Simba: Towards High-Fidelity and Geometrically-Consistent Point Cloud Completion via Transformation Diffusion

Lirui Zhang[1*], **Zhengkai Zhao**[1*], **Zhi Zuo**[1], **Pan Gao**[1†], **Jie Qin**[1†]

[1]Nanjing University of Aeronautics and Astronautics
Nanjing, Jiangsu, China
{lirui.zhang, zhengkai.zhao, zuozhi, pan.gao, jie.qin}@nuaa.edu.cn

Abstract

Point cloud completion is a fundamental task in 3D vision. A persistent challenge in this field is simultaneously preserving fine-grained details present in the input while ensuring the global structural integrity of the completed shape. While recent works leveraging local symmetry transformations via direct regression have significantly improved the preservation of geometric structure details, these methods suffer from two major limitations: (1) These regression-based methods are prone to overfitting which tend to memorize instant-specific transformations instead of learning a generalizable geometric prior. (2) Their reliance on point-wise transformation regression lead to high sensitivity to input noise, severely degrading their robustness and generalization. To address these challenges, we introduce **Simba**, a novel framework that reformulates point-wise transformation regression as a distribution learning problem. Our approach integrates symmetry priors with the powerful generative capabilities of diffusion models, avoiding instance-specific memorization while capturing robust geometric structures. Additionally, we introduce a hierarchical Mamba-based architecture to achieve high-fidelity upsampling. Extensive experiments across the PCN, ShapeNet, and KITTI benchmarks validate our method's state-of-the-art (SOTA) performance.

Code — https://github.com/I2-Multimedia-Lab/Simba

Introduction

Point clouds, as a fundamental 3D representation, are integral to numerous applications, from autonomous driving to robotics and augmented reality (Geiger, Lenz, and Urtasun 2012; Cadena et al. 2017; Li et al. 2023a; Zuo et al. 2025). Point clouds captured in real-world environments are often incomplete due to occlusion and limited sensor range, making point cloud completion a critical area of research (Yuan et al. 2018; Li et al. 2023b, 2024).

Existing methods often struggle to preserve input details while maintaining global structural integrity (Yuan et al. 2018; Wen et al. 2021; Xie et al. 2020). Leveraging symmetry to enforce structural consistency has emerged as a powerful approach, and the explicit modeling of geometric priors has shown great potential (Schiebener et al. 2016; Cui et al. 2023). However, existing approaches to modeling symmetry have notable limitations: On one hand, many methods assume global axisymmetry, limiting the ability to capture local or partial symmetries (Zhang et al. 2023; Ma et al. 2023). Recent work like SymmCompletion (Yan et al. 2025) has shown the promise of learning point-wise local symmetry transformations, but its reliance on **direct regression** is a critical limitation: (1) they tend to overfit the training distribution, memorizing specific transformation patterns instead of learning generalized geometric alignment rules; (2) they are highly sensitive to occlusions and noise, as per-point predictions can fragment global structure. These limitations severely hinder generalization to real-world or unseen data distributions (Geiger, Lenz, and Urtasun 2012).

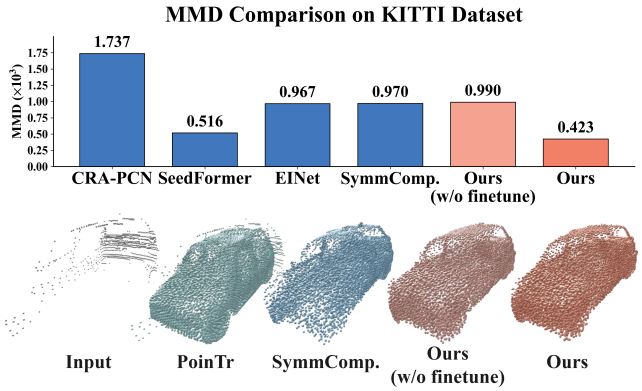

Figure 1: Strong cross-domain generalizability on KITTI. Our model, trained on synthetic data, is competitive without finetuning and achieves superior performance with it.

How to leverage the powerful symmetry priors encoded in affine and translation matrices while avoiding the risk of the network merely memorizing specific transformation patterns remains an open problem. We observe that diffusion models possess strong generative capabilities, enabling diverse sampling. By integrating diffusion with transformation matrices, we can effectively utilize geometric priors without overfitting to fixed solutions.

To this end, we propose a novel direction that leverages the generative power of diffusion models. We argue

*These authors contributed equally.
†Corresponding authors.

that prior applications of diffusion, which directly operate on point coordinates (Lyu et al. 2021), are suboptimal as they can wash out fine details present in the partial input. To explore symmetric geometric priors completely, we reframe the completion task and propose a novel generative paradigm: **instead of diffusing points, we diffuse a field of geometric transformations**. Our framework, Simba, is the first to learn the conditional distribution of point-wise affine transformations. By iteratively denoising a low-dimensional transformation vector, our model generates a robust and geometrically plausible structural prior, which, when applied to the original keypoints, constructs a complete shape while inherently preserving its details. As shown in Figure 1, our model, both with and without fine-tuning, outperforms competing baselines on the real-world KITTI vehicle completion task. This achievement underscores the strong cross-domain generalizability of our approach and its powerful synthetic-to-real transfer capabilities.

Specifically, we propose a two-stage learning framework. In Stage 1, we pre-train SymmGT, a supervision network that generates target transformation matrices for the subsequent stage. In Stage 2, we employ a diffusion model, termed Symmetry-Diffusion (Sym-Diffuser), which conditions on partial input features to generate transformation fields. This design fully exploits the generative capacity of diffusion models to capture symmetric geometric priors. Subsequently, a Mamba-based refinement (MBA-Refiner) network is introduced to progressively enhance and upsample the coarse completions into high-fidelity outputs.

