Point-PC: Point Cloud Completion Guided by Prior Knowledge via Causal Inference

Anonymous

Abstract

Point cloud completion aims to recover raw point clouds captured by scanners from partial observations caused by occlusion and limited view angles. Many approaches utilize a partial-complete paradigm in which missing parts are directly predicted by a global feature learned from partial inputs. This makes it hard to recover details because the global feature is unlikely to capture the full details of all missing parts. In this paper, we propose a novel approach to point cloud completion called Point-PC, which uses a memory network to retrieve shape priors and designs an effective causal inference model to choose missing shape information as supplemental geometric information to aid point cloud completion. Specifically, we propose a memory operating mechanism where the complete shape features and the corresponding shapes are stored in the form of "key-value" pairs. To retrieve similar shapes from the partial input, we also apply a contrastive learning-based pre-training scheme to transfer features of incomplete shapes into the domain of complete shape features. Moreover, we use backdoor adjustment to get rid of the confounder, which is a part of the shape prior that has the same semantic structure as the partial input. Experimental results on the ShapeNet-55, PCN, and KITTI datasets demonstrate that Point-PC performs favorably against the state-of-the-art methods.

1 Introduction

With more people using 3D scanners and RGB-D cameras, 3D vision has become one of the most popular topics for research in recent years [Han *et al.*, 2019; Han *et al.*, 2017; Han *et al.*, 2018a; Han *et al.*, 2018b]. Among all the 3D descriptors [Wang *et al.*, 2018; Xie *et al.*, 2020a; Qi *et al.*, 2017; Park *et al.*, 2019], the point cloud stands out because of its remarkable ability to render spatial structure at a lower computational cost. However, due to occlusion, view angles, and limitations of sensor resolution, raw point clouds are usually sparse and defective [Wen *et al.*, 2021; Wen *et al.*, 2020; Wen *et al.*, 2022]. Consequently, point cloud completion becomes essential.

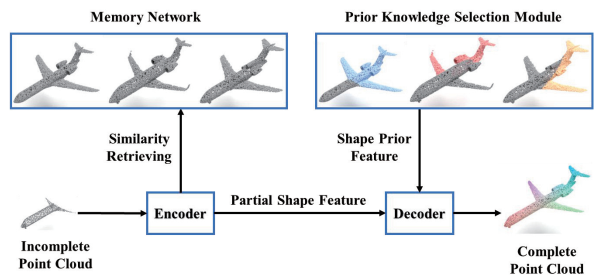

Figure 1: Point-PC is proposed for point cloud completion. Point-PC proposes a novel paradigm that finds similar shape information as prior knowledge to help the model handle the point cloud completion problem. Furthermore, our approach also selects geometric information from shape priors (blue, red, and yellow points) guided by causal inference.

Benefiting from large-scale point cloud datasets [Chang *et al.*, 2015; Yuan *et al.*, 2018; Geiger *et al.*, 2013], massively efficient learning-based methods for point cloud completion have emerged. The pioneering work is PCN [Yuan *et al.*, 2018] which encoded the input shape into a global feature and decoded it using a folding operation. Following an encoder-decoder pattern, several methods such as NSFA [Zhang *et al.*, 2020b] and GRNet [Xie *et al.*, 2020b] have emerged. Later work focuses on the decoding part of making point clouds with more geometric structures. SA-Net [Wen *et al.*, 2020] and PFNet [Huang *et al.*, 2020] increased the density of point clouds hierarchically. Such a coarse-to-fine pattern achieves better performance since more constraints are imposed on the generation process.

Most recent methods incorporate geometry-aware modules into a transformer-based structure. PoinTr [Yu *et al.*, 2021] used a KNN model to facilitate transformers, which can better leverage the inductive bias about 3D geometric structures. CompleteDT [Li *et al.*, 2022] integrated useful local information into the generation operations by enhancing the correlation of neighboring points in the proposed dense augment inference transformers. These two methods formulate the point cloud completion task as a set-to-set translation task, where complex dependency is learned among the point groups. Many approaches used the same framework to handle the point cloud completion problem [Li *et al.*, 2022; Zhang *et al.*, 2022; Cao *et al.*, 2022]. However, there are two drawbacks to the paradigm: 1) An incomplete shape is hard to learn detailed structure information and build a clear relationship between the complete point cloud model; 2) A global feature like this is spread out and does not keep much fine-grained information for the up-sample phase. Because of this, geometry-aware models can not learn complex structures if they know less about geometry.

To deal with this problem, we propose a new memory-based framework for completing point clouds (Point-PC). This framework uses a memory network to get shape priors and an effective causal inference model to choose missing shape information as additional geometric information to help complete point clouds. First, we construct an operating strategy to store, write, and read the memory. Specifically, we store the memory in a "key-value" pair. The key can be updated according to the similarity between the value and the corresponding ground truth. In order to achieve the best prior knowledge information, we construct a causal graph to remove the unrelated shape information of prior shapes. The obtained causal graph model removes the partial shape information and it only saves the missing structural information to help the final decoder obtain a more complete point cloud. Experimental results on the ShapeNet-55, PCN, and KITTI datasets demonstrate that Point-PC performs favorably against the state-of-the-art methods.

The main contributions of our work are as follows:

- We propose a novel memory-based 3D point cloud completion network, Point-PC, to supplement geometric information explicitly from prior knowledge.
- We introduce causal inference to further refine the shape prior, so as to eliminate the distraction of irrelevant information.
- We apply qualitative and quantitative experiments on ShapeNet-55, PCN, and KITTI datasets, which shows that Point-PC improves the accuracy and plausibility of point cloud completion.

