Building-PCC: Building Point Cloud Completion Benchmarks

Weixiao Gao[1],*Ravi Peters[2], Jantien Stoter[1]

[1] Delft University of Technology, The Netherlands - (w.gao-1,j.e.stoter)@tudelft.nl
[2] 3DGI, Zoetermeer, The Netherlands - ravi.peters@3dgi.nl

Keywords: Point Cloud Completion, Deep Learning, Benchmarks, Chamfer Distance.

Abstract

With the rapid advancement of 3D sensing technologies, obtaining 3D shape information of objects has become increasingly convenient. Lidar technology, with its capability to accurately capture the 3D information of objects at long distances, has been widely applied in the collection of 3D data in urban scenes. However, the collected point cloud data often exhibit incompleteness due to factors such as occlusion, signal absorption, and specular reflection. This paper explores the application of point cloud completion technologies in processing these incomplete data and establishes a new real-world benchmark Building-PCC dataset, to evaluate the performance of existing deep learning methods in the task of urban building point cloud completion. Through a comprehensive evaluation of different methods, we analyze the key challenges faced in building point cloud completion, aiming to promote innovation in the field of 3D geoinformation applications. Our source code is available at `https://github.com/tudelft3d/Building-PCC-Building-Point-Cloud-Completion-Benchmarks.git`.

1. Introduction

The rapid advancement of 3D sensing technology has simplified acquiring 3D object information, making data acquisition increasingly convenient. LiDAR, compared with stereo vision and structured light, can capture 3D information accurately from long distances (Giancola et al., 2018). For collecting urban data, LiDAR has been widely used (Wang et al., 2019), including airborne, terrestrial, and mobile laser scanning. However, these devices still face issues such as occlusion, signal absorption, reflection, and sensor resolution, which result in incomplete point clouds (Fei et al., 2022). Thus, completing these point clouds to restore full structures is crucial, especially for applications related to 3d geoinformation, such as 3D reconstruction (Nan and Wonka, 2017), semantic understanding (Song et al., 2017), change detection (Czerniawski et al., 2021), and autonomous driving (Bai et al., 2020).

The task of point cloud completion was first proposed in computer vision and graphics fields. Early research inferred the geometric properties of missing areas from observed regions using methods such as surface reconstruction (Berger et al., 2014), symmetry analysis (Mitra et al., 2006), or template matching (Nan et al., 2012). With the development of deep learning technology, methods based on voxels (Xie et al., 2020), encoders-decoders (Yuan et al., 2018), and transformers (Yu et al., 2021, Chen et al., 2023) have been proposed, aiming to recover the complete geometric shape of objects from incomplete data. These learning-based methods can capture higher-dimensional features without relying on prior assumptions, offering significant advantages over traditional methods.

However, despite these advancements, the practical application of these methods predominantly involves testing on CAD model datasets such as ShapeNet (Chang et al., 2015), revealing a gap in their evaluation within real-world contexts. Existing evaluations in real-world settings often concentrate on specific categories such as cars in datasets like KITTI (Geiger et al., 2012), lacking comprehensive evaluation for large-scale real-world scenes. To bridge this gap, this paper introduces real-world data benchmarks based on airborne LiDAR point clouds to improve testing scenarios for deep learning-based completion methods.

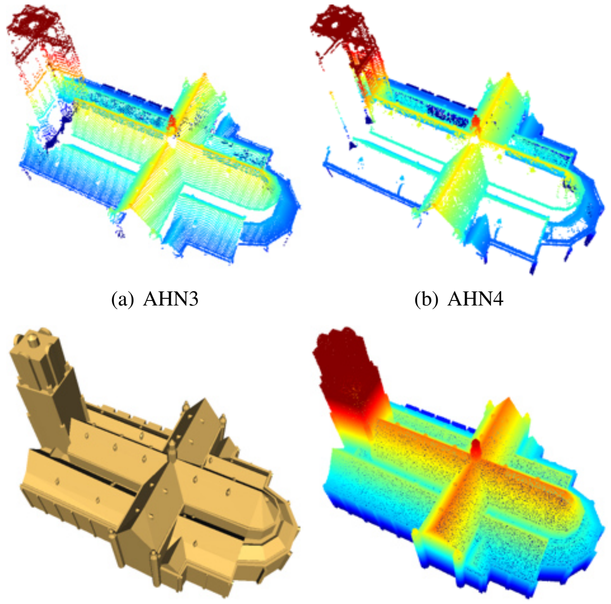

(a) AHN3
(b) AHN4
(c) LoD2 model
(d) Sampled points

Figure 1. (a) and (b) display partial point clouds captured by airborne LiDAR, corresponding to AHN3 and AHN4, respectively. (c) presents a manually created LoD2 building model, while (d) illustrates point clouds sampled from (c).

