MonoArt: Progressive Structural Reasoning for Monocular Articulated 3D Reconstruction

Haitian Li*, Haozhe Xie*, Junxiang Xu, Beichen Wen, Fangzhou Hong, and Ziwei Liu ✉

S-Lab, Nanyang Technological University, 637335 Singapore
https://lihaitian.com/MonoArt

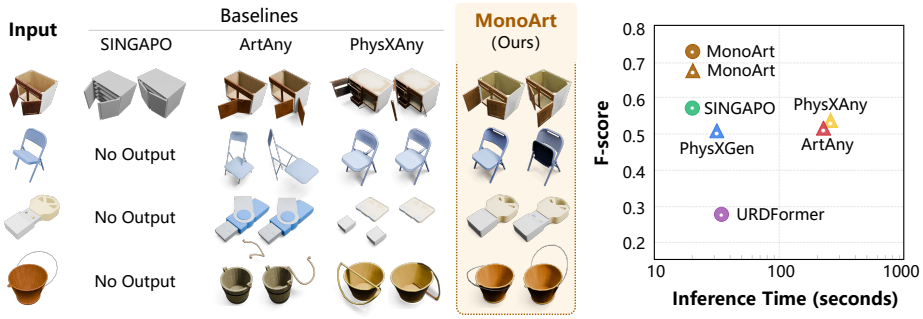

Fig. 1: (Left) Qualitative results of SINGAPO [31], Articulate-Anything (ArtAny) [23], PhysX-Anything (PhysXAny) [3], and MonoArt on diverse objects. **(Right)** F-score vs. inference time on the PartNet-Mobility [59] test set. Circles indicate models evaluated on 7 categories, while triangles denote models supporting all 46 categories.

Abstract. Reconstructing articulated 3D objects from a single image requires jointly inferring object geometry, part structure, and motion parameters from limited visual evidence. A key difficulty lies in the entanglement between motion cues and object structure, which makes direct articulation regression unstable. Existing methods address this challenge through multi-view supervision, retrieval-based assembly, or auxiliary video generation, often sacrificing scalability or efficiency. We present MonoArt, a unified framework grounded in progressive structural reasoning. Rather than predicting articulation directly from image features, MonoArt progressively transforms visual observations into canonical geometry, structured part representations, and motion-aware embeddings within a single architecture. This structured reasoning process enables stable and interpretable articulation inference without external motion templates or multi-stage pipelines. Extensive experiments on PartNet-Mobility demonstrate that MonoArt achieves state-of-the-art performance in both reconstruction accuracy and inference speed. The framework further generalizes to robotic manipulation and articulated scene reconstruction.

Keywords: Monocular Articulated 3D Reconstruction · Progressive Structural Reasoning · Kinematic Estimation

* Equal Contribution ✉ Corresponding Author

1 Introduction

Reconstructing articulated 3D objects from visual observations remains a fundamental challenge in computer vision and graphics. Such objects (*e.g.*, laptops, cabinets) are ubiquitous in daily environments, and generating high-fidelity, interactable 3D assets at scale is increasingly vital for applications in robotics [1, 13, 19, 63, 65] and scene synthesis [57, 61, 62, 67]. Despite recent progress in 3D object generation [12, 22, 52, 60], producing high-quality articulated assets still requires extensive manual effort. Unlike rigid reconstruction, articulated reconstruction demands understanding both an object's structural composition and the kinematic relationships among its parts.

Articulated 3D object modeling [8, 16, 26, 27, 72] has attracted increasing attention in recent years, driving significant advances in reconstruction. Existing approaches can be broadly classified into three categories. Most existing methods [17, 32, 41, 56, 58] rely on multiple images captured from videos or frame sequences indicating different articulation states (*e.g.*, open and closed). While achieving higher reconstruction accuracy, they require multiple motion states of the same object, which are not always readily available in practice. To relax this constraint, subsequent approaches [9, 23, 31] take a single image as input but reconstruct 3D objects by retrieving and assembling parts from pre-built asset libraries, often leading to texture misalignment and geometric inaccuracies. Very recent works move beyond retrieval and attempt single-image reconstruction of articulated objects by generating auxiliary videos [38], leveraging vision-language models [3, 4], or relying on predefined motion directions to infer articulation cues [15, 28]. While these approaches demonstrate the potential of single-image articulation modeling, the former is complex and computationally expensive, whereas the latter depends on handcrafted priors that limit generalization. This reliance on external cues reflects the lack of intrinsic 3D understanding, making it difficult to infer part composition and spatial relationships directly from a single image.

To address these limitations, we propose **MonoArt**, an end-to-end framework for monocular articulated 3D reconstruction grounded in progressive structural reasoning. Instead of directly regressing articulation parameters from image features, MonoArt progressively constructs structured part representations and lifts them into motion-aware embeddings, enabling stable and interpretable kinematic prediction. Specifically, **TRELLIS-based 3D Generator** first produces a canonical 3D shape from the input image, providing a stable geometric foundation. **Part-aware Semantic Reasoner** then lifts geometry-aligned point features into globally contextualized part-level embeddings through triplane aggregation and transformer-based refinement under 3D structural supervision. These part-aware features are subsequently processed by a **Dual-Query Motion Decoder**, which decouples spatial motion anchors and semantic part representations, and performs iterative refinement to reason about component-level motion patterns. Finally, **Kinematic Estimator** predicts part-level articulation parameters, including the part centroid, mask, joint type, axis, origin, and motion limits, and infers the kinematic tree structure, yielding a coherent

articulated 3D reconstruction. As illustrated in Figure 1, MonoArt achieves both higher F-score and substantially lower inference time than existing approaches, demonstrating that explicit structural priors enable not only more accurate but also more efficient articulated reconstruction.

The contributions are summarized as follows:

- We show that embedding 3D structural priors simplifies single-image articulated reconstruction by eliminating reliance on video generation, handcrafted motion templates, or vision-language priors, enabling reliable part decomposition and articulation reasoning.
- We propose MonoArt, a unified end-to-end framework that progressively reasons from geometry to kinematics, disentangling shape recovery, part-aware encoding, motion decoding, and kinematic regression for stable and physically meaningful articulation inference.
- MonoArt achieves state-of-the-art performance on PartNet-Mobility in both geometric and articulation metrics while significantly reducing inference time. The framework further generalizes to robotic manipulation and indoor 3D reconstruction tasks.

2 Related Work

Articulated Object Modeling. Articulated object modeling aims to recover object geometry together with part structure and motion from visual observations. Early multi-view methods [50, 56] adopt neural implicit representations to model category-level articulated shapes as deformations of a canonical template, enabling view synthesis but without explicit part or kinematic modeling. Later works introduce explicit part decomposition and motion reasoning. PARIS [32] aligns reconstructions across articulation states to estimate rigid parts and transformations. DTA [58] and ArticulatedGS [17] jointly model geometry, segmentation, and articulation parameters from multi-view RGB-D or Gaussian Splatting, producing motion-ready digital twins with high geometric fidelity. More recent approaches reduce input requirements by leveraging generative or language priors. FreeArt3D [7] optimizes geometry from sparse articulated views, while SINGAPO [31], NAP [24], and MeshArt [14] predict articulation trees and synthesize articulated parts. Articulate-Anything [23] formulates the task as vision-language reasoning to infer symbolic part hierarchies, and PhysX-Anything [3] further incorporates VLM priors to predict physically plausible structures and interactions. These methods improve scalability and controllability, often at the expense of precise instance-level reconstruction.