2 Related Work

2.1 Point Cloud Completion

Most recent state-of-the-art completion methods focus on the decoding process of recovering fine details instead of providing sufficient geometric guidance from partial inputs in the encoding process [Xiang *et al.*, 2021; Xie *et al.*, 2020b; Tchapmi *et al.*, 2019]. The first learning-based work PCN [Yuan *et al.*, 2018] generates a coarse completion based on a learned global feature and is then upsampled combined with the assumption that a 3D object lies on a 2D-manifold. Later researches focus on mitigating mature learning-based structures. Some previous methods [Liu *et al.*, 2019b; Liu *et al.*, 2019a] voxelized the point cloud into binary voxels to migrate 3D convolutions, which cubically increased the computational cost, whereas other methods [Huang *et al.*, 2020; Mandikal and Babu, 2019] process coordinates directly by Multi-Layer Perceptrons, yet losing geometric information with pooling-based aggregation operations. These two kinds of completion methods ignore relation and context across points, thus failing to preserve regional information of local patterns. To solve this problem, TopNet [Tchapmi *et al.*, 2019] constrains the point completion process as the growth of a hierarchical rooted tree where several child points are projected by a parent point in a feature expansion layer. On the other hand, SnowflakeNet [Xiang *et al.*, 2021] models point cloud completion procedure as the generation of a snowflake. Furthermore, by breaking the point cloud into several sequential patches, transformer-based methods [Guo *et al.*, 2021; Yu *et al.*, 2021; Zhou *et al.*, 2022] are proved to efficiently handle large-scale point cloud and enhance relations between neighboring points, which outperform and dominate the research prospect. Nevertheless, upsample and expansion modules among the aforementioned methods are based on a global feature vector due to its simplicity, which prevents them from precisely capturing the detailed geometries and structures of 3D shapes, therefore it is unable for these methods to arrange the well-structured point splitting in local regions. In order to integrate more geometric information explicitly, we utilize a memory network to provide rich structural details and enhance neighboring relations to recover local regions.

2.2 Memory Network

The Memory Network [Weston *et al.*, 2015] was initially presented in dialog systems to save scene information and realize the functionality of long-term memory. However, the original design of the Memory Network just vectorizes and saves the original text without proper modification, thus limiting the promotion of the model. Further works [Sukhbaatar *et al.*, 2015; Liu and Perez, 2017] reinforce the Memory Network so that it can be trained in an end-to-end way. Hierarchical Memory Network [Chandar *et al.*, 2016; Xu *et al.*, 2016] stores and searches memory in a hierarchical structure to speed up calculations when implementing large-scale memory. Key-Value Memory Network [Miller *et al.*, 2016] stores memory slots in a "key-value" pair where the key module is responsible for scoring the degree of correlation between memory and queries, while the value module is responsible for weighting and summing the values of the memory to obtain the output. In our work, we further extend the application of "key-value" structured memory into point cloud completion and reveal its ability for preserving high-quality geometry details through a well-designed pre-training method.

2.3 Causal Inference

Causal Inference was first introduced by [Pearl, 2000]. Recent research [Hu *et al.*, 2021; Niu *et al.*, 2020] has shown that causal inference is beneficial to various fields in computer vision. VC R-CNN [Wang *et al.*, 2020] proposes that observational bias causes the model to make predictions based on co-occurrence information while ignoring some common-sense causal relationships, and attempts to extract a visual feature that contains common sense through causal intervention. CONTA [Zhang *et al.*, 2020a] attributes the cause of the ambiguous boundaries of pseudo-masks to the confounding context, and uses backdoor adjustment to eliminate the confounder and generate better pixel-level pseudo masks by

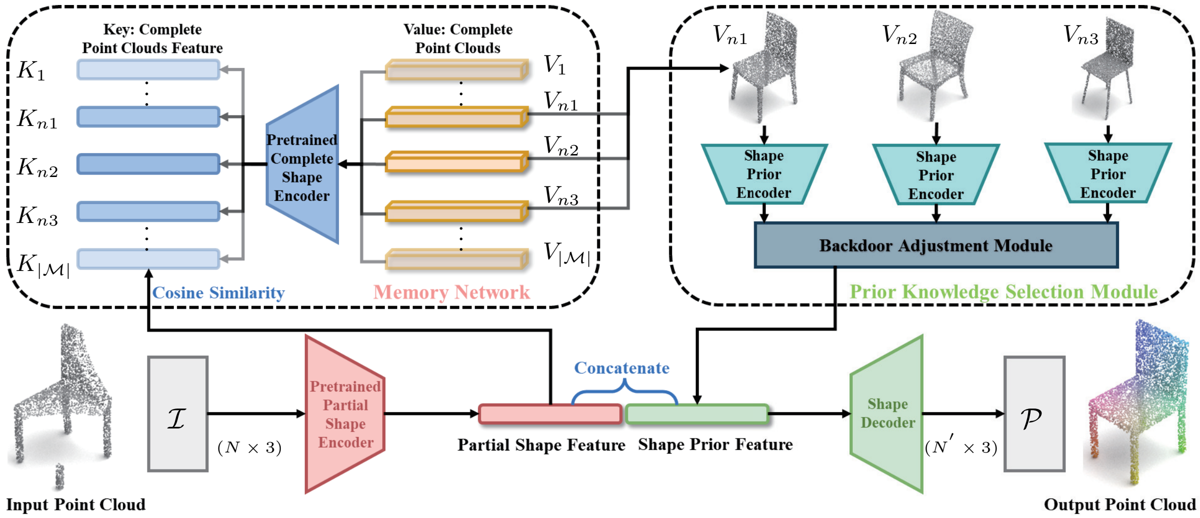

Figure 2: The overall architecture of Point-PC, which consists of four main modules: (i) pre-trained partial shape encoder, (ii) memory network, (iii) prior knowledge selection module, and (iv) shape decoder. The pre-trained encoder extracts feature from the partial input, which is then fed into memory. The memory network retrieves shape priors with sufficient geometric information. Moreover, the prior knowledge selection module selects useful information from the prior shapes. The shape decoder takes the concatenation of the partial shape feature and the shape prior feature to generate the complete point cloud.

using only image-level labels. Ifsl [Yue *et al.*, 2020] argues that the pre-trained knowledge is essentially a confounder that causes spurious correlations between the sample features and class labels in the support set and removes the confounding bias using the backdoor adjustment. To our best knowledge, we are the first to introduce causal inference to point cloud completion. We introduce a causal feature fusion strategy to mitigate the confounding effect in shape priors. It encourages the decoder to pay more attention to causal features, which will also enhance the robustness of the memory network.

