StructureGS: Structure-aware Gaussian Splatting for Articulated Object Reconstruction

Gahye Lee, Gyoonseo Kim, Wonjong Jang, Jooeun Son, and Seungyong Lee

POSTECH, South Korea
{gahye0509, nevermore, wonjong, jeson, leesy}@postech.ac.kr

Abstract. Reconstructing articulated objects with multiple movable parts is essential for understanding object structure and enabling physical interaction. However, this reconstruction task poses significant challenges due to the entanglement of geometry, appearance, and motion parameters during optimization. Existing methods rely primarily on photometric supervision, which commonly fails to disentangle these interdependent components, resulting in poor part decomposition with blurred boundaries and geometric artifacts. To address this limitation, we introduce *StructureGS*, a reconstruction framework for articulated objects that integrates structure-aware guidance into 3D Gaussian Splatting. Our approach leverages oriented bounding boxes of object parts to enforce two key structural properties: *spatial coherence*, which constrains each part's geometry to remain compact and spatially coherent within its designated region, and *structural connectivity*, which enforces physically plausible contact relationships between adjacent parts. These properties are realized through structure-aware losses that inject explicit structural constraints into the optimization process. Extensive experiments demonstrate that our method achieves state-of-the-art performance in articulated object reconstruction, producing high-quality results with well-defined part geometries.

1 Introduction

Articulated objects with multiple movable parts are omnipresent in human-made environments. Accurate reconstruction of these objects is crucial for embodied AI systems [3, 30, 32] to correctly understand object structure and plan physical interaction. Beyond capturing static geometry, such reconstruction must also recover the underlying kinematic structure, which defines how different parts move relative to each other with constraints on their possible configurations.

Unlike typical object reconstruction [2, 22, 26, 29, 34] that focuses solely on static geometry and appearance, reconstructing objects with a kinematic structure poses significant challenges due to the need to jointly optimize multiple interdependent components: per-part *geometry*, *appearance*, and *motion parameters*. These components are inherently entangled, as the observed appearance depends on the underlying geometry, while accurate geometry recovery relies

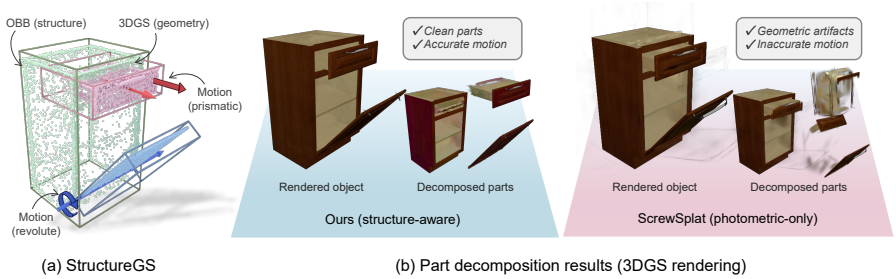


Fig. 1: *StructureGS.* (a) Each part is represented by an explicit OBB (structure) and 3D Gaussians (geometry), with prismatic or revolute motion parameters. (b) Structure-aware optimization yields cleaner part decomposition and more accurate motion estimation than a photometric-only method (ScrewSplat [12]).

on understanding how parts are spatially structured and move relative to each other. This entanglement often hinders accurate estimation of each component, resulting in suboptimal reconstruction quality.

Recent works on articulated object reconstruction represent geometry and appearance using neural radiance fields [8, 9, 18, 24, 35] or Gaussian splatting [7, 11, 12, 17, 20, 21], and parameterize part motions as rigid transformations [1, 6, 15, 19, 27, 37]. These methods rely primarily on photometric reconstruction objectives to update 3D model parameters, and may recover high-quality renderings of the *whole* object. However, without explicit structural guidance, photometric supervision alone often entangles geometry, appearance, and motion parameters during reconstruction, leading to poor *part* decomposition with blurred boundaries and geometric artifacts, as shown in Fig. 1.

To address this problem, we introduce structure-aware guidance for the reconstruction framework. In this work, we use the term *structure* to refer to both the geometry of each part and the contact relationships between adjacent parts, which together characterize how an articulated object maintains a physically plausible configuration. Our structure-aware guidance enforces two properties: *spatial coherence*, which constrains each part's geometry to remain compact and spatially coherent within its designated region, and *structural connectivity*, which enforces physically plausible contact relationship between adjacent parts. To effectively encode structural information, we leverage an oriented bounding box (OBB) for each articulated part. The OBB explicitly defines the oriented extent of each part, providing spatial boundary information for reasoning about its geometric coherence. Furthermore, its convexity facilitates efficient distance computation, allowing for the direct evaluation of connectivity between parts.