Airborne point cloud data, known for its high precision, is commonly used for building reconstruction in large-scale urban scenes. For instance, initiatives such as 3D BAG (Peters et al., 2022) and Helsinki3D (Helsinki, 2019) leverage airborne point cloud data as their foundational input, integrating this with

[*] Corresponding author

GIS-based building footprints to reconstruct building models. These models adhere to the CityGML standards (Kolbe et al., 2005), achieving detailed representations at either Level of Detail 1 (LoD1) or Level of Detail 2 (LoD2) (see Figure 1(c)). These models can be applied to various fields (Biljecki et al., 2015) such as urban planning, energy estimation, fluid dynamics simulation, and city management. However, due to factors such as tree occlusion, glass adsorption, and varying viewpoints, the collected data often have missing parts (see Figure 1(a) and 1(b)), which critically affect the quality of 3D reconstruction of the building models. Therefore, this research conducts a thorough evaluation of the performance of state-of-the-art, deep learning-based approaches in the task of completing building point clouds.

The objective of this paper is to introduce a new real-world benchmark dataset, namely Building-PCC, specifically designed for the task of building point cloud completion. We collected 50,000 building instances from two cities in the Netherlands, The Hague and Rotterdam, correlating each with two sets of airborne point cloud data, namely AHN3 and AHN4, alongside ground truth data sampled from manually reconstructed 3D building models. Compared to existing datasets, our dataset is based on real-world scenarios and presents a higher level of challenge owing to the intricate and diverse nature of these real-world settings. On this basis, this study comprehensively evaluates the existing deep learning-based 3D completion methods on our dataset. We delve into the critical challenges encountered in building point cloud completion from various perspectives, with the goal of fostering innovation within the realm of 3D geoinformation applications.

2. Related work

This section begins with a comprehensive overview of the datasets frequently employed in 3D point cloud completion tasks. It then proceeds to offer an in-depth exploration of deep learning-based techniques for achieving 3D shape completion.

2.1 Point cloud completion datasets

Point cloud completion tasks leverage datasets that can be categorized into two primary types: synthetic and real-world scene datasets. Among these, synthetic datasets such as ShapeNet (Chang et al., 2015), ModelNet (Vishwanath et al., 2009), and Completion3D (Tchapmi et al., 2019) are extensively utilized. These datasets consist of Computer-Aided Design (CAD) models that span a diverse array of object categories, including but not limited to airplanes, ships, furniture, and electronic devices. The generation of complete point clouds within these datasets involves either uniform sampling or Poisson disk sampling techniques applied to the model surfaces, standardizing the maximum resolution at 16,384 sampling points per model (Yuan et al., 2018). To obtain incomplete point coulds, a process of simulating 2.5D depth images from various viewpoints and subsequently back-projecting these images into the three-dimensional space is employed to achieve sampling.

In the realm of real-world scene datasets, the KITTI (Geiger et al., 2012) dataset stands out as a primary resource. Acquired using Mobile Laser Scanning (MLS) technology across actual road environments, KITTI was initially crafted for the purpose of autonomous driving research. It encompasses 22 point cloud sequences and 11 semantically annotated categories. Within the scope of point cloud completion tasks, the vehicle category frequently serves as the chosen subject for testing. The inherent sparsity of point clouds within the KITTI dataset often results in the geometric structure of the captured object being incomplete. As a consequence, a common approach among researchers is to initially train their models on synthetic datasets featuring car models. Subsequently, they utilize the incomplete vehicle point clouds from the KITTI dataset, performing uniform sampling to create the requisite datasets for further training and predictive analysis.

It is important to highlight that all datasets undergo a normalization process before being fed into the point cloud completion network. This is a crucial step designed to mitigate the adverse effects of outlier samples. Specifically for KITTI data, each vehicle is marked with a bounding box that provides a reference coordinate system for normalization. While most test datasets, aside from KITTI car datasets, consist of synthetic data, they often fall short in capturing the diversity found in real-world settings, particularly within urban scenes. To address this shortfall, this research introduces a comprehensive airborne point cloud completion benchmark dataset tailored for buildings, thereby aiming to bridge this notable gap.

2.2 3D shape completion

In the domain of 3D shape completion, methods leveraging deep learning are primarily categorized into three strategic approaches: point cloud-based, multimodality-based, and those employing Generative Adversarial Networks (GANs).

The point cloud-based completion approach is intricately divided into three distinct methodologies: 3D convolution-based methods, encoder-decoder frameworks, and anchor-based strategies. Techniques rooted in 3D convolution typically require the construction of voxel grids or distance fields (Xie et al., 2020), offering high accuracy at the cost of significant computational resources and challenges in capturing the intricate details of objects. Encoder-decoder-based methods (Yuan et al., 2018) adeptly extract both global and local 3D features from the point clouds using the encoder, while the decoder refines the prediction of the object's complete point cloud from a coarse to a fine resolution. The anchor-based approach (Chen et al., 2023, Yu et al., 2023), on the other hand, determines the positions of anchor points by learning the local features of the input, followed by an upsampling process that leverages these anchor points to reconstruct the object's full shape, integrating both local and global features. This method may also incorporate attention mechanisms to further boost the performance of the network. This research is dedicated to examining the performance of these methods in achieving accurate completion of building point clouds.

Multimodality-based methods (Zhang et al., 2021) leverage image data from single or multiple views combined with partial point clouds for completion. These images can provide auxiliary information about the global shape of the object. Notably, such images usually have a pure white background, containing only the target object. However, the complex nature of urban environments poses significant challenges in acquiring object instances paired with both registered point clouds and segmented image data, crucial for constructing comprehensive multimodality datasets for completion.

