This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

ArtLLM: Generating Articulated Assets via 3D LLM

Penghao Wang[1,2,*] Siyuan Xie[1] Hongyu Yan[2,3] Xianghui Yang[2]
Jingwei Huang[2,†] Chunchao Guo[2,†] Jiayuan Gu[1,†]
[1]ShanghaiTech University [2]Tencent Hunyuan [3]HKUST
https://authoritywang.github.io/artllm

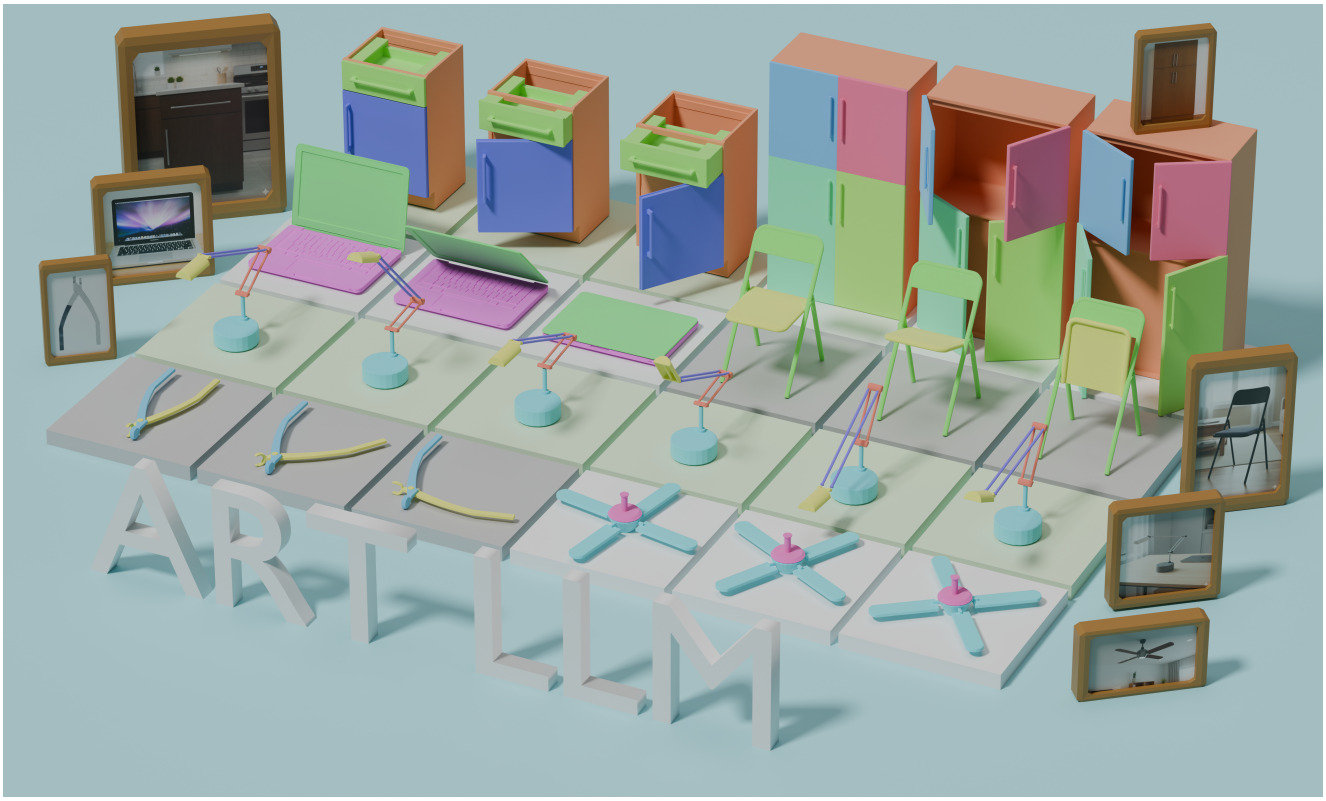

Figure 1. We propose **ArtLLM**, a novel framework capable of rapidly generating articulation assets from images or text. By using a 3D LLM to jointly predict part layouts and joints, and integrating state-of-the-art part generation methods, our approach can produce high-quality, physically grounded articulation assets.

Abstract

Creating interactive digital environments for gaming, robotics, and simulation relies on articulated 3D objects whose functionality emerges from their part geometry and kinematic structure. However, existing approaches remain fundamentally limited: optimization-based reconstruction methods require slow, per-object joint fitting and typically handle only simple, single-joint objects, while retrieval-based methods assemble parts from a fixed library, leading to repetitive geometry and poor generalization. To address these challenges, we introduce ArtLLM, a novel framework for generating high-quality articulated assets directly from complete 3D meshes. At its core is a 3D multimodal large language model trained on a large-scale articulation dataset curated from both existing articulation datasets and procedurally generated objects. Unlike prior work, ArtLLM autoregressively predicts a variable number of parts and joints, inferring their kinematic structure in a unified manner from the object's point cloud. This articulation-aware layout then conditions a 3D generative model to synthesize high-fidelity part geometries. Experiments on the PartNet-Mobility dataset show that ArtLLM significantly outperforms state-of-the-art methods in both

*This work is done while interning with Tencent Hunyuan.
†Corresponding authors.

part layout accuracy and joint prediction, while generalizing robustly to real-world objects. Finally, we demonstrate its utility in constructing digital twins, highlighting its potential for scalable robot learning.

1. Introduction

The creation of interactive digital worlds for gaming, robotics, and simulation fundamentally relies on assets that can be manipulated and animated. These articulated objects, from doors and drawers to complex machinery, derive their functionality from their underlying part-based geometry and kinematic structures. Generating articulated assets automatically is essential for scaling up content creation, enabling realistic robot training in simulation [47], and enriching the interactivity of virtual environments.

Recent efforts in articulated object generation have primarily followed two distinct paradigms. One line of work [25, 29, 30, 39] focuses on optimization-based reconstruction from multi-view images or videos, leveraging neural representations like NeRF [36] or 3DGS [14] to estimate joint parameters and geometry. However, these approaches are often hampered by slow, per-object optimization, tend to produce low-fidelity geometry, and are typically constrained to simple objects with only a single joint. Other approaches [26, 27] train feedforward networks on existing datasets to directly predict part layouts and joint parameters. While offering fast inference, these methods are usually constrained to retrieving parts from a fixed, predefined database, which severely limits their ability to produce novel shapes and results in geometrically repetitive assets.

While generating high-quality articulated objects remains difficult, general 3D object generation [55, 66] has seen immense progress, enabling high-fidelity synthesis from various inputs. Recent extensions support part-level generation [23, 46, 59, 62]. Yet, a fundamental limitation persists: a disconnect between geometry and motion. These models are unaware of the underlying kinematic structures that dictate how parts should move, leading to a potential mismatch between a part's visual semantics and its intended mechanical role. It clearly indicates that a unified approach, capable of jointly reasoning over geometry and articulation, is required.