Based on the OBB representation, we design two complementary loss functions: *part fitting loss* and *part contact loss*. The part fitting loss encourages spatial coherence by constraining the reconstructed geometry of a part to remain within a tight bounding box. This drives tightly bounded part representations, reducing scattered and spatially incoherent geometry. The part contact

loss promotes structural connectivity by leveraging the Separating Axis Theorem (SAT) [4], which identifies shape intersection by examining projection overlaps across candidate separating axes. By computing inter-OBB distances using SAT, our loss encourages physically plausible contact between adjacent parts.

Building upon the structure-aware losses, we present *StructureGS*, a novel framework for articulated object reconstruction, where each part is represented as a set of 3D Gaussian splats [10]. Importantly, in our framework, the motion of each part, shared by all Gaussians belonging to that part, is parameterized and optimized using its OBB. Given multi-view RGB images capturing an object before and after articulation, our part fitting and part contact losses, together with a photometric loss, provide explicit structural guidance for jointly optimizing per-part geometry, appearance, and motion parameters. This structure-aware design enables high-quality reconstruction with well-defined part boundaries and physically plausible articulated configurations, and provides robustness even under sparse observations, enhancing its practical applicability in real-world capture scenarios.

In summary, our main contributions are:

- We present *StructureGS*, a framework integrating structure-aware guidance into 3D Gaussian Splatting for high-quality reconstruction of articulated objects with clear part boundaries.
- We introduce structure-aware guidance via two OBB-based loss functions, *part fitting loss* and *part contact loss*, that enforce *spatial coherence* and *structural connectivity* of parts during optimization.
- Our method achieves state-of-the-art performance on benchmarks with clear part decomposition and accurate motion estimation. It also provides robust reconstructions with sparse views and generalizes well to real-world objects.

2 Related Work

Articulated object reconstruction Articulated object reconstruction methods can be categorized into *holistic* and *part-wise* approaches. Holistic methods reconstruct the object as a single continuous representation, where part structure is implicitly captured by latent variables or deformation fields. In contrast, part-wise methods decompose the object as a set of explicit parts, each with its own geometry and motion parameters. Table 1 summarizes the key characteristics of representative methods.

Holistic deformation-based methods, such as A-SDF [23], represent the object's shape and articulation within a single signed distance function by learning separate latent codes for shape and pose. Similarly, REACTO [31] employs a skeleton-driven deformation field to explain appearance changes across a video sequence. These approaches can produce plausible whole-object reconstructions under large deformations, but they lack explicit control over part-level geometry and often struggle to recover clean part boundaries.

Alternatively, part-wise approaches model an articulated object as a composition of multiple parts with part-level geometry and rigid motions [8, 9, 12,

Method	Input	Representation	Multi-part	Part-wise	Joint type	Structure-awareness
A-SDF [23]	PC	NeRF				
REACTO [31]	RGB					
PARIS [18]	RGB			✓	✓	
SPLART [16]	RGB	Gaussian Splatting				
ArtGS [20]	RGBD		✓		✓	
ScrewSplat [12]	RGB		✓	✓	✓	
ArticulatedGS [7]	RGB		✓		✓	
Ours	RGB		✓	✓	✓	✓

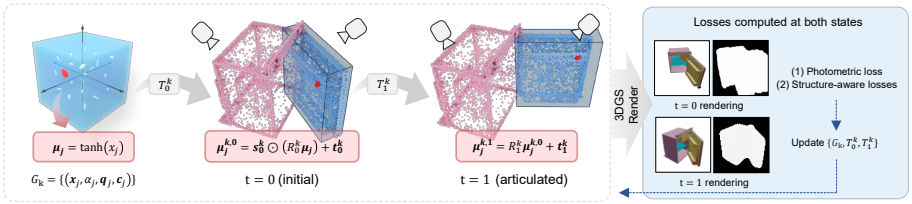
Table 1: Comparison of articulated object reconstruction methods. *Multi-part* indicates support for arbitrary k parts; *Part-wise* denotes per-part representation rather than a holistic one; *Joint type* means no predefined joint type information is required; *Structure-awareness* reflects the use of structural constraints beyond photometric supervision.

18]. PARIS [18] jointly reconstructs part-level radiance fields and motions from multi-view captures. Within the realm of 3D Gaussian Splatting (3DGS), ArticulatedGS [7] and ArtGS [20] discover part structures through deformation-aware clustering and skeleton-guided spectral clustering of Gaussian splats, respectively. Other recent works focus on mobility: SPLART [16] infers part assignments from per-Gaussian mobility estimates, while ScrewSplat [12] groups Gaussians via screw-motion parameters. In this work, we adopt the part-wise paradigm, explicitly representing each movable part with a set of 3D Gaussians and an oriented bounding box to jointly recover its structure and motion.