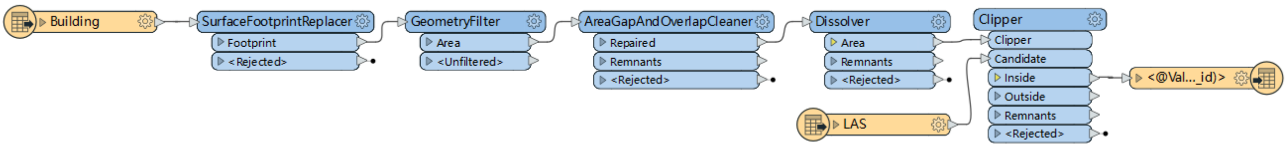


Figure 2. FME workflow for extracting building point cloud instances.

Currently, the performance improvement of multimodality methods in this area is limited. Therefore, this paper focuses on exploring point cloud-based methods for completing the shape of buildings.

GAN-based approaches (Xie et al., 2021) leverage the discriminator's implicit learning capability to evaluate the accuracy of complete point clouds produced by the generator. This technique may integrate the previously mentioned encoder-decoder architecture or utilize multimodality data. While capable of generating complete predictions, the outcomes may present multiple interpretations, potentially resulting in point clouds of the same object but with varying styles or details. Given the stringent accuracy requirements for 3D data in disciplines such as geoinformation and geodesy, GAN-based techniques may not always be the optimal choice for completing building point clouds. This is due to the potential for the generated results to diverge from the true geometric configuration of the original structures.

3. The Building-PCC datasets

In this section, we detail the construction process of the Building-PCC dataset. Our goal is to pair individual building models with their sampling points and corresponding partial point clouds of the actual scanned buildings. In subsection 3.1, we discuss the specifics of the research area and data collection. In subsection 3.2, we showcase our approach for extracting complete building instances from both point clouds and models, as well as conducting point cloud sampling.

3.1 Data collection

We selected The Hague and Rotterdam in the Netherlands as our main research areas, which together host approximately 350,000 buildings of various types. The city of The Hague provides LoD2 building models as open data based on the CityGML standard, from which we collected data for the year 2022. Similarly, the city of Rotterdam also provided LoD2 building models in DWG (from AutoCAD) and SKP (from SketchUp) formats. It is worth mentioning that these building models were semi-automatically constructed by professional modelers using 3D modeling software and GIS data, offering accuracy superior to automatic reconstruction methods. We randomly selected about 50,000 building models from these two cities as our sample, each model containing a unique BAG identifier, the identifier used in the Dutch national geographic database to uniquely identify a building.

For point cloud data, we relied on the Actueel Hoogtebestand Nederland (AHN), a nationwide elevation dataset of the Netherlands. The AHN dataset, collected via LiDAR technology, provides accurate terrain and elevation information. We used data from both AHN3 and AHN4 versions, where AHN3 offers a point density of approximately 8 points per square meter, and AHN4 increases the point density to approximately 10 points per square meter, providing more detailed elevation maps. It is important to note that although AHN3 and AHN4 offer rich information, they do not contain segmentation information for building instances.

3.2 Building instance extraction

Our goal is to extract individual building models for sampling and match them with corresponding real scans. By utilizing the BAG identifiers mentioned in Section 3.1, we are able to precisely extract individual buildings from the LoD2 building models discussed earlier. Using FME software, we achieved this process and converted all models to obj format for sampling. These models may contain holes, inner faces, or might be non-manifold or non-watertight, which could affect the quality and precision of subsequent point cloud sampling. Therefore, we used Easy3d (Nan, 2021), an open-source libraries, to automatically correct these errors. To obtain the ground truth, we employed Poisson Disk Sampling and fixed the number of sampling points at 16,384 per building model. Compared to Uniform Sampling, Poisson Disk Sampling generates a more natural and irregular distribution of sampling points, avoiding the aggregation of sample points and ensuring the randomness and uniformity of sampling.

For AHN point clouds, considering that manually segmenting each building is very time-consuming and automatic segmentation methods based on machine learning require a large amount of ground truth annotation data, we implemented a building instance segmentation pipeline based on LoD2 models (see Figure 2). First, we used the Overhead Shadow technique to project 3D buildings to extract each building model's footprint. Then, we corrected topological errors such as gaps and overlaps in the footprint and dissolved footprints with common boundaries. Finally, we projected the point cloud to 2D and clipped it using the generated footprint to retain points within the footprint. Thus, we acquired complete point clouds for each building, along with two types of partial point clouds derived from AHN3 and AHN4 datasets, to evaluate the performance of point cloud completion methods.

4. Benchmarks

In this section, we embark on an in-depth analysis of the performance of representative deep learning methods applied to the Building-PCC dataset.

4.1 Implementation details

In this study, we deployed an experimental setup equipped with an AMD Ryzen Threadripper 1920X 12-core processor, 32GB RAM, and an Nvidia Geforce RTX 3090 graphics card with 24GB of VRAM. All test networks were developed and implemented using the Pytorch framework. We randomly selected approximately 50,000 buildings as the subject of our study, with 30,000 used to construct the training set and 10,000 each allocated to the validation and test sets, supporting model training and performance evaluation. During the training phase, for each building, we randomly choose either an incomplete point cloud from the corresponding AHN3 or AHN4 datasets to serve as the input. Additionally, the point cloud data for each building underwent normalization processing.