3 Our Approach

The overall architecture of Point-PC is shown in Figure 2, which consists of four modules: pre-train encoder, memory network, prior knowledge selection module, and shape decoder. We will detail each of our designs in the following.

3.1 Memory Network

The memory network aims to learn the dependency of partial and complete shapes in feature space and produce the prior shapes. Denote the input set of partial point clouds as $S = \{\mathcal{I}_i\}_{i=1}^{|S|}$, where $\mathcal{I}_i \in \mathbb{R}^{N \times 3}$ represents each point in the object, N is the point number of a shape. We construct the memory network in a "key-value-age" formation. The "key" and "value" represent complete shape features and the corresponding 3D shapes, respectively. The "age" indicates how long the corresponding "key-value" pair has been established. Therefore, the memory item is denoted as $\mathcal{M} = (K_i, V_i, A_i)_{i=1}^{|\mathcal{M}|}$, where $|\mathcal{M}|$ is the size of the memory.

Compared with other methods, the memory network utilizes the "key" and "value" to improve the effectiveness of prior shapes. Meanwhile, the "key" and "value" can also be updated by the training data and improve the relevance of obtaining prior information. Next, we will introduce the model update and retrieval process in two parts.

Update Strategy

K_i is extracted through the pre-trained complete shape encoder from V_i, which can be denoted as F^{V_i}. It is worth noting that the updating strategy only works at training because we take the training set as our external knowledge base, which can not be available during testing.

We compute the cosine similarity between $F^{\mathcal{I}}$ and F^{V_i} to match a "key-value" pair as follows:

$$Sim_{key}\left(F^{\mathcal{I}}, F^{V_i}\right) = \frac{F^{\mathcal{I}} \cdot F^{V_i}}{\|F^{\mathcal{I}}\| \, \|F^{V_i}\|}. \quad (1)$$

To measure whether it is a valid match, we adopt Chamfer Distance [Yuan *et al.*, 2018] as the similarity measurement between the corresponding ground truth $\mathcal{V}$ and the value V_i in 3D space. If the Chamfer Distance Sim_{value} exceeds a threshold δ(discussed in the ablation study), it is a positive match and vice versa. For a positive match, the value V_{n_0} stays unchanged, while the key $F^{V_{n_0}}$ is updated as below:

$$F^{V_{n_0}} = \frac{F^{\mathcal{I}} + F^{V_{n_0}}}{\left\|F^{\mathcal{I}} + F^{V_{n_1}}\right\|}, \quad (2)$$

where $n_0 = \arg\max_i Sim_{key}\left(F^{\mathcal{I}}, F^{V_i}\right)$. In the meantime, except for the corresponding age A_{n_0} to be set to zero, all the other ages should be increased by one. For a negative match, $\mathcal{V}$ is read into the memory and should overwrite the oldest slot as follows:

$$K_{n_1} = F^{\mathcal{I}}, V_{n_1} = \mathcal{V}, \quad (3)$$

where n_1 depends on $n_1 = \arg\max_i (A_i)$. The ages here are updated in the same way as mentioned above. In this way, the memory network reinforces its reception ability with similar shapes, saves the unknown shapes, and refreshes the oldest memory slot.

Algorithm 1 Update and Query Strategy

Input: partial point cloud feature $F^{\mathcal{I}}$
Hyper-parameter: similarity threshold δ
Output: shape priors V_{n_i}

```
Let i = 0.
while i ≤ |M| − 1 do
    Compute Sim_key(F^I, F^{V_i}) by Eq. 1.
    if (Sim_value(V, V_{n_1}) ≥ δ) then
        Let n_0 = arg max_i Sim_key(F^I, F^{V_i}),
        Update K_{n_0} by Eq. 2,
        Set A_{n_0} = 0, A_i = A_i + 1 (i ≠ n_0).
    else
        Let n_1 = arg max_i (A_i).
        Update K_{n_1} and V_{n_1} by Eq. 3,
        Set A_{n_1} = 0, A_i = A_i + 1 (i ≠ n_1).
    end if
end while
return V_{n_i} by Eq. 4.
```

Query Strategy

We propose a query strategy for obtaining shape priors that are rich in geometric information for completion and very similar to the partial input. These shape priors are the values in the memory, which are complete point clouds. To fix the number of shape priors fed forward, we retrieve $\hat{k}$ shapes through top-$\hat{k}$ keys with the largest similarity for convenience, which can be formulated as:

$$V = \left[V_{n_i} | n_i = \arg\max_i Sim_{key}\left(F^{\mathcal{I}}, F^{V_i}\right) \right]. \quad (4)$$

The simplified update and query process is described in Algorithm 1.

3.2 Pre-training Scheme

The pre-training scheme aims to minimize the distance between partial point clouds and complete point clouds, as well as enhance the consistency of partial shape features. Given the complete shape denoted as $\mathcal{S}_i \in \mathbb{R}^{N\times 3}$, where N is the number of points, we render the corresponding partial ones $\mathcal{I}_{i,n_1}$ and $\mathcal{I}_{i,n_2}$ in different viewpoints and crop different numbers of n_1 and n_2 points. We provide a visualization of the overall pre-training scheme in the supplementary material.

