The Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24)

# PointAttN: You Only Need Attention for Point Cloud Completion

**Jun Wang**[1], **Ying Cui**[1], **Dongyan Guo**[1*], **Junxia Li**[2], **Qingshan Liu**[2], **Chunhua Shen**[3]
[1]Zhejiang University of Technology
[2]Nanjing University of Information Science and Technology
[3]Zhejiang University
{wangj, cuiying, guodongyan}@zjut.edu.cn
junxiali99@163.com, qsliu@nuist.edu.cn, chunhuashen@zju.edu.cn

## Abstract

Point cloud completion referring to completing 3D shapes from partial 3D point clouds is a fundamental problem for 3D point cloud analysis tasks. Benefiting from the development of deep neural networks, researches on point cloud completion have made great progress in recent years. However, the explicit local region partition like kNNs involved in existing methods makes them sensitive to the density distribution of point clouds. Moreover, it serves limited receptive fields that prevent capturing features from long-range context information. To solve the problems, we leverage the cross-attention and self-attention mechanisms to design novel neural network for point cloud completion with implicit local region partition. Two basic units Geometric Details Perception (GDP) and Self-Feature Augment (SFA) are proposed to establish the structural relationships directly among points in a simple yet effective way via attention mechanism. Then based on GDP and SFA, we construct a new framework with popular encoder-decoder architecture for point cloud completion. The proposed framework, namely PointAttN, is simple, neat and effective, which can precisely capture the structural information of 3D shapes and predict complete point clouds with detailed geometry. Experimental results demonstrate that our PointAttN outperforms state-of-the-art methods on multiple challenging benchmarks. Code is available at: https://github.com/ohhhyeahhh/PointAttN

## Introduction

Point cloud completion is the task of estimating a complete shape of an object from its incomplete observation. It plays an important role in 3D computer vision since the raw data captured by existing 3D sensors are usually incomplete and sparse due to factors such as occlusion, limited sensor resolution and light reflection. The unordered and unstructured point cloud data makes the task a challenging problem.

Benefiting from the recent advances of deep learning, point cloud completion has achieved remarkable progress. Current popular point cloud completion methods (Wang et al. 2020; Wen et al. 2021b; Huang et al. 2021; Alliegro et al. 2021; Wang et al. 2022b; Sun et al. 2022) mainly revolve around the design of an encoder-decoder architecture for complete point clouds generation. The famous point feature extractor PointNet (Charles et al. 2017) and its variant PointNet++ (Qi et al. 2017) are widely taken as the encoders in the mainstream methods (Tchapmi et al. 2019; Huang et al. 2020; Pan et al. 2021a; Xiang et al. 2021; Yu et al. 2021). Consequently, the idea of kNN (or its variant ball query) is introduced to build the local spatial relationships among points for local feature learning. The parameter of kNN is fixed and set empirically induced by the density of the point clouds. However, the densities of different point clouds are different, and even within the same point cloud, the points are not uniformly distributed in different local region. Thus, using the fixed parameter to process point clouds is far from ideal, which makes it hard to depict a general local geometric structure of points in local regions. In order to learn the structural features and long-range correlations among local regions, PointTr (Yu et al. 2021) proposes to adopt Transformers (Vaswani et al. 2017) to build an encoder-decoder architecture. SeedFormer (Zhou et al. 2022) further integrates seed features into the generation process of PointTr with a new shape representation. Wang et al.(Wang et al. 2022a) investigate grouping local features to improve the performance of completion. FBNet (Yan et al. 2022) proposes to reuse the high-level information for low-level feature learning via a feedback network, and leverages a cross transformer to build the connections between the present and subsequent features. SnowflakeNet (Xiang et al. 2021) focuses on the decoding process and introduces skip-transformer to integrate the spatial relationships across different levels of decoding. The attention mechanism with Transformer architecture in these methods has shown benefits on capturing the structure features in point clouds. However, kNNs are still used in these methods to capture the geometric relation in the point cloud.

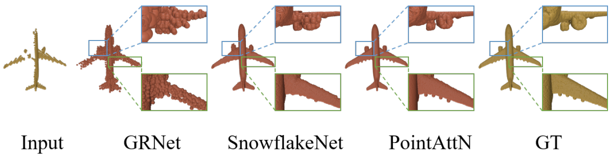

Figure 1: Visual comparisons of point cloud completion. Our PointAttN can produce high-quality complete shape with accurate geometric details.

[*]Corresponding author: Dongyan Guo

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In this paper, we investigate eliminating the above-mentioned issue of kNNs, and show that a properly designed encoder-decoder architecture with only MLP and attention mechanism can achieve leading performance in point cloud completion. Different from existing methods that adopt explicit local feature learner from local neighborhoods to capture local geometry, the proposed framework, termed PointAttN, focuses on capturing both local and global geometric context in an implicit manner via the self-attention and cross-attention mechanisms. PointAttN adopts the encoder-decoder architecture to process the point completion task in a coarse-to-fine manner. It mainly consists of three parts: a feature extractor module for local geometric structure and global shape feature capturing, a seed generator module for coarse point cloud generation, and a point generator module to produce the fine-grained point cloud. To construct the three modules, we specifically design two basic units, one is a geometric details perception unit (GDP) and another is a self-feature augment unit (SFA). The GDP unit adopts the cross-attention to establish the relationships of points between the input point cloud feature and its down-sampled point cloud feature, which allows each point feature in the down-sampled point cloud to perceive the geometric detail features in the original point cloud. By eliminating the concept of explicit local feature region partition like kNNs, GDP can adaptively capture the local geometry structure of the point cloud, which can alleviate the influence of local point density and achieves more precise geometric details for reconstructing fine-grained point information. The SFA unit establishes the relationship among points in its input point cloud by introducing self-attention, which allows each point feature in the cloud to augment its global perceptibility. By cascading SFAs in the coarse-to-fine decoding process, we can progressively reveal the spatial structural and context information of the 3D shape in each processing step to produce more precise shape structure. Considering its ability of capturing global information, SFA can also be adopted in the encoding step, which is used to enhance the global structure perception ability of the feature model. With such designs, the proposed framework PointAttN is simple and neat, which only includes the down-sampling operation of farthest point sampling (FPS), multilayer perceptron (MLP) and attention layers. As shown in Figure 1, our PointAttN can generate fine-grained shapes with precise geometric details. In summary, our main contributions are:

- We propose a novel framework PointAttN for point cloud completion. As far as we know, PointAttN is the first time to eliminate the explicit local region operations such as kNNs in Transformer-based methods. It alleviates the influence of data density distribution and achieves high-quality complete shapes with precise geometrical details.
- We propose two basic and essential units GDP and SFA for constructing the framework. They establish the relationships among points in a very simple yet effective way via attention mechanism. Moreover, the proposed units and modules can be easily incorporated into other networks to enhance the capability of feature representation and fine-grained shape generation for completion.
- Without bells and whistles, the proposed method achieves leading performance for point cloud completion on challenging benchmarks such as Completion3D, PCN, ShapeNet-55/34 and KITTI.