3D Part Segmentation. 3D part segmentation aims to decompose objects into semantic parts. Early learning-based methods [18,29,30,37,45–47,54,55,71] focus on fully supervised point- or mesh-level classification using curated 3D datasets [6, 10, 42, 70]. While effective, these approaches are limited by the scale and diversity of part annotations. To improve open-world generalization and enable zero-shot capabilities, recent methods leverage 2D foundation vision models

and complementary 2D segmentation priors [21,25,36,44,48,69]. PartSLIP [35], PartSLIP++ [73], and ZeroPS [66] transfer image-language and segmentation priors to 3D via multi-view reasoning or prompt-based inference, while PartDistill [53], SaMesh [51], and SAMPart3D [68] distill 2D foundation features into geometric representations. More recently, scaling-based approaches train feed-forward 3D segmentation models with large-scale part annotations. Find3D [40] leverages foundation models for pseudo-label generation, PartField [33] learns ambiguity-aware continuous feature fields, and P3-SAM [39] and PartSAM [74] demonstrate that large-scale part supervision yields strong point-wise representations for part prompting.

3 Our Approach

3.1 Overview

MonoArt reconstructs articulated 3D objects from a single image by progressively transforming visual observations into geometry-aware, part-aware, and motion-aware representations. As shown in Fig. 2, the framework consists of four components: TRELLIS-based 3D Generator (Sec. 3.2), Part-Aware Semantic Reasoner (Sec. 3.3), Dual-Query Motion Decoder (Sec. 3.4), and Kinematic Estimator (Sec. 3.5).

Given an input image $\mathbf{I}$, TRELLIS-based 3D Generator reconstructs a canonical geometry $\mathbf{O}$ using a frozen TRELLIS backbone and produces geometry-aligned latent features $\mathbf{Z}$. Based on $(\mathbf{O}, \mathbf{Z})$, Part-Aware Semantic Reasoner derives part-aware features $\mathbf{H}$ that encode explicit part decomposition guided by 3D structural annotations. Then, Dual-Query Motion Decoder initializes position and content queries $(\mathbf{Q}_p^0, \mathbf{Q}_c^0)$ from $(\mathbf{H}, \mathbf{F}_{\text{geo}})$ and iteratively refines them to obtain motion-aware representations $(\mathbf{Q}_p^L, \mathbf{Q}_c^L)$, jointly reasoning about part localization and motion semantics. Finally, Kinematic Estimator transforms the refined queries into explicit articulation parameters, namely part masks $\mathbf{m}_m$, motion types $\mathbf{m}_t$, motion origins $\mathbf{m}_o$, motion axes $\mathbf{m}_a$, and motion limits $\mathbf{m}_l$, while inferring the kinematic hierarchy to produce an articulated 3D representation.

3.2 TRELLIS-based 3D Generator

Given a single RGB image $\mathbf{I}$, MonoArt first reconstructs a canonical 3D object geometry using TRELLIS [60] as a frozen 3D generation backbone. TRELLIS predicts a structured sparse voxel latent representation $\mathbf{Z} \in \mathbb{R}^{N_z \times N_z \times N_z \times d_1}$, where N_z denotes the voxel grid resolution and each active voxel stores a d_1-dimensional feature while empty regions remain inactive, forming a spatially structured sparse volume. The latent $\mathbf{Z}$ is subsequently decoded by the mesh decoder of TRELLIS to produce an explicit 3D mesh $\mathbf{O}$, which serves as the canonical geometry for downstream part reasoning and articulation inference.

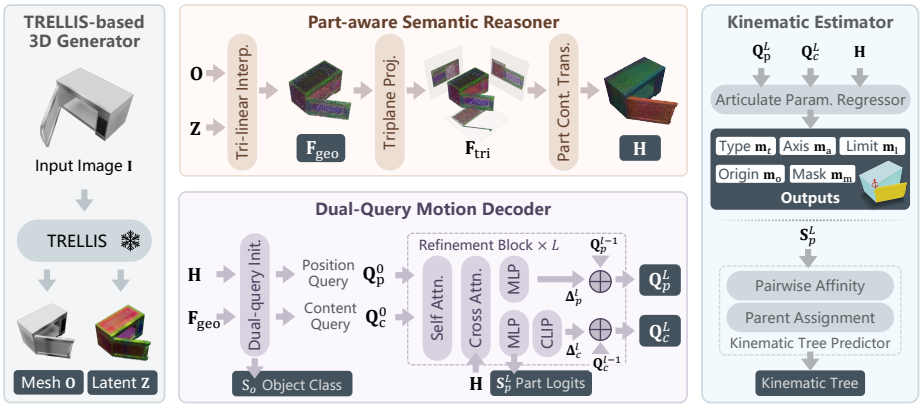

Fig. 2: Overview of MonoArt. TRELLIS-based 3D Generator reconstructs a canonical shape from a single image. Part-Aware Semantic Reasoner derives tri-plane-based part embeddings. Dual-Query Motion Decoder performs iterative motion reasoning, and Kinematic Estimator predicts part-level articulation parameters (motion type, origin, axis, limits) and infers the kinematic tree structure. Note that "Attn.", "Interp.", "Proj.", "Cont.", "Trans.", and "Init." represent "Attention", "Interpolation", "Projection", "Contrast", "Transformer", and "Initialization", respectively. $\oplus$ and $\otimes$ denote element-wise addition and matrix multiplication, respectively.

3.3 Part-Aware Semantic Reasoner

Our goal is to derive part-aware point features $\mathbf{H}$ from the canonical geometry $\mathbf{O}$ and the sparse voxel latent $\mathbf{Z}$ predicted by TRELLIS-based 3D generator, so that downstream modules can reason about object parts and their motions.

Tri-linear Interpolation. We sample M point sets $\{\mathbf{p}_m\}_{m=1}^{M}$ on the surface of $\mathbf{O}$, where $\mathbf{p}_m \in \mathbb{R}^3$. Given the sparse voxel feature volume $\mathbf{Z}$, we obtain point-aligned features by tri-linear interpolation over the neighboring voxels:

$$\mathbf{f}_m = \mathrm{TrilinearInterp}(\mathbf{Z}, \mathbf{p}_m), \qquad \mathbf{F}_{\mathrm{geo}} = \{\mathbf{f}_m\}_{m=1}^{M}. \tag{1}$$

Here, each point $\mathbf{p}_m$ is first mapped to its corresponding continuous voxel coordinate in the $N_z \times N_z \times N_z$ grid. The feature $\mathbf{f}_m \in \mathbb{R}^{d_1}$ is then computed as the weighted combination of the eight neighboring voxel features according to standard tri-linear interpolation [64]. This converts the discrete voxel latent $\mathbf{Z}$ into continuous surface-aligned features $\mathbf{F}_{\mathrm{geo}} \in \mathbb{R}^{M \times d_1}$.