To this end, we present a framework that generates articulated assets by first predicting their geometric layouts as well as kinematic structures and then synthesizing their geometry. The centerpiece is a 3D articulation language model (**ArtLLM**), which autoregressively outputs a tokenized blueprint of the object's part layout and kinematic relationships, given an input point cloud. This blueprint then guides a part-aware generative model to synthesize high-fidelity geometries, overcoming the reliance on fixed databases. Furthermore, we introduce a post-processing step to optimize the predicted joint limit, so that the resulting articulation is physically plausible and collision-free. Trained on a large, curated dataset of articulated objects, ArtLLM offers a scalable and effective solution that avoids the slow optimization of reconstruction methods and the geometric limitations of retrieval-based approaches.

We evaluate our method on the PartNet-Mobility [54] dataset and compare it with state-of-the-art approaches. Our model achieves superior performance in predicting part placement, joint accuracy, and kinematic relationship modeling. Unlike retrieval-based methods, it generates accurate and novel part geometries and generalizes well to real-world images, effectively reconstructing articulated assets and enabling realistic digital twins, which highlight its potential to bridge perception and generation for scalable robot learning.

2. Related Work

3D Generation Early approaches leveraged 2D foundation models for 3D generation, including techniques based on SDS optimization [40] and multi-view image synthesis [28, 31, 43]. Later, LRM series [8, 45, 57, 67], introduced a feed-forward paradigm to achieve fast 3d generation. However, their representation is not inherently 3D, which constrained the generation fidelity. More recently, 3DShape2VecSet [66] and Trellis [55] respectively introduced native 3D generation representations based on point and voxel, laying the foundation for native 3D generative models. Several works such as [15, 18–20, 68] have further advanced high-quality 3D generation with DiT [38]. Building upon these 3D foundation models, many studies have explored downstream 3D generation tasks, including part-level generation [6, 59, 61, 62, 69], scene-level generation [10, 63], diverse condition control generation [11, 58], editable generation [65]. Collectively, these advancements enable flexible and expressive 3D generation across diverse conditions, establishing robust 3D foundation models that benefit multiple application domains.

3D Large Language Models Inspired by the success of VLMs [1, 24], enabling LLMs to understand 3D content has become an urgent and important research direction. Recent works [9, 41, 56] have pioneered the direct integration of 3D representations into language models, enabling native 3D reasoning. Beyond conversational understanding, subsequent studies have broadened the capabilities of 3D LLMs to tasks such as 3D generation [48, 51, 64], scene grounding [35]. Collectively, these works demonstrate the practical value and strong potential of 3D LLMs in advancing multimodal understanding and generation tasks.

Articulation Assets Generation Rapidly generating articulated object assets from images or text is crucial for building digital twins and advancing robotic simulation. Early efforts [2, 7, 12, 13, 37, 50, 54] relied on manual annota-

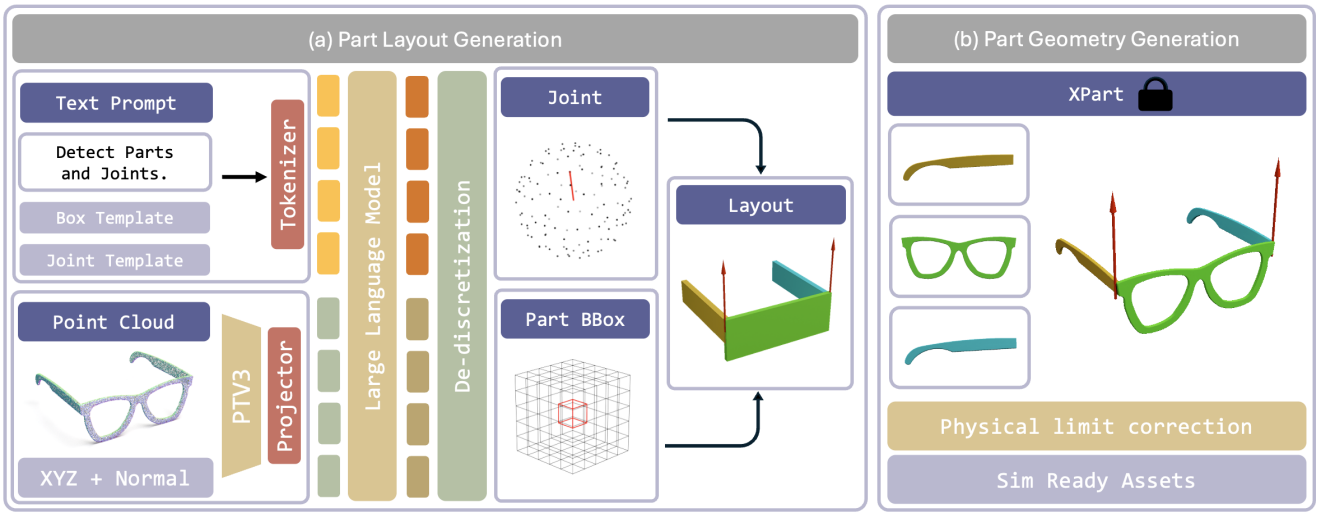

Figure 2. **Architecture Overview.** Given an input point cloud, ArtLLM first predicts a tokenized articulation blueprint that specifies **part layouts and kinematic structures**. This blueprint then conditions a part-aware generative model to synthesize high-fidelity **link geometries**, followed by a physics-based joint-limit correction module refines the articulation, producing simulation-ready articulated assets.

tion to construct large-scale part-level and articulated object datasets, laying the foundation for this field. Recent methods [25, 29, 30, 39] employ per-object optimization to reconstruct articulated objects. However, these approaches suffer from low optimization speed, dependence on dense multi-view or video inputs, and limited scalability. Other approaches [5, 16, 22, 26, 27, 52] generate articulated objects by retrieving existing parts from pre-built libraries, which restricts geometric novelty. To enable novel geometry generation, some studies [17, 44] perform surface reconstruction from generated SDFs, but the results often lack quality. With the emergence of 3D foundation models, methods such as [3, 4, 32] have achieved articulated object generation, though they are typically limited to single-joint structures. Additionally, some works [21, 34] leverage the strong reasoning ability of LLMs for articulated object modeling. Yet, they directly predict float parameters and are trained on limited data, leading to poor generalization. Their reliance on point cloud based mesh reconstruction further constrains output quality. In contrast, our approach trains a 3D LLM with a well-designed template and data quantization strategy, enabling accurate prediction of part layouts and joint parameters. Combined with state-of-the-art part generation models, our method produces high-quality articulated objects with high-fidelity geometric structures.