## Related Work

The objective of point cloud completion is to forecast the complete shape of a 3D object from a partial point cloud. Current methodologies, predicated on deep neural networks, solve this task through an encoder-decoder architecture. However, an inherent challenge lies in capturing the intricate topological details in these unordered point clouds. Pioneering works (Dai, Qi, and Nießner 2017; Groueix et al. 2018; Stutz and Geiger 2018; Thomas et al. 2019; Wang, Ang, and Hee Lee 2021) map the point cloud to a voxel grid, and then use 3D convolution to extract features. GRNet (Xie et al. 2020) further proposes a gridding reverse module to map voxels and complete the point cloud in voxel mesh. However, due to the cubic nature of the voxel mesh, features of the point cloud surface cannot be properly represented.

With the success of PointNet (Charles et al. 2017) that directly processes 3D coordinates, many researchers leverage it as the feature encoder and pay special attention on the decoding process to produce complete point clouds (Tchapmi et al. 2019; Huang et al. 2020; Sipiran et al. 2022). However, since PointNet directly processes all the points with max pooling to obtain global features, the local structures among points are not learned by the network, which lead to the loss of shape details during decoding. To solve the problem, NSFA (Zhang, Yan, and Xiao 2020) proposes to explore the functionality of multi-scale features from different layers to enhance the performance. CRN (Wang, Ang, and Lee 2020) proposes a cascaded refinement network to synthesize the detailed object shapes by considering both the local details of partial input with the global shape information together. Inspired by the different receptive fields across multiple levels of CNNs, PointNet++ (Qi et al. 2017) proposes to process a set of points sampled in a metric space in a hierarchical fashion, where the ball query (an invariant of kNN) is used to guarantee local neighborhoods. By leveraging its success representation of local shape features, recent works with encoder-decoder architecture that adopt PointNet++ to construct feature extractors have shown great progress in point cloud completion (Pan et al. 2021a; Wen et al. 2020; Xiang et al. 2021; Zhu et al. 2023). However, the prefixed partition of local regions involved by kNNs make these methods sensitive to the density of point clouds. Moreover, the involved limited receptive fields prevent the feature extractor from achieving better local and global structural information of the point cloud.

Compared with the limited receptive fields of CNNs, Transformer (Vaswani et al. 2017) characterized by the attention mechanism shows its advantages in long-range interaction capturing (Dosovitskiy et al. 2020; Carion et al. 2020; Guo et al. 2021a; Zhang et al. 2022). Inspired by their success, researchers try to introduce the transformer framework into point cloud analysis tasks (Guo et al. 2021b; Pan et al. 2021b; Wen et al. 2020; Zhao et al. 2021; Yan et al.

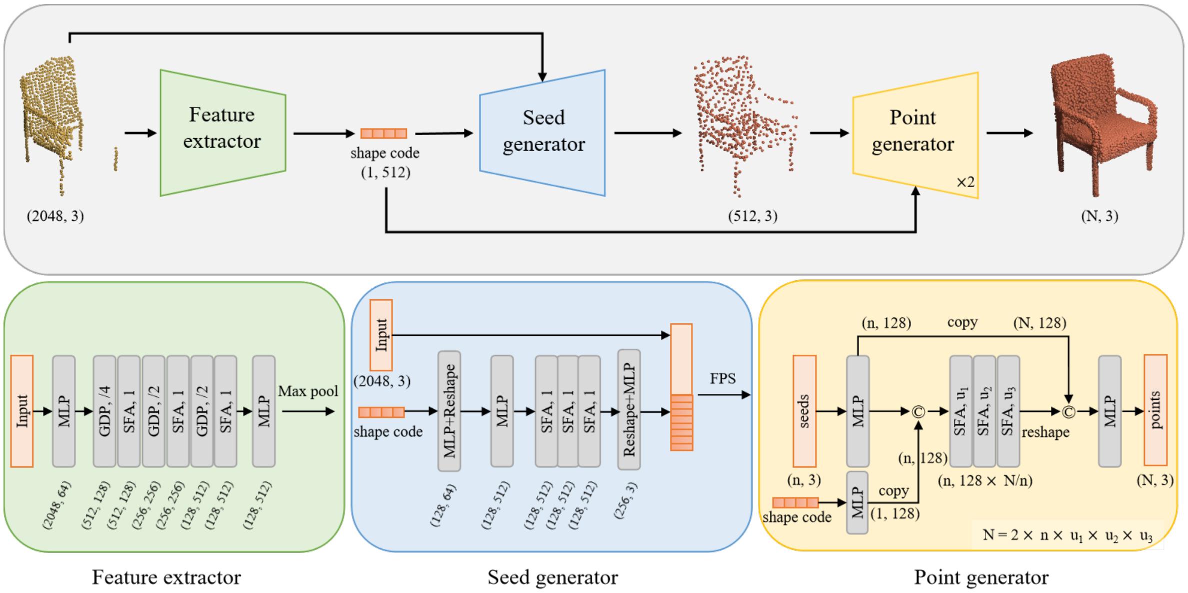

Figure 2: The overall framework of our PointAttN. Here © denotes a concatenation operation. For easy understanding, we take 2048 as the number of input partial points to draw the illustration. Point clouds with other number of partial points can also be processed by retraining the model.