Extensive experiments on multiple benchmarks demonstrate that our method achieves state-of-the-art (SOTA) performance. In addition, we conduct evaluations on the real-world KITTI dataset, further validating the generalizability and effectiveness of our approach. Our main contributions are summarized as follows:

- We propose ***Simba***, a novel framework that formulates point cloud completion as a conditional generative task over a field of geometric transformations.
- We are the ***first*** to employ a diffusion model, termed ***Sym-Diffuser***, to learn the distribution of affine transformations, providing a robust representation to ensure the geometric consistency of the completed shape.
- We design the ***MBA-Refiner***, a cascaded Mamba-based architecture, to progressively refine the coarse output, enabling high-fidelity progressive upsampling.

Related Work

Point Cloud Completion

Foundational works such as PointNet (Qi et al. 2017a) and PointNet++ (Qi et al. 2017b) enabled end-to-end learning on point clouds, laying the groundwork for point cloud completion networks. Early methods like PCN (Yuan et al. 2018) and FoldingNet (Yang et al. 2018) adopted a canonical coarse-to-fine paradigm, learning global shape priors to generate complete surfaces. This approach was further refined by architectures such as SnowflakeNet (Xiang et al. 2021), which incorporated sophisticated decoders to produce more uniform and detailed point distributions. With the advent of transformers (Vaswani et al. 2017), recent methods (Zhao et al. 2021; Li et al. 2023a; Rong et al. 2024; Wang et al. 2024; Nunes et al. 2024; Yu et al. 2024) have focused on capturing long-range dependencies within point sets, achieving state-of-the-art performance in completion tasks. However, the quadratic complexity of transformers presents a major bottleneck in computational efficiency. To alleviate this issue, 3DMambaComplete (Li, Yang, and Fei 2024) introduced the Mamba (Gu and Dao 2023; Liang et al. 2024) architecture into point cloud completion, significantly reducing computational overhead.

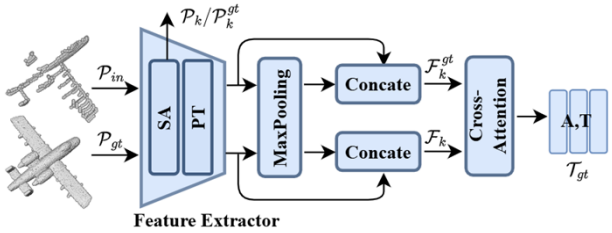

Figure 2: **Stage 1: SymmGT pre-training architecture.** The network regresses transformation field $\mathcal{T}_{gt}$ from partial input and complete GT point clouds.

Diffusion models (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020) have inspired several novel paradigms for 3D synthesis. These include directly denoising 3D coordinates, as in PDR (Lyu et al. 2021), leveraging 2D priors through optimization (SDS) (Kasten, Rahamim, and Chechik 2023), or fusing outputs from feed-forward image-to-3D models like PCDreamer (Wei et al. 2025; Li, Zhu, and Wei 2025). Despite strong generative capability, such approaches often incur high computational costs, slow inference, and difficulty in faithfully preserving input details during fusion.

Symmetry Priors in Point Cloud Completion

Symmetry serves as a fundamental geometric prior for completing structured objects, particularly man-made shapes. Early approaches like GTNet (Zhao et al. 2020) relied on global symmetry assumptions, limiting their effectiveness on partial or asymmetrical inputs. SymmCompletion (Yan et al. 2025) advanced this by learning point-wise local affine transformations between observed and missing regions. However, deterministic transformation regression from partial observations remains ill-posed and prone to overfitting on training-specific patterns.

In contrast, our method formulates point-wise transformation prediction as a distribution learning problem, leveraging the intrinsic diversity of diffusion models to mitigate overfitting. Furthermore, we introduce a cascaded Mamba-based refinement network block combined with a hierarchical upsampling strategy to enhance computational efficiency without sacrificing performance.

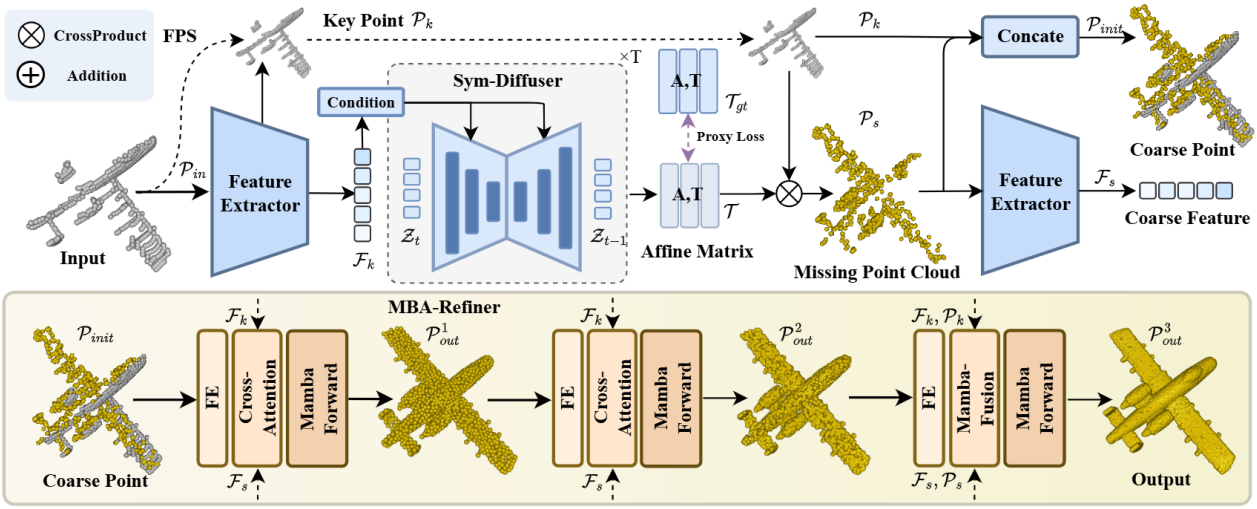

Figure 3: **Stage 2: The Simba coarse-to-fine architecture.** Our framework comprises two core components: a Symmetry-Diffusion Module **(Sym-Diffuser)** that generates a field of geometric transformations to produce a structurally-complete coarse shape from partial input, and a novel **MBA-Refiner** decoder that progressively refines this coarse representation to yield the final high-fidelity output point cloud. **FE** denotes Feature Extractor.