Intra-modality Learning

Suppose that $\mathcal{I}_{i,n_1}$ and $\mathcal{I}_{i,n_2}$ are fed into the partial shape encoder E_K to extract features $F^K_{i,n_1}, F^K_{i,n_2} \in \mathbb{R}^{1\times C}$, where C is the feature dimension. Following the NT-Xent loss in SimCLR [Chen *et al.*, 2020], given a positive pair (F^K_{i,n_1}, F^K_{i,n_2}), we treat the other $2(N-1)$ examples within a minibatch as negative examples, where N is the size of the minibatch. The intra-modality contrastive loss $\mathcal{L}_{intra}$ can be formulated as:

$$l_{\text{intra}}\left(i; n_1, n_2\right) = -\log \frac{Sim_{pos}(i; n_1, n_2)}{Sim_{neg}(i; n_1, n_2)}, \quad (5)$$

$$\mathcal{L}_{\text{intra}} = \frac{1}{2N} \sum_{i=1}^{N} \left(l_{\text{intra}}\left(i; n_1, n_2\right) + l_{\text{intra}}\left(i; n_2, n_1\right) \right), \quad (6)$$

where $Sim_{pos}(i; n_1, n_2)$ and $Sim_{neg}(i; n_1, n_2)$ represent the positive and negative cosine similarity between the same partial inputs but with a different incomplete pattern. The cosine similarity function is defined as follows:

$$\begin{aligned} Sim_{pos}(i; n_1, n_2) &= \exp\left(sim\left(F^K_{i,n_1}, F^K_{i,n_2}\right)/\tau \right), \\ Sim_{neg}(i; n_1, n_2) &= \sum_{j=1}^{N} \mathbb{I}_{[j\neq i]} \exp\left(sim\left(F^K_{i,n_1}, F^K_{j,n_1}\right)/\tau \right) \\ &+ \sum_{j=1}^{N} \exp\left(sim\left(F^K_{i,n_1}, F^K_{j,n_2}\right)/\tau \right), \end{aligned} \quad (7)$$

where $\mathbb{I}_{[j\neq i]} \in \{0, 1\}$ is an indicator function evaluating to 1 if $j \neq i$ and τ is the temperature parameter which we set to 0.1.

Cross-modality Learning

Considering that the partial shape features should keep consistent with the complete shape features, for each $\mathcal{S}_i$, we extract features $F^V_i \in \mathbb{R}^{1\times C}$ by the complete shape encoder E_V. Together with the partial shape features F^K_i, the cross-modality contrastive loss $\mathcal{L}_{cross}$ is indicated as follows:

$$l_{\text{intra}}\left(i; K, V\right) = -\log \frac{Sim_{pos}(i; K, V)}{Sim_{neg}(i; K, V)}, \quad (8)$$

$$\mathcal{L}_{\text{cross}} = \frac{1}{2N} \sum_{i=1}^{N} \left(l_{\text{cross}}(i; K, V) + l_{\text{cross}}(i; V, K) \right) \quad (9)$$

where $Sim_{pos}(i; K, V)$ and $Sim_{neg}(i; K, V)$ represent the positive and negative cosine similarity between the partial and complete shape features. The overall pre-training loss function $\mathcal{L}_{pre}$ is the sum of the intra-modality and cross-modality loss $\mathcal{L}_{pre} = \mathcal{L}_{intra} + \mathcal{L}_{cross}$.

3.3 Prior Knowledge Selection Module

We exploit causal theory [Pearl, 2013] to dig out the true causality of the extracted features and generated 3D shapes. The causal graph is shown as Figure 3.

We list the following explanations for the causalities among the four variables shown in Figure 3:

- $M \rightarrow I$. Since the retrieved shapes share the same semantic structures as the partial inputs, this causal effect is naturally established.
- $I \rightarrow C \leftarrow M$. The variable C denotes the causal feature that is truly responsible for the completion result. We not only keep the original part I but also add M as the supplementary information.
- $C \rightarrow Y$. The causality reflects the intrinsic association of the feature space and 3D coordinate space.

Investigating the causal graph above, we recognize a backdoor path between M and I, *i.e.*, $M \rightarrow I$, wherein the M plays a role of confounder between I and C. This backdoor path will cause I to create a false correlation with Y even if I is not the only one directly linked to Y, resulting in generating low-quality shapes. Hence, it is crucial to cut off the backdoor path.

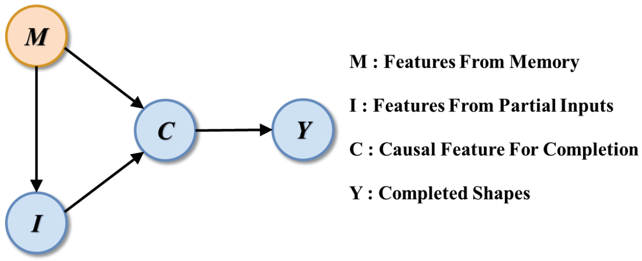

Figure 3: Causal graph for Backdoor Adjustment Module. Circles represent variables, and arrows represent causal relationships from one variable to another.

Backdoor Adjustment

Instead of modeling the confounded $P(Y|I)$ in Figure 3, we need to eliminate the backdoor path. According to causal theory, we exploit the do-calculus on the variable M to remove the backdoor path by estimating $P_B(Y|I) = P(Y|do(I))$ which stratifies the confounder M. We then obtain the following derivations:

- The features extracted from memory will not be affected by cutting off the backdoor path. Thus, $P(m) = P_B(m)$.
- C has nothing to do with the causal effect between the variable M and I, which we can get $P_B(C|I,m) = P(C|I,m)$.
- After the causal intervention, the variable m is independent from I, for which we have $P_B(m) = P_B(m|I)$.

B refers to the case when the backdoor path is cut off, and $m \in M$ denotes the confounder sets. Driven by the derivations above, the backdoor adjustment for Figure 3 can be written as:

$$\begin{aligned} P(Y \mid do(I)) &= \sum_{m \in M} P_B(Y|I,m) P_B(m|I) \\ &= \sum_{m \in M} P_B(Y|I,m) P_B(m) \\ &= \sum_{m \in M} P(Y|I,m) P(m), \end{aligned} \tag{10}$$

where $P(Y|I,m)$ represents the conditional probability given the partial shape feature and confounder; $P(m)$ is the prior probability of the confounder.

Module Design

Driven by Eq. 10, we design the prior knowledge selection module to alleviate the confounding effect in shape priors. Our implementation idea is stratifying the confounder and pairing the partial shape feature with every stratification. Towards this end, we make the implicit intervention on feature-wise sampling. Suppose that $\mathcal{H}$ is the index set of the dimensions of the concatenated shape prior feature from the last layer of the shape prior encoder. We divide $\mathcal{H}$ into n equal-size disjoint subsets, e.g., the output feature dimension of the shape prior encoder is 384, if we select top-3 shape priors and $n = 6$, the i-th set will be a feature dimension index set of size $1152/6 = 192$, i.e., $\mathcal{H}_m = 192(m-1)+1, ..., 192m$.