2022). Observing that the extracted features from PointNet neglect the geometric relationship within the point clouds due to the max-pooling operation, PCTMA-Net (Lin et al. 2021) stacks a number of Transformer encoders after PointNet to capture the local context within a point cloud and exploits its local geometric details. PMP-Net++ (Wen et al. 2022) incorporates PointNet++ with Transformer to enhance the learned point features for point cloud completion. SA-Net (Wen et al. 2020) introduces the skip-attention mechanism to fuse local region features from encoder into point features of decoder, which enables more detailed geometry information preserving for decoding. On the other hand, SnowflakeNet(Xiang et al. 2021) and PointTr(Yu et al. 2021) pay special attention on the decoding process via Transformer architectures. FBNet (Yan et al. 2022) proposes a feedback network to refine completion shapes across time steps and leverage the cross-transformer to build the connections between the present and subsequent features. Wang et al. (Wang et al. 2022a) investigate grouping local features via point feature matching, neighbor-pooling and upsampling to improve the completion performance. To improve the ability of detail preservation and recovery, SeedFormer (Zhou et al. 2022) proposes a new shape representation algorithm Patch Seeds to integrate seed features into the generation process of PointTr. These methods have highlighted the potential of transformers in accomplishing point cloud completion. However, due to memory constraints or explicit local partition demand, kNNs are still employed within these methods. In this work, we propose that the need of kNNs can be obviated by a strategically designed framework, which optimally leverages the strengths of both self-attention and cross-attention mechanisms. Concurrently, we demonstrate that a much simpler architecture can achieve superior performance compared to state-of-the-art methods for point cloud completion.

## Proposed Method

The overall framework of the proposed PointAttN is illustrated in Figure 2. The framework adopts a popular encoder-decoder architecture for point cloud completion. It mainly consists of three modules, a feature extractor for shape feature encoding, a seed generator and a point generator for coarse-to-fine generation of the complete shape. Two basic units, namely Geometric Details Perception (GDP) and Self-Feature Augment (SFA), are essentially designed to construct the three modules. GDP and SFA are built upon the core concept of Transformer (Vaswani et al. 2017), i.e., incorporating multi-head cross-attention and self-attention mechanisms, yet the encoder-decoder structure of Transformer is not adopted in the design. In the following, we first introduce the two basic units and then describe the three modules based on them in details.

### Geometric Details Perception

Establishing local spatial relationship among unordered points for feature learning is a critical and fundamental challenge in point cloud completion. Current kNNs-based local shape models are influenced by density variation and offer limited receptive fields, hindering precise capture of the point cloud's structural information. To address the issues, we use a cross-attention mechanism, establishing relationships between the original and down-sampled point clouds. Figure 3 illustrates the cross-attention map, indicating each down-sampled point's corresponding local region in the input cloud. The operation offers an implicit local region partition manner, which provides an adaptable receptive field

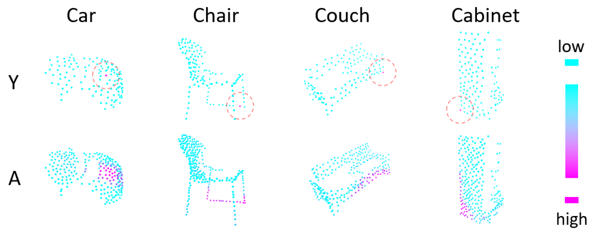

Figure 3: Cross-attention map between an input point cloud (Row A) and its down-sampled point cloud (Row Y). The heatmap in Row A shows attention weights of an indexed point in Row Y ( the point representing in red).

compared to explicit kNNs partition. Based on it, we design the Geometric Details Perception (GDP) unit to adaptively aggregate information from unordered points, which can effectively model the point cloud's geometric features.

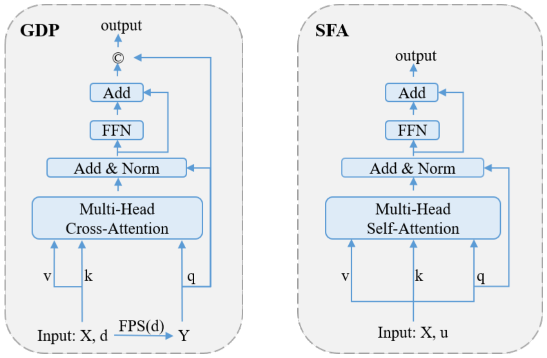

Figure 4: The detailed structure of GDP (left) and SFA (right). Here © denotes a concatenation operation.

The structure of GDP is illustrated in the left of Figure 4. GDP receives an input $(X, d)$, where $X$ is a matrix of size $n \times c$ and each row of $X$ can be considered as a feature vector corresponding to a point, $d$ is the down-sampling ratio. By applying an FPS operation (Qi et al. 2017), we can get a down-sampled point cloud feature matrix $Y$ of size $n/d \times c$. Then we leverage the multi-head cross-attention in the form of residual to learn the feature matrix $F$:

$$\begin{aligned} F &= Norm(Q + MultiHead(Q, K, V)), \\ Q &= YW^Q, K = XW^{KV}, V = XW^{KV}, \end{aligned} \tag{1}$$

where $MultiHead(\cdot)$ is performed similarly to (Vaswani et al. 2017), $W^Q \in \mathbb{R}^{c \times c}$ and $W^{KV} \in \mathbb{R}^{c \times c}$ are linear transformation matrices. Through this operation, each point in $Y$ can adaptively aggregate features from $X$, encompassing similarity in shape and proximity in distance, thus the local geometric structure can be perceived in the model.

In order to enhance the fitting ability of the model, we use a feed forward network (FFN) (Vaswani et al. 2017) to further update $F$. The output of GDP can be formulated as

$$GDP(X, d) = Concat(F + FFN(X), Y), \tag{2}$$

where $Concat(\cdot)$ denotes a concatenation operation.

## Self-Feature Augment

While GDP enables perception of local geometric details, we then need to address another issue for predicting the complete point cloud, that is, revealing the global shape information of 3D objects from incomplete point clouds. As illustrated in Figure 5, the self-attention mechanism demonstrates its global context association ability among points. We leverage it to establish global spatial relationships of points and design a Self-Feature Augment (SFA) unit to infer the complete geometry of 3D shapes.