Methodology

The two-stage architecture of our Simba framework is shown in Figures 2 and 3. Stage 1 (Fig 2) pre-trains a **SymmGT** network to generate target transformation matrices for diffusion supervision. Stage 2 (Fig 3) comprises two core components: a Symmetry-Diffusion Module **(Sym-Diffuser)** and a cascaded Mamba-Based Refinement network **(MBA-Refiner)**. The Sym-Diffuser produces a point-wise affine transformation field to generate a coarse yet structurally complete point cloud from partial input. The MBA-Refiner progressively refines and upsamples this coarse prediction through cascaded processing to synthesize the final high-fidelity output.

Pre-training SymmGT (Stage 1)

We observe that symmetric geometric priors inherently capture strong geometric awareness, which can serve as an effective inductive bias for shape completion. Inspired by (Yan et al. 2025), we incorporate such priors into our generative framework. In this section, our target is to generate the transformation matrix that guides the diffusion process. Specifically, we pre-train a base network, SymmGT. As shown in Figure 2, given a partial input point cloud $\boldsymbol{\mathcal{P}}_{in}$ and a complete ground truth (GT) point cloud $\boldsymbol{\mathcal{P}}_{gt}$, it first samples a set of keypoints $\boldsymbol{\mathcal{P}}_k$ from $\boldsymbol{\mathcal{P}}_{in}$ and extracts the keypoint features $\boldsymbol{\mathcal{F}}_k$ from the partial input and a global feature $\boldsymbol{\mathcal{F}}_{gt}$ from the GT point cloud by a **shared Feature Extractor**, which consists of a Set Abstraction (SA) layer (Qi et al. 2017b) and a Point Transformer block (Zhao et al. 2021). These features are fused via cross-attention to regress a target transformation field, denoted as $\boldsymbol{\mathcal{T}}_{gt} \in \mathbb{R}^{K\times 12}$. This field is composed of a point-wise affine matrix $\boldsymbol{A}_i \in \mathbb{R}^{3\times 3}$ and a translation vector $\boldsymbol{T}_i \in \mathbb{R}^3$. The transformation field $\boldsymbol{\mathcal{T}}$ is then applied to the input keypoints $\boldsymbol{\mathcal{P}}_k$ to construct a coarse but complete point cloud $\boldsymbol{\mathcal{P}}_{init}$. The network is trained by minimizing the Chamfer Distance (L_{CD}) between this reconstructed coarse shape and the ground truth:

$$\mathcal{L}_{\text{stage1}} = L_{CD}(\boldsymbol{\mathcal{P}}_k \cup \{\boldsymbol{A}_i\mathbf{p}_i + \boldsymbol{T}_i\}, \boldsymbol{\mathcal{P}}_{gt}), \qquad (1)$$

where $(\boldsymbol{A}_i, \boldsymbol{T}_i) \in \boldsymbol{\mathcal{T}}$, $\mathbf{p}_i \in \boldsymbol{\mathcal{P}}_k$, and k denotes the keypoint set. In Stage 2, the frozen SymmGT is used solely to generate this target field $\boldsymbol{\mathcal{T}}gt$, which serves as the clean data target $\mathcal{Z}_0$ for training our Sym-Diffuser.

Completion with Simba (Stage 2)

While SymmCompletion (Yan et al. 2025) shows notable improvements on datasets like PCN and ShapeNet, its performance on real-world data remains limited. We attribute this to the inherent limitations of its regression-based approach, namely overfitting and high sensitivity to occlusions and noise. To address these fundamental issues, we propose a novel generative paradigm: instead of diffusing points, we diffuse a field of geometric transformations by leveraging the generative power of Diffusion Models.

Symmetry-Diffusion Module (Sym-Diffuser) The Sym-Diffuser is designed to generate a high-quality, structurally sound coarse completion by producing a field of symmetric transformations. Given an input point cloud $\boldsymbol{\mathcal{P}}_{in}$, we employ a conditional diffusion model, Sym-Diffuser, to generate its corresponding transformation field. The model learns the conditional distribution $p(\boldsymbol{\mathcal{T}}|\boldsymbol{\mathcal{F}}_k)$, where $\boldsymbol{\mathcal{F}}_k$ are features extracted from the input keypoints.

During training, we leverage the ground truth transformation field $\boldsymbol{\mathcal{T}}_{gt}$ from Stage 1 as clean data, denoted $\boldsymbol{\mathcal{Z}}_0$. We simulate the standard forward diffusion process, a sequential Markov chain that progressively corrupts $\boldsymbol{\mathcal{Z}}_0$ with Gaussian

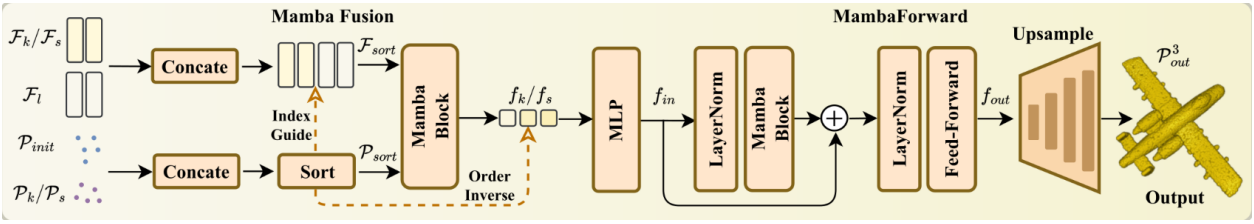

Figure 4: **Detailed architecture of the Mamba Fusion and MambaForward module.** It employs a **Mamba Fusion** block to coherently merge multi-source features and point coordinates. A core **MambaForward**, integrated within a feed-forward network, then progressively refines and upsamples the geometry to produce a high-fidelity output.

noise over $T = 100$ timesteps. This process creates a series of noisy versions $\boldsymbol{\mathcal{Z}}_t$, where the noise level at any timestep t is governed by a predefined variance schedule. This allows for efficient, direct sampling of any noisy version $\boldsymbol{\mathcal{Z}}_t$ from $\boldsymbol{\mathcal{Z}}_0$ via a closed-form expression.