- $P(Y|I,m) = P_\phi(Y|cat(F_I, [F_V]_c))$, where F_I and F_V are the partial shape feature and the concatenated shape prior feature, respectively. $[F_V]_c$ is a feature selector which selects the dimensions of F_V according to the index set c. Note that $c = \{k | k \in \mathcal{H}_m \cap \mathcal{S}_t\}$, where $\mathcal{S}_t$ is an index set whose corresponding absolute values in F_V are larger than the threshold t. And ϕ represents the parameters of the shape decoder.
- $P(m) = 1/n$, where we assume a uniform prior distribution for the adjusted features. n is the number of confounder set.

Thus, the overall feature-wise adjustment is:

$$P(Y \mid do(I)) = \frac{1}{n} \sum_{m \in M} P_\phi(Y|cat(F_I, [F_V]_c)). \tag{11}$$

To optimize the ϕ in the above Eq. 11, we propose a slightly modified L1 Chamfer Distance loss guided by the backdoor adjustment. Let $\mathcal{G}$ be the notation of high-resolution ground truth, and $\mathcal{P}$ be the notation of the completed prediction. The $\mathcal{L}_{caus}$ can be written as:

$$\mathcal{P} = \Phi(cat(F_I, [F_V]_c)), \tag{12}$$

$$\mathcal{L}_{caus} = \frac{1}{n} \sum_{m \in M} \left(CD - \ell_1(\mathcal{P}, \mathcal{G}) \right), \tag{13}$$

where Φ represents the shape decoder, and $cat(\cdot,\cdot)$ denotes the concatenate operation. The Eq. 13 pushes the predictions of such intervened partial-complete probability to be invariant and stable across different stratifications, due to the shared causal features.

We follow the existing works [Yu *et al.*, 2021] to use the L1 Chamfer Distance [Fan *et al.*, 2016] as a quantitative measurement for the quality of output. Apart from generating $\mathcal{P}$, Point-PC also predicts local centers $\mathcal{C}$ of the completed point cloud. For each prediction, the L1 Chamfer Distance loss function between the central point set and the ground truth $\mathcal{G}$ is calculated as:

$$\mathcal{L}_{recon} = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \min_{g \in \mathcal{G}} \|c-g\|_1 + \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \min_{c \in \mathcal{C}} \|g-c\|_1. \tag{14}$$

The final objective function can be defined as the sum of the losses: $\mathcal{L} = \lambda \mathcal{L}_{caus} + (1-\lambda)\mathcal{L}_{recon}$, where λ is a hype-parameter used to control the contribution of different losses in the optimization process.

4 Experiment

In this section, we first present the experimental results on ShapeNet-55/34 [Yu *et al.*, 2021], PCN [Yuan *et al.*, 2018], and KITTI [Geiger *et al.*, 2013]. Then, we visualize and analyze the results for both our method and several baseline methods. Finally, we also provide detailed ablation studies of our method.

4.1 Results on ShapeNet-55

Following the evaluation setting in [Yu *et al.*, 2021], 8 fixed viewpoints are selected, and the number of points in the partial point cloud is set to 2,048, 4,096, and 6,144 (25%, 50%,

Methods	Table	Chair	Airplane	Car	Sofa	Birdhouse	Bag	Remote	Keyboard	Rocket	CD-S	CD-M	CD-H	CD-Avg	F-Score@1%
FoldingNet	2.53	2.81	1.43	1.98	2.48	4.71	2.79	1.44	1.24	1.48	2.67	2.66(-0.01)	4.05(+1.38)	3.12	0.082
PCN	2.13	2.29	1.02	1.85	2.06	4.5	2.86	1.33	0.89	1.32	1.94	1.96(+0.02)	4.08(+2.14)	2.66	0.133
TopNet	2.21	2.53	1.14	2.18	2.36	4.83	2.93	1.49	0.95	1.32	2.26	2.16(-0.10)	4.30(+2.26)	2.91	0.126
PFNet	3.95	4.24	1.81	2.53	3.34	6.21	4.96	2.91	1.29	2.36	3.83	3.87(+0.04)	7.97(+4.10)	5.22	0.339
GRNet	1.63	1.88	1.02	1.64	1.72	2.97	2.06	1.09	0.89	1.03	1.35	1.71(+0.36)	2.85(+1.50)	1.97	0.238
PoinTr	**0.81**	**0.95**	**0.44**	**0.91**	**0.79**	**1.86**	**0.93**	**0.53**	**0.38**	**0.57**	**0.58**	**0.88**(+0.30)	**1.79**(+1.21)	**1.09**	**0.464**
Point-PC	1.16	1.26	0.58	1.05	1.19	2.14	1.58	0.68	0.53	0.79	1.16	1.23(+0.07)	2.04(+0.88)	1.48	0.426

Table 1: Quantitative results of our methods and several baselines on ShapeNet-55. Detailed results for each method on 10 selected categories are reported, as well as the overall results on 55 categories. CD-S, CD-M, and CD-H represent the CD-ℓ_2 results under the simple, moderate, and hard settings, respectively. Red/green numbers represent increments of CD-ℓ_2 results compared to results under the CD-S setting.

and 75% of the whole point cloud), which divides the testing stage into three difficulty degrees of simple, moderate, and hard (denoted as CD-S, CD-M, and CD-H). As shown in Table. 1, Point-PC achieves an average CD-ℓ_2(multiplied by 1000) of 1.48 and F-Score@1% of 0.426 on ShapeNet-55, which shows the effectiveness of Point-PC encountering diverse categories of objects. It is worth noting that the increments of CD-ℓ_2 under CD-M(+0.07) and CD-H(+0.88) strategy demonstrate that Point-PC better deals with diverse incompleteness levels and diverse incomplete patterns compared to the state-of-the-art methods. Furthermore, we report the results for categories with sufficient(first 5 columns) and insufficient(following 5 columns) training samples. Point-PC performs evenly despite the training sample imbalance. Quantitative results on ShapeNet-55 show that Point-PC can generate complete point clouds in a variety of situations.