Photometric supervision and its limitation Photometric supervision has been the primary objective used in previous articulated object reconstruction methods. However, although appearance changes in articulated objects are solely induced by part motions, photometric supervision cannot distinguish whether the appearance changes have been caused by geometry change or motion. As a result, the optimizer often incorrectly adjusts the geometry to account for appearance changes caused by motion, rather than accurately updating kinematic parameters, which leads to blurred part boundaries or geometry leaking across parts.

To mitigate this ambiguity, several approaches incorporate auxiliary priors to better estimate articulation or part geometry. Skeleton-driven methods such as REACTO [31] use predefined joint hierarchies to guide deformation. Clustering-based approaches including ArticulatedGS [7], ArtGS [20], and SPLART [16] form part groups using deformation similarity, spectral analysis, or per-Gaussian mobility. ScrewSplat [12] constrains the space of rigid motions through a screw-motion formulation.

These methods provide valuable supplementary supervision for identifying how an articulated object moves or how its elements should be grouped. However, they do not constrain the structure of parts, such as shapes, boundaries, and physical contacts, which is critical for robust reconstruction of articulated objects. In this paper, we address this issue by introducing structure-aware losses that explicitly promote coherent part shapes and plausible part connectivity.



(a) Canonical Gaussians and part transformations for two states (b) StructureGS optimization

Fig. 2: Overview of StructureGS. (a) Each part P_k is represented by canonical Gaussians G_k, which are transformed to two states: an initial state ($t = 0$) and an articulated state ($t = 1$). (b) For each state, the transformed Gaussians are rendered with the 3DGS renderer, and photometric and structure-aware losses are computed to jointly optimize G_k, T_0^k, and T_1^k.

3 Structure-Aware Gaussian Splatting

Given multi-view RGB images $\{\mathbf{I}_v^t\}$ and corresponding camera parameters $\{\mathbf{C}_v\}$ of an articulated object captured in two distinct configurations, the initial state at $t = 0$ and an articulated state at $t = 1$, our objective is to jointly reconstruct the part-wise 3D geometry, appearance, and motion parameters that faithfully explain the observations across both states. Similar to prior articulated reconstruction works [7, 16, 18, 20], we model the articulated object as K rigid parts $\mathcal{P} = \{P_0, P_1, \ldots, P_{K-1}\}$, where P_0 is a fixed base and each movable part P_k ($k \geq 1$) transforms rigidly between the two states.

3.1 Part Representation

Our part representation is designed to disentangle intrinsic geometry from extrinsic pose. Each part P_k consists of Gaussian primitives in a canonical coordinate frame, coupled with transformation parameters that determine its location, orientation, and scale in the world space.

Canonical Gaussian primitives Each part P_k is represented by a group G_k consisting of N_k Gaussian primitives defined in a normalized canonical space, along with transformation parameters $\{T_0^k, T_1^k\}$, where T_0^k is an affine transformation that maps the canonical space to the world space at $t = 0$, and T_1^k is a rigid transformation that maps the initial state to the articulated state:

$$G_k = \{(\boldsymbol{x}_j, \alpha_j, \boldsymbol{q}_j, \boldsymbol{c}_j)\}_{j=1}^{N_k} \cup \{T_0^k, T_1^k\},$$

where $\boldsymbol{x}_j \in \mathbb{R}^3$ is the canonical position, $\alpha_j \in [0, 1]$ is the opacity, $\boldsymbol{q}_j$ parameterizes the covariance through rotation and scale, and $\boldsymbol{c}_j$ denotes spherical harmonic coefficients for view-dependent appearance. To ensure that Gaussian positions remain within the normalized canonical space, we constrain them to the cube $[-1, 1]^3$ using:

$$\boldsymbol{\mu}_j = \tanh(\boldsymbol{x}_j), \qquad \boldsymbol{\mu}_j \in [-1, 1]^3.$$

This canonical representation decouples the local geometric details of each part from its global pose and spatial extent, facilitating explicit modeling and optimization of part structure through bounding box parameters (Fig. 2(a), right).