Our diffusion model is trained to reverse this process. Internally, it functions as a noise predictor $\boldsymbol{\epsilon}_\theta$ that estimates the noise in a given input $\boldsymbol{\mathcal{Z}}_t$. This allows us to recover the predicted clean field $\hat{\boldsymbol{\mathcal{T}}}_\theta$ by algebraically inverting the forward process with the predicted noise. Inspired by Consistency Models (Song et al. 2023), our training objective is formulated as a weighted Mean Squared Error (MSE) between the predicted clean field $\hat{\boldsymbol{\mathcal{T}}}_\theta$ and the ground truth $\boldsymbol{\mathcal{T}}_{gt}$. A weighting function $\lambda(t)$ modulates the contribution of each timestep, prioritizing learning at different signal-to-noise ratios.

$$\mathcal{L}_{\text{proxy}} = \mathbb{E}_{t,\boldsymbol{\mathcal{Z}}_0,\boldsymbol{\epsilon}} \left[\lambda(t) \left\| \boldsymbol{\mathcal{T}}_{gt} - \hat{\boldsymbol{\mathcal{T}}}_\theta(\boldsymbol{\mathcal{Z}}_t, t, \boldsymbol{\mathcal{F}}_k) \right\|^2 \right] \quad (2)$$

At inference, the Sym-Diffuser takes a random Gaussian vector $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{N_k \times 12}$ and, conditioned on the features $\boldsymbol{\mathcal{F}}_k$, iteratively denoises it to produce a clean transformation field $\hat{\boldsymbol{\mathcal{T}}} \in \mathbb{R}^{N_k \times 12}$. Following (Yan et al. 2025), this field is composed of a point-wise affine matrix $\boldsymbol{A}_i \in \mathbb{R}^{3\times3}$ and a translation vector $\boldsymbol{T}_i \in \mathbb{R}^3$ for each keypoint $\mathbf{p}_i \in \boldsymbol{\mathcal{P}}_k$. The symmetric keypoints $\boldsymbol{\mathcal{P}}_s$ are then constructed by applying these transformations as follows:

$$\boldsymbol{\mathcal{P}}_s = \{\boldsymbol{A}_i\mathbf{p}_i + \boldsymbol{T}_i \mid \mathbf{p}_i \in \boldsymbol{\mathcal{P}}_k, (\boldsymbol{A}_i, \boldsymbol{T}_i) \in \hat{\boldsymbol{\mathcal{T}}}\} \quad (3)$$

The initial coarse completion is formed by the union of the partial keypoints and the generated missing part, $\boldsymbol{\mathcal{P}}_{init} = \boldsymbol{\mathcal{P}}_k \cup \boldsymbol{\mathcal{P}}_s \in \mathbb{R}^{2N_k\times3}$.

Cascaded Refinement with MBA-Refiner The coarse completion $\boldsymbol{\mathcal{P}}_{init}$, encoded by FE to $\boldsymbol{\mathcal{F}}_l$, is refined and upsampled by the MBA-Refiner, a three-block cascade balancing performance and efficiency. Each block follows a consistent design: Feature Fusion to integrate guidance, followed by MambaForward for refinement and upsampling. The blocks differ in their fusion strategy, tailored to point density. The refinement is guided by partial keypoint features $\boldsymbol{\mathcal{F}}_k$ and symmetric point features $\boldsymbol{\mathcal{F}}_s$.

The MBA-Refiner employs different feature fusion strategies across its three blocks to adapt to varying computational constraints at different point densities.

Blocks 1-2: Cross-Attention Fusion. At lower point densities ($l = 0, 1$), we employ cross-attention fusion for performance. The base features $\boldsymbol{\mathcal{F}}_l$ from the previous layer are refined by separately attending to each guidance source via Multi-head Cross-Attention (MCA). Specifically, $\boldsymbol{\mathcal{F}}_l$ attends to $\boldsymbol{\mathcal{F}}_k$ and $\boldsymbol{\mathcal{F}}_s$ independently, and the resulting context-aware features are concatenated and processed through a Multi-Layer Perceptron (MLP), denoted as a fusion function $\boldsymbol{\psi}$, to produce the unified feature set $\boldsymbol{f}_{in}^l$ for subsequent refinement, where $[\cdot]$ denotes the concatenation operation:

$$\boldsymbol{f}_{in}^l = \boldsymbol{\psi}\left(\left[\text{MCA}(\boldsymbol{\mathcal{F}}_l, \boldsymbol{\mathcal{F}}_g) \right]_{g\in\{k,s\}} \right) \quad (4)$$

Block 3: Mamba Fusion. At the highest point density ($l = 2$), where the $\mathcal{O}(N^2)$ complexity of attention is prohibitive, we employ an efficient Mamba-based fusion (MFusion) strategy as illustrated in Figure 4. The base features $\boldsymbol{\mathcal{F}}_l$ from the previous layer are fused with guidance features ($\boldsymbol{\mathcal{F}}_k$ and $\boldsymbol{\mathcal{F}}_s$) through spatial ordering and Mamba processing, then processed to produce the unified feature set $\boldsymbol{f}_{in}^3$ for subsequent refinement:

$$\boldsymbol{f}_{in}^3 = \boldsymbol{\psi}\left(\left[\text{MFusion}(\boldsymbol{\mathcal{F}}_l, \boldsymbol{\mathcal{F}}_g) \right]_{g\in\{k,s\}} \right) \quad (5)$$

Shared Refinement with MambaForward. After the fusion stage, all blocks utilize a shared-architecture **MambaForward** module for final feature refinement and upsampling. As shown in Figure 4, this module takes the fused features from the corresponding layer, $\boldsymbol{f}_{in}^l$, and processes them through a sequence of operations including an MLP, a Mamba block with residual connections, and an upsampling layer to directly produce the refined point cloud:

$$\boldsymbol{\mathcal{P}}_{out}^l = \text{MambaForward}(\boldsymbol{f}_{in}^l) \quad (6)$$

This shared design ensures consistent and powerful refinement across the cascade, achieving $2\times$ upsampling in Blocks 1-2 and $4\times$ in Block 3.