3. Method

This section introduces our framework for generating articulated assets from point clouds. First, our novel 3D articulation language model (**ArtLLM**) autoregressively predicts the object's kinematic structure, outputting a tokenized representation (Sec.3.1). This model is trained on a large-scale, diverse collection of articulated data (Sec.3.2). Next, the previously predicted structural blueprint conditions a part-aware generative model to synthesize high-fidelity, coherent part geometries (Sec.3.3). Finally, a joint-range optimization step ensures the resulting asset is physically plausible and collision-free (Sec. 3.4). See Figure 2 for the overall pipeline.

3.1. 3D Articulation Language Model (ArtLLM)

Kinematic structures of articulated objects are often specified in the *Unified Robotics Description Format* (URDF), which utilizes an XML schema. Thus, we reformulate 3D articulation understanding as a language modeling problem to leverage the powerful reasoning and sequence modeling capabilities of Large Language Models (LLMs). We represent an object's entire kinematic structure—including its constituent parts, their layout, and joint parameters—as a unified sequence of discrete tokens. The autoregressive approach naturally accommodates objects with varying numbers and types of parts, while also allowing us to leverage the rich semantic and structural priors learned by the LLM. Furthermore, this token-based representation is inherently flexible and easily extensible.

Input Representation Our model operates on a point cloud representation, allowing it to flexibly handle various input modalities. For text or image inputs, we first leverage off-the-shelf generative models (e.g., Hunyuan3D 2.5 [15], TripoSG [20]) to produce an initial 3D mesh. For mesh inputs, whether generated or provided directly, we uniformly sample 32,768 surface points with their corresponding normals. To ensure the consistency of normals, we pre-process the

mesh to be watertight before sampling.

Bridging 3D Geometry and Language To bridge the modality gap between the input point cloud and our LLM, we employ an encoder-projector architecture inspired by the success of vision-language models [1, 24] as well as 3D-language modeling [35, 41, 56]. We choose Point Transformer v3 [53] as our point cloud encoder due to its powerful yet efficient design. Following SpatialLM [35], we augment the final layer's features with position embeddings to preserve crucial spatial information. These augmented features are then projected by a simple two-layer MLP for modality alignment. The Qwen3 [60] 0.6B model is used as our language model backbone.

Generating Articulation as Languages We formulate the articulated structure of an object as a language sequence composed of part and joint definitions. We design a concise yet informative text template to regularize outputs.

Each part is assigned with an *id* and parameterized by its 3D axis-aligned bounding box (AABB):

$$\text{bbox\_}id = \text{BBox}(x_{min}, y_{min}, z_{min}, x_{max}, y_{max}, z_{max}), \quad (1)$$

Similarly, kinematic joints are defined using a structured format that encodes their type, connectivity, and parameters. We support four primitive types, including *Revolute*, *Continuous*, *Prismatic*, and *Screw*. For example, a revolute joint is written as:

$$\text{joint\_}id = Revolute\text{Joint}(parent, child, dir, pos, limit), \quad (2)$$

where $parent$ and $child$ link are integer IDs of the connected parts; dir and pos are 3D vectors defining the joint's rotation axis and origin; and $limit$ is a tuple specifying the motion range.

The model is designed to autoregressively generate the full sequence, by first predicting all part bounding boxes, followed by a separator token, and then all joint definitions. This ordering ensures that joint prediction is conditioned on the complete part layout, improving structural coherence.

Quantized Predictions While structured templates simplify the generation task, LLMs are fundamentally designed to predict tokens from a discrete vocabulary, making direct regression of continuous values prone to numerical instability. To address this, we convert all continuous geometric and kinematic parameters into a discrete, token-based representation through quantization. This approach allows us to frame the entire articulation prediction problem within a robust language modeling paradigm.

For each part's bounding box, we quantize its coordinate from a normalized range of $[-1, 1]$ into discrete 128 bins per axis:

$$\hat{c}_{\text{min}} = \left\lfloor \frac{(c_{\text{min}}+1)}{2} \times 128 \right\rfloor, \hat{c}_{\text{max}} = \left\lceil \frac{(c_{\text{max}}+1)}{2} \times 128 \right\rceil, \quad (3)$$

where c represents a original continuous coordinate value and $\hat{c}$ is its its corresponding quantized bin index.

We apply a similar discretization to joint parameters. The joint origin is quantized into 128 bins per axis, identical to the bounding box coordinates. For joint limits, we discretize rotational angles into 48 bins over $[-2\pi, 2\pi]$ and translational distances into 64 bins over $[-2, 2]$. For the joint axis, we create a discrete 128-entry codebook. Motivated by the observation that most joint axes align with the coordinate axes, our codebook is constructed hierarchically. We first sample points uniformly from the unit circles on the XY, YZ, and XZ planes. Then, additional points are obtained via Farthest Point Sampling (FPS) on a Fibonacci sphere. Each point on the unit sphere corresponds to a rotation axis. This codebook design provides dense coverage for axis-aligned directions while maintaining the flexibility to represent other orientations.

Multi-Task and Multi-Stage SFT To effectively train our model, we draw inspiration from multi-task learning [42], which improves performance by leveraging shared representations with related auxiliary tasks. The articulation prediction naturally decomposes into two core sub-problems: identifying the geometric layout of its parts and inferring the kinematic relationships (joints) between them. We define three supervised fine-tuning (SFT) tasks:

1. **Part Layout Prediction:** Predicts only the part bounding boxes from the point cloud.
2. **Kinematic Prediction:** Predicts the joints, conditioned on both the point cloud and the ground-truth part layout.
3. **End-to-End Articulation Prediction:** Predicts both parts and joints from the input point cloud.

Furthermore, we propose a progressive, two-stage training strategy that effectively integrates multi-task learning. The first stage is to establish a robust geometric foundation for our 3D encoder, and we train model only on the *Task 1* (Part layout prediction). In addition, we initialize the point encoder with weights from P3SAM [33], a model pretrained on large-scale part segmentation. This stage provides our encoder with a strong prior for identifying part-level geometry. In the second stage, we initialize the point encoder and projector with the weights from the first stage. We then perform SFT on model using all three tasks. This refines the model's understanding by training it to focus on kinematic reasoning.

This multi-task, multi-stage training strategy is crucial as we verify in ablation studies (Sec. 4.4). By first grounding the 3D encoder on part-centric feature learning, we establish a high-quality weight initialization that significantly stabilizes the subsequent, more complex multi-task SFT. It effectively decouples geometric understanding from kinematic reasoning during initial training, enabling the model to learn their intricate relationship more robustly.

Table 1. Statistics of our curated dataset for ArtLLM training.