Triplane Projection. To incorporate global spatial context while preserving geometric structure, we project the geometry-aligned point features $\mathbf{F}_{\mathrm{geo}}$ onto three orthogonal planes (XY, YZ, ZX) following the triplane formulation [5]. Specifically, each surface point $\mathbf{p}_m$ is orthographically projected onto the three planes according to its 3D coordinates, and its feature $\mathbf{f}_m$ is accumulated to the corresponding 2D grid locations via bilinear interpolation. This yields three feature maps of resolution $N_t \times N_t$, forming $\mathbf{F}_{\mathrm{tri}} \in \mathbb{R}^{3 \times N_t \times N_t \times d_1}$.

Part Contrast Transformer. The triplane features $\mathbf{F}_{\mathrm{tri}}$ are flattened into a token sequence $\mathbf{T} \in \mathbb{R}^{3N_t^2 \times d_1}$ and processed by a multi-layer self-attention

Transformer to capture global interactions across planes, producing refined tokens $\mathbf{T}'$. The refined tokens are reshaped back into triplane feature maps $\mathbf{F}'_{\text{tri}} \in \mathbb{R}^{3\times N_t\times N_t\times d_1}$. For each surface point $\mathbf{p}_m$, we project its 3D coordinate onto the three orthogonal planes and query the refined triplane feature maps via a tri-plane sampling operation. This yields the updated point embedding

$$\mathbf{h}_m = \text{MLP}(\text{TriQuery}(\mathbf{F}'_{\text{tri}}, \mathbf{p}_m)), \qquad \mathbf{H} = \{\mathbf{h}_m\}_{m=1}^{M}, \tag{2}$$

where TriQuery projects $\mathbf{p}_m$ onto the XY/YZ/ZX planes, bilinearly samples each plane feature map, and concatenates the three sampled features. Then, the MLP then lifts the aggregated feature from dimension d_1 to d_2. These embeddings encode part-aware structural representations and are supervised by 3D part annotations using triplet loss [49].

3.4 Dual-Query Motion Decoder

Articulated reasoning requires jointly modeling what constitutes a movable part and where its motion is spatially anchored. To this end, we adopt a dual-query formulation that disentangles semantic representation and geometric localization via two complementary query types: a content query $\mathbf{Q}_c \in \mathbb{R}^{N_q\times d_2}$ encoding part semantics, and a position query $\mathbf{Q}_p \in \mathbb{R}^{N_q\times 3}$ representing spatial motion anchors, where N_q denotes the number of dual queries. The dual queries are initialized from global object context and subsequently refined through L stacked refinement blocks in an iterative manner.

Dual-query Initialization. Given the part-aware point embeddings $\mathbf{H}$ and geometry-aligned features $\mathbf{F}_{\text{geo}}$, we first aggregate global object context by applying global pooling over $\mathbf{H}$ and $\mathbf{F}_{\text{geo}}$, followed by feature concatenation to obtain an object-level representation. This aggregated feature is projected to generate the initial content queries $\mathbf{Q}_c^0 \in \mathbb{R}^{N_q\times d_2}$ and position queries $\mathbf{Q}_p^0 \in \mathbb{R}^{N_q\times 3}$, together with an auxiliary object-category prediction S_o.

Refinement Block. The refinement block iteratively updates the dual queries over L layers to progressively refine articulation hypotheses. Given the initialized position query $\mathbf{Q}_p^0 \in \mathbb{R}^{N_q\times 3}$ and content query $\mathbf{Q}_c^0$, each layer first applies self-attention to model inter-part interactions, followed by cross-attention where the queries attend to the visual feature $\mathbf{H}$ to retrieve image evidence. For the l-th layer, the position query $\mathbf{Q}_p^l$ represents spatial motion anchors that localize candidate articulation points. It is updated via a residual scheme $\mathbf{Q}_p^l = \mathbf{Q}_p^{l-1} + \Delta_p^l$, allowing progressive refinement of spatial motion anchors. In parallel, the content query is also updated in a residual manner $\mathbf{Q}_c^l = \mathbf{Q}_c^{l-1} + \Delta_c^l$, where Δ_c^l captures semantic refinement derived from self- and cross-attention. Based on the refined content queries, we predict per-query part class logits $\mathbf{S}_p^l$, which provide structural supervision and are further used to retrieve semantic prototypes from frozen CLIP text embeddings. The retrieved prototypes are fused into the content branch to enhance part-level semantic consistency.

Query Confidence Estimation. Since the number of dual queries N_q defines an upper bound on articulated components, some queries may correspond to invalid part hypotheses. To address this, we predict a confidence score $c_i \in [0, 1]$ from the refined content embedding $\mathbf{Q}_{c,i}^L$ of the i-th query, indicating the reliability of its part hypothesis. During training, confidence supervision is derived from Hungarian matching between predicted part masks and ground-truth components. Matched queries are assigned confidence targets proportional to their mask overlap, while unmatched queries are treated as null hypotheses with zero confidence. At inference time, queries with confidence below a threshold are discarded, allowing the model to automatically determine the number of parts.

3.5 Kinematic Estimator

Given $\mathbf{Q}_p^L$, $\mathbf{Q}_c^L$, part logits $\mathbf{S}_p^L$, and point embedding $\mathbf{H}$, Kinematic Estimator predicts articulation parameters and a kinematic tree.

Articulate Parameter Regressor. The refined dual queries are $\mathbf{Q}_p^L \in \mathbb{R}^{N_q \times 3}$ and $\mathbf{Q}_c^L \in \mathbb{R}^{N_q \times d_2}$, where $\mathbf{Q}_p^L$ is interpreted as part centroid, *i.e.*, the spatial center of an articulated component.
Part mask is obtained by query-point matching. Specifically, we compute the affinity between content queries and point features $\mathbf{H} \times \mathbb{R}^{M \times d_2}$:

$$\mathbf{m}_m = \mathbf{Q}_c^L \mathbf{H}^\top, \qquad \mathbf{m}_m \in \mathbb{R}^{N_q \times M}. \tag{3}$$

Each row of $\mathbf{m}_m$ indicates the soft assignment of surface points to a query-induced articulated part.
Joint parameters are regressed from query-level representations. We integrate $\mathbf{Q}_p^L$, $\mathbf{Q}_c^L$, and part-level features $\mathbf{H}$ into a unified per-query representation, which is processed by lightweight MLP heads to predict physically interpretable joint parameters:

- **Joint type** $\mathbf{m}_t \in \mathbb{R}^{N_q \times N_t}$, where N_t denotes the number of predefined motion categories (*e.g.*, fixed, revolute, prismatic, and continuous);
- **Joint axis** $\mathbf{m}_a \in \mathbb{R}^{N_q \times 3}$, representing the unit direction vector of the predicted joint axis;
- **Joint pivot** $\mathbf{m}_o \in \mathbb{R}^{N_q \times 3}$, denoting the pivot point through which the motion axis passes;
- **Joint limits** $\mathbf{m}_l \in \mathbb{R}^{N_q \times 2}$, parameterized by a center and a symmetric span.