Overall Training Objective

Our Simba framework employs a two-stage training strategy. In Stage 1, SymmGT is pre-trained using Chamfer Distance loss to generate target transformation fields. In Stage 2, the complete framework is trained end-to-end with a

Methods	Avg CD-ℓ_1	Plane	Cabinet	Car	Chair	Lamp	Sofa	Table	Watercraft	F-Score@1%
PCN(3DV 2018)	9.64	5.50	22.70	10.63	8.70	11.00	11.34	11.68	8.59	0.695
PoinTr(ICCV2021)	8.38	4.75	10.47	8.68	9.39	7.75	10.93	7.78	7.29	-
SnowflakeNet(ICCV2021)	7.21	4.29	9.16	8.08	7.89	6.07	9.23	6.55	6.40	0.801
SeedFormer(ECCV2022)	6.74	3.85	9.05	8.06	7.06	5.21	8.85	6.05	5.85	0.818
ProxyFormer(CVPR2023)	6.77	4.01	9.01	7.88	7.11	5.35	8.77	6.03	5.98	-
AdaPoinTr(TPAMI2023)	6.53	3.68	8.82	7.47	6.85	5.47	8.35	5.80	5.76	0.845
SVDFormer(ICCV2023)	6.54	3.62	8.79	7.46	6.91	5.33	8.49	5.90	5.83	0.841
CRA-PCN(AAAI2024)	6.39	3.59	8.70	7.50	6.70	**5.06**	8.24	5.72	5.64	-
SymmCompletion(AAAI2025)	6.47	3.67	8.74	7.47	6.86	5.11	8.41	5.88	5.66	0.840
PointCFormer(AAAI2025)	6.41	3.53	8.73	7.32	**6.68**	5.12	8.34	5.86	5.74	0.855
DC-PCN(AAAI2025)	6.46	3.65	8.75	7.48	6.71	5.35	8.28	5.76	5.71	0.850
PCDreamer(CVPR2025)	6.52	**3.51**	**8.62**	**6.92**	6.91	5.66	8.31	6.27	5.90	**0.856**
Ours (Simba)	**6.34**	3.63	8.67	7.34	6.74	5.09	**8.17**	**5.62**	**5.51**	0.853

Table 1: Quantitative comparison on the PCN dataset. We report per-category Chamfer Distance (ℓ_1 CD $\times 10^3 \downarrow$) and average F-Score@1% $\uparrow$. Specifically, **Avg CD-ℓ_1** (average ℓ_1 Chamfer Distance) reflects reconstruction accuracy; **F-Score@1%** measures the geometric similarity. Results are from original papers or carefully reproduced under identical experimental settings.

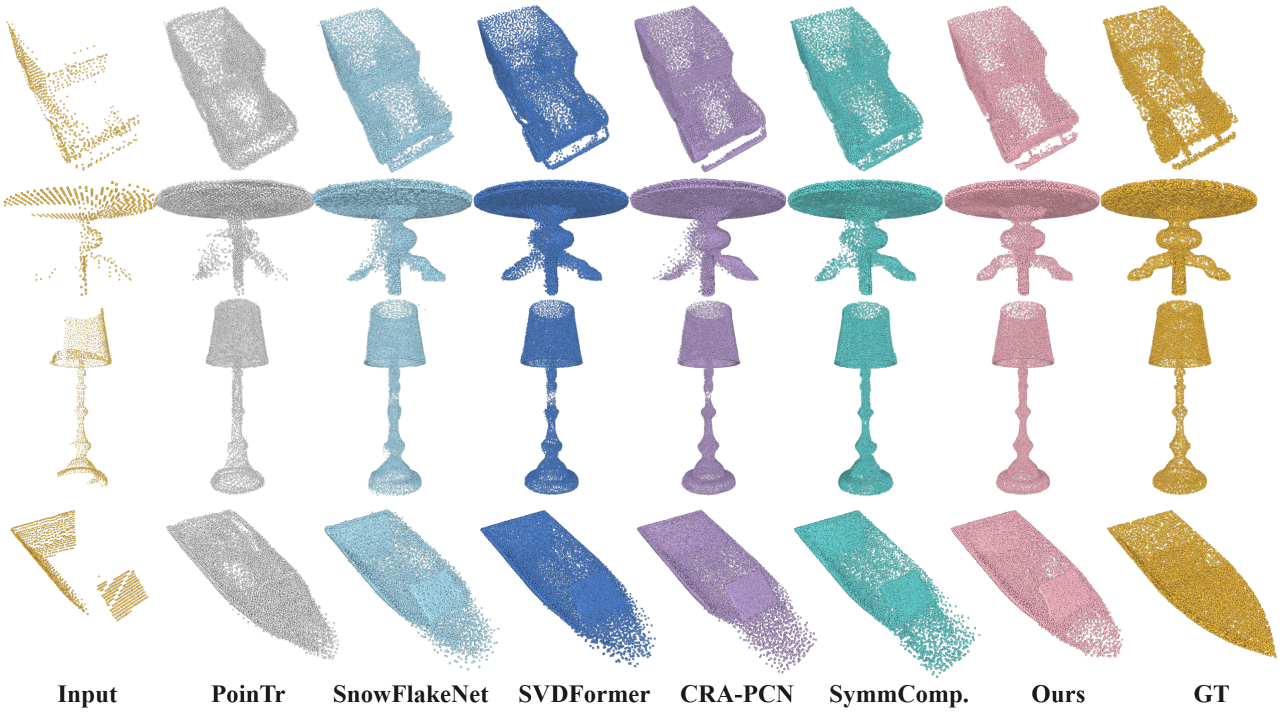

Figure 5: Qualitative comparison on the PCN dataset. Our **Simba** achieves superior geometric consistency and preserves fine details where other methods struggle.

composite objective that supervises both Sym-Diffuser and MBA-Refiner simultaneously:

$$\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{proxy}} + \sum_{l=1}^{3} L_{CD}(\mathcal{P}^l_{out}, \mathcal{P}_{gt}) \tag{7}$$

where $\mathcal{L}_{\text{proxy}}$ supervises the Sym-Diffuser module as defined in Eq. 2, and the Chamfer Distance terms supervise the MBA-Refiner cascade at each refinement stage $\mathcal{P}^l_{out}$. This multi-stage supervision ensures effective joint optimization of both generative and refinement components.