The qualitative comparison results are shown in Figure. 4. The proposed Point-PC performs better with fine details than the other methods. For example, in the bottle category, Point-PC predicts a more smooth and more regular structure of bottle edges compared with the other methods. Moreover, Point-PC retains the original details of the partial shapes. In the fifth column of Figure. 4, Point-PC not only generates the incomplete lamp bracket with a clear structure but also keeps the texture of the lamp shade, which makes it a more plausible completion than the other methods. Consequently, Point-PC effectively learns the geometric information based on the existing partial shape, retrieves similar shape priors based on the learned information and reconstructs complete shapes with more regular arrangements and surface smoothness.

4.2 Results on ShapeNet-34

We utilize ShapeNet-34 to evaluate the performance of Point-PC on novel objects from categories that do not appear in the training phase. As shown in Table.2, our method achieves the best scores of 0.444 F-Score@1% on 34 seen categories and 0.406 F-Score@1% on 21 unseen categories. In particular, we observe fewer gaps between the results of 34 seen categories and 21 unseen categories under each difficulty setting, which demonstrates the superiority of shape priors offered by the memory network. We also provide the visual comparison with GRNet on novel categories in Figure.5, which show the effectiveness of Point-PC in this more challenging setting.

4.3 Results on PCN

We compare several SOTA methods on the PCN dataset. The related experimental results are shown in Table.3. Our proposed method stands out and produces the best results in 3 out

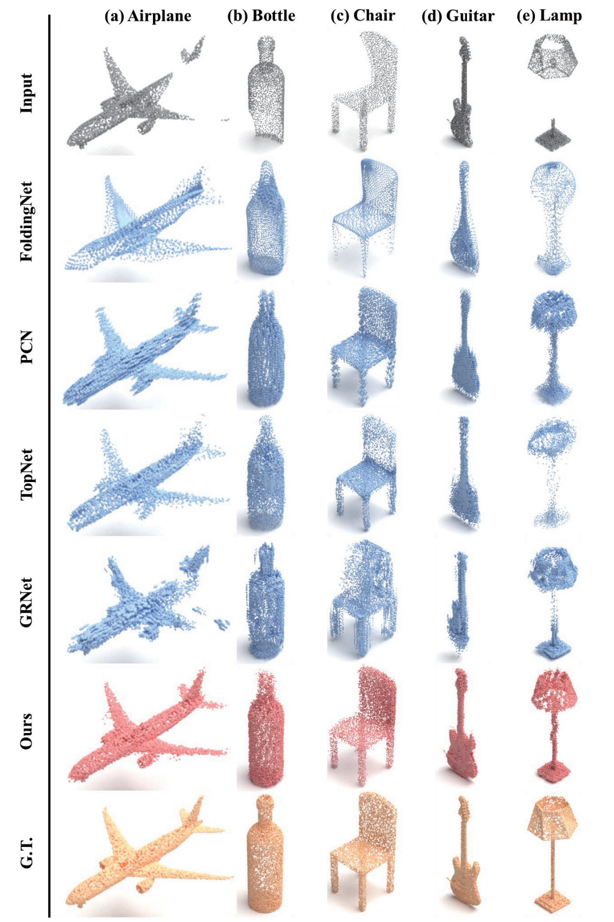

Figure 4: Qualitative results on ShapeNet-55 benchmark. All methods above take samples in the first row as inputs and generate complete point clouds.

of 8 categories. In terms of average CD-ℓ_1, Point-PC achieves second-best score of 8.50, which illustrate that Point-PC performs favorably against state-of-the-art completion networks.

4.4 Results on KITTI Benchmark

We report both the results of Fidelity and MMD metrics in Table.4 on the KITTI dataset. The Fidelity measures the average distance between points in the input and their nearest neighbors in the output, representing how well the input is preserved. MMD is the Chamfer Distance between the completion result and the closest ground truth in ShapeNetCars, indicating how much the reconstruction resembles a typical car. Observed in Table.4, Point-PC shows better generaliza-

Methods	34 seen categories					21 unseen categories				
	CD-S	CD-M	CD-H	CD-Avg	F1	CD-S	CD-M	CD-H	CD-Avg	F-Score@1%
FoldingNet	1.86	1.81	3.38	2.35	0.139	2.76	2.74	5.36	3.62	0.095
PCN	1..87	1.81	2.97	2.22	0.154	3.17	3.08	5.29	3.85	0.101
TopNet	1.77	1.61	3.54	2.31	0.171	2.62	2.43	5.44	3.5	0.121
PFNet	3.16	3.19	7.71	4.68	0.347	5.29	5.87	13.33	8.16	0.322
GRNet	1.26	1.39	2.57	1.74	0.251	1.85	2.25	4.87	2.99	0.216
PoinTr	**0.76**	**1.05**	**1.88**	**1.23**	0.421	**1.04**	**1.67**	3.44	**2.05**	0.384
Point-PC	1.17	1.46	2.21	1.61	**0.444**	1.62	2.05	**3.15**	2.27	**0.406**

Table 2: Quantitative results on ShapeNet-34 evaluated as CD-ℓ_2(multiplied by 1000) and F-Score@1%.