4.2 Representative baselines

To comprehensively assess the performance of existing point cloud completion pipelines on real-world dataset, we carefully selected eight representative methods as the solid baseline for the Building-PCC benchmark. These baselines encompass the point cloud-based deep learning strategies discussed in Section 2.2. A concise overview of these baseline methods is as follows:

- **PCN**: As the first architecture in the field of point cloud completion to introduce neural networks for direct processing of 3D coordinates, PCN's encoder transforms the input data into a high-dimensional global feature vector, while its decoder employs the concept of a folding network, simulating the deformation of a two-dimensional plane to map 2D grid points into 3D space (Yuan et al., 2018).
- **FoldingNet**: This end-to-end deep autoencoder was originally designed to tackle the challenges of unsupervised learning on point clouds. Its core mechanism folds a 2D grid onto the underlying surface of a 3D object. For point cloud completion tasks, FoldingNet utilizes an encoder similar to PCN's, but introduces a folding-based decoder (Yang et al., 2017).
- **TopNet**: Proposes a decoder based on a tree topology structure, aimed at constructing structured data by generating subsets of the point cloud, facilitating the completion process (Tchapmi et al., 2019).
- **GRNet**: Introduces a 3D grid as an intermediate representation to regularize unordered point clouds, enhancing the structure and contextual information of the point cloud through a gridding residual network to achieve complete cloud reconstruction (Xie et al., 2020).
- **SnowflakeNet**: Simulates the process of completing point clouds as a snowflake-like growth in space, using multi-layer snowflake points for deconvolution to generate locally compact and structured data with high-detail geometries (Xiang et al., 2021).
- **PoinTr**: Transforms the point cloud completion task into a set-to-set translation task, employing a transformer to precisely model the local geometric relations within the structure, addressing the inductive biases in 3D geometry (Yu et al., 2021).
- **AnchorFormer**: Introduces the use of anchors to dynamically capture the regional information of objects. These anchors are dispersed to observed and unobserved positions by estimating specific offsets, forming a sparse set with downsampled points from the input observation data, and finally reconstructing fine-grained 3D structures through a point morphing scheme (Chen et al., 2023).
- **AdaPoinTr**: Building on PoinTr, it adds an adaptive query generation mechanism and a denoising module to effectively deal with issues such as discontinuous and uneven appearance (Yu et al., 2023).

These methods showcase the diversity and innovation within the point cloud completion technology, providing a solid foundation for understanding and advancing the field.

4.3 Evaluation metrics

Our evaluation methodology is aligned with previous studies (Yu et al., 2023), employing the mean Chamfer Distance (CD) with the L1-norm ($CD\text{-}l_1$) as the primary metric to quantify the differences between the predicted building point cloud P and the ground truth G as defined in Equation (1). Note that a lower $CD\text{-}l_1$ value indicates smaller differences between the predicted data and the ground truth, which corresponds to better performance.

$$d_{CD}(P,G) = \frac{1}{|P|}\sum_{p\in P}\min_{g\in G}\|p-g\| + \frac{1}{|G|}\sum_{g\in G}\min_{p\in P}\|g-p\| \quad (1)$$

Additionally, we incorporate the F-Score as an alternate evaluation metric. The precision $P(d)$, recall $R(d)$, and $F\text{-}Score(d)$ for a point cloud completion at a given threshold d are defined as Equations (2) to (4). In our experimental setup, we establish the threshold d at 1%.

$$P(d) = \frac{1}{|P|}\sum_{p\in P}\left[\min_{g\in G}\|p-g\| < d\right] \quad (2)$$

$$R(d) = \frac{1}{|G|}\sum_{g\in G}\left[\min_{p\in P}\|g-p\| < d\right] \quad (3)$$

$$F\text{-}Score(d) = \frac{2P(d)R(d)}{P(d)+R(d)} \quad (4)$$

4.4 Results

In this section, we showcase the outcomes of the methods discussed in Section 4.2 on the Building-PCC dataset, detailed in Table 1 and Figure 3 for quantitative and qualitative analyses, respectively. For the quantitative aspect, 10,000 partial building point clouds from each of the AHN3 and AHN4 datasets served as input for evaluating the overall performance of each method. As delineated in Table 1's left section, AnchorFormer, PoinTr, and AdaPoinTr significantly outperform others in terms of average $CD\text{-}l_1$, with PoinTr leading at an impressive 1.40. These methods also show robustness across AHN3 and AHN4 inputs, maintaining consistent average Chamfer Distances.

AdaPoinTr, with its advanced denoising module, excels on the $F\text{-}Score$ metric, achieving 0.679 and 0.725 on AHN3 and AHN4, respectively, edging out the runner-up, AnchorFormer, by approximately 0.048 and 0.040. Overall, AdaPoinTr stands out in comprehensive performance, closely following PoinTr in average $CD\text{-}l_1$ by less than 0.02, yet far surpasses other methods in $F\text{-}Score$.