Experiments

Implementation Details

Our framework is implemented in PyTorch, and all experiments were conducted on four NVIDIA RTX 4090 GPUs.

Datasets and Evaluation Metrics

Datasets. Our experiments are conducted on three widely-used public datasets, covering both synthetic and real-world scenarios. **PCN** (Yuan et al. 2018) is a classic benchmark for point cloud completion, consisting of 8 categories from ShapeNet (Chang et al. 2015). We follow its official train/-

Method	ShapeNet-55				ShapeNet-34 (Seen)				ShapeNet-21 (Unseen)			
	CD-S	CD-M	CD-H	CD-Avg	CD-S	CD-M	CD-H	CD-Avg	CD-S	CD-M	CD-H	CD-Avg
FoldingNet	2.67	2.66	4.05	3.12	1.86	1.81	3.38	2.35	2.76	2.74	5.36	3.62
PCN	1.94	1.96	4.08	2.66	1.87	1.81	2.97	2.22	3.17	3.08	5.29	3.85
PoinTr	0.67	1.05	2.02	1.25	0.76	1.05	1.88	1.23	1.04	1.67	3.44	2.05
SeedFormer	0.50	0.77	1.49	0.92	0.48	0.70	1.30	0.83	0.61	1.07	2.35	1.34
AdaPoinTr	0.49	0.69	**1.24**	0.81	0.48	0.63	**1.07**	0.73	0.61	**0.96**	**2.11**	1.23
SVDFormer	0.48	0.70	1.30	0.83	0.46	0.65	1.13	0.75	0.61	1.05	2.19	1.28
CRA-PCN	0.48	0.71	1.37	0.85	0.45	0.65	1.18	0.76	0.55	0.97	2.19	1.24
Ours (Simba)	**0.45**	**0.66**	1.25	**0.79**	**0.43**	**0.59**	1.08	**0.70**	**0.54**	0.97	2.18	**1.23**

Table 2: Comparison of point cloud completion performance on ShapeNet-55, ShapeNet-34 (seen), and ShapeNet-21 (unseen) datasets. All metrics report the L2 Chamfer Distance ($\times 10^3$); lower is better. Best performance in each block is marked in **bold**.

Metric	crapcn	SeedFormer	EINet	SymmComp.	Ours
MMD	1.737	0.516	0.967	0.970	**0.423**

Table 3: Quantitative comparison on the KITTI dataset using MMD ($\times 10^3$). Lower is better.

validation/test split. **ShapeNet-55/34** (Yu et al. 2021) is a large-scale dataset derived from ShapeNet.We evaluate generalization on all 55 categories and the 34-seen/21-unseen split. **KITTI** (Geiger et al. 2013) provides real-world LiDAR vehicle scans. We use this dataset to assess the robustness and generalization performance of our model on sparse, noisy, and partial data from a different domain.

Evaluation Metrics. We employ standard metrics to quantitatively evaluate the completion quality. For synthetic datasets (PCN and ShapeNet), we use **Chamfer Distance (CD)** and **F-Score**. We report L1-CD for PCN and L2-CD for ShapeNet, following common practice. For a more fine-grained analysis on ShapeNet, we also report the L2-CD on three difficulty levels: Simple (S), Moderate (M), and Hard (H). The F-Score (1% threshold) measures surface accuracy and is less sensitive to outliers.

Results on PCN Dataset

Quantitative Analysis. We evaluate our **Simba** on the PCN benchmark (Yuan et al. 2018). Table 1 presents the quantitative comparison against state-of-the-art methods in terms of L1 Chamfer Distance (CD) and F1-Score. Simba achieves state-of-the-art performance on several categories (e.g., Sofa, Table, Watercraft). In other categories, our accuracy is also competitive with existing leading approaches. Most importantly, we obtain the best overall performance in terms of average Chamfer Distance. Furthermore, our method demonstrates satisfactory precision on the F-Score@1% metric.

Notably, our model demonstrates superior performance over SymmCompletion.We attribute this to transformation diffusion: SymDiffuser learns more robust symmetric priors than deterministic regressions.

Qualitative Analysis. Figure 5 shows completions with strong geometric consistency and high fidelity. For complex shapes like 'Table' and 'Car', our model faithfully reconstructs intricate features, such as distinct table legs and the car's smooth, complete body. The results are free from the distorted geometry and fragmented artifacts that are prevalent in outputs from CAR-PCN, PoinTr, and SVDFormer. Simba's proficiency is not limited to large-scale coherence; it also excels at preserving fine-grained details. It restores the 'Watercraft' with a pristine, continuous hull, unlike the noisy results from SnowFlakeNet, and accurately reproduces the 'Lamp's' ornate stem, a challenging feature where competing methods often resort to coarse approximations. This balanced performance validates our method's ability to generate completions that are both globally plausible and locally precise.

Results on ShapeNet Datasets

We conduct experiments on ShapeNet-55 with its 34 seen/21 unseen category split to evaluate model performance and generalization capability.