Transformation to world space The transformation parameters $T_0^k = (\boldsymbol{s}_0^k, R_0^k, \boldsymbol{t}_0^k)$ define an affine transformation that maps the canonical Gaussians to the world coordinate system at the initial state $t = 0$. For the j-th Gaussian in part k, the world position is computed as:

$$\boldsymbol{\mu}_j^{k,0} = \boldsymbol{s}_0^k \odot (R_0^k \boldsymbol{\mu}_j) + \boldsymbol{t}_0^k,$$

where $\boldsymbol{s}_0^k \in \mathbb{R}^3_{>0}$ represents per-axis scaling, $R_0^k \in SO(3)$ is the rotation, and $\boldsymbol{t}_0^k \in \mathbb{R}^3$ is the translation. The element-wise product $\odot$ with the scale vector effectively parameterizes an oriented bounding box that encodes the part's spatial extent and orientation. The covariance matrix is similarly transformed as:

$$\boldsymbol{\Sigma}_j^{k,0} = R_0^k \,\mathrm{diag}(\boldsymbol{s}_0^k)\, \boldsymbol{\Sigma}_j \,\mathrm{diag}(\boldsymbol{s}_0^k)\, (R_0^k)^\top.$$

Transformation to articulated state For the articulated state $t = 1$, we apply a rigid transformation $T_1^k = (R_1^k, \boldsymbol{t}_1^k)$ to the initial position of part P_k at $t = 0$:

$$\boldsymbol{\mu}_j^{k,1} = R_1^k \boldsymbol{\mu}_j^{k,0} + \boldsymbol{t}_1^k \quad \text{and} \quad \boldsymbol{\Sigma}_j^{k,1} = R_1^k \boldsymbol{\Sigma}_j^{k,0} (R_1^k)^\top,$$

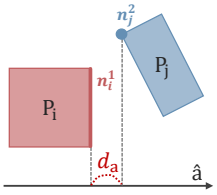
where $R_1^k \in SO(3)$ and $\boldsymbol{t}_1^k \in \mathbb{R}^3$ capture the part's rigid motion between states (Fig. 2(a), middle and left).

3.2 Part Contactness

A fundamental property of articulated objects is that connected parts remain in physical contact. We exploit this structural constraint by computing inter-part distances directly from the bounding box parameters equipped in our part representation.

Bounding box extraction Since each part is defined in a normalized canonical space $[-1, 1]^3$, transformation $T_0^k = (\boldsymbol{s}_0^k, R_0^k, \boldsymbol{t}_0^k)$ naturally induces the oriented bounding box for part P_k at the initial state $t = 0$. The OBB is centered at $\boldsymbol{t}_0^k$ with half-extent $\boldsymbol{s}_0^k$ and orientation defined by R_0^k. Similarly, the OBB at the articulated state $t = 1$ can be obtained by applying T_1^k to the OBB in the initial state. This direct correspondence between part parameters and bounding boxes enables efficient contact evaluation without explicit mesh extraction.

Distance computation via Separating Axis Theorem To compute the distance between two parts P_i and P_j, we employ the Separating Axis Theorem [4], which provides an efficient method for determining the distance between two convex objects by projecting both objects onto a set of candidate axes.



For two OBBs, the complete set of candidate axes consists of 15 directions:

$$\mathcal{A} = \{\boldsymbol{n}_i^1, \boldsymbol{n}_i^2, \boldsymbol{n}_i^3, \boldsymbol{n}_j^1, \boldsymbol{n}_j^2, \boldsymbol{n}_j^3\} \cup \{\boldsymbol{n}_i^p \times \boldsymbol{n}_j^q\}_{p,q=1}^3,$$

comprising the three face normals of each OBB (obtained from the columns of R_0^i and R_0^j), and the nine pairwise cross products of their edge directions. For each axis $\boldsymbol{a} \in \mathcal{A}$, we project both bounding boxes onto the normalized axis $\hat{\boldsymbol{a}} = \boldsymbol{a}/\|\boldsymbol{a}\|$, yielding one-dimensional intervals with centers c_i and c_j and extents r_i and r_j. The signed distance along this axis is:

$$d_{\boldsymbol{a}} = |c_i - c_j| - (r_i + r_j),$$

where $c_i = \boldsymbol{t}_0^i \cdot \hat{\boldsymbol{a}}$ and $r_i = \sum_{d\in\{x,y,z\}} s_{0,d}^i |\boldsymbol{n}_i^d \cdot \hat{\boldsymbol{a}}|$. The distance between two OBBs is then:

$$d(P_i, P_j) = \max_{\mathbf{a}\in\mathcal{A}} d_{\mathbf{a}}, \tag{1}$$

whose positive values indicate separation, zero indicates contact, and negative values indicate overlap. This differentiable distance metric enables efficient enforcement of contact constraints during optimization (see Sec. 4).