To further illustrate these models' capabilities, we conducted tests on six representative buildings for each method, presenting these findings on the right section of Table 1 and in Figure 3. The right section of Table 1 echoes the quantitative analysis

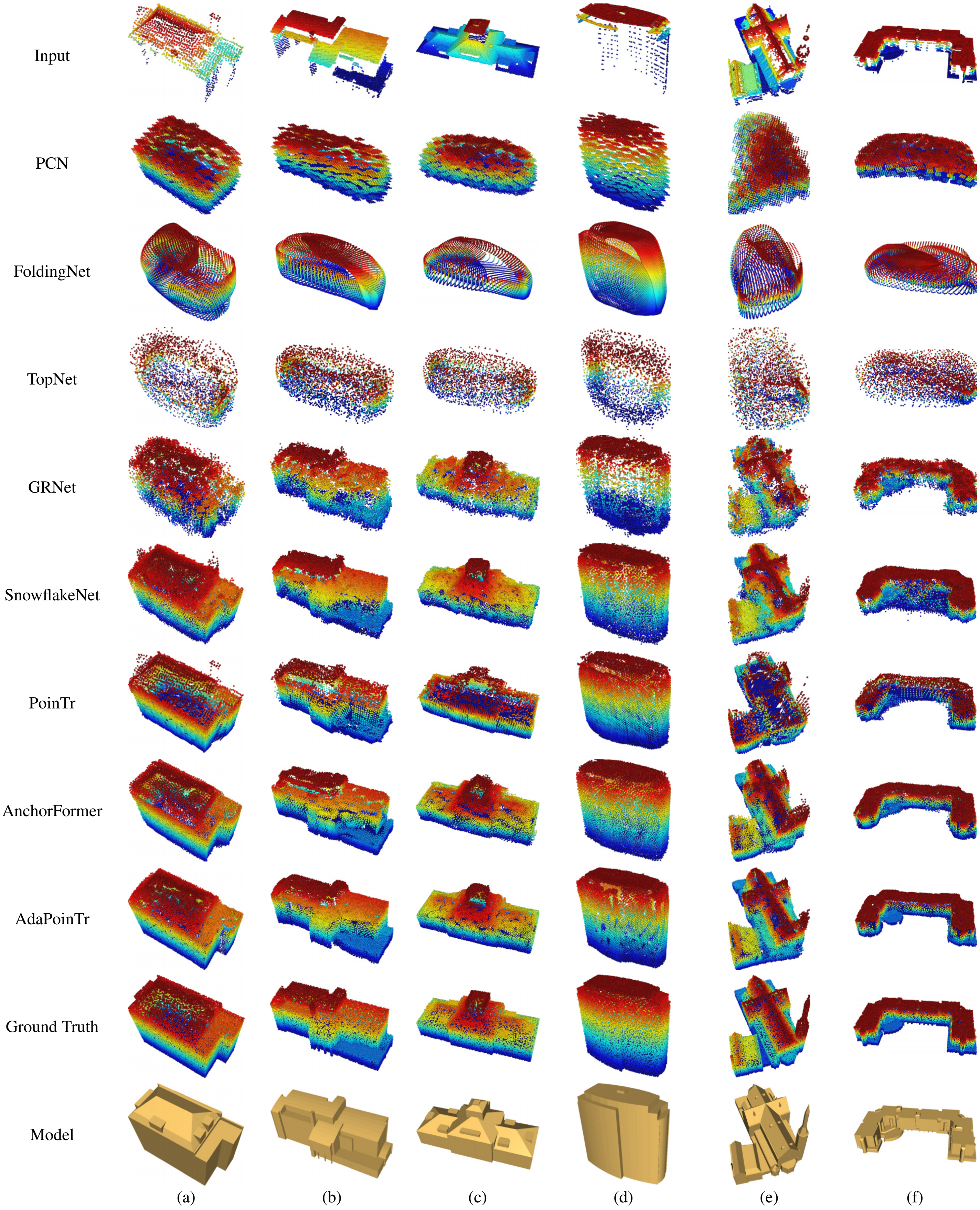


Figure 3. Six visual examples of point cloud completion results by different approaches on the Building-PCC dataset. The color coding, ranging from blue to red, represents the height field, with blue indicating lower elevations and red signifying higher elevations.

AHN3/4 Methods	$CD\text{-}l_1$	$F\text{-}Score$	$CD\text{-}l_1$	$F\text{-}Score$	(a)	(b)	(c)	(d)	(e)	(f)	$CD\text{-}l_1$
PCN	6.09	0.374	6.14	0.416	14.21	14.36	14.56	11.60	11.77	12.98	13.25
FoldingNet	4.20	0.328	5.63	0.352	10.31	15.47	16.03	15.60	15.04	14.98	14.57
TopNet	6.42	0.262	6.44	0.267	14.61	12.80	12.67	11.24	11.38	15.56	13.04
GRNet	5.21	0.399	4.89	0.463	10.90	16.39	16.83	14.56	14.86	16.29	14.97
SnowflakeNet	6.61	0.588	6.60	0.629	16.24	20.92	21.08	15.88	15.91	16.53	17.76
PoinTr	**1.40**	0.511	**1.40**	0.585	**2.81**	**5.00**	5.21	**7.58**	7.99	**11.39**	**6.66**
AnchorFormer	1.46	0.631	1.46	0.685	2.89	5.26	5.29	7.82	**7.91**	11.70	6.81
AdaPoinTr	1.42	**0.679**	1.41	**0.725**	2.87	5.17	**5.18**	7.75	8.04	11.66	6.78