Quantitative Analysis. The quantitative results are presented in Table 2. On the full **ShapeNet-55** benchmark, Simba demonstrates strong performance, achieving a competitive average L2-CD of **0.79** ($\times 10^3$) and outperforming most prior works. This supports the effectiveness of our approach across diverse categories. More importantly, the results on the **ShapeNet-34/21** split highlight the superior generalization ability of our framework. Our model not only achieves state-of-the-art performance on the 34 seen categories but also maintains a strong lead on the 21 unseen categories. This indicates our learned generative prior is more robust and generalizable than deterministic regression, enabling better performance on novel object classes.

Results on Real-World Data

Quantitative and Qualitative Analysis on KITTI. To assess real-world performance, we evaluate on the KITTI dataset (Geiger et al. 2013), which contains sparse and noisy LiDAR scans. This is a challenging domain-generalization test since we train only on synthetic data. As reported in Table 3, **Simba** achieves highly competitive performance, highlighting the robustness of our transformation-based generative approach. Figure 1 further show that our method

ID	Configuration	CD-ℓ_1 ($\times 10^3$) $\downarrow$
A1	**Diffusion Model (Ours)**	**6.34**
A2	Transformer Regression	6.48
B1	**3-layer: [2×, 2×, 4×]**	**6.34**
B2	1-layer: [16×]	6.70
B3	2-layer: [2×, 8×]	6.56
B4	2-layer: [4×, 4×]	6.52

Table 4: Ablation study on PCN: (A) prediction module and (B) progressive upsampling strategy. All upsampling variants share a total $16\times$ upsampling factor.

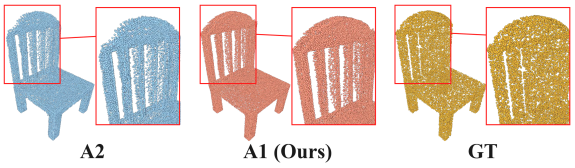

Figure 6: Ablation study on the prediction module (A).

ID	Fusion Strategy per Stage	Memory	CD-ℓ_1
C1	**[CA, CA, MFusion]** (Ours)	14.7 GB	**6.34**
C2	[MLP, MLP, MFusion]	12.1 GB	6.49
C3	[CA, CA, MLP]	**12.0 GB**	6.41
C4	[CA, CA, CA]	16.4 GB	6.35
C5	[MFusion, MFusion, MFusion]	13.8 GB	6.43

Table 5: Ablation study on PCN for MBA-Refiner fusion strategies (C). Different fusion combinations are compared using CD-ℓ_1 scores ($\times 10^3$). CA denotes Cross-Attention and MFusion denotes Mamba Fusion.

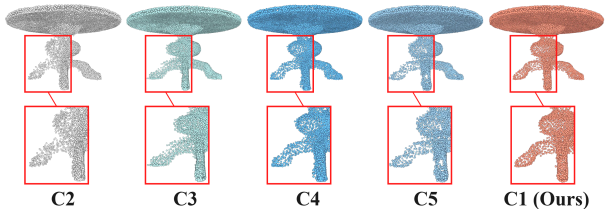

Figure 7: Qualitative comparison of fusion strategies (C).

produces structurally plausible vehicle shapes, avoiding the floating artifacts and scale inconsistencies that plague many competing approaches when facing a domain gap.

Ablation Studies and Analysis

We conduct ablation studies on the PCN dataset to analyze our three components: the diffusion-based transformation prediction, the progressive upsampling strategy, and the heterogeneous MBA-Refiner architecture.

Analysis of the Transformation Prediction Module. As shown in Table 4, our diffusion-based transformation prediction (A1) clearly outperforms a conventional Transformer regressor (A2), achieving a CD score of 6.34 versus 6.48. The visual results in Figure 6 further reinforce this finding; our model reconstructs a geometrically coherent shape, while the regressor produces noticeable and distracting structural artifacts. Overall, these results validate that learning a generative distribution of transformations is more robust, overcoming the critical overfitting to which direct deterministic regression is prone.

Analysis of the Progressive Upsampling Strategy We analyze our progressive upsampling strategy in Table 4, comparing different structures that all yield a total $16\times$ upsampling factor. Our three-level upsampling (**B1**) with a gradual $[2\times, 2\times, 4\times]$ schedule achieves the best performance (6.34 CD-L1). In contrast, a single-level, aggressive $16\times$ upsampling (**B2**) performs worst (6.70), highlighting the difficulty of direct coarse-to-fine mapping. The two-level variants confirm this trend: the balanced $[4\times, 4\times]$ schedule (**B4**) significantly outperforms the unbalanced $[2\times, 8\times]$ strategy (**B3**). This demonstrates that a gradual, multi-level refinement is crucial, and our three-level upsampling approach provides an optimal structure for learning complex geometries.

Analysis of the MBA-Refiner Architecture. We ablate our MBA-Refiner to validate its heterogeneous cascade design. As shown in Table 5, our full model (**C1**), which uses Cross-Attention in early stages and Mamba in the final stage, sets the performance benchmark with a Chamfer Distance of **6.34**. First, we test homogeneous architectures. A cascade using only Cross-Attention (**C4**) exhibits high memory consumption while achieving a slightly inferior score of **6.35**, demonstrating computational inefficiency. Conversely, using only Mamba blocks (**C5**) is more memory-efficient but results in a higher error of **6.43**, proving Cross-Attention is vital for initial feature processing. We then assess simpler fusion methods. Replacing our sequence-based fusion with an MLP-based approach (**C2**) or using an MLP in the final block (**C3**) also harms performance, yielding inferior scores of **6.49** and **6.41** respectively, despite their lower memory footprints. Figure 7 visually confirms these findings, showing that both the MLP-based (**C2**) and all-Mamba (**C5**) variants generate distorted geometries, unlike the high-fidelity structure recovered by our model. These results confirm that our heterogeneous cascade (**C1**) strikes an optimal balance between performance and computational efficiency.