Kinematic Tree Predictor. The articulated objects are modeled as tree-structured kinematic graphs. We use $i \in \{1, ..., N_q\}$ to index queries. The part logits are given by $\mathbf{S}_p^L \in \mathbb{R}^{N_q \times N_c}$, and the predicted category distribution of part i is denoted as $\mathbf{s}_i \in \mathbb{R}^{N_c}$.
Pairwise affinity is computed to model potential parent-child relations among predicted parts. For each ordered pair (i, j), we compute a semantic attachment score using a learnable compatibility matrix $\mathbf{C} \in \mathbb{R}^{N_c \times N_c}$:

$$\mathbf{S}_{i,j} = \mathbf{s}_i^\top \mathbf{C} \mathbf{s}_j. \tag{4}$$

This bilinear formulation captures category-level attachment priors in a data-driven manner. To obtain the probability that part j serves as the parent of part i, we normalize the scores over all candidate parents:

$$P(j|i) = \text{Softmax}(\mathbf{S}_{i,j}). \tag{5}$$

Parent assignment selects, for each part i, the parent with the highest attachment probability $P(j|i)$, optionally including a learnable root node as a candidate parent. To ensure a valid kinematic hierarchy, we enforce a single-root, cycle-free constraint during structure construction.

4 Experiments

4.1 Evaluation Protocol

Dataset. We use PartNet-Mobility [59] as the benchmark, which contains approximately 2K articulated objects with part-level geometry and joint annotations, covering fixed, prismatic, revolute, and continuous joints. We adopt two evaluation splits: **(1)** a 7-category setting following SINGAPO [31] (Storage, Table, Refrigerator, Dishwasher, Oven, Washer, and Microwave), and **(2)** a full 46-category setting following PhysX-Anything [3] to evaluate large-scale multi-class generalization.

Metrics. Following FreeArt3D [7], we evaluate both motion-aware geometric reconstruction quality and kinematic prediction accuracy.
Geometric reconstruction quality includes four metrics: Chamfer Distance (CD), F-Score, PSNR, and CLIP similarity. For each shape, we uniformly sample six articulation states along the predicted motion range and generate the corresponding meshes using the estimated joint parameters. All metrics are computed at each state and averaged across states. Predicted meshes are aligned with ground-truth meshes using [34] prior to evaluation. CD and F-Score (threshold 0.05) are computed on 100k uniformly sampled surface points per aligned mesh. For appearance, we render both predicted and ground-truth meshes at the six articulation states, sampling 10 random viewpoints per state from a unit sphere (60 images in total), and compute the average PSNR and CLIP similarity [48] using ViT-L/14@336px.
Kinematic prediction accuracy includes three metrics: Type Accuracy, Axis Direction Error, and Pivot Distance Error. Predicted parts are first matched to ground-truth parts with bipartite matching to establish one-to-one correspondence. Type Accuracy measures joint type classification correctness. Axis Direction Error e_{axis} is defined as the angular deviation between predicted and ground-truth motion axes $\mathbf{a}_p$ and $\mathbf{a}_g$ for both revolute and prismatic joints:

$$e_{\text{axis}} = \min\left(\arccos\left(\frac{\mathbf{a}_p \cdot \mathbf{a}_g}{\|\mathbf{a}_p\|_2\|\mathbf{a}_g\|_2}\right), \arccos\left(\frac{-\mathbf{a}_p \cdot \mathbf{a}_g}{\|\mathbf{a}_p\|_2\|\mathbf{a}_g\|_2}\right)\right). \tag{6}$$

Table 1: Quantitative comparison on the PartNet-Mobility dataset. CD is scaled by $\times 10^{-2}$. Type Acc. (%) denotes joint type classification accuracy. Axis Err. (rad) represents joint axis direction error.. Pivot Err. is joint pivot distance error in normalized object coordinates. Best results are highlighted in bold.

Method	Geometry				Kinematics		
	CD ↓	F-Score ↑	PSNR ↑	CLIP ↑	Type Acc. ↑	Axis Err. ↓	Pivot Err. ↓
Partial 7 classes							
URDFormer [9]	4.73	0.275	12.43	0.845	35.22	1.324	0.404
SINGAPO [31]	1.26	0.572	15.22	0.870	77.12	0.493	0.201
MonoArt	**0.77**	**0.728**	**17.55**	**0.926**	**88.26**	**0.209**	**0.085**
All 46 classes							
ArtAny [23]	2.07	0.514	16.44	0.866	43.32	0.440	0.347
PhysXGen [2]	3.06	0.501	16.38	0.859	46.82	0.941	0.208
PhysXAny [3]	1.88	0.531	17.07	0.880	63.35	**0.289**	0.173
MonoArt	**1.25**	**0.670**	**18.55**	**0.907**	**67.47**	0.423	**0.108**

Pivot Distance Error e_{pivot} measures the distance between the predicted joint pivot $\mathbf{o}_p$ to the ground-truth pivot $\mathbf{o}_g$:

$$e_{\text{pivot}} = \frac{|(\mathbf{o}_p - \mathbf{o}_g) \cdot (\mathbf{a}_p \times \mathbf{a}_g)|}{|\mathbf{a}_p \times \mathbf{a}_g|}. \tag{7}$$

All geometric quantities are evaluated in normalized object coordinates.

4.2 Implementation Details

Hyperparameters. We sample $M = 100{,}000$ surface points from the reconstructed mesh for part-aware reasoning. The structured voxel latent resolution is $N_z = 64$. The tri-plane resolution is $N_t = 128$ with feature dimension $d_1 = 8$, and the lifted point embedding dimension is $d_2 = 448$. We use $N_q = 100$ dual queries and a 6-layer dual-query motion decoder ($L = 6$).

Training Procedure. We adopt a four-phase training strategy: **1)** Warm up Part-Aware Semantic Reasoner with triplet supervision to learn motion-aware part embeddings. **2)** Freeze Part-Aware Semantic Reasoner and train the dual-query initialization branch using object-category supervision. **3)** Jointly optimize Part-Aware Semantic Reasoner, Dual-query Motion Decoder, and Articulation Parameter Regressor. **4)** Freeze all preceding modules and train Kinematic Tree Predictor to model parent-child relations. Additional training hyperparameters and loss configurations are provided in the Appendix.