4 Optimization with Structural Guidance

For articulated object reconstruction with meaningful part decomposition, in addition to the photometric loss, we introduce two differentiable losses for structural guidance: *part fitting loss* for spatial coherence of parts and *part contact loss* for structural connectivity of parts (see Fig. 2).

4.1 Loss Functions

Photometric image loss Following 3D Gaussian Splatting [10], we employ the standard photometric image loss:

$$\mathcal{L}_{\text{image}} = (1 - \lambda_{\text{ssim}})\mathcal{L}_1 + \lambda_{\text{ssim}}\mathcal{L}_{\text{D-SSIM}}, \tag{2}$$

where $\mathcal{L}_1$ measures the L1 distance between the rendered and ground-truth images, and $\mathcal{L}_{\text{D-SSIM}}$ is the structural dissimilarity term.

Part fitting loss We enforce spatially coherent parts by ensuring each bounding box tightly encloses its part geometry with minimal unused interior volume. To quantify this constraint, we first generate structure images that visualize each part's spatial extent: we uniformly sample points within the canonical cube $[-1, 1]^3$, transform them to the world space using T_0^k for $t = 0$ (or additionally apply T_1^k for $t = 1$), and render them as white pixels on a black background. The rendered structure image is a binary image that represents the spatial occupancy of each part's bounding box, where 1 (white) indicates occupied regions. Based on

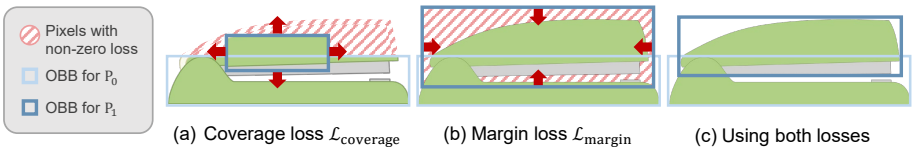


Fig. 3: Illustration of part fitting loss. (a) P_1's OBB is fully inside the part (margin loss = 0), but does not cover the entire part, incurring coverage loss. (b) P_1's OBB fully covers the part (coverage loss = 0), but includes regions outside the part, incurring margin loss. (c) Optimizing both losses yields a tight OBB.

the structure images, we formulate the part fitting loss with two complementary objectives:

$$\mathcal{L}_{\text{fit}} = \lambda_{\text{coverage}}\mathcal{L}_{\text{coverage}} + \lambda_{\text{margin}}\mathcal{L}_{\text{margin}}.$$

The coverage loss $\mathcal{L}_{\text{coverage}}$ ensures that the union of all part structure images fully covers the object region in the rendered image:

$$\mathcal{L}_{\text{coverage}} = \|(1 - \mathbf{I}^{\text{all}}_{\text{struct}}) \odot \mathbf{M}\|_1,$$

where $\mathbf{I}^{\text{all}}_{\text{struct}}$ is the union of structure images rendered from OBBs of all parts, $\mathbf{M}$ is the object mask in the rendered image, and $\odot$ denotes element-wise multiplication. This term penalizes regions of the object not covered by any OBB.

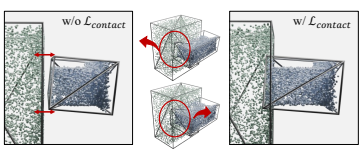
Minimizing $\mathcal{L}_{\text{coverage}}$ alone may cause OBBs to expand excessively beyond object boundaries. We therefore introduce a margin loss that penalizes regions of each part's structure image that lie outside the object mask:

$$\mathcal{L}_{\text{margin}} = \frac{1}{K}\sum_{k=0}^{K-1} \|\mathbf{I}^{k}_{\text{struct}} \odot (1 - \mathbf{M})\|_1, \tag{3}$$

where $\mathbf{I}^{k}_{\text{struct}}$ denotes the structure image of part P_k. This term encourages tight fitting of OBBs by penalizing the unused void space within each bounding box. For concave parts, reaching the tightest feasible bound does not necessarily drive the margin loss to zero, as unavoidable empty regions may remain within their corresponding OBBs. Fig. 3 provides a visual illustration of the part fitting loss.

Part contact loss To enforce connectivity, we penalize separation between adjacent parts using the distance $d(\cdot,\cdot)$ computed by Eq. 1:

$$\mathcal{L}_{\text{contact}} = \lambda_{\text{contact}} \sum_{(i,j)\in\mathcal{E}} \max(0, d(P_i, P_j))^2, \tag{4}$$



where $\mathcal{E}$ denotes the set of adjacent pairs, formed between the static base part P_0 and each movable part P_k ($k \geq 1$). Since OBBs encode part structures rather than exact occupied volumes, valid articulated layouts can involve OBB overlap, as in a drawer nested inside a cabinet body. Accordingly, the max operator penalizes only separation, not overlap.

