixed decreasing ratio.

Table 1. The effect of searching radius on Completion3D dataset.

Searching Radius	CD-Avg	EMD-Avg
$[1.0, 1.0, 1.0, 1.0, 1.0]$	6.94	3.01
$[1.0, 1.0, 0.5, 0.5, 0.25]$	6.68	2.83
$[1.0, 1.0, 0.1, 0.1, 0.01]$	**6.59**	**2.75**

*Corresponding author: Yushi.Li@xjtlu.edu.cn

B.2. Deformation Step

By gradually decreasing the searching radius, our model exhibits a coarse-to-fine completion process. We further conduct experiments on the effectiveness of the deformation step. The quantitative results in Table 2 prove that our method benefits from the increasing number of steps, which successfully refines the point clouds throughout the deformation. However, the improvement is also limited by information redundancy with more than 5 steps.

Table 2. The effect of deformation step on Completion3D dataset.

Step	Searching Radius	CD-Avg	EMD-Avg
3	$[1.0, 0.1, 0.01]$	7.42	3.10
4	$[1.0, 1.0, 0.1, 0.01]$	6.84	2.89
5	$[1.0, 1.0, 0.1, 0.1, 0.01]$	**6.59**	**2.75**
6	$[1.0, 1.0, 0.1, 0.1, 0.01, 0.01]$	6.67	2.78

B.3. Complexity Analysis

We list the complexity analysis of our WalkFormer with other methods in Table 3. We report the number of parameters (Params) and theoretical computation cost (FLOPs) for space and time complexity. Despite involving multiple steps in the completion process, our method has relatively low Params and FLOPs among the methods in the table. We provide two versions of the WalkFormer, one complete version (step=5) yields the lowest EMD, and another version (step=4) outperforms most methods with reduced computational demands. This strikes a balance between the overhead and performance.

Table 3. The space and time complexity analysis on PCN dataset in terms of the number of parameters (Params) and theoretical computational cost (FLOPs).

Methods	Params	FLOPs	CD-Avg	EMD-Avg
PCN [7]	6.84M	14.69G	9.64	2.99
GRNet [6]	76.71M	25.88G	8.83	2.83
PMP-Net++ [4]	5.92M	**4.61G**	7.56	2.42
Snowflake [5]	19.32M	10.32G	7.21	2.20
SeedFormer [9]	**3.20M**	29.61G	**6.74**	2.14
WalkFormer (step=4)	7.32M	10.01G	6.97	2.18
WalkFormer (step=5)	9.21M	12.89G	6.79	**2.12**

C. More Visualization Results

We provide additional completion results on the PCN dataset, as depicted in Figure 2. Qualitative comparison with PCN [7], PMP-Net++ [4], and SeedFormer [9], shows the better visual performance of our method.

References

[1] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[2] 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

[3] Xin Wen, Peng Xiang, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. Pmp-net: Point cloud completion by learning multi-step point moving paths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7443–7452, 2021. 1

[4] Xin Wen, Peng Xiang, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. Pmp-net++: Point cloud completion by transformer-enhanced multi-step point moving paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):852–867, 2022. 2

[5] Peng Xiang, Xin Wen, Yu-Shen Liu, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Zhizhong Han. Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5499–5509, 2021. 2

[6] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Jiageng Mao, Shengping Zhang, and Wenxiu Sun. Grnet: Gridding residual network for dense point cloud completion. In *European Conference on Computer Vision*, pages 365–381, 2020. 2

[7] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 International Conference on 3D Vision (3DV)*, pages 728–737. IEEE, 2018. 2

[8] 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. 1

[9] Haoran Zhou, Yun Cao, Wenqing Chu, Junwei Zhu, Tong Lu, Ying Tai, and Chengjie Wang. Seedformer: Patch seeds based point cloud completion with upsample transformer. In *European Conference on Computer Vision*, pages 416–432, 2022. 2