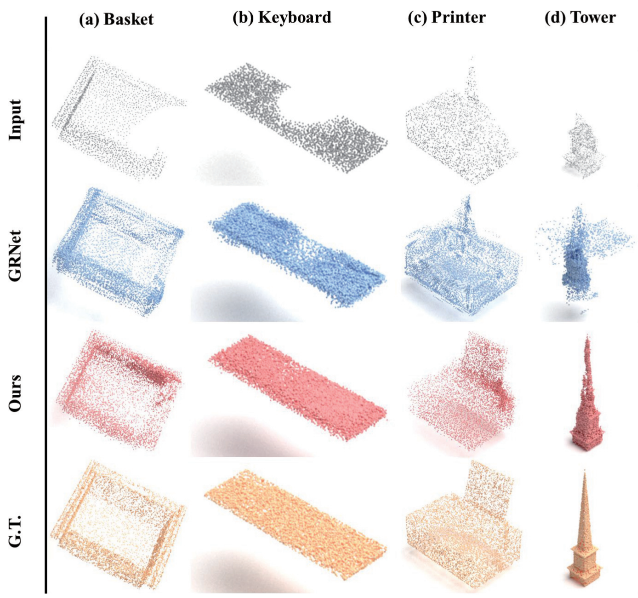

Figure 5: Quantitative results on objects of novel categories that do not appear in the training set. We show the input partial point cloud and the ground truth as well as the prediction of GRNet and Point-PC.

tion ability compared with previous methods, achieving a Fidelity of 0.398 and MMD of 0.527. Qualitative results can be found in the supplementary material. Compared with other public datasets, the KITTI dataset is composed of a sequence of real-world scans. The points in the data are more sparse and lack regularity, which brings greater challenges to data completion. However, our approach achieves the best performance, which further proves the necessity of prior knowledge for guiding the point cloud completion.

4.5 Model Design Analysis

To examine the effectiveness of our designs, we conduct detailed ablation studies. The results of the novel modules of Point-PC are shown in Table.5. The baseline model A refers to a geometry-aware transformer encoder and a foldingnet-based decoder in an "encoder-decoder" pattern. This model generates poor results. We then add the memory network and improves the baseline by 4.84 in the CD-ℓ_1 metric, which means that the memory network provides more geometric information to improve the performance. However, due to the lack of consistent representational learning of complete and partial shapes, the relevance of prior information cannot be guaranteed. Thus, it did not get the best results. When applying well-designed pre-training with intra- and cross-modality

Methods	Air	Cab	Car	Cha	Lam	Sof	Tab	Wat	CD-Avg
FoldingNet	9.49	15.8	12.61	15.55	16.41	15.97	13.65	14.99	14.31
AtlasNet	6.37	11.94	10.1	12.06	12.37	12.99	10.33	10.61	10.85
PCN	5.50	22.70	10.63	**8.70**	11.00	11.34	11.68	8.59	9.64
TopNet	7.61	13.31	10.9	13.82	14.44	14.78	11.22	11.12	12.15
MSN	5.60	11.90	10.30	10.20	10.70	11.60	9.60	9.90	10.00
GRNet	6.45	10.37	9.45	9.41	7.96	10.51	8.44	8.04	8.83
PoinTr	**4.75**	10.47	8.68	9.39	**7.75**	10.93	**7.78**	**7.29**	**8.38**
Point-PC	4.89	**10.20**	**8.56**	9.24	8.65	**9.70**	8.62	8.14	8.50

Table 3: Quantitative results on the PCN dataset. We report detailed results on each category and the average results under the CD-ℓ_1(multiplied by 1000) metric.

	FoldingNet	AtlasNet	PCN	TopNet	MSN	PFNet	GRNet	Point-PC
Fidelity	7.467	1.759	2.235	5.354	0.434	1.137	0.816	**0.398**
MMD	0.537	2.108	1.366	0.636	2.259	0.792	0.568	**0.527**

Table 4: Quantitative results on the KITTI dataset under the metrics of Fidelity Distance and MMD(Minimal Matching Distance). Lower is better.

learning (model C), we observe another improvement of 0.71 in the Chamfer distance, which is a sign of retrieving more relevant shape priors. By adding the prior knowledge select module to Point-PC, the performance can be further improved, achieving an average CD-ℓ_2 of 8.5, which indicates that the causal model effectively removes existing structural information and save missing shape information to improve the integrity of the fused representation. The ablation study clearly demonstrates the effectiveness of key components in Point-PC. The ablation studies on the number of shape priors and the similarity threshold δ can be found in the supplementary material.

Model	Memory Network	Pre-train Scheme	PKS Module	CD-AVG	F-Score@1%
A	×	×	×	15.37	0.109
B	✓	×	×	10.53	0.541
C	✓	✓	×	9.82	0.623
D	✓	✓	✓	**8.50**	**0.709**

Table 5: Ablation study on the PCN dataset. We investigate different designs including the Memory Network, the pre-train scheme, and the prior knowledge selection module(PKS Module).

5 Conclusion

In this paper, we propose a novel approach to point cloud completion called Point-PC, which proposes a new memory-based architecture to search prior shapes and designs an effective causal inference model to choose missing shape information as supplemental geometric information to aid point cloud completion. Specifically, the update mechanism of the memory network can optimize the retrieval distance based on the training data, thereby improving the accuracy of the prior shape. To our best knowledge, this is the first work to introduce a causal graph into the point cloud completion task, which effectively filters shape information from previous shapes and preserves missing shape information to improve the integrity and ultimate performance of the fused representation. Comprehensive experiments show the effectiveness and superiority of Point-PC compared to state-of-the-art competitors.

References

[Cao *et al.*, 2022] Rui Cao, Kaiyi Zhang, Yang Chen, Ximing Yang, and Cheng Jin. Point cloud completion via multi-scale edge convolution and attention. *Proceedings of the 30th ACM International Conference on Multimedia*, 2022.

[Chandar *et al.*, 2016] A. P. Sarath Chandar, Sungjin Ahn, H. Larochelle, Pascal Vincent, Gerald Tesauro, and Yoshua Bengio. Hierarchical memory networks. *ArXiv*, abs/1605.07427, 2016.

[Chang *et al.*, 2015] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, L. Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *ArXiv*, abs/1512.03012, 2015.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.

[Fan *et al.*, 2016] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3d object reconstruction from a single image. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2463–2471, 2016.

[Geiger *et al.*, 2013] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32:1231 – 1237, 2013.

[Guo *et al.*, 2021] Meng-Hao Guo, Junxiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph Robert Martin, and Shimin Hu. Pct: Point cloud transformer. *Comput. Vis. Media*, 7:187–199, 2021.

[Han *et al.*, 2017] Zhizhong Han, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Shuhui Bu, Junwei Han, and C. L. Philip Chen. Boscc: Bag of spatial context correlations for spatially enhanced 3d shape representation. *IEEE Transactions on Image Processing*, 26:3707–3720, 2017.