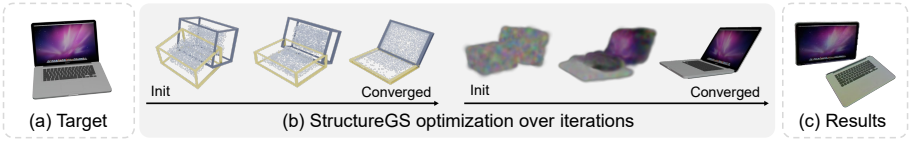


Fig. 4: Structure-aware optimization over iterations. Part-wise 3D Gaussian splats constrained by OBBs progressively align with their corresponding object parts.

Extent regularization To encourage the OBB scale components to adapt to the underlying part geometry, we introduce an axis-weighted scale regularization:

$$\mathcal{L}_{\text{ext}} = -\lambda_{\text{ext}} \sum_{k=0}^{K-1} \sum_{i=1}^{3} w_k^{(i)} \log(s_{0,i}^k), \tag{5}$$

where $s_{0,i}^k$ is the i-th scale component of the k-th OBB, and $\mathbf{w}_k = \text{softmax}(\boldsymbol{\beta}_k)$ are per-axis weights with learnable parameters $\boldsymbol{\beta}_k \in \mathbb{R}^3$. The softmax weights act as a differentiable selector over the OBB scale components, emphasizing scale directions whose expansion does not increase the overall objective. Consequently, the OBB expands primarily along under-covered directions rather than scaling uniformly.

Combining structure-aware and photometric image losses across temporal states $t \in \{0, 1\}$, along with extent regularization, our final optimization objective is:

$$\mathcal{L}_{\text{total}} = \sum_{t \in \{0,1\}} \left(\mathcal{L}_{\text{image}}^t + \mathcal{L}_{\text{fit}}^t + \mathcal{L}_{\text{contact}}^t \right) + \mathcal{L}_{\text{ext}}. \tag{6}$$

4.2 Optimization Process

We initialize Gaussians by randomly sampling 10,000 primitives per part within the canonical space $[-1, 1]^3$. For OBB parameters, we initialize the boxes using coarse point clouds estimated for the two articulation states with VGGT [33]. We use nearest-neighbor distances between the two states to separate static and moving regions, cluster the moving points into dynamic parts with DBSCAN [5], and fit an OBB to each resulting part to initialize T_0^k. The articulation transforms T_1^k are initialized as identity. The initialization provides only a coarse structural cue and does not require accurate part segmentation. The proposed structure-aware losses subsequently optimize both part geometry and motion, making the method tolerant to imperfect initial bounding boxes. We provide detailed initialization procedures and robustness analysis in the supplementary material.

Optimization proceeds for 10,000 iterations using Adam optimizer [13] with progressive scheduling. Structure-aware losses ($\mathcal{L}_{\text{fit}}$, $\mathcal{L}_{\text{contact}}$, $\mathcal{L}_{\text{ext}}$) are active from the start to establish part decomposition. We introduce $\mathcal{L}_{\text{image}}$ at iteration 400 and disable $\mathcal{L}_{\text{contact}}$ and $\mathcal{L}_{\text{ext}}$ at iteration 1500, once the part structures are

sufficiently established. During optimization, we apply standard 3DGS densification and pruning. More details on optimization schedules, hyperparameters, and architectural details are provided in the supplementary material.

In essence, we reparameterize each part as a canonical Gaussian set in the unit cube centered at the origin. Before rendering, the canonical Gaussians are first transformed to the initial state by the OBB transform T_0^k, and then mapped to the articulated state by the articulation transform T_1^k. This differentiable parameterization allows photometric and structure-aware losses to jointly update the Gaussians, OBB parameters, and articulation parameters, guiding the reconstruction toward coherent articulated parts.

Fig. 4 visualizes the optimization process. The left columns show part-wise bounding boxes and their associated Gaussian primitive sets, while the right columns show rendered Gaussians. As optimization progresses, the structure-aware losses guide the Gaussian sets toward separated parts with cleaner boundaries.

5 Experiments

5.1 Setup

Benchmark datasets We evaluate our method on two primary articulated object benchmarks. The PARIS dataset [18] comprises 10 synthetic and 2 real-world objects, each consisting of a static base and a single movable part. The DTA-Multi dataset [36] contains 2 synthetic multi-part objects, each consisting of a static base and multiple movable parts. For both datasets, objects are captured in two distinct articulation states, each with 100 multi-view RGB images at an 800×800 resolution from cameras uniformly distributed over a hemisphere. To assess robustness under limited observations, we additionally introduce a sparse-view setting using only 10 uniformly sampled views per state from PARIS. Additional scalability experiments on more complex articulated objects with 4–7 parts are provided in the supplementary material.