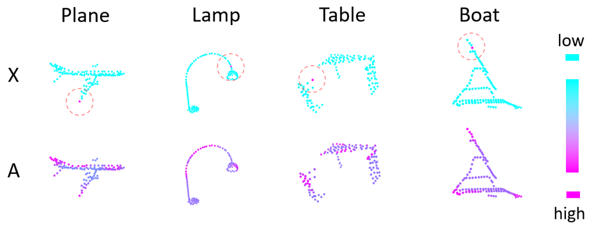

Figure 5: Self-attention map superimposed on the input point cloud. The heatmap in Row A shows attention weights of an indexed point in Row X (the point representing in red). Self-attention emphasizes the global context association relationship.

The structure of SFA is shown in the right of Figure 4. SFA receives an input $(X, u)$, where $X$ is a matrix of $n \times c$, $u$ is an up-sampling ratio. SFA integrates the information from different points of $X$ by applying the multi-head self-attention in the form of residual:

$$\begin{aligned} F &= Norm(Q + MultiHead(Q, K, V)), \\ Q &= XW^{Qu}, K = XW^{KVu}, V = XW^{KVu}, \end{aligned} \tag{3}$$

where $W^{Qu} \in \mathbb{R}^{c \times uc}$ and $W^{KVu} \in \mathbb{R}^{c \times uc}$ are linear transformation matrices. The dimension of point features is increased by $u$ times after the linear transformation operation.

Similar to GDP, an FFN is also ultilized here, thus the output of SFA can be computed as

$$SFA(X, u) = F + FFN(X). \tag{4}$$

Accordingly, the output of SFA is a matrix of size $n \times uc$.

Considering its ability of capturing global context information, SFA can also be employed for feature extraction to enhance the model's feature representation ability.

## PointAttN for Point Cloud Completion

Our PointAttN framework leverages the widely-used encoder-decoder architecture for point cloud completion. The encoder, serving as Feature Extractor, is constructed upon both GDP and SFA units to capture precise geometric information of 3D shapes. Meanwhile, the decoder modules, including Seed Generator and Point Generator are constructed upon cascade SFAs to produce complete point cloud in a coarse-to-fine decoding manner.

**Feature Extractor.** The feature extractor (FE) takes the partial point cloud as input to generate a shape code that captures both local geometry details and global context of the object for the following coarse-to-fine shape decoding. As shown in the lower left of Figure 2, a multi-layer perceptron (MLP) translates the unordered points into feature vectors. Then by alternately stacking three GDP and three SFA units, both local geometry and global shape information can be progressively embedded during the down-sampling process. GDP units, with a down-sampling ratio of $4, 2, 2$, adaptively aggregate local geometry details from the partial point cloud. Concurrently, SFA post each GDP unit enhances the global information perception. The up-sampling ratio of each SFA is set to 1 to maintain the same feature dimensions. After the stacked units, we use an MLP followed by a max pooling operation to produce the shape code.

**Seed Generator.** The seed generator (SG) aims to produce a sparse yet complete point cloud. As shown in the lower middle of Figure 2, the shape code is first decoded into a feature matrix via an MLP and reshape operation, with each matrix row representing a feature vector of a point. After extending point feature dimensions by another MLP, three SFA units are cascaded to enhance the feature ability of perceiving the target shape structure. Sparse points are produced by splitting point features through reshape and MLP, which transforms feature vectors into 3D coordinates. These points are merged with the input partial point cloud and undergo down-sampling via an FPS operation to yield a sparse point cloud, which is served as the seed for the point generator module.

**Point Generator.** The point generator (PG) module uses the shape code and the seed point cloud to generate a fine-grained point cloud. As shown in the lower right of Figure 2, the seed point cloud and shape code are separately processed through MLP to yield two point feature matrices $F_1$ and $F_2$ (matching $F_1$'s row count through self-copy), which are then integrated through point-wise concatenation. By that, each point feature in the integrated matrix maintains both local geometric details and global shape structure of the target. In the next, three cascading SFA units incrementally up-sample the point features, which are then split through reshaping and concatenated with $F_1$ (again matching row counts via self-copy). Dense points are generated by transforming the concatenated features into 3D coordinates via MLP. During implement, two PG modules are cascaded to construct the fine decoding network.

**Training Loss.** During training, we utilize Chamfer distance (CD) as the metric distance of point cloud to formulate the loss function. Suppose the seed point cloud, the output point clouds of the two cascaded point generators are denoted as $\mathcal{P}_0$, $\mathcal{P}_1$, $\mathcal{P}_2$, respectively. Meanwhile, the groud-truth point cloud is down-sampled through FPS to obtain three sub-clouds $\mathcal{S}_0$, $\mathcal{S}_1$, $\mathcal{S}_2$, which respectively share the same density of $\mathcal{P}_0$, $\mathcal{P}_1$ and $\mathcal{P}_2$. Then the loss of the model can be defined as follows:

$$\mathcal{L} = \sum_{i=0}^{2} \lambda_i d_{CD}(\mathcal{P}_i, \mathcal{S}_i). \tag{5}$$

where $d_{CD}$ is the Chamfer distance loss. In the implementation, each $\lambda_i$ is set as 1.

# Experiments

## Dataset and Implementation Details

To evaluate the effectiveness of our PointAttN, we conduct comprehensive experiments on multiple challenging benchmarks, including Completion3D (Tchapmi et al. 2019), PCN (Yuan et al. 2018), ShapeNet-55/34(Yu et al. 2021) and KITTI (Geiger et al. 2013). For fair comparison, we follow the common protocols of each dataset for training and testing. The proposed framework is implemented in Python with PyTorch and trained on 4 NVIDIA 2080Ti GPUs. Models are trained with Adam optimizer by totally 400 epochs, while the learning rate is initialized to 1E-4 and decayed by 0.7 every 40 epochs. The batch size is set to 32.