Dataset Source	#Objects	#Categories
PartNet-Mobility [54]	2168	43
PhysX3D [2]	7672	23
Infinite-Mobility [22]	10833	13
Total	**20673**	**43**

3.2. Training Corpus for ArtLLM

To training ArtLLM, we construct a new large-scale dataset by aggregating and refining existing articulation datasets as well as procedurally generated data. Our dataset comprises objects from established benchmarks: PartNet-Mobility [54] and PhysX3D [2]. To enhance scale and diversity, we supplement these with 12k synthetic assets generated via the procedural method of Infinite-Mobility [22].

We then perform data preprocessing on the collected raw articulation assets, including:

- *Filtering*: We remove objects with more than 20 joints and exclude categories containing excessively small parts (e.g., keyboard, remote). Small components, such as buttons, are also filtered according to part volume thresholds.
- *Structure Simplification*: All fixed joints are removed, and their connected links are merged. For screw joints, which are usually represented as a combination of revolute and prismatic joints in URDF files, we merge them into a single screw joint to reduce prediction complexity.
- *Normalization*: All joint parameters are transformed into the global coordinate frame, and normalized to the range $[-0.9, 0.9]$ together with geometry.
- *Normal Correction*: For models in PartNet-Mobility [54] with incorrect surface normals, we apply watertight reconstruction to obtain accurate surface normals.

This pipeline yields a dataset of **20,673 articulated objects across 43 categories**. Table 1 shows its statistics.

3.3. Part-Aware Geometry Synthesis

Our framework generates a structural blueprint—a layout of part bounding boxes—that can be seamlessly integrated with recent part-based generative models. Methods like OmniPart [62] and XPart [59] are particularly well-suited, as they can condition geometry synthesis on bounding box inputs. For this work, we adopt XPart as our geometry generation backbone.

However, when predicted bounding boxes do not perfectly encompass the ground-truth part geometry, generated parts might be truncated or incomplete. To mitigate this, we introduce a robust **bounding box expansion** step that ensures complete geometric coverage. First, we iterate through every point in the input object's point cloud. Any point not contained within any predicted bounding box is assigned to its nearest box based on Euclidean distance.

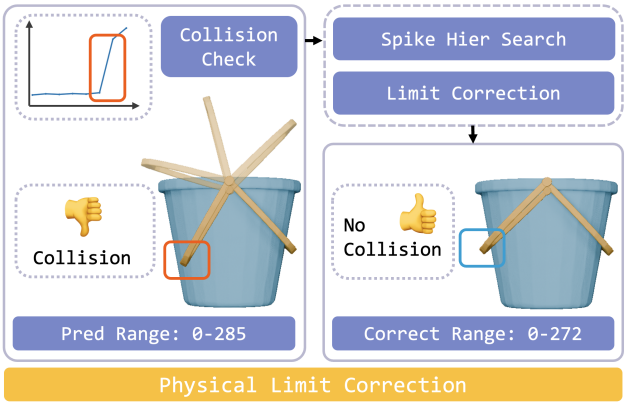

Figure 3. **Physical limit calcualtion.** Illustration for our physical based limit correction process.

Subsequently, each bounding box is expanded just enough to tightly enclose all points newly assigned to it. This simple yet effective mechanism guarantees that the entire object point cloud is covered, preventing geometric artifacts and ensuring the fidelity of the final generated parts.

3.4. Physically-Constrained Joint Limit Correction

When predicting joint limits, the model relies solely on the geometric state at a single timestep, which limits its ability to perceive dynamic motion. This can lead to inter-part collisions during articulation, thereby compromising physical realism. To address this issue, we introduce a post-processing correction step that refines joint limits based on collision detection.

Our method is illustrated in Figure 3. For a given revolute joint, we articulate its child part through its initially predicted range and compute the collision volume against all other static parts at discrete steps. Significant collisions manifest as sharp spikes in the derivative of this collision volume with respect to the joint angle. We first identify a coarse angular window containing a spike and then perform a hierarchical search within this window to pinpoint the precise angle of initial contact. This angle is then set as the refined, collision-free joint limit. A similar procedure is applied to prismatic joints based on translational distance.

By incorporating this physics-based refinement, we effectively prevent self-collisions. This enhances the stability and realism of the asset in physical simulations, making it more reliable for downstream applications like robotic manipulation.

4. Experiments

In this section, we compare our approach with the current state-of-the-art articulation object generation methods to demonstrate its superior performance. We also conduct ablation studies on the designed modules to validate the soundness of our design. Finally, we further verify the prac-