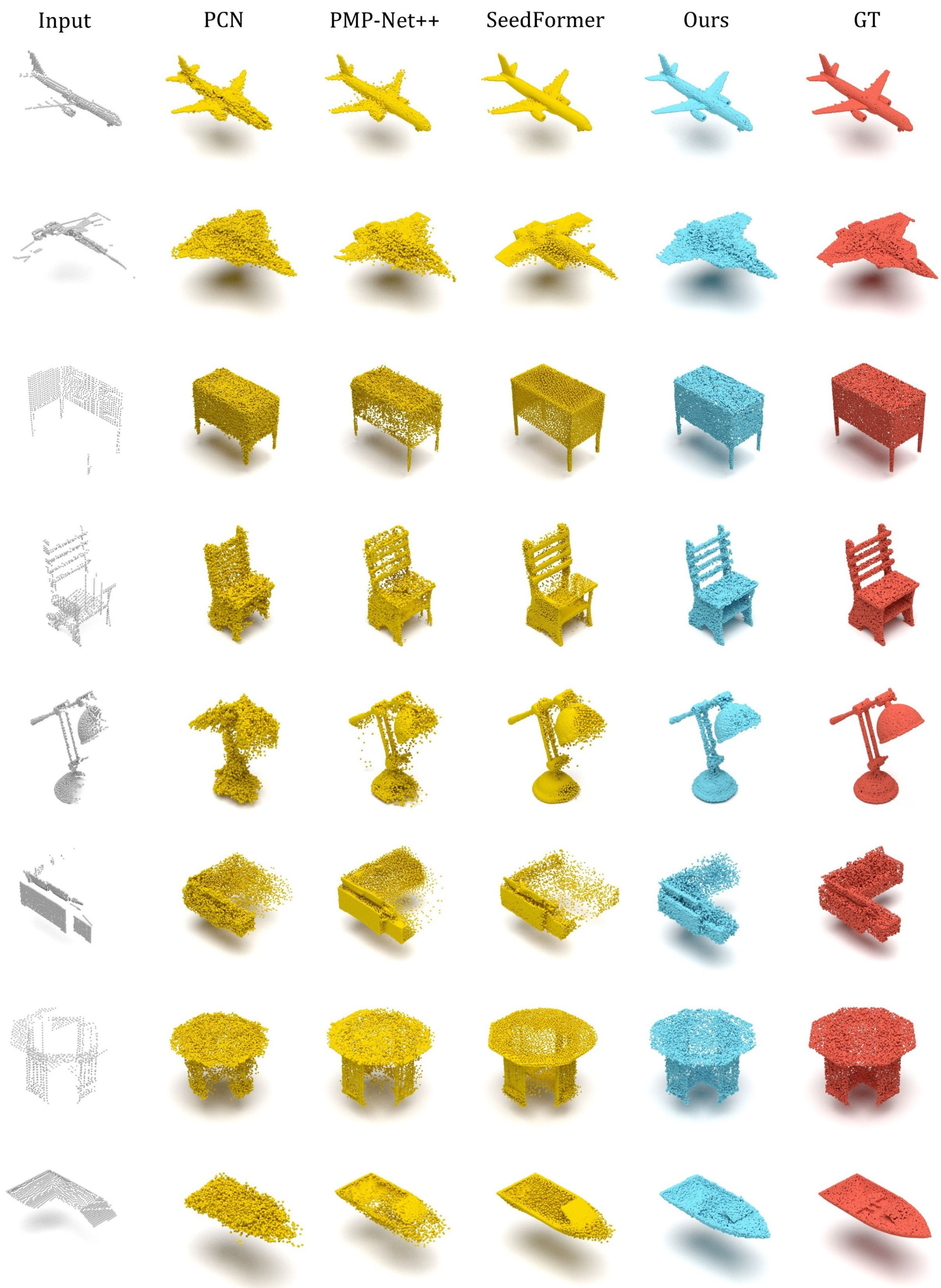

Figure 2. Visual comparison of point cloud completion results on the PCN dataset.