## Comparisons on Different Datasets

**Completion3D.** To align with previous works, we use the specified training set of Completion3D to train the model and take the L2 Chamfer distance (CD) as the metric. The results are shown in Table 1. Our PointAttN ranks first and surpasses the second-ranked method SnowflakeNet (Xiang et al. 2021) by a large reduction of 12.8% (6.63 vs. 7.60, the following data percentages are calculated in the same way) in terms of average CD. Compared with methods like VRC-Net (Pan et al. 2021a) and SnowflakeNet (Xiang et al. 2021) that also adopt the same coarse-to-fine decoding strategy, our PointAttN achieves significant improvement, which owing to the proposed GDP and SFA units that help to capture both local and global geometry information of the shape.

| Methods | Avg | Plane | Cabinet | Car | Chair | Lamp |
|---|---|---|---|---|---|---|
| GRNet(eccv20) | 10.64 | 6.13 | 16.90 | 8.27 | 12.23 | 10.22 |
| SoftPool++(ijcv22) | 9.36 | 4.59 | 15.82 | 6.78 | 11.41 | 8.82 |
| SCRN(tpami21) | 9.13 | 3.35 | 12.81 | 7.78 | 9.88 | 10.12 |
| VRC-Net(cvpr21) | 8.12 | 3.94 | 10.93 | 6.44 | 9.32 | 8.32 |
| PMP-Net++(tpami22) | 7.97 | **3.25** | 12.25 | 7.62 | 8.71 | 7.64 |
| SnowflakeNet(iccv21) | 7.60 | 3.48 | 11.09 | 6.90 | 8.75 | 8.42 |
| PointAttN | **6.63** | 3.28 | **10.77** | **6.13** | **7.14** | **5.92** |

Table 1: Point cloud completion results on Completion3D in terms of CD-L2 (lower is better). “Avg” denotes the average CD on all 8 categories of Competion3D. Only five categories are shown due to width limitation.

**PCN.** For a fair comparison, we follow the same split settings with PCN(Yuan et al. 2018) during experiments and take the L1 Chamfer Distance as the metric. The quantitative comparisons with the state-of-the-art methods are shown in Table 2. Our method ranks second only a bit behind the first ranked SeedFormer (Zhou et al. 2022) in terms of average CD, yet it outperforms SeedFormer on multiple categories like Cabinet and Car. Notably, it surpasses Transformer-based methods such as FBNet (Yan et al. 2022), SnowflakeNet (Xiang et al. 2021) and PointTr (Yu et al. 2021). The comparison demonstrates the effectiveness of our kNN-free attention-based framework in producing complete shapes.

A comparison of visualization results is shown in Figure 6, listing four cases with varying un-uniform density distributions. Compared with Transformer-based methods SnowflakeNet (Xiang et al. 2021) and SeedFormer (Yu et al. 2021) that still adopt explicit kNNs, our PointAttN achieves much more precise geometric structures and fine-grained details, as demonstrated in the completion of lamp poles that have extreme density changes. It is owing to that our kNN-free framework alleviate the sensitivity to the density distribution of point cloud. Moreover, by eliminating the limited receptive field of explicit local partition, our method better perceives both geometric details and global shape structures during the process.

| Methods | Avg | Plane | Cabinet | Car | Chair | Lamp |
|---|---|---|---|---|---|---|
| GRNet(eccv20) | 8.83 | 6.45 | 10.37 | 9.45 | 9.41 | 7.96 |
| PointTr(iccv21) | 8.38 | 4.75 | 10.47 | 8.68 | 9.39 | 7.75 |
| SCRN(tpami21) | 8.29 | 4.80 | 9.94 | 9.31 | 8.78 | 8.66 |
| PMP-Net++(tpami22) | 7.56 | 4.39 | 9.96 | 8.53 | 8.09 | 6.06 |
| SnowflakeNet(iccv21) | 7.21 | 4.29 | 9.16 | 8.08 | 7.89 | 6.07 |
| GTNet(ijcv23) | 7.15 | 4.17 | 9.33 | 8.38 | 7.66 | 5.49 |
| FBNet(eccv22) | 6.94 | 3.99 | 9.05 | 7.90 | 7.38 | 5.82 |
| SeedFormer(eccv22) | **6.74** | **3.85** | 9.05 | 8.06 | **7.06** | **5.21** |
| PointAttN | 6.84 | 3.88 | **9.01** | **7.60** | 7.28 | 5.97 |

Table 2: Point cloud completion results on PCN in terms of CD-L1 (lower is better). "Avg" denotes the average CD on all 8 categories of PCN. Only five categories are shown due to width limitation. The best results are highlighted in bold.

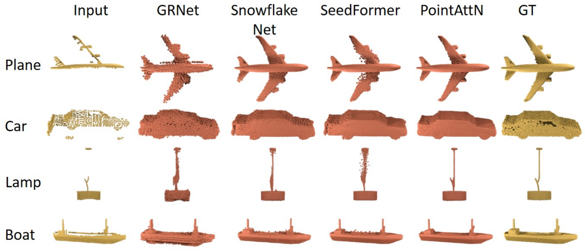

Figure 6: Visual comparisons on PCN. Our method shows its superiority on handling un-uniform density in point clouds, achieving more accurate and fine-grained completions.

**ShapeNet-55/34.** To evaluate the completion performance under different poses and the generality to unseen classes, we conduct experiments on the ShapeNet-55/34 benchmarks. During experiments, we follow the training and evaluation settings of PointTr(Yu et al. 2021). The results are shown in Table 3, with CD-S, CD-M and CD-H indicating the CD results for Simple, Moderate and Hard levels of incompleteness, respectively. As shown in the first four rows of Table 3, our PointAttN can effectively handles diverse viewpoints. Compared with state-of-the-art methods like PointTr and SeedFormer, it achieves reductions of at least 6%, 14.28%, and 27.35% in CD-L2 across the simple, moderate and hard completion levels. Besides, our method also achieve robust performance on unseen classes. As shown in the last four rows of Table 3, it ranks second that only a bit lower to SeedFormer. However, for the hard level (CD-H) completion task (75% missing points), our method surpasses SeedFormer by 5.11% reduction in CD-L2, which demonstrates the generality of the proposed method to unseen classes and hard level in-completions.