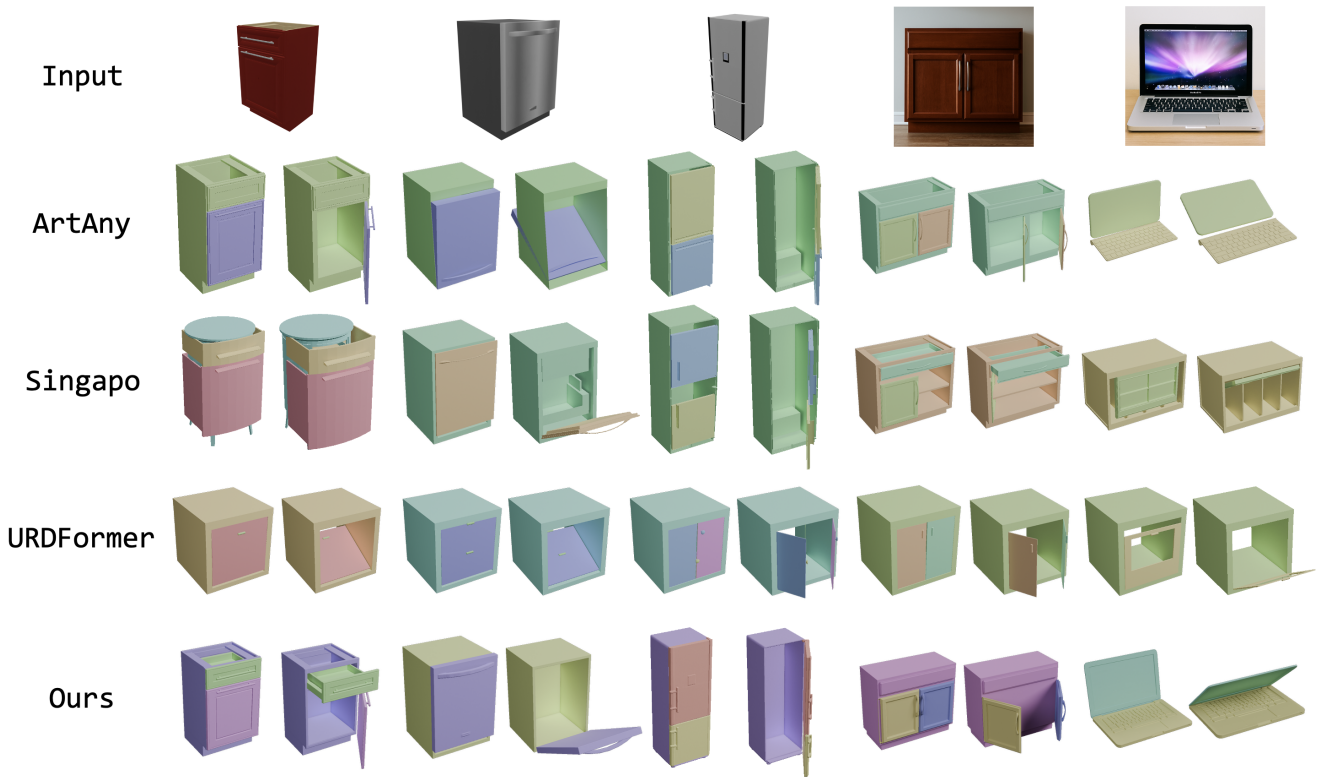

Figure 4. **Qualitative Comparison**. Baseline methods rely on retrieving parts from a fixed asset library, hence often fail to recover accurate geometry and frequently generate incorrect articulations with mismatched joint types or misaligned joint positions. In contrast, our approach produces geometry that closely matches the input and recovers correct, coherent articulations.

ticality of our method in real robotic scenarios.

4.1. Evaluation Dataset

We adopt the PartNet-Mobility [54] dataset for evaluation. Following the data split proposed in SINGAPO [26], we select 7 categories (Storage, Table, Refrigerator, Dishwasher, Oven, Washer, and Microwave), comprising a total of 77 objects, as our test set. These objects are held out during training. In addition, we adopt several real-world images, including objects outside these seven categories, to further assess the model's generalization ability.

4.2. Evaluation Metrics

For the model's predicted results and the ground-truth data, we first perform scale and offset alignment of the objects, along with coordinate conversion (e.g., from z-up to y-up). Next, similar to [26], we apply the Hungarian matching algorithm to match parts based on the distances between their centers. Finally, using the established part correspondences, we can derive the matching relationships for the joints.

To thoroughly evaluate the model's capability, we adopt multiple metrics for part prediction and articulation prediction. For part layout, we compute the mIoU between parts. For articulation prediction, we first compute the joint type accuracy, then follow FreeArt3D [3] to adopt angle and minimum distance betwe axis to evaluate the axis prediction. Finally, we evaluate the accuracy of limit prediction by computing the IoU of the predicted limit ranges. We also evaluate the kinematic hierarchy by constructing a directed graph from links and joints, and computing graph accuracy.

4.3. Comparisons

Comparison Methods We conduct a comprehensive comparison between our method and recent state-of-the-art baselines, including URDFormer [5], Singapo [26], and Articulate-Anything [16]. Since URDFormer is trained only on five categories, we restrict our evaluation to these categories and explicitly denote this setting in the table. Besides, both Singapo and URDFormer treat handles as separate parts and connect them to parent components using fixed joints. For a consistent evaluation protocol, we remove these fixed joints and merge the corresponding child and parent links. For Articulate-Anything, we evaluate the model with the GPT-4o API. To ensure fairness, we also remove the ground-truth object parts from the retrieval library to ensure fairness. As for our method, we adopt Hunyuan3D 3.0 to generate accurate geometry from the input image, and sample surface points as the input of our pipeline.

Table 2. **Quantitative Comparison.** We evaluate all methods on the seven categories of the PartNet-Mobility dataset (splitted by SINGAPO [26]). Metrics are computed per category, and the final score is obtained by an average over all seven categories. * denotes retraining on our dataset. Our method attains high performance, highlighting its strong ability to recover accurate articulated structures.

Method	mIoU	CD	Type Acc	Joint-Axis-Err	Joint-Pivot-Err	Joint-Range-IoU	Graph Acc	Time(s)
Ours	**0.6884**	**0.028**	**0.9084**	**0.1271**	**0.0801**	0.7398	**0.7741**	**19**
ArtAny	0.3381	0.072	0.8457	0.4529	0.5361	**0.8653**	0.6142	522
Singapo	0.4330	0.044	0.7649	0.2445	0.2567	0.5256	0.4564	84
Singapo*	0.4705	0.048	0.9065	0.2463	0.1465	0.6184	0.6851	84
urdformer	0.1225	0.249	0.6068	0.7377	0.6095	0.7032	0.0791	183

Results As shown in Fig 4, our method is able to generate articulated objects whose shapes closely match the input images. In contrast, URDFormer [5] relies on a fixed assumption that objects consist of an external frame and internal components, which leads to results that differ significantly from the ground truth appearance. Moreover, it fails to accurately predict the number of articulated parts and their rotation directions, resulting in very limited articulation structures. Although Articulate Anything [16] and SINGAPO [26] can retrieve nearly identical geometry from their part databases, both methods exhibit clear limitations in joint prediction: Articulate Anything often misidentifies the axis direction, producing incorrect motion types, while SINGAPO frequently predicts inaccurate part scales and axis positions, causing the retrieved parts to be misaligned or poorly sized. On real-world examples, URDFormer still produces highly limited results. Articulate Anything can reconstruct almost identical geometry, but continues to suffer from substantial errors in axis localization. SINGAPO fails to retrieve accurate geometry and often misses essential parts.

Quantitative results in Table 2 show that our method achieves a clear advantage over existing approaches in part layout prediction, joint accuracy, and hierarchical structure modeling. Although Articulate Anything performs well with rule-based limit prediction, it suffers from large axis-position errors. SINGAPO achieves reasonable performance on limit prediction but is fundamentally constrained by its small training category set. After retraining the SINGAPO model, all metrics showed significant improvement, but a noticeable gap remains compared to our method, demonstrating the effectiveness of our model architecture. Furthermore, we report the inference time of each method, demonstrating that our method is also significantly faster, providing an efficient and scalable pipeline for generating articulated assets for large-scale simulation environments.

4.4. Ablation

We further conducted ablation studies to demonstrate the effectiveness of our proposed components. Due to the large computation cost, we train the model only on the PartNet-Mobility [54] training set for 30 epochs for each ablation variant. To more comprehensively evaluate the model's capability, we additionally select 2 objects in each category of PartNet-Mobility based on the split of SINGAPO [26], totaling 144 objects.

Table 3. **Quantitative Ablation.** Ablation experiments evaluating the impact of our key components.