4.3 Main Results

We compare MonoArt with state-of-the-art monocular articulated object reconstruction methods, including URDFormer [9], SINGAPO [31], Articulate-

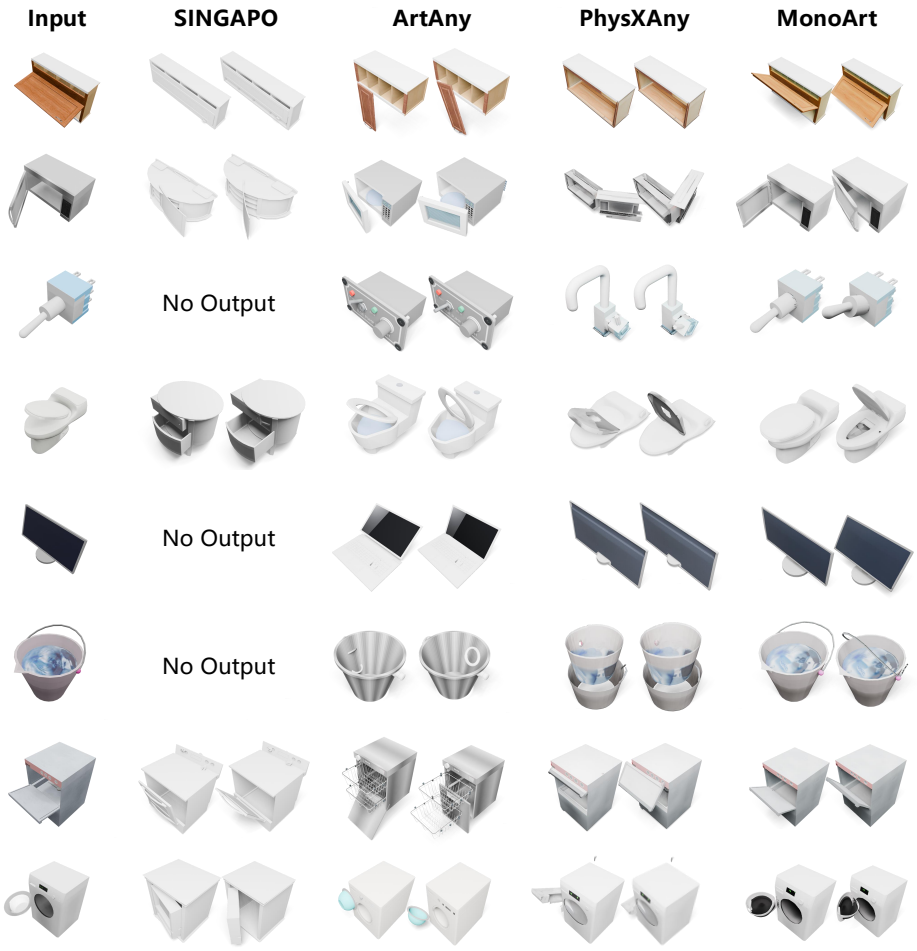

Fig. 3: Qualitative results on the test set of PartNet-Mobility. ArtAny and PhysXAny denote Articulate-Anything and PhysXAnything, respectively. For each object, we show the reconstructed geometry under two sampled articulated states.

Anything [23], PhysXGen [2], and PhysXAnything [3]. For retrieval-based baselines (Articulate-Anything and SINGAPO), we remove ground-truth test shapes from their retrieval database to ensure fair comparison and avoid data leakage.

PartNet-Mobility Benchmark. On PartNet-Mobility [59], MonoArt achieves the best overall performance in both geometry reconstruction and kinematic prediction. For the partial 7 classes, it substantially outperforms prior methods with clear margins in reconstruction quality and articulation estimation, particularly showing large gains in joint type accuracy and significant reductions in axis and pivot errors. On the full 46 classes, MonoArt remains consistently superior, delivering the strongest overall reconstruction fidelity while achieving the highest joint type accuracy and the lowest pivot error. Notably, it reduces pivot error

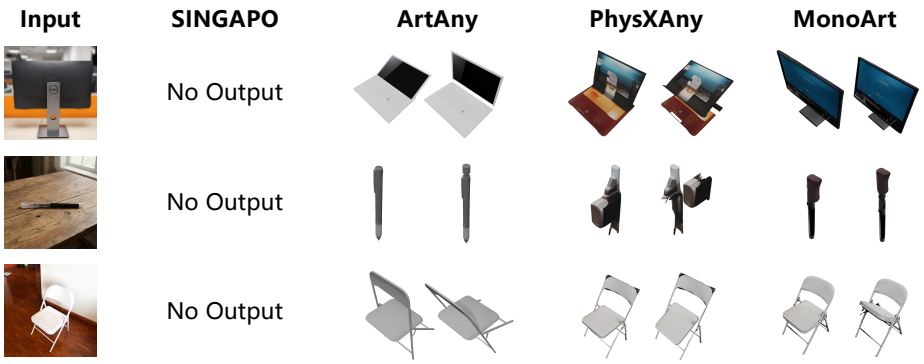

Fig. 4: Qualitative results on in-the-wild images. ArtAny and PhysXAny denote Articulate-Anything and PhysXAnything, respectively. For each object, we show the reconstructed geometry under two sampled articulated states.

by more than 40%, demonstrating robust structural reasoning across diverse articulated categories. Qualitative results in Fig. 3 further show that MonoArt produces more faithful geometry and more accurate joint predictions, leading to more plausible articulated motion.

Real-World Generalization. To evaluate real-world generalization, we collect approximately 100 in-the-wild images from the Internet using category keywords, covering common everyday articulated objects. As shown in Fig. 4, MonoArt produces coherent geometry and plausible articulation across diverse real-world appearances, despite being trained primarily on synthetic data. We conduct a user study with 20 participants to evaluate geometric and kinematic quality of the generated articulations. Participants rate rendered videos on two 1-5 scales. Our method achieves the highest scores (4.63 / 4.37), outperforming PhysX-Anything (3.34 / 3.12), SINGAPO (2.55 / 2.87), Articulate-Anything (2.72 / 2.60), PhysXGen (2.53 / 2.46), and URDFormer (1.37 / 1.49).

4.4 Ablation Study

We perform ablations on MonoArt to study key design choices in Part-Aware Semantic Reasoner, Dual-Query Motion Decoder, and Kinematic Estimator. All experiments are conducted on PartNet-Mobility (46 classes), evaluating their impact on geometry and kinematics.

Part-Aware Semantic Reasoner. As shown in Table 2, we ablate the design of the Part-Aware Semantic Reasoner. Removing the reasoner significantly degrades both geometry and kinematics, especially joint type accuracy and pivot prediction, confirming its critical role in motion-aware part reasoning. In this variant, voxel features and lifted point embeddings are directly fused via trilinear interpolation followed by an MLP. To make the learned feature $\mathbf{H}$ part-aware, we supervise the reasoner with a triplet loss that enforces part-level feature separation. Replacing it with cross-entropy supervision leads to clear performance drops, and removing the loss further degrades results. Triplet supervision

Table 2: Ablation of Part-Aware Semantic Reasoner. CD is scaled by $\times 10^{-2}$. Type Acc. (%) is joint classification accuracy. Axis Err. (rad) and Pivot Err. denote axis direction and pivot distance errors. Note that "CE" and "Tri." denote "Cross-Entropy" and "Triplet", respectively.

Enabled	Loss	Geometry			Kinematics		
		CD ↓	F-Score ↑	PSNR ↑	Type Acc. ↑	Axis Err. ↓	Pivot Err. ↓
✗	✗	1.74	0.626	17.96	24.72	0.549	0.237
✓	✗	1.63	0.643	17.74	41.60	0.922	0.323
✓	CE	1.49	0.648	17.71	57.74	1.029	0.302
✓	Tri.	**1.25**	**0.670**	**18.55**	**67.47**	**0.423**	**0.108**

Table 3: Ablation of Dual-Query Motion Decoder. CD is scaled by $\times 10^{-2}$. Type Acc. (%) is joint classification accuracy. Axis Err. (rad) and Pivot Err. denote axis direction and pivot distance errors. Note that "DQI." indicates whether Dual-Query Initialization is enabled. "Res." denotes residual updates in the position and/or content query branches. L is the number of refinement blocks.