| Evaluation | | GRNet | PointTr | SeedFormer | GTNet | PointAttN |
|---|---|---|---|---|---|---|
| 55 categories | CD-S | 1.35 | 0.58 | 0.50 | **0.45** | 0.47 |
| | CD-M | 1.71 | 0.88 | 0.77 | 0.66 | **0.66** |
| | CD-H | 2.85 | 1.79 | 1.49 | 1.30 | **1.17** |
| | Avg | 1.97 | 1.09 | 0.92 | 0.80 | **0.77** |
| 34 seen categories | CD-S | 1.26 | 0.76 | **0.48** | 0.51 | 0.51 |
| | CD-M | 1.39 | 1.05 | 0.70 | 0.73 | **0.70** |
| | CD-H | 2.57 | 1.88 | 1.30 | 1.40 | **1.23** |
| | Avg | 1.74 | 1.23 | 0.83 | 0.88 | **0.81** |
| 21 unseen categories | CD-S | 1.85 | 1.04 | **0.61** | 0.78 | 0.76 |
| | CD-M | 2.25 | 1.67 | **1.07** | 1.22 | 1.15 |
| | CD-H | 4.87 | 3.44 | 2.35 | 2.56 | **2.23** |
| | Avg | 2.99 | 2.05 | **1.34** | 1.52 | 1.38 |

Table 3: Point cloud completion results on ShapeNet-55/34 in terms of CD-L2. The best results are highlighted in bold.

**KITTI.** To further evaluate the effectiveness of our PointAttN, we show its performance in real-world scenarios on the KITTI dataset. Following the experimental settings of existing works (Yu et al. 2021; Xie et al. 2020), we fine-tune our model ( prertrained on the PCN dataset) on ShapeNetCars (Yuan et al. 2018) and report the results in terms of MMD and Fidelity metrics, as shown in Table 4. PointTr (Yu et al. 2021) preserves all the input points in the completed point cloud, leading to a Fidelity score of 0.00. Both SnowflakeNet (Xiang et al. 2021) and SeedFormer (Zhou et al. 2022) utilize the partial matching loss from (Wen et al. 2021a) to maintain the shape structure of the input point cloud, resulting in Fidelity scores of 0.110 and 0.151, respectively. Beyond that, our method achieves the best completion results on KITTI. Compare with state-of-the-art Transformer-based methods, our PointAttN surpasses SeedFormer by 2.33% and PointTr by 4.18% reductions on MMD. Compared with GRNet (Xie et al. 2020), our PointAttN surpasses it by 11.27% and 17.65% reductions on MMD and Fidelity, respectively. While the benchmarks of KITTI are real scans with highly non-uniform distribution, the substantial improvement in both MMD and Fidelity demonstrates the robustness of our PointAttN against point cloud density variation.

Figure 7 presents visual comparisons against GRNet and PointTr across five hard cases. PointAttN successfully recovers a complete car point cloud in case A and smoothly produces detailed components in case D. In case E, only PointAttN accurately recreates the car's rear view mirrors. Analyzing cases A, B, and D, PointAttN proves its superiority in maintaining car contours. It also delivers more precise results than the kNN-based PointTr, achieving fine-grained details like windows, mirrors, and tires. It is attributed to the adaptive local receptive fields and the global associative capability of the proposed GDP and SFA units, which enables PointAttN to perform accurate 3D object completion.

| *1000 | SnowflakeNet | GRNet | PointTr | SeedFormer | PointAttN |
|---|---|---|---|---|---|
| Fidelity | 0.110 | 0.816 | 0.000 | 0.151 | **0.672** |
| MMD | 0.907 | 0.568 | 0.526 | 0.516 | **0.504** |

Table 4: Point cloud completion results on KITTI in terms of MMD and Fidelity. The best results are highlighted in bold.

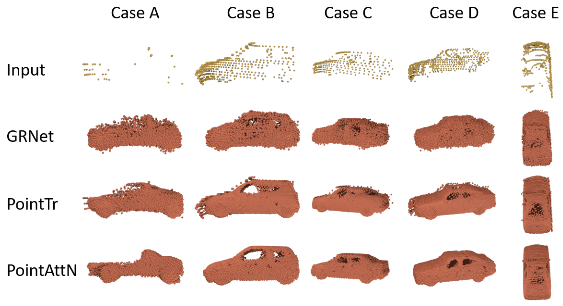

Figure 7: Visual comparisons on the KITTI dataset. Our PointAttN can produce fine-grained results with much more accurate details (e.g. windows, rear view mirrors and tires).

## Ablation Studies

To evaluate the effectiveness of the proposed units and framework for completion, we construct detailed ablation studies of each part in PointAttN on the Completion3D dataset. Five network variations are designed: (1) *FE w/o GDP*: to verify the effectiveness of our GDP for feature extraction, we replace all the GDP units in the FE module with FPS, and set the up-sampling ratio $u$ of SFAs in FE to 2 for model structure preservation; (2) *FE w/o SFA*: to verify the effectiveness of our SFA for feature extraction, we remove all the SFA units in FE and keep other structures unchanged; (3) *FE+FoldingNet*: to verify the effectiveness of the designed FE module, we use it to replace the backbone of FoldingNet (Yang et al. 2018); (4) *FE+SG+SPD*: to verify the effectiveness of the designed FE and SG modules, we replace the PG module with the SPD decoder in SnowflakeNet (Xiang et al. 2021); (5) *PN+SG+PG*: to verify the effectiveness of the designed SG and PG modules, we replace the FE module with PointNet++ (Qi et al. 2017).