Real-world data To demonstrate practical usefulness of our method, we introduce a custom real-world dataset of four articulated objects (kettle, stand, carrier, and hole punch) that exhibit challenging features: textureless surfaces, non-box-like geometries, and subtle articulations. As a practical setup, we utilize only 10 images per state captured via a consumer smartphone. Camera poses and object masks are obtained using off-the-shelf tools, COLMAP [28] and SAM [14], respectively, so our evaluation reflects the estimation noise encountered in practical settings.

Baselines We compare our method against three representative baselines using their official implementations: PARIS [18], which explicitly decomposes part-wise neural radiance fields; ArticulatedGS [7], which discovers part structures via deformation-aware clustering on canonical 3D Gaussians; and ScrewSplat [12], which models continuous screw motions using soft Gaussian assignments. For a

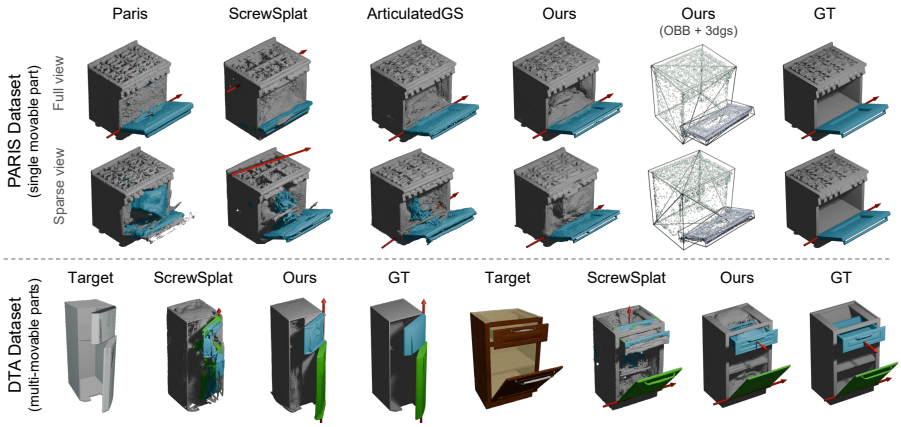


Fig. 5: Qualitative comparison on PARIS and DTA datasets under full- and sparse-view settings. Reconstructed geometry, fitted OBBs with per-part 3DGS, and estimated joint axes (red) are shown; gray denotes static parts, while blue and green indicate movable parts. Additional results are provided in the supplementary material.

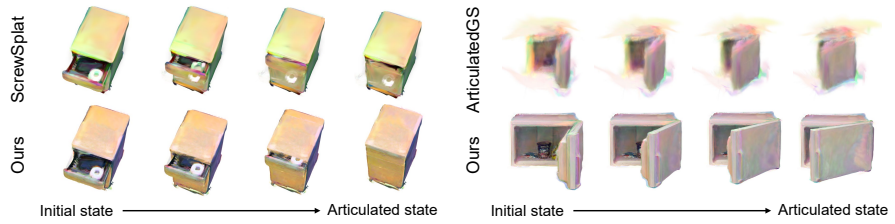


Fig. 6: Articulation animation on PARIS real data from initial to articulated states.

fair comparison, we adapt ScrewSplat to the two-state setting according to its original protocol, evaluating multiple parsimony weights per scene to report the best achievable result.

Evaluation metrics We comprehensively evaluate our method across three aspects: geometry, kinematics, and rendering. Following [7,18], geometric accuracy is measured via the Chamfer-L_1 distance (CD) using $10,000$ sampled points, reported for the whole object (CD-w), static base (CD-s), and movable part (CD-m), all scaled by 10^3. For kinematics, we evaluate joint axis accuracy using angular error (in degrees) for all joints and position error specifically for revolute joints. Motion magnitude is assessed via geodesic and translation errors for revolute and prismatic motions, respectively. Finally, novel view synthesis is evaluated using PSNR, SSIM, and LPIPS.

Table 2: Quantitative results on the PARIS dataset under full (100 views) and sparse (10 views) settings. Scores are averaged over 8 revolute and 2 prismatic scenes for each joint type; per-scene results are provided in the supplementary material. Angle error is in degrees (°); CD is scaled by ×1000.