DQI.	Res.	L	Geometry			Kinematics		
			CD ↓	F-Score ↑	PSNR ↑	Type Acc. ↑	Axis Err. ↓	Pivot Err. ↓
✗	Both	6	1.67	0.622	18.11	44.06	0.472	0.329
✓	Q_p^l	6	1.73	0.640	17.81	66.41	0.523	0.181
✓	Q_c^l	6	1.29	0.663	17.80	60.88	0.506	0.184
✓	Both	0	1.70	0.652	17.94	62.65	0.640	0.186
✓	Both	1	1.71	0.652	17.91	63.12	0.608	0.189
✓	Both	3	1.67	0.655	17.88	66.38	0.524	0.157
✓	Both	6	**1.25**	**0.670**	**18.55**	**67.47**	**0.423**	**0.108**
✓	Both	9	1.59	0.659	17.95	66.81	0.475	0.161

Table 4: Ablation of Kinematic Estimator. CD is scaled by $\times 10^{-2}$. Type Acc. (%) is joint classification accuracy. Axis Err. (rad) and Pivot Err. denote axis direction and pivot distance errors.

$\mathbf{Q}_p^L$	$\mathbf{H}$	Geometry			Kinematics		
		CD ↓	F-Score ↑	PSNR ↑	Type Acc. ↑	Axis Err. ↓	Pivot Err. ↓
✗	✓	1.65	0.652	17.92	67.20	0.499	0.191
✓	✗	2.35	0.573	18.11	27.14	0.882	0.283
✓	✓	**1.25**	**0.670**	**18.55**	**67.47**	**0.423**	**0.108**

consistently achieves the best performance across all metrics, highlighting the importance for discriminative and motion-consistent part representations.

Dual-Query Motion Decoder. As shown in Table 3, we analyze key design choices of the Dual-Query Motion Decoder. Disabling Dual-Query Initialization (DQI) and randomly initializing $\mathbf{Q}_p^0$ and $\mathbf{Q}_c^0$ degrades both geometry and

Fig. 5: Robot manipulation with generated articulated objects. MonoArt reconstructions are directly imported into IsaacSim for contact-rich interaction.

Fig. 6: Articulated scene reconstruction. MonoArt augments SAM3D [11] with object-level articulation recovery to produce articulated, operable 3D scenes.

kinematics, showing the importance of informed query initialization. Applying residual updates to only one branch is suboptimal, while updating both position and content queries achieves the best performance, highlighting the need for joint refinement. Increasing the number of refinement layers improves results up to $L = 6$, while a deeper model (*e.g.*, $L = 9$) leads to performance degradation.

Kinematic Estimator. As shown in Table 4, we ablate the Kinematic Estimator. We regress the joint origin using a residual formulation, $\mathbf{m}_o = \mathbf{Q}_p^L + \Delta_o$. Directly predicting $\mathbf{m}_o$ without this anchor degrades both geometry and kinematics, showing the benefit of centroid-based residual prediction. Excluding the point embedding $\mathbf{H}$ also leads to significant performance drops, indicating its importance for parameter regression. Using both $\mathbf{Q}_p^L$-based residual prediction and $\mathbf{H}$ achieves the best results.

4.5 Applications

Robot Manipulation. MonoArt extends beyond static 3D reconstruction by inferring articulation axes, joint types, and motion limits that are directly usable for robotic control. These structured kinematic priors convert monocular observations into actionable parameters, enabling motion reasoning and interaction planning without manual joint annotation. To evaluate downstream utility, we

import the articulated objects reconstructed from in-the-wild images in Fig. 4 into IsaacSim [43]. As shown in Fig. 5, the simulation-ready assets can be directly manipulated by a Franka robot arm for contact-rich tasks such as grasping and opening, without additional modeling. This demonstrates a practical real-to-sim pipeline that produces physically plausible articulated models for robotic interaction and policy learning.

Articulated Scene Reconstruction. Methods such as MIDI [20] and SAM 3D [11] provide static scene reconstructions with per-object masks and 6D poses. Building on these outputs, we reconstruct each masked object instance with MonoArt to recover both geometry and articulation parameters. The reconstructed articulated objects are then placed back into the scene using their estimated 6D poses, yielding a coherent articulated scene without additional manual modeling. As shown in Fig. 6, this simple object-level augmentation converts rigid scene reconstructions into functionally operable environments, where articulated objects preserve their kinematic structure while remaining consistent with the global layout.

4.6 Discussion

Runtime. All timings are measured on a single NVIDIA A6000 GPU, excluding I/O time, and averaged over 100 runs. As shown in Fig. 1, recent methods require 229.9s (Articulate-Anything [23]), 256.8s (PhysXAnything [3]), 31.6s (PhysX-Gen [2]), 34.1s (URDFormer [9]), and 19.6s (SINGAPO [31]) per instance. In comparison, MonoArt requires 20.5 seconds per instance. Among this, 18.2s is spent on TRELLIS-based 3D reconstruction, while articulation reasoning and post-processing introduce only marginal overhead.

Limitations. While MonoArt demonstrates strong performance in articulated object reconstruction and motion reasoning, several limitations remain. MonoArt can struggle with very small parts attached to large objects (*e.g.*, tiny buttons). Due to uniform point sampling over the entire shape, such components may receive only sparse coverage, making their features less distinctive and more prone to over-smoothing. Consequently, articulations under extreme scale imbalance can be difficult to reliably segment and parameterize. MonoArt also relies on learned structural priors over part-whole relationships, which may not fully generalize to objects with novel topologies or uncommon articulation patterns. For such unseen object configurations, the predicted motion parameters, including motion axes and ranges, may be less accurate, even when part segmentation remains reasonable.

5 Conclusion

In this work, we present **MonoArt**, a unified framework for monocular articulated 3D reconstruction grounded in progressive structural reasoning. Instead of depending on multi-view supervision, retrieval libraries, or auxiliary

video synthesis, MonoArt formulates monocular articulated reconstruction as a progressive structural reasoning process. By explicitly modeling geometry, part structure, and motion in a unified framework, it achieves accurate and efficient articulation inference without handcrafted motion priors or external pipelines. Extensive experiments demonstrate that MonoArt achieves state-of-the-art performance in both reconstruction accuracy and inference speed on PartNet-Mobility. Beyond object-level reconstruction, the framework generalizes effectively to robotic manipulation and articulated scene reconstruction, highlighting its practical applicability.

Acknowledgements

This study is supported by the Ministry of Education, Singapore, under its MOE AcRF Tier 2 (MOE-T2EP20223-0002). This research is also supported by cash and in-kind funding from NTU S-Lab and industry partner(s).