	IoU	TA	JAE	JPE	JRI	GA
Full	**0.473**	**0.898**	**0.141**	0.135	**0.582**	**0.780**
A	0.352	0.823	0.277	0.235	0.575	0.775
B	0.464	0.825	0.289	**0.131**	0.510	0.737
C	0.412	0.894	0.142	0.138	0.577	0.754
D	0.463	0.890	0.143	0.175	0.511	0.780

In Experiment A, directly predicting continuous part layouts and articulation parameters significantly weakens the model's ability to infer coordinate and direction-related attributes, highlighting the difficulty of auto-regressive continuous prediction. In Experiment B, removing the multi-task setup slightly improves axis-direction prediction but degrades all other metrics, indicating that multi-task learning with varied difficulty reinforces part and articulation understanding. In Experiment C, eliminating random scaling and rotation leads to lower part-IoU performance, showing that 3D augmentation enhances spatial perception of part position and scale. In Experiment D, removing the multi-stage training and using P3SAM [33] initialized point encoders lowers both part and joint prediction accuracy, demonstrating that pretraining on part-layout prediction yields superior encoder initialization.

For the physical constraint–based limit correction proposed in the part geometry generation stage, we show qualitative results in Figure 5. In several cases where predicted limits cause self-collision, our correction strategy effectively adjusts the limits to ensure that the articulated parts remain collision-free. This further improves the stability and realism of the generated assets when used for training in simulation environments.

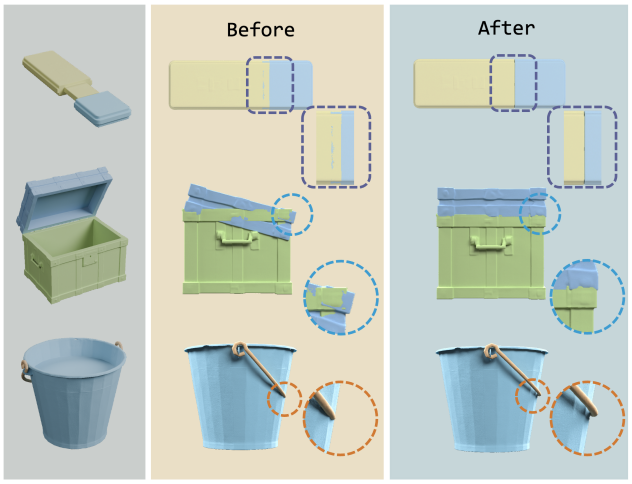

Figure 5. Qualitative result for physical limit correction. Before correction, the predicted joint ranges cause noticeable self-collisions during articulation. After applying our physics-based limit refinement, the articulated parts move smoothly without collision, yielding physically plausible and stable motion.

4.5. Application in Robotic Area

To further demonstrate the value of our proposed method for generating articulated object assets in robotics learning, we conducted a real2sim evaluation. We first teleoperated a Franka Panda robot arm equipped with a Robotiq gripper in the real world to perform several tasks, and recorded the full pose sequence of the execution. Next, we used our ArtLLM pipeline to reconstruct the real objects as articulated assets suitable for simulation, placed them into a simulated scene, and replayed the recorded pose sequence with the simulated robot. By assessing whether the simulated objects exhibit the same articulation behavior as in the real environment, we evaluate how faithfully our generated assets preserve real-world articulation properties.

We tested three tasks: closing a laptop, closing a box, and moving a bucket handle. We employed Hunyuan3D 3.0 to reconstruct accurate 3D object geometry from video frames, and used our pipeline to generate URDF-format articulated assets. In SAPIEN [54], we arranged the generated assets and the robot in appropriate positions and executed the replay. As shown in Fig 6, all three tasks were successfully reproduced in the simulator, demonstrating the high fidelity of our articulated asset generation in terms of structure, joint constraints, and resulting motion behavior.

5. Limitation

Although our method efficiently generates high-quality articulated object assets, it has several limitations. First, though training on a large curated dataset, the diversity of object categories is still limited. As a result, the model generalizes well to common household items but struggles with more complex categories such as vehicles or robots. Future work could incorporate open-vocabulary approaches like Kinematify [49] to expand category coverage and enable broader object modeling. Second, the framework does not jointly model physical properties. Training on large-scale datasets with annotated physical attributes could equip the model with physics-aware prediction capabilities, which we leave for future work.

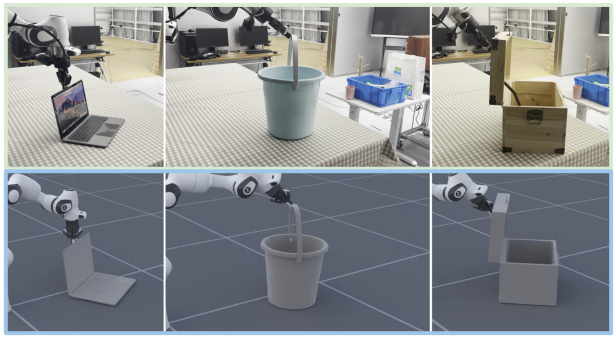

Figure 6. We teleoperate a Franka Panda robot to execute three articulation tasks and record its pose trajectories. Using ArtLLM, we reconstruct the corresponding articulated assets from real scenes and replay the trajectories in simulation. The simulated objects reproduce the real articulation behavior, showing that our generated assets accurately capture real-world kinematics and joint constraints.

6. Conclusion

In this work, we present ArtLLM, an efficient framework capable of rapidly generating articulated objects from various modalities such as a single image or text description. Our approach models part layouts and articulation parameters in an autoregressive, language-based manner, enabling flexible representation of objects with varying numbers of joints and topologies. The proposed data discretization design effectively improves numerical stability in next-token prediction. Furthermore, ArtLLM integrates seamlessly with existing high-quality part generation modules, allowing us to synthesize geometrically detailed and diverse parts, thus overcoming the low-novel geometry limitation common in retrieval-based methods. In addition, our physical constraint-based limit correction mechanism effectively mitigates mesh collisions, producing physically grounded assets that are valuable for robotic simulation and related downstream tasks. Overall, ArtLLM provides a stable and generalizable pipeline for articulated object generation, with strong potential to narrow the real-to-simulation gap, accelerate the creation of high-fidelity digital twins, thus enable scalable robot learning.