Group	Metric	Type	PARIS full views (100)				PARIS sparse views (10)			
			PARIS	ScrewSplat	ArticulatedGS	Ours	PARIS	ScrewSplat	ArticulatedGS	Ours
Motion	Ang Err	All	5.635	7.787	0.091	**0.088**	32.840	11.537	8.553	**0.701**
	Pos Err	Revolute	0.385	0.139	0.018	**0.001**	0.162	0.181	0.126	**0.007**
	Geo Dist	Revolute	85.52	32.20	0.761	**0.066**	71.84	39.96	12.89	**1.305**
		Prismatic	1.062	0.200	0.200	**0.001**	0.14	0.401	0.270	**0.042**
Geometry	CD-s	All	5.528	38.948	2.119	**1.558**	157.46	36.521	2.044	**1.957**
	CD-m	All	67.935	32.202	1.585	**1.400**	164.93	133.50	62.700	**1.982**
	CD-w	All	5.895	13.935	1.908	**0.919**	84.632	31.892	19.847	**1.320**
Photometry	PSNR ↑	All	29.324	31.377	37.336	**43.39**	22.363	28.135	23.320	**33.264**
	SSIM ↑	All	0.948	0.965	0.985	**0.995**	0.878	0.952	0.910	**0.969**
	LPIPS ↓	All	0.097	0.046	0.039	**0.014**	0.160	0.064	0.095	**0.052**

5.2 Comparisons

Our method achieves state-of-the-art performance in both dense- and sparse-view settings, with consistent improvements across geometry, motion, and photometric metrics. Notably, baselines exhibit inconsistency in metrics: a low whole-object Chamfer Distance often obscures severe part-level geometric errors (Table 2) and incorrect motion estimates (Fig. 5). This suggests that photometric-only optimization can converge to physically implausible states that satisfy visual observations but misrepresent true articulation. Moreover, this ambiguity between geometry and motion is amplified in under-constrained sparse-view settings.

Consequently, during animation, baseline reconstructions suffer from ghosting artifacts and blurry boundaries caused by flawed part decomposition (Fig. 6). In contrast, our structure-aware formulation jointly regularizes part decomposition and motion, ensuring physically valid articulation and robust reconstructions regardless of view density.

We further validate our method's scalability on the DTA dataset, which features more complex, multi-part kinematic structures. As shown in Table 3 and Fig. 5, our approach maintains high-quality reconstructions and lower errors across both scenes. While ScrewSplat achieves a marginally lower error for one joint in the *Storage* scene, it suffers from significant geometric and kinematic inaccuracies elsewhere. Specifically, it fails to decouple the two movable parts, merging them into a single component and propagating errors that degrade the overall reconstruction. Further comparisons with ScrewSplat on ArtGS-Multi dataset in the supplementary material show consistent gains on more complex 4–7 part articulated objects.

Finally, in the full-view setting, our method converges in approximately 13 minutes per scene, achieving comparable efficiency to ArticulatedGS (13 min) and ScrewSplat (15 min), while being substantially faster than PARIS (40 min).

Table 3: Quantitative results on the DTA dataset, reported per part. Superscripts † and ‡ denote part 0 and part 1, respectively. F denotes failure; '-' indicates not applicable due to prismatic geometry.

Object	Method	Ang Err†	Ang Err‡	Pos Err†	Pos Err‡	Geo Dist†	Geo Dist‡	CD-s	CD-m†	CD-m‡	CD-w
Fridge	ScrewSplat	15.12	0.347	0.024	0.007	9.640	15.12	25.43	321.01	34.93	8.369
	Ours	**0.445**	**0.119**	**0.003**	**0.002**	**1.301**	**0.217**	**5.888**	**0.736**	**1.078**	**4.699**
Storage	ScrewSplat	**0.052**	89.86	**0.003**	-	60.30	-	136.1	180.9	299.5	38.15
	Ours	2.165	**0.542**	0.017	-	**0.292**	-	**3.336**	**0.553**	**4.184**	**3.238**

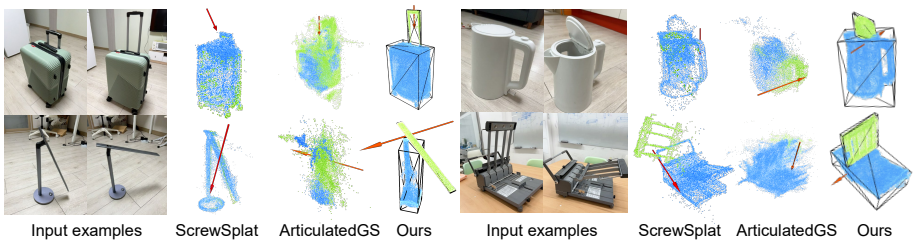