Table 1. Comparative evaluation of point cloud completion methods on Building-PCC datasets. The left section quantifies the overall performance across the entire test dataset, with inputs derived separately from AHN3 and AHN4 partial point clouds (with 10,000 samples respectively) , using the mean L1 Chamfer Distance ($CD\text{-}l_1$) and mean $F\text{-}Score$ metrics. The right section details the performance for selected individual building samples, referenced in Figure 3, providing specific $CD\text{-}l_1$ values for each sample labeled (a) to (f). The last column displays the average $CD\text{-}l_1$ across all samples. Optimal performances are denoted in bold for emphasis.

based on $CD\text{-}l_1$, affirming that AnchorFormer, PoinTr, and AdaPoinTr maintain superior performance, particularly PoinTr. Figure 3 reveals that while PCN, FoldingNet, and TopNet exhibit notable deviations from the ground truth, the rest successfully reconstructs the overall geometric structure of the buildings. Among these, AdaPoinTr and SnowflakeNet distinctly excel in capturing building details. Nonetheless, the predicted point clouds of all methods exhibit varying degrees of incompleteness in the results and loss of sharp features and fine details compared to the ground truth. These shortcomings might limit their use in downstream applications. Therefore, in Section 5, we elaborate on the specific reasons for these issues and potential solutions.

5. Challenges

In this section, we delve into challenges faced by point cloud completion methods when dealing with real-world building point clouds. These challenges include imbalanced datasets, limitations of completion structures, and issues of point cloud normalization. Through detailed analysis, we aim to provide insights for researchers on how to effectively utilize point cloud completion methods in practical applications.

5.1 Impact of imbalanced datasets

In classification or semantic segmentation tasks, data imbalance manifests as a discrepancy in sample numbers across categories within the training dataset. This results in models performing better on categories with more samples and worse on underrepresented categories, affecting the overall model performance. A strategy to counter this, is data augmentation to balance sample numbers across categories, preventing model overfitting to dominant data categories. In point cloud completion tasks, while no data imbalance issue from semantic labels exists, an uneven distribution of local geometric structures within point clouds does. Datasets like ShapeNet can augment data by increasing point clouds for objects of the same category in different styles. But for real-world building point clouds, incomplete areas are mostly on the bottoms and facades, not on the roofs. This results in models more easily completing geometric structures of these areas, causing inaccuracies in predicting complete geometric structures of missing roof parts (see Figure 4).

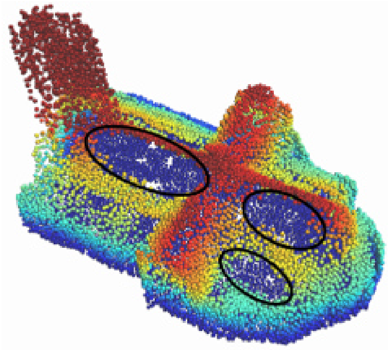


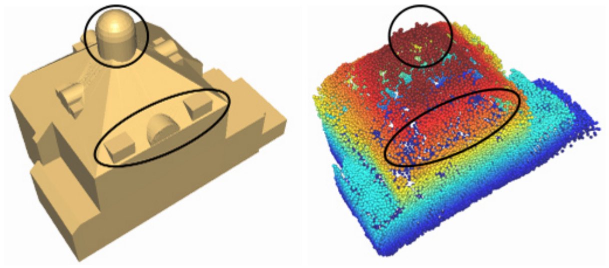
Figure 4. AdaPoinTr prediction example using an AHN4 partial point cloud as input (refer to Figure 1(b)). The area within the black circle highlights the roof parts that were not successfully predicted. The color coding from blue to red indicates elevation from low to high.

In response to these issues, we propose two potential research directions. First, we can create synthetic datasets for building point cloud completion, such as using tools like HELIOS++ (Winiwarter et al., 2022) to simulate laser scans of large-scale virtual urban scenes, including various urban objects like vegetation, street furniture, and building models, to generate building point clouds. By controlling the quantity, location, and attributes of different categories of objects, we can produce a diverse set of incomplete building point clouds. Alternatively, direct simulation scans of LoD2 building models to obtain representative incomplete data can be performed, with random deletion of points to simulate incompleteness. The second approach involves the use of multimodal data, such as orthophotos or multi-view images. Although collecting and matching these data types can be challenging, in theory, the photo-consistency of images can assist in better perceiving the geometric structural consistency.

5.2 Limitations on fine details

In Section 4.2, we discussed representative methods, but these generally exhibit loss of details and over-smoothed sharp features when predicting complete building point clouds. These issues often lead to reduced accuracy of the point cloud, thereby affecting the quality of downstream applications such as point cloud-based building model reconstruction.