References

1. Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L.X., Tanner, J., Vuong, Q., Walling, A., Wang, H., Zhilinsky, U.: π_0: A vision-language-action flow model for general robot control. In: RSS (2025)
2. Cao, Z., Chen, Z., Pan, L., Liu, Z.: PhysX-3D: physical-grounded 3D asset generation. In: NeurIPS (2025)
3. Cao, Z., Hong, F., Chen, Z., Pan, L., Liu, Z.: Physx-Anything: Simulation-ready physical 3D assets from single image. In: CVPR (2026)
4. Cao, Z., Liu, Y., Li, H., Yao, R., Hong, F., Chen, Z., Pan, L., Liu, Z.: PhysX-Omni: Unified simulation-ready physical 3D generation for rigid, deformable, and articulated objects. arXiv 2605.21572 (2026)
5. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., Mello, S.D., Gallo, O., Guibas, L.J., Tremblay, J., Khamis, S., Karras, T., Wetzstein, G.: Efficient geometry-aware 3D generative adversarial networks. In: CVPR (2022)
6. Chang, A.X., Funkhouser, T.A., Guibas, L.J., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An information-rich 3D model repository. arXiv 1512.03012 (2015)
7. Chen, C., Liu, I., Wei, X., Su, H., Liu, M.: FreeArt3D: Training-free articulated object generation using 3D diffusion. In: SIGGRAPH Asia (2025)
8. Chen, H., Lan, Y., Chen, Y., Pan, X.: ArtiLatent: Realistic articulated 3D object generation via structured latents. In: SIGGRAPH Asia (2025)
9. Chen, Q., Walsman, A., Memmel, M., Mo, K., Fang, A., Fox, D., Gupta, A.: URDFormer: A pipeline for constructing articulated simulation environments from real-world images. In: RSS (2024)
10. Chen, X., Golovinskiy, A., Funkhouser, T.A.: A benchmark for 3D mesh segmentation. ACM TOG **28**(3), 73 (2009)

11. Chen, X., CHU, F.J., Gleize, P., Liang, K.J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., Lin, A., Liu, J.W., Ma, Z., Sagar, A., Song, B., Wang, X., Yang, J., Zhang, B., Dollár, P., Gkioxari, G., Feiszli, M., Malik, J.: SAM 3D: 3Dfy anything in images. In: CVPR (2026)
12. Chen, Z., Tang, J., Dong, Y., Cao, Z., Hong, F., Lan, Y., Wang, T., Xie, H., Wu, T., Saito, S., Pan, L., Lin, D., Liu, Z.: 3DTopia-XL: scaling high-quality 3D asset generation via primitive diffusion. In: CVPR (2025)
13. Fei, X., Xu, Z., Fang, H., Zhang, T., Shao, L.: T(R,O) grasp: Efficient graph diffusion of robot-object spatial transformation for cross-embodiment dexterous grasping. In: ICRA (2026)
14. Gao, D., Siddiqui, Y., Li, L., Dai, A.: MeshArt: Generating articulated meshes with structure-guided transformers. In: CVPR (2025)
15. Gao, M., Pan, Y., Gao, H., Zhang, Z., Li, W., Dong, H., Tang, H., Yi, L., Zhao, H.: PartRM: modeling part-level dynamics with large cross-state reconstruction model. In: CVPR (2025)
16. Geng, C., He, G., Gao, Y., Zhang, Y., Wu, S., Wu, J.: NeuROK: Generative 4D neural object kinematics. In: CVPR (2026)
17. Guo, J., Xin, Y., Liu, G., Xu, K., Liu, L., Hu, R.: ArticulatedGS: self-supervised digital twin modeling of articulated objects using 3D gaussian splatting. In: CVPR (2025)
18. Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D.: MeshCNN: a network with an edge. ACM TOG **38**(4), 90:1–90:12 (2019)
19. Hou, Y., Ma, J., Sun, H., Wu, F.: Effective offline robot learning with structured task graph. IEEE Robotics and Automation Letters **9**(4), 3633–3640 (2024)
20. Huang, Z., Guo, Y., An, X., Yang, Y., Li, Y., Zou, Z., Liang, D., Liu, X., Cao, Y., Sheng, L.: MIDI: multi-instance diffusion for single image to 3D scene generation. In: CVPR (2025)
21. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W., Dollár, P., Girshick, R.B.: Segment anything. In: ICCV (2023)
22. Lai, Z., Zhao, Y., Liu, H., Zhao, Z., Lin, Q., Shi, H., Yang, X., Yang, M., Yang, S., Feng, Y., Zhang, S., Huang, X., Luo, D., Yang, F., Yang, F., Wang, L., Liu, S., Tang, Y., Cai, Y., He, Z., Liu, T., Liu, Y., Jiang, J., Linus, Huang, J., Guo, C.: Hunyuan3D 2.5: towards high-fidelity 3D assets generation with ultimate details. arXiv 2506.16504 (2025)
23. Le, L., Xie, J., Liang, W., Wang, H., Yang, Y., Ma, Y.J., Vedder, K., Krishna, A., Jayaraman, D., Eaton, E.: Articulate-Anything: automatic modeling of articulated objects via a vision-language foundation model. In: ICLR (2025)
24. Lei, J., Deng, C., Shen, W.B., Guibas, L.J., Daniilidis, K.: NAP: neural 3D articulated object prior. In: NeurIPS (2023)
25. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J., Chang, K., Gao, J.: Grounded language-image pre-training. In: CVPR (2022)
26. Li, R., Yao, Y., Zheng, C., Rupprecht, C., Lasenby, J., Wu, S., Vedaldi, A.: Particulate: Feed-forward 3D object articulation. In: CVPR (2026)
27. Li, R., Yao, Y., Zhou, M., Zheng, C., Rupprecht, C., Lasenby, J., Wu, S., Vedaldi, A.: Instruct-Particulate: Scaling feed-forward 3D object articulation with kinematic control. arXiv 2606.14699 (2026)
28. Li, R., Zheng, C., Rupprecht, C., Vedaldi, A.: Puppet-Master: scaling interactive video generation as a motion prior for part-level dynamics. In: ICCV (2025)