Conclusion

We introduce Simba, a novel paradigm for point cloud completion which reformulates the task as learning the distribution of geometric transformations via diffusion. This is achieved via two core components: (1) a Symmetry-Diffusion mechanism (Sym-Diffuser) to address the overfitting and noise-sensitivity issues of prior direct-regression methods, and (2) a cascaded Mamba-based architecture (MBA-Refiner) for high-fidelity upsampling. Extensive experiments demonstrate that Simba learns robust transformation representations, enabling strong cross-domain generalizability on the real-world KITTI benchmark, achieving state-of-the-art performance on multiple benchmarks. Consequently, Simba establishes a new direction for diffusion-based point cloud completion.

Acknowledgements

This work was supported in part by the Natural Science Foundation of China (No. 62272227 and No. U25A20533).

References

Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; and Leonard, J. J. 2017. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6): 1309–1332.

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*.

Cui, R.; Qiu, S.; Anwar, S.; Liu, J.; Xing, C.; Zhang, J.; and Barnes, N. 2023. P2C: Self-Supervised Point Cloud Completion from Single Partial Clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 14351–14360.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32: 1231–1237.

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

Gu, A.; and Dao, T. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Kasten, Y.; Rahamim, O.; and Chechik, G. 2023. Point cloud completion with pretrained text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 36: 12171–12191.

Li, A.; Zhu, Z.; and Wei, M. 2025. GenPC: Zero-shot Point Cloud Completion via 3D Generative Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1308–1318.

Li, S.; Gao, P.; Tan, X.; and Wei, M. 2023a. Proxyformer: Proxy alignment assisted point cloud completion with missing part sensitive transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9466–9475.

Li, S.; Gao, P.; Tan, X.; and Xiang, W. 2023b. RLGrid: Reinforcement Learning Controlled Grid Deformation for Coarse-to-Fine Point Could Completion. *IEEE Transactions on Multimedia*, 1–16.

Li, Y.; Yang, W.; and Fei, B. 2024. 3dmambacomplete: Exploring structured state space model for point cloud completion. *arXiv preprint arXiv:2404.07106*.

Li, Z.; Gao, P.; You, K.; Yan, C.; and Paul, M. 2024. Global attention-guided dual-domain point cloud feature learning for classification and segmentation. *IEEE Transactions on Artificial Intelligence*.

Liang, D.; Zhou, X.; Xu, W.; Zhu, X.; Zou, Z.; Ye, X.; Tan, X.; and Bai, X. 2024. Pointmamba: A simple state space model for point cloud analysis. *Advances in neural information processing systems*, 37: 32653–32677.

Lyu, Z.; Kong, Z.; Xu, X.; Pan, L.; and Lin, D. 2021. A conditional point diffusion-refinement paradigm for 3d point cloud completion. *arXiv preprint arXiv:2112.03530*.

Ma, C.; Chen, Y.; Guo, P.; Guo, J.; Wang, C.; and Guo, Y. 2023. Symmetric Shape-Preserving Autoencoder for Unsupervised Real Scene Point Cloud Completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13560–13569.

Nunes, L.; Marcuzzi, R.; Mersch, B.; Behley, J.; and Stachniss, C. 2024. Scaling diffusion models to real-world 3d lidar scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14770–14780.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 652–660.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

Rong, Y.; Zhou, H.; Yuan, L.; Mei, C.; Wang, J.; and Lu, T. 2024. Cra-pcn: Point cloud completion with intra-and inter-level cross-resolution transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 4676–4685.

Schiebener, D.; Schmidt, A.; Vahrenkamp, N.; and Asfour, T. 2016. Heuristic 3D object shape completion based on symmetry and scene context. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 74–81.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y.; Dhariwal, P.; Chen, M.; and Sutskever, I. 2023. Consistency Models. In *NeurIPS 2023 Workshop on Diffusion Models*. Workshop paper.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, J.; Cui, Y.; Guo, D.; Li, J.; Liu, Q.; and Shen, C. 2024. Pointattn: You only need attention for point cloud completion. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 38, 5472–5480.

Wei, G.; Feng, Y.; Ma, L.; Wang, C.; Zhou, Y.; and Li, C. 2025. Pcdreamer: Point cloud completion through multi-view diffusion priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 27243–27253.

Wen, X.; Xiang, P.; Han, Z.; Cao, Y.-P.; Wan, P.; Zheng, W.; and Liu, Y.-S. 2021. Pmp-net: Point cloud completion by learning multi-step point moving paths. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7443–7452.

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 *Proceedings of the IEEE/CVF international conference on computer vision*, 5499–5509.

Xie, H.; Yao, H.; Zhou, S.; Mao, J.; Zhang, S.; and Sun, W. 2020. Grnet: Gridding residual network for dense point cloud completion. In *European conference on computer vision*, 365–381. Springer.

Yan, H.; Li, Z.; Luo, K.; Lu, L.; and Tan, P. 2025. Symm-Completion: High-Fidelity and High-Consistency Point Cloud Completion with Symmetry Guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 9094–9102.

Yang, Y.; Feng, C.; Shen, Y.; and Tian, D. 2018. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 206–215.

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

Yu, Z.; Zhang, R.; Ying, J.; Yu, J.; Hu, X.; Luo, L.; Cao, S.-Y.; and Shen, H.-L. 2024. Context and geometry aware voxel transformer for semantic scene completion. *Advances in Neural Information Processing Systems*, 37: 1531–1555.

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. IEEE.

Zhang, S.; Liu, X.; Xie, H.; Nie, L.; Zhou, H.; Tao, D.; and Li, X. 2023. Learning Geometric Transformation for Point Cloud Completion. *International Journal of Computer Vision*, 1–21.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 16259–16268.

Zhao, S.; Gao, C.; Shao, Y.; Li, L.; Yu, C.; Ji, Z.; and Sang, N. 2020. Gtnet: Generative transfer network for zero-shot object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 12967–12974.

Zuo, Z.; Zhuang, C.; Gao, P.; Qin, J.; Feng, H.; and Sebe, N. 2025. Uni4D: A Unified Self-Supervised Learning Framework for Point Cloud Videos. arXiv:2504.04837.