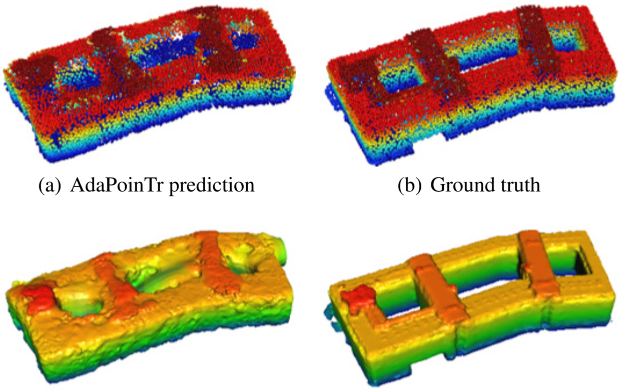
Detail loss refers to algorithms' inability to accurately reproduce small building components, like chimneys, dormers, and roof decorations, which are relatively small geometric structures (see Figure 5). The primary reason is the low resolution of predicted point clouds, i.e., a fixed number of predicted points (such as 16,384 points per building),



(a) LoD2 building model (b) AdaPoinTr prediction

Figure 5. An example of details loss in the predict building point cloud using AdaPoinTr. The black circle emphasizes the absence of detailed roof superstructures.

insufficient to cover large-scale buildings with many details comprehensively. Although increasing the number of predicted points or predicting only missing parts can mitigate this issue, the number of points cannot be indefinitely increased due to memory limitations. Another strategy is to decompose large building point clouds into subsets, predict complete structures for subsets, and then stitch predictions together.



(a) AdaPoinTr prediction (b) Ground truth

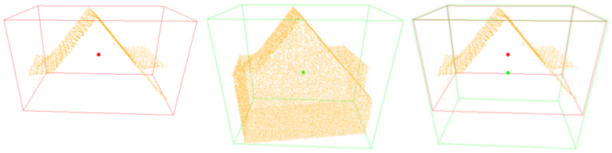
(c) AdaPoinTr mesh (d) Ground truth mesh

Figure 6. An example of over-smoothed sharp features. The Poisson surface reconstruction is used to extract surface mesh from the point clouds. The color transition from blue to red signifies a shift from lower to higher elevation levels.

As for the over-smoothed sharp features, they refer to the excessively smoothed sharp corners or sharp edges of buildings. These sharp features primarily arise at the intersections of piece-wise planar surfaces of buildings, such as where the facade meets the roof. As illustrated in Figure 6, we employed Poisson surface reconstruction on both the AdaPoinTr-predicted point clouds and the ground truth data to highlight the disparities in sharp features. It is evident that the mesh reconstructed from AdaPoinTr-predicted point clouds exhibits more over-smoothed areas compared to the ground truth mesh. The over-smoothed sharp features is primarily due to the fact that the loss functions of these methods are mostly based on the Chamfer Distance, which is a point-to-point distance measure. For urban buildings that conform to the Manhattan world assumption, this measure cannot ensure that the distance error from each point to the local plane is minimized. Therefore, we suggest incorporating the consideration of the distance from points to local planes into the loss function, aiming to address the issue of over-smoothed sharp features to a certain extent.

5.3 Normalization issues

Although normalization is widely regarded as a means to speed up training, enhance model convergence, and improve generalization capabilities, it has become a problematic issue in the field of point cloud completion. This is because existing methods typically normalize partial point clouds by aligning them with the ground truth, adjusting the partial point clouds based on the center and scale of the ground truth. However, in practical applications, especially for building point cloud completion tasks, this method is not always feasible, as not all buildings have readily available ground truth or models for reference. If the partial point clouds cannot be accurately aligned with the ground truth, it may lead to failures in model training or prediction. Figure 7 illustrates the disparities in center position and scale between the partial point cloud and the ground truth point cloud.



(a) (b) (c)

Figure 7. An example of normalization in point cloud completion: (a) displays the partial point cloud with its center point and bounding box outlined in red. (b) presents the ground truth point cloud, marked with a center point and bounding box in green. (c) overlays (a) with the center point and bounding box from (b), emphasizing the disparities in center position and scale between the partial and ground truth point clouds.

To address this issue, a potential strategy is to develop a network capable of capturing the global features of point clouds, thereby predicting the center position and scale of the point clouds. Alternatively, we could consider relying on external data resources, such as GIS data, to assist in the normalization process. These approaches have the potential to establish a more effective and robust normalization framework for the task of point cloud completion.

6. Conclusions

This study established the real-world benchmark Building-PCC dataset and evaluated the performance of current deep learning-based point cloud completion technologies in building point cloud completion. The results demonstrate that, despite deep learning methods excelling in capturing local and global geometric features, the quality of results in real-world scenarios is poor and fails to meet downstream applications' needs. By comparing different technologies on the dataset, we revealed the impact of dataset imbalance, limitations on details, and normalization issues on point cloud completion. Finally, we propose solutions to these challenges, including developing networks that capture global features, improving loss functions, and utilizing external data to assist with imbalance and normalization, offering new directions for future research.

References

Bai, L., Zhao, Y., Elhousni, M., Huang, X., 2020. DepthNet: Real-time LiDAR point cloud depth completion for autonomous vehicles. *IEEE Access*, 8, 227825–227833.