29. Li, X., Yang, J., Zhang, F.: Laplacian mesh transformer: Dual attention and topology aware network for 3D mesh classification and segmentation. In: ECCV (2022)
30. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B.: PointCNN: Convolution on x-transformed points. In: NeurIPS (2018)
31. Liu, J., Iliash, D., Chang, A.X., Savva, M., Amiri, A.M.: SINGAPO: single image controlled generation of articulated parts in objects. In: ICLR (2025)
32. Liu, J., Mahdavi-Amiri, A., Savva, M.: PARIS: part-level reconstruction and motion analysis for articulated objects. In: ICCV (2023)
33. Liu, M., Uy, M.A., Xiang, D., Su, H., Fidler, S., Sharp, N., Gao, J.: PARTFIELD: learning 3D feature fields for part segmentation and beyond. In: ICCV (2025)
34. Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., Su, H.: One-2-3-45++: Fast single image to 3D objects with consistent multi-view generation and 3D diffusion. In: CVPR (2024)
35. Liu, M., Zhu, Y., Cai, H., Han, S., Ling, Z., Porikli, F., Su, H.: PartSLIP: Low-shot part segmentation for 3D point clouds via pretrained image-language models. In: CVPR (2023)
36. Liu, X., Sun, X., Xie, H., Li, Z., Li, R., Zhang, S.: Multi-view consistent 3D panoptic scene understanding. In: AAAI (2025)
37. Liu, X., Xie, H., Zhang, S., Yao, H., Ji, R., Nie, L., Tao, D.: 2D semantic-guided semantic scene completion. IJCV **133**(3), 1306–1325 (2025)
38. Lu, R., Liu, Y., Tang, J., Ni, J., Wang, Y., Wan, D., Zeng, G., Chen, Y., Huang, S.: DreamArt: generating interactable articulated objects from a single image. In: SIGGRAPH Asia (2025)
39. Ma, C., Li, Y., Yan, X., Xu, J., Yang, Y., Wang, C., Zhao, Z., Guo, Y., Chen, Z., Guo, C.: P3-SAM: native 3D part segmentation. arXiv 2509.06784 (2025)
40. Ma, Z., Yue, Y., Gkioxari, G.: Find any part in 3D. In: ICCV (2025)
41. Mandi, Z., Weng, Y., Bauer, D., Song, S.: Real2Code: reconstruct articulated objects via code generation. In: ICLR (2025)
42. Mo, K., Zhu, S., Chang, A.X., Yi, L., Tripathi, S., Guibas, L.J., Su, H.: PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In: CVPR (2019)
43. NVIDIA: Isaac Lab: A GPU-accelerated simulation framework for multi-modal robot learning. arXiv 2511.04831 (2025)
44. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P., Li, S., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jégou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. TMLR **2024** (2024)
45. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep learning on point sets for 3D classification and segmentation. In: CVPR (2017)
46. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: NIPS (2017)
47. Qian, G., Li, Y., Peng, H., Mai, J., Hammoud, H., Elhoseiny, M., Ghanem, B.: PointNeXt: Revisiting pointnet++ with improved training and scaling strategies. In: NeurIPS (2022)
48. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021)
49. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: A unified embedding for face recognition and clustering. In: CVPR (2015)

50. Song, C., Wei, J., Foo, C.S., Lin, G., Liu, F.: REACTO: reconstructing articulated objects from a single video. In: CVPR (2024)
51. Tang, G., Zhao, W., Ford, L., Ben-Haim, D., Zhang, P.: Segment any mesh: Zero-shot mesh part segmentation via lifting segment anything 2 to 3D. arXiv 2408.13679 (2024)
52. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: DreamGaussian: generative gaussian splatting for efficient 3D content creation. In: ICLR (2024)
53. Umam, A., Yang, C., Chen, M., Chuang, J., Lin, Y.: PartDistill: 3D shape part segmentation by vision-language model distillation. In: CVPR (2024)
54. Wang, R., Patil, A.G., Yu, F., Zhang, H.: Active coarse-to-fine segmentation of moveable parts from real images. In: ECCV (2024)
55. Wang, R., Zhang, H.: RESAnything: Attribute prompting for arbitrary referring segmentation. In: NeurIPS (2025)
56. Wei, F., Chabra, R., Ma, L., Lassner, C., Zollhöfer, M., Rusinkiewicz, S., Sweeney, C., Newcombe, R.A., Slavcheva, M.: Self-supervised neural articulated shape and appearance models. In: CVPR (2022)
57. Wen, B., Xie, H., Chen, Z., Hong, F., Liu, Z.: 3D scene generation: A survey. arXiv 2505.05474 (2025)
58. Weng, Y., Wen, B., Tremblay, J., Blukis, V., Fox, D., Guibas, L.J., Birchfield, S.: Neural implicit representation for building digital twins of unknown articulated objects. In: CVPR (2024)
59. Xiang, F., Qin, Y., Mo, K., Xia, Y., Zhu, H., Liu, F., Liu, M., Jiang, H., Yuan, Y., Wang, H., Yi, L., Chang, A.X., Guibas, L.J., Su, H.: SAPIEN: A simulated part-based interactive environment. In: CVPR (2020)
60. Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3D latents for scalable and versatile 3D generation. In: CVPR (2025)
61. Xie, H., Chen, Z., Hong, F., Liu, Z.: CityDreamer: compositional generative model of unbounded 3D cities. In: CVPR (2024)
62. Xie, H., Chen, Z., Hong, F., Liu, Z.: Compositional generative model of unbounded 4D cities. IEEE TPAMI **48**(1), 312–328 (2026)
63. Xie, H., Wen, B., Zheng, J., Chen, Z., Hong, F., Diao, H., Liu, Z.: DynamicVLA: A vision-language-action model for dynamic object manipulation. arXiv 2601.22153 (2026)
64. Xie, H., Yao, H., Zhou, S., Mao, J., Zhang, S., Sun, W.: GRNet: gridding residual network for dense point cloud completion. In: ECCV (2020)
65. Xu, M., Zhang, T., Liu, T., Chen, Z., Han, X., Liu, Z.: Kinema4d: Kinematic4d world modeling for spatiotemporal embodied simulation. arXiv 2603.16669 (2026)
66. Xue, Y., Chen, N., Liu, J., Sun, W.: ZeroPS: High-quality cross-modal knowledge transfer for zero-shot 3D part segmentation. In: 3DV (2025)
67. Yang, Y., Jia, B., Zhi, P., Huang, S.: PhyScene: physically interactable 3D scene synthesis for embodied AI. In: CVPR (2024)
68. Yang, Y., Huang, Y., Guo, Y., Lu, L., Wu, X., Lam, E.Y., Cao, Y., Liu, X.: SAMPart3D: Segment any part in 3D objects. arXiv 2411.07184 (2024)
69. Yang, Z., Qiu, X., Zhao, X., Wang, X., Zhang, Q., Xiao, J.: Frequency-aware affinity for weakly supervised semantic segmentation. In: CVPR (2026)
70. Yi, L., Kim, V.G., Ceylan, D., Shen, I., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., Guibas, L.J.: A scalable active framework for region annotation in 3D shape collections. ACM TOG **35**(6), 210:1–210:12 (2016)
71. Zhao, H., Jiang, L., Jia, J., Torr, P.H.S., Koltun, V.: Point transformer. In: ICCV (2021)

72. Zhou, M., Li, R., Lyu, X., Song, Z., Huang, Z., Zheng, C., Rupprecht, C., Vedaldi, A., Wu, S.: Articraft: An agentic system for scalable articulated 3D asset generation. arXiv 2605.15187 (2026)
73. Zhou, Y., Gu, J., Li, X., Liu, M., Fang, Y., Su, H.: PartSLIP++: Enhancing low-shot 3D part segmentation via multi-view instance segmentation and maximum likelihood estimation. arXiv 2312.03015 (2023)
74. Zhu, Z., Wan, L., Xu, R., Zhang, Y., Chen, H., Dou, Z., Lin, C., Liu, Y., Wei, M.: PartSAM: A scalable promptable part segmentation model trained on native 3D data. In: ICLR (2026)