7. Acknowledgement

This work was supported by the Shanghai Pujiang Program (24PJA080), the MoE Key Lab of Intelligent Perception and Human-Machine Collaboration (ShanghaiTech University), the Shanghai Frontiers Science Center of Human-centered Artificial Intelligence, and the HPC Platform of ShanghaiTech University. We gratefully acknowledge the invaluable discussion and feedback provided by **Chunshi Wang**, **Junliang Ye**, **Yunhan Yang** from the Tencent Hunyuan3D Team, **Xinyu Lian** from the Shanghai AI Lab, and **Kaixin Yao**, **Zhehao Shen** from ShanghaiTech University.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2, 4

[2] Ziang Cao, Zhaoxi Chen, Liang Pan, and Ziwei Liu. Physx-3d: Physical-grounded 3d asset generation. *arXiv preprint arXiv:2507.12465*, 2025. 2, 5

[3] Chuhao Chen, Isabella Liu, Xinyue Wei, Hao Su, and Minghua Liu. Freeart3d: Training-free articulated object generation using 3d diffusion. *arXiv preprint arXiv:2510.25765*, 2025. 3, 6

[4] Honghua Chen, Yushi Lan, Yongwei Chen, and Xingang Pan. Artilatent: Realistic articulated 3d object generation via structured latents. *arXiv preprint arXiv:2510.21432*, 2025. 3

[5] Zoey Chen, Aaron Walsman, Marius Memmel, Kaichun Mo, Alex Fang, Karthikeya Vemuri, Alan Wu, Dieter Fox, and Abhishek Gupta. Urdformer: A pipeline for constructing articulated simulation environments from real-world images. *arXiv preprint arXiv:2405.11656*, 2024. 3, 6, 7

[6] Lihe Ding, Shaocong Dong, Yaokun Li, Chenjian Gao, Xiao Chen, Rui Han, Yihao Kuang, Hong Zhang, Bo Huang, Zhanpeng Huang, et al. Fullpart: Generating each 3d part at full resolution. *arXiv preprint arXiv:2510.26140*, 2025. 2

[7] Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7081–7091, 2023. 2

[8] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 2

[9] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023. 2

[10] Zehuan Huang, Yuan-Chen Guo, Xingqiao An, Yunhan Yang, Yangguang Li, Zi-Xin Zou, Ding Liang, Xihui Liu, Yan-Pei Cao, and Lu Sheng. Midi: Multi-instance diffusion for single image to 3d scene generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23646–23657, 2025. 2

[11] Team Hunyuan3D, Bowen Zhang, Chunchao Guo, Haolin Liu, Hongyu Yan, Huiwen Shi, Jingwei Huang, Junlin Yu, Kunhong Li, Penghao Wang, et al. Hunyuan3d-omni: A unified framework for controllable generation of 3d assets. *arXiv preprint arXiv:2509.21245*, 2025. 2

[12] Team Hunyuan3D, Bowen Zhang, Chunchao Guo, Dongyuan Guo, Haolin Liu, Hongyu Yan, Huiwen Shi, Jiaao Yu, Jiachen Xu, Jingwei Huang, et al. Hy3d-bench: Generation of 3d assets. *arXiv preprint arXiv:2602.03907*, 2026. 2

[13] Denys Iliash, Hanxiao Jiang, Yiming Zhang, Manolis Savva, and Angel X Chang. S2o: Static to openable enhancement for articulated 3d objects. *arXiv preprint arXiv:2409.18896*, 2024. 2

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[15] Zeqiang Lai, Yunfei Zhao, Haolin Liu, Zibo Zhao, Qingxiang Lin, Huiwen Shi, Xianghui Yang, Mingxin Yang, Shuhui Yang, Yifei Feng, et al. Hunyuan3d 2.5: Towards high-fidelity 3d assets generation with ultimate details. *arXiv preprint arXiv:2506.16504*, 2025. 2, 3

[16] Long Le, Jason Xie, William Liang, Hung-Ju Wang, Yue Yang, Yecheng Jason Ma, Kyle Vedder, Arjun Krishna, Dinesh Jayaraman, and Eric Eaton. Articulate-anything: Automatic modeling of articulated objects via a vision-language foundation model. *arXiv preprint arXiv:2410.13882*, 2024. 3, 6, 7

[17] Jiahui Lei, Congyue Deng, William B Shen, Leonidas J Guibas, and Kostas Daniilidis. Nap: Neural 3d articulated object prior. *Advances in Neural Information Processing Systems*, 36:31878–31894, 2023. 3

[18] Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman3d: High-fidelity mesh generation with 3d native generation and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024. 2

[19] Weiyu Li, Xuanyang Zhang, Zheng Sun, Di Qi, Hao Li, Wei Cheng, Weiwei Cai, Shihao Wu, Jiarui Liu, Zihao Wang, et al. Step1x-3d: Towards high-fidelity and controllable generation of textured 3d assets. *arXiv preprint arXiv:2505.07747*, 2025.

[20] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *arXiv preprint arXiv:2502.06608*, 2025. 2, 3

[21] Zhe Li, Xiang Bai, Jieyu Zhang, Zhuangzhe Wu, Che Xu, Ying Li, Chengkai Hou, and Shanghang Zhang. Urdf-anything: Constructing articulated objects with 3d multimodal language model. *arXiv preprint arXiv:2511.00940*, 2025. 3

[22] Xinyu Lian, Zichao Yu, Ruiming Liang, Yitong Wang, Li Ray Luo, Kaixu Chen, Yuanzhen Zhou, Qihong Tang,

Xudong Xu, Zhaoyang Lyu, et al. Infinite mobility: Scalable high-fidelity synthesis of articulated objects via procedural generation. *arXiv preprint arXiv:2503.13424*, 2025. 3, 5

[23] Yuchen Lin, Chenguo Lin, Panwang Pan, Honglei Yan, Yiqiang Feng, Yadong Mu, and Katerina Fragkiadaki. Partcrafter: Structured 3d mesh generation via compositional latent diffusion transformers. *arXiv preprint arXiv:2506.05573*, 2025. 2

[24] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 2, 4

[25] Jiayi Liu, Ali Mahdavi-Amiri, and Manolis Savva. Paris: Part-level reconstruction and motion analysis for articulated objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 352–363, 2023. 2, 3

[26] Jiayi Liu, Denys Iliash, Angel X Chang, Manolis Savva, and Ali Mahdavi-Amiri. Singapo: Single image controlled generation of articulated parts in objects. *arXiv preprint arXiv:2410.16499*, 2024. 2, 3, 6, 7

[27] Jiayi Liu, Hou In Ivan Tam, Ali Mahdavi-Amiri, and Manolis Savva. Cage: Controllable articulation generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17880–17889, 2024. 2, 3

[28] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10072–10083, 2024. 2

[29] Yu Liu, Baoxiong Jia, Ruijie Lu, Chuyue Gan, Huayu Chen, Junfeng Ni, Song-Chun Zhu, and Siyuan Huang. Videoartgs: Building digital twins of articulated objects from monocular video. *arXiv preprint arXiv:2509.17647*, 2025. 2, 3

[30] Yu Liu, Baoxiong Jia, Ruijie Lu, Junfeng Ni, Song-Chun Zhu, and Siyuan Huang. Artgs: Building interactable replicas of complex articulated objects via gaussian splatting. *arXiv preprint arXiv:2502.19459*, 2025. 2, 3