Berger, M., Tagliasacchi, A., Seversky, L. M., Alliez, P., Levine, J. A., Sharf, A., Silva, C. T., 2014. State of the art in surface reconstruction from point clouds. *35th Annual Conference of the European Association for Computer Graphics, Eurographics 2014-State of the Art Reports*, The Eurographics Association.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4), 2842–2889.

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H. et al., 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

Chen, Z., Long, F., Qiu, Z., Yao, T., Zhou, W., Luo, J., Mei, T., 2023. Anchorformer: Point cloud completion from discriminative nodes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13581–13590.

Czerniawski, T., Ma, J. W., Leite, F., 2021. Automated building change detection with amodal completion of point clouds. *Automation in construction*, 124, 103568.

Fei, B., Yang, W., Chen, W.-M., Li, Z., Li, Y., Ma, T., Hu, X., Ma, L., 2022. Comprehensive review of deep learning-based 3d point cloud completion processing and analysis. *IEEE Transactions on Intelligent Transportation Systems*.

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 3354–3361.

Giancola, S., Valenti, M., Sala, R., 2018. *A survey on 3D cameras: Metrological comparison of time-of-flight, structured-light and active stereoscopy technologies*. Springer.

Helsinki, C., 2019. Helsinki's 3D city models. `https://www.hel.fi/helsinki/en/administration/information/general/3d`. Accessed: 2020-11-25.

Kolbe, T. H., Gröger, G., Plümer, L., 2005. Citygml: Interoperable access to 3d city models. *Geo-information for disaster management*, Springer, 883–899.

Mitra, N. J., Guibas, L. J., Pauly, M., 2006. Partial and approximate symmetry detection for 3d geometry. *ACM Transactions on Graphics (ToG)*, 25(3), 560–568.

Nan, L., 2021. Easy3D: a lightweight, easy-to-use, and efficient C++ library for processing and rendering 3D data. *Journal of Open Source Software*, 6(64), 3255.

Nan, L., Wonka, P., 2017. Polyfit: Polygonal surface reconstruction from point clouds. *Proceedings of the IEEE International Conference on Computer Vision*, 2353–2361.

Nan, L., Xie, K., Sharf, A., 2012. A search-classify approach for cluttered indoor scene understanding. *ACM Transactions on Graphics (TOG)*, 31(6), 1–10.

Peters, R., Dukai, B., Vitalis, S., van Liempt, J., Stoter, J., 2022. Automated 3D reconstruction of LoD2 and LoD1 models for all 10 million buildings of the Netherlands. *Photogrammetric Engineering & Remote Sensing*, 88(3), 165–170.

Song, S., Yu, F., Zeng, A., Chang, A. X., Savva, M., Funkhouser, T., 2017. Semantic scene completion from a single depth image. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1746–1754.

Tchapmi, L. P., Kosaraju, V., Rezatofighi, H., Reid, I., Savarese, S., 2019. Topnet: Structural point cloud decoder. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 383–392.

Vishwanath, K. V., Gupta, D., Vahdat, A., Yocum, K., 2009. Modelnet: Towards a datacenter emulation environment. *2009 IEEE Ninth International Conference on Peer-to-Peer Computing*, IEEE, 81–82.

Wang, Y., Chen, Q., Zhu, Q., Liu, L., Li, C., Zheng, D., 2019. A survey of mobile laser scanning applications and key techniques over urban areas. *Remote Sensing*, 11(13), 1540.

Winiwarter, L., Pena, A. M. E., Weiser, H., Anders, K., Sánchez, J. M., Searle, M., Höfle, B., 2022. Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. *Remote Sensing of Environment*, 269, 112772.

Xiang, P., Wen, X., Liu, Y.-S., Cao, Y.-P., Wan, P., Zheng, W., Han, Z., 2021. Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer. *Proceedings of the IEEE/CVF international conference on computer vision*, 5499–5509.

Xie, C., Wang, C., Zhang, B., Yang, H., Chen, D., Wen, F., 2021. Style-based point generator with adversarial rendering for point cloud completion. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4619–4628.

Xie, H., Yao, H., Zhou, S., Mao, J., Zhang, S., Sun, W., 2020. Grnet: Gridding residual network for dense point cloud completion. *European Conference on Computer Vision*, Springer, 365–381.

Yang, Y., Feng, C., Shen, Y., Tian, D., 2017. Foldingnet: Interpretable unsupervised learning on 3d point clouds. *arXiv preprint arXiv:1712.07262*, 2(3), 5.

Yu, X., Rao, Y., Wang, Z., Liu, Z., Lu, J., Zhou, J., 2021. Pointr: Diverse point cloud completion with geometry-aware transformers. *Proceedings of the IEEE/CVF international conference on computer vision*, 12498–12507.

Yu, X., Rao, Y., Wang, Z., Lu, J., Zhou, J., 2023. AdaPoinTr: Diverse Point Cloud Completion with Adaptive Geometry-Aware Transformers. *arXiv preprint arXiv:2301.04545*.

Yuan, W., Khot, T., Held, D., Mertz, C., Hebert, M., 2018. Pcn: Point completion network. *2018 international conference on 3D vision (3DV)*, IEEE, 728–737.

Zhang, X., Feng, Y., Li, S., Zou, C., Wan, H., Zhao, X., Guo, Y., Gao, Y., 2021. View-guided point cloud completion. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15890–15899.