[Han *et al.*, 2018a] Zhizhong Han, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Shuhui Bu, Junwei Han, and C. L. Philip Chen. Deep spatiality: Unsupervised learning of spatially-enhanced global and local 3d features by deep neural network with coupled softmax. *IEEE Transactions on Image Processing*, 27:3049–3063, 2018.

[Han *et al.*, 2018b] Zhizhong Han, Mingyang Shang, Yu-Shen Liu, and Matthias Zwicker. View inter-prediction gan: Unsupervised representation learning for 3d shapes by learning global shape memories to support local view predictions. In *AAAI Conference on Artificial Intelligence*, 2018.

[Han *et al.*, 2019] Zhizhong Han, Chao Chen, Yu-Shen Liu, and Matthias Zwicker. Shapecaptioner: Generative caption network for 3d shapes by learning a mapping from parts detected in multiple views to sentences. *Proceedings of the 28th ACM International Conference on Multimedia*, 2019.

[Hu *et al.*, 2021] Xinting Hu, Kaihua Tang, Chunyan Miao, Xiansheng Hua, and Hanwang Zhang. Distilling causal effect of data in class-incremental learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3956–3965, 2021.

[Huang *et al.*, 2020] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. Pf-net: Point fractal network for 3d point cloud completion. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7659–7667, 2020.

[Li *et al.*, 2022] Jun Li, Shangwei Guo, Zhengchao Lai, Xiantong Meng, and Shaokun Han. Completedt: Point cloud completion with dense augment inference transformers. *ArXiv*, abs/2205.14999, 2022.

[Liu and Perez, 2017] Fei Liu and Julien Perez. Gated end-to-end memory networks. In *Conference of the European Chapter of the Association for Computational Linguistics*, 2017.

[Liu *et al.*, 2019a] Yongcheng Liu, Bin Fan, Shiming Xiang, and Chunhong Pan. Relation-shape convolutional neural network for point cloud analysis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8887–8896, 2019.

[Liu *et al.*, 2019b] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. Point-voxel cnn for efficient 3d deep learning. *ArXiv*, abs/1907.03739, 2019.

[Mandikal and Babu, 2019] Priyanka Mandikal and R. Venkatesh Babu. Dense 3d point cloud reconstruction using a deep pyramid network. *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1052–1060, 2019.

[Miller *et al.*, 2016] Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. *ArXiv*, abs/1606.03126, 2016.

[Niu *et al.*, 2020] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xiansheng Hua, and Ji rong Wen. Counterfactual vqa: A cause-effect look at language bias. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12695–12705, 2020.

[Park *et al.*, 2019] Jeong Joon Park, Peter R. Florence, Julian Straub, Richard A. Newcombe, and S. Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 165–174, 2019.

[Pearl, 2000] Judea Pearl. Causality: Models, reasoning and inference. 2000.

[Pearl, 2013] Judea Pearl. Interpretation and identification of causal mediation. *ERN: Other Econometrics: Econometric Model Construction*, 2013.

[Qi *et al.*, 2017] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.

[Sukhbaatar *et al.*, 2015] Sainbayar Sukhbaatar, Arthur D. Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *NIPS*, 2015.

[Tchapmi *et al.*, 2019] Lyne P. Tchapmi, Vineet Kosaraju, Hamid Rezatofighi, Ian D. Reid, and Silvio Savarese. Topnet: Structural point cloud decoder. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 383–392, 2019.

[Wang *et al.*, 2018] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, W. Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *European Conference on Computer Vision*, 2018.

[Wang *et al.*, 2020] Tan Wang, Jianqiang Huang, Hanwang Zhang, and Qianru Sun. Visual commonsense r-cnn. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10757–10767, 2020.

[Wen *et al.*, 2020] Xin Wen, Tianyang Li, Zhizhong Han, and Yu-Shen Liu. Point cloud completion by skip-attention network with hierarchical folding. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1936–1945, 2020.

[Wen *et al.*, 2021] Xin Wen, Zhizhong Han, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Yu-Shen Liu. Cycle4completion: Unpaired point cloud completion using cycle transformation with missing region coding. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13075–13084, 2021.

[Wen *et al.*, 2022] Xin Wen, Peng Xiang, Yaru 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*, 45:852–867, 2022.

[Weston *et al.*, 2015] Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *CoRR*, abs/1410.3916, 2015.

[Xiang *et al.*, 2021] 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. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5479–5489, 2021.

[Xie *et al.*, 2020a] Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *International Journal of Computer Vision*, 128:2919 – 2935, 2020.

[Xie *et al.*, 2020b] Haozhe Xie, Hongxun Yao, Shangchen Zhou, Jiageng Mao, Shengping Zhang, and Wenxiu Sun. Grnet: Gridding residual network for dense point cloud completion. *ArXiv*, abs/2006.03761, 2020.

[Xu *et al.*, 2016] Jiaming Xu, Jing Shi, Yiqun Yao, Suncong Zheng, Bo Xu, and Bo Xu. Hierarchical memory networks for answer selection on unknown words. In *COLING*, 2016.

[Yu *et al.*, 2021] Xumin Yu, Yongming Rao, Ziyi Wang, Zuyan Liu, Jiwen Lu, and Jie Zhou. Pointr: Diverse point cloud completion with geometry-aware transformers. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12478–12487, 2021.

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

[Yue *et al.*, 2020] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xiansheng Hua. Interventional few-shot learning. *ArXiv*, abs/2009.13000, 2020.

[Zhang *et al.*, 2020a] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xiansheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *ArXiv*, abs/2009.12547, 2020.

[Zhang *et al.*, 2020b] Wenxiao Zhang, Qingan Yan, and Chunxia Xiao. Detail preserved point cloud completion via separated feature aggregation. *ArXiv*, abs/2007.02374, 2020.

[Zhang *et al.*, 2022] Ziyu Zhang, Yi Yu, and Fei peng Da. Partial-to-partial point generation network for point cloud completion. *IEEE Robotics and Automation Letters*, 7:11990–11997, 2022.

[Zhou *et al.*, 2022] Hao 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*, 2022.