[31] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9970–9980, 2024. 2

[32] Ruijie Lu, Yu Liu, Jiaxiang Tang, Junfeng Ni, Yuxiang Wang, Diwen Wan, Gang Zeng, Yixin Chen, and Siyuan Huang. Dreamart: Generating interactable articulated objects from a single image. *arXiv preprint arXiv:2507.05763*, 2025. 3

[33] Changfeng Ma, Yang Li, Xinhao Yan, Jiachen Xu, Yunhan Yang, Chunshi Wang, Zibo Zhao, Yanwen Guo, Zhuo Chen, and Chunchao Guo. P3-sam: Native 3d part segmentation. *arXiv preprint arXiv:2509.06784*, 2025. 4, 7

[34] Zhao Mandi, Yijia Weng, Dominik Bauer, and Shuran Song. Real2code: Reconstruct articulated objects via code generation. *arXiv preprint arXiv:2406.08474*, 2024. 3

[35] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling. *arXiv preprint arXiv:2506.07491*, 2025. 2, 4

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[37] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 2

[38] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 2

[39] Weikun Peng, Jun Lv, Cewu Lu, and Manolis Savva. Generalizable articulated object reconstruction from casually captured rgbd videos. *arXiv preprint arXiv:2506.08334*, 2025. 2, 3

[40] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2

[41] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *European Conference on Computer Vision*, pages 214–238. Springer, 2024. 2, 4

[42] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 4

[43] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2

[44] Jiayi Su, Youhe Feng, Zheng Li, Jinhua Song, Yangfan He, Botao Ren, and Botian Xu. Artformer: Controllable generation of diverse 3d articulated objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1894–1904, 2025. 3

[45] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024. 2

[46] Jiaxiang Tang, Ruijie Lu, Zhaoshuo Li, Zekun Hao, Xuan Li, Fangyin Wei, Shuran Song, Gang Zeng, Ming-Yu Liu, and Tsung-Yi Lin. Efficient part-level 3d object generation via dual volume packing. *arXiv preprint arXiv:2506.09980*, 2025. 2

[47] Marcel Torne Villasevil, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. In *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024. 2

[48] Chunshi Wang, Junliang Ye, Yunhan Yang, Yang Li, Zizhuo Lin, Jun Zhu, Zhuo Chen, Yawei Luo, and Chunchao Guo. Part-x-mllm: Part-aware 3d multimodal large language model. *arXiv preprint arXiv:2511.13647*, 2025. 2

[49] Jiawei Wang, Dingyou Wang, Jiaming Hu, Qixuan Zhang, Jingyi Yu, and Lan Xu. Kinematify: Open-vocabulary synthesis of high-dof articulated objects. *arXiv preprint arXiv:2511.01294*, 2025. 8

[50] Penghao Wang, Yiyang He, Xin Lv, Yukai Zhou, Lan Xu, Jingyi Yu, and Jiayuan Gu. Partnext: A next-generation dataset for fine-grained and hierarchical 3d part understanding. *arXiv preprint arXiv:2510.20155*, 2025. 2

[51] Zhengyi Wang, Jonathan Lorraine, Yikai Wang, Hang Su, Jun Zhu, Sanja Fidler, and Xiaohui Zeng. Llama-mesh: Unifying 3d mesh generation with language models. *arXiv preprint arXiv:2411.09595*, 2024. 2

[52] Ruiqi Wu, Xinjie Wang, Liu Liu, Chunle Guo, Jiaxiong Qiu, Chongyi Li, Lichao Huang, Zhizhong Su, and Ming-Ming Cheng. Dipo: Dual-state images controlled articulated object generation powered by diverse data. *arXiv preprint arXiv:2505.20460*, 2025. 3

[53] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4840–4851, 2024. 4

[54] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11097–11107, 2020. 2, 5, 6, 7, 8

[55] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21469–21480, 2025. 2

[56] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer, 2024. 2, 4

[57] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. In *European Conference on Computer Vision*, pages 1–20. Springer, 2024. 2

[58] Hongyu Yan, Kunming Luo, Weiyu Li, Yixun Liang, Shengming Li, Jingwei Huang, Chunchao Guo, and Ping Tan. Posemaster: Generating 3d characters in arbitrary poses from a single image. *arXiv preprint arXiv:2506.21076*, 2025. 2

[59] Xinhao Yan, Jiachen Xu, Yang Li, Changfeng Ma, Yunhan Yang, Chunshi Wang, Zibo Zhao, Zeqiang Lai, Yunfei Zhao, Zhuo Chen, et al. X-part: high fidelity and structure coherent shape decomposition. *arXiv preprint arXiv:2509.08643*, 2025. 2, 5

[60] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 4

[61] Yunhan Yang, Yuan-Chen Guo, Yukun Huang, Zi-Xin Zou, Zhipeng Yu, Yangguang Li, Yan-Pei Cao, and Xihui Liu. Holopart: Generative 3d part amodal segmentation. *arXiv preprint arXiv:2504.07943*, 2025. 2

[62] Yunhan Yang, Yufan Zhou, Yuan-Chen Guo, Zi-Xin Zou, Yukun Huang, Ying-Tian Liu, Hao Xu, Ding Liang, Yan-Pei Cao, and Xihui Liu. Omnipart: Part-aware 3d generation with semantic decoupling and structural cohesion. *arXiv preprint arXiv:2507.06165*, 2025. 2, 5

[63] Kaixin Yao, Longwen Zhang, Xinhao Yan, Yan Zeng, Qixuan Zhang, Lan Xu, Wei Yang, Jiayuan Gu, and Jingyi Yu. Cast: Component-aligned 3d scene reconstruction from an rgb image. *ACM Transactions on Graphics (TOG)*, 44(4): 1–19, 2025. 2

[64] Junliang Ye, Zhengyi Wang, Ruowen Zhao, Shenghao Xie, and Jun Zhu. Shapellm-omni: A native multimodal llm for 3d generation and understanding. *arXiv preprint arXiv:2506.01853*, 2025. 2

[65] Junliang Ye, Shenghao Xie, Ruowen Zhao, Zhengyi Wang, Hongyu Yan, Wenqiang Zu, Lei Ma, and Jun Zhu. Nano3d: A training-free approach for efficient 3d editing without masks. *arXiv preprint arXiv:2510.15019*, 2025. 2

[66] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions On Graphics (TOG)*, 42(4):1–16, 2023. 2

[67] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024. 2

[68] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024. 2

[69] Longwen Zhang, Qixuan Zhang, Haoran Jiang, Yinuo Bai, Wei Yang, Lan Xu, and Jingyi Yu. Bang: Dividing 3d assets via generative exploded dynamics. *ACM Transactions on Graphics (TOG)*, 44(4):1–21, 2025. 2