In all experiments, we replace only the relevant parts, keeping all other settings unchanged. Table 5 shows the ablation results. By comparing PointAttN with *FE w/o GDP*, it can be found that the use of GDP reduces the average CD by 10.6% (6.63 vs. 7.42), which verifies its importance for feature extraction. The comparison of PointAttN with *FE w/o SFA* shows that SFA also improves the performance by enhancing the global information in feature extraction. *FE+FoldingNet* reduces the average CD by 56.6% (8.28 vs. 19.07) when compared to the baseline FoldingNet. Such a large improvement demonstrates the effectiveness of our feature extractor for point cloud completion. *FE+SG+SPD* achieves a 3.8% lower average CD (7.31 vs. 7.60) than the baseline SnowflakeNet. Compared with SA-Net (Wen et al. 2020) that uses PointNet++ as the backbone, *PN+SG+PG* reduces the average CD by 35.6% (7.23 vs. 11.22), demonstrating the capability of our decoding modules designed with SFAs. Moreover, each variation, when compared with PointAttN, indicates the unique contribution of each corresponding part to performance improvement, proving that the framework is neatly designed.

| Evaluation | Avg | Plane | Cabinet | Car | Chair | Lamp |
|---|---|---|---|---|---|---|
| *FE w/o GDP* | 7.42 | 2.99 | 13.03 | 6.46 | 7.51 | 6.61 |
| *FE w/o SFA* | 6.81 | 3.43 | 11.62 | 6.79 | 7.11 | 6.32 |
| *FE+FoldingNet* | 8.28 | 3.33 | 11.54 | 7.5 | 8.62 | 8.4 |
| *FE+SG+SPD* | 7.31 | 3.19 | 11.49 | 7.26 | 7.78 | 7.13 |
| *PN+SG+PG* | 7.23 | 3.44 | 11.5 | 6.3 | 7.54 | 6.74 |
| SA-Net | 11.22 | 5.27 | 14.45 | 7.78 | 13.67 | 13.53 |
| FoldingNet | 19.07 | 12.83 | 23.01 | 14.88 | 25.69 | 21.79 |
| SnowflakeNet | 7.60 | 3.48 | 11.09 | 6.90 | 8.75 | 8.42 |
| PointAttN | 6.63 | 3.28 | 10.77 | 6.13 | 7.14 | 5.92 |

Table 5: Ablation studies on the Completion3D dataset. Here SA-Net, FoldingNet and SnowflakeNet are listed as baselines for intuitive comparison (lower value is better).

## Complexity Analysis

To evaluate the efficiency of the proposed method, we perform extensive experimental comparisons of model performance, computational cost, memory usage and parameters on the PCN dataset. The results are shown in Table 6, which demonstrate that our PointAttN can achieve a good balance between the performance and the model cost. For model parameters, our method has only 3M more than SnowflakeNet(Xiang et al. 2021) but much fewer than the other methods. Besides the substantial performance improvement, our method still maintains a comparable high speed of 21.41 ms for inference time. Compared with the kNN-based methods, the memory usage of PointAttN is comparable to SnowflakeNet and only a quarter of PointTr(Yu et al. 2021), which requires huge memory for position embedding operations.

| Evaluation | Avg(CD) | Params(M) | Times(ms) | VRAM(G) |
|---|---|---|---|---|
| GRNet | 8.83 | 76.708 | 10.35 | 2.354 |
| SnowflakeNet | 7.21 | 19.318 | 13.45 | 1.886 |
| PointTr | 8.38 | 31.242 | 17.95 | 7.911 |
| PointAttN | 6.84 | 22.308 | 21.41 | 2.046 |

Table 6: Model complexity analysis on the PCN dataset.

## Conclusions

In this paper, we present a novel encoder-decoder framework, namely PointAttN, for point cloud completion. By fully employing cross-attention and self-attention mechanisms, our method have eliminated the need for explicit local region division like kNNs and directly established local- and long-range structural relationships of unordered points to perceive the geometry details and global context of 3D point clouds. The framework is neatly designed without any complicated operations. Extensive comparisons and ablation studies are conducted to demonstrate the superiority of our proposed PointAttN, which outperforms the state-of-the-art methods on many challenging benchmarks.

## Acknowledgements

This work was supported in part by the National Key R&D Program of China (No. 2022ZD0118700), the National Natural Science Foundation of China(No. 62102364, No. 62002325, No. 61825601, No. 62272235), and the Natural Science Foundation of Zhejiang Province (No. LY22F020016, No. LY22F030015).

## References

Alliegro, A.; Valsesia, D.; Fracastoro, G.; Magli, E.; and Tommasi, T. 2021. Denoise and Contrast for Category Agnostic Shape Completion. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4627–4636. https://doi.org/10.1109/CVPR46437.2021.00460.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-End Object Detection with Transformers. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 213–229. Springer. https://doi.org/10.1007/978-3-030-58452-8_13.

Charles, R. Q.; Su, H.; Kaichun, M.; and Guibas, L. J. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 77–85. https://doi.org/10.1109/CVPR.2017.16.

Dai, A.; Qi, C. R.; and Nießner, M. 2017. Shape Completion Using 3D-Encoder-Predictor CNNs and Shape Synthesis. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6545–6554. https://doi.org/10.1109/CVPR.2017.693.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint*. https://arxiv.org/pdf/2010.11929.pdf.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11): 1231–1237. https://doi.org/10.1177/0278364913491297.

Groueix, T.; Fisher, M.; Kim, V. G.; Russell, B. C.; and Aubry, M. 2018. A Papier-Mâché Approach to Learning 3D Surface Generation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 216–224. https://doi.org/10.1109/CVPR.2018.00030.

Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; and Shen, C. 2021a. Graph Attention Tracking. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9538–9547. https://doi.org/10.1109/CVPR46437.2021.00942.

Guo, M.-H.; Cai, J.-X.; Liu, Z.-N.; Mu, T.-J.; Martin, R. R.; and Hu, S.-M. 2021b. Pct: Point cloud transformer. *Comp. Visual Media*, 7(2): 187–199. https://doi.org/10.1007/s41095-021-0229-5.

Huang, T.; Zou, H.; Cui, J.; Yang, X.; Wang, M.; Zhao, X.; Zhang, J.; Yuan, Y.; Xu, Y.; and Liu, Y. 2021. RFNet: Recurrent Forward Network for Dense Point Cloud Completion. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 12488–12497. https://doi.org/10.1109/ICCV48922.2021.01228.

Huang, Z.; Yu, Y.; Xu, J.; Ni, F.; and Le, X. 2020. PF-Net: Point Fractal Network for 3D Point Cloud Completion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7659–7667. https://doi.org/10.1109/CVPR42600.2020.00768.

Lin, J.; Rickert, M.; Perzylo, A.; and Knoll, A. 2021. PCTMA-Net: Point Cloud Transformer with Morphing Atlas-based Point Generation Network for Dense Point Cloud Completion. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5657–5663. https://doi.org/10.1109/IROS51168.2021.9636483.

Pan, L.; Chen, X.; Cai, Z.; Zhang, J.; Zhao, H.; Yi, S.; and Liu, Z. 2021a. Variational Relational Point Completion Network. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8520–8529. https://doi.org/10.1109/CVPR46437.2021.00842.

Pan, X.; Xia, Z.; Song, S.; Li, L. E.; and Huang, G. 2021b. 3D Object Detection with Pointformer. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7459–7468. https://doi.org/10.1109/CVPR46437.2021.00738.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Sipiran, I.; Mendoza, A.; Apaza, A.; and Lopez, C. 2022. Data-Driven Restoration of Digital Archaeological Pottery with Point Cloud Analysis. *International Journal of Computer Vision*. https://doi.org/10.1007/s11263-022-01637-1.

Stutz, D.; and Geiger, A. 2018. Learning 3D Shape Completion from Laser Scan Data with Weak Supervision. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1955–1964. https://doi.org/10.1109/CVPR.2018.00209.

Sun, B.; Kim, V. G.; Aigerman, N.; Huang, Q.; and Chaudhuri, S. 2022. PatchRD: Detail-Preserving Shape Completion by Learning Patch Retrieval and Deformation. In *Computer Vision – ECCV 2022*, 503–522. ISBN 978-3-031-20062-5.

Tchapmi, L. P.; Kosaraju, V.; Rezatofighi, H.; Reid, I.; and Savarese, S. 2019. TopNet: Structural Point Cloud Decoder. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 383–392. https://doi.org/10.1109/CVPR.2019.00047.

Thomas, H.; Qi, C. R.; Deschaud, J.-E.; Marcotegui, B.; Goulette, F.; and Guibas, L. J. 2019. KPConv: Flexible and Deformable Convolution for Point Clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; Luxburg, U. V.;

Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wang, X.; Ang, M. H.; and Hee Lee, G. 2021. Voxel-based Network for Shape Completion by Leveraging Edge Generation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 13169–13178. https://doi.org/10.1109/ICCV48922.2021.01294.

Wang, X.; Ang, M. H.; and Lee, G. H. 2020. Cascaded Refinement Network for Point Cloud Completion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 787–796. https://doi.org/10.1109/CVPR42600.2020.00087.

Wang, Y.; Tan, D. J.; Navab, N.; and Tombari, F. 2020. SoftPoolNet: Shape Descriptor for Point Cloud Completion and Classification. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 70–85. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-58580-8_5.

Wang, Y.; Tan, D. J.; Navab, N.; and Tombari, F. 2022a. Learning local displacements for point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1568–1577.

Wang, Y.; Tan, D. J.; Navab, N.; and Tombari, F. 2022b. SoftPool++: An Encoder–Decoder Network for Point Cloud Completion. *International Journal of Computer Vision*, 130(5): 1145–1164. https://doi.org/10.1007/s11263-022-01588-7.

Wen, X.; Han, Z.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Liu, Y.-S. 2021a. Cycle4Completion: Unpaired Point Cloud Completion Using Cycle Transformation With Missing Region Coding. In *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Wen, X.; Li, T.; Han, Z.; and Liu, Y.-S. 2020. Point Cloud Completion by Skip-Attention Network With Hierarchical Folding. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1936–1945. https://doi.org/10.1109/CVPR42600.2020.00201.

Wen, X.; Xiang, P.; Han, Z.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Liu, Y.-S. 2021b. PMP-Net: Point Cloud Completion by Learning Multi-step Point Moving Paths. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7439–7448. https://doi.org/10.1109/CVPR46437.2021.00736.

Wen, X.; Xiang, P.; Han, Z.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Liu, Y.-S. 2022. PMP-Net++: Point Cloud Completion by Transformer-Enhanced Multi-step Point Moving Paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. https://doi.org/10.1109/TPAMI.2022.3159003.

Xiang, P.; Wen, X.; Liu, Y.-S.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Han, Z. 2021. SnowflakeNet: Point Cloud Completion by Snowflake Point Deconvolution with Skip-Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 5479–5489. https://doi.org/10.1109/ICCV48922.2021.00545.

Xie, H.; Yao, H.; Zhou, S.; Mao, J.; Zhang, S.; and Sun, W. 2020. GRNet: Gridding Residual Network for Dense Point Cloud Completion. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 365–381. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-58545-7_21.

Yan, X.; Yan, H.; Wang, J.; Du, H.; Wu, Z.; Xie, D.; Pu, S.; and Lu, L. 2022. Fbnet: Feedback network for point cloud completion. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, 676–693. Springer.

Yang, Y.; Feng, C.; Shen, Y.; and Tian, D. 2018. FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 206–215. https://doi.org/10.1109/CVPR.2018.00029.

Yu, X.; Rao, Y.; Wang, Z.; Liu, Z.; Lu, J.; and Zhou, J. 2021. PoinTr: Diverse Point Cloud Completion with Geometry-Aware Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 12478–12487. https://doi.org/10.1109/ICCV48922.2021.01227.

Yuan, W.; Khot, T.; Held, D.; Mertz, C.; and Hebert, M. 2018. PCN: Point Completion Network. In *2018 International Conference on 3D Vision (3DV)*, 728–737. https://doi.org/10.1109/3DV.2018.00088.

Zhang, W.; Dong, Z.; Liu, J.; Yan, Q.; Xiao, C.; et al. 2022. Point cloud completion via skeleton-detail transformer. *IEEE Transactions on Visualization and Computer Graphics*.

Zhang, W.; Yan, Q.; and Xiao, C. 2020. Detail Preserved Point Cloud Completion via Separated Feature Aggregation. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 512–528. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-58595-2_31.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P.; and Koltun, V. 2021. Point Transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 16239–16248. https://doi.org/10.1109/ICCV48922.2021.01595.

Zhou, H.; Cao, Y.; Chu, W.; Zhu, J.; Lu, T.; Tai, Y.; and Wang, C. 2022. Seedformer: Patch seeds based point cloud completion with upsample transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, 416–432. Springer.

Zhu, Z.; Chen, H.; He, X.; Wang, W.; Qin, J.; and Wei, M. 2023. SVDFormer: Complementing Point Cloud via Self-view Augmentation and Self-structure Dual-generator. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 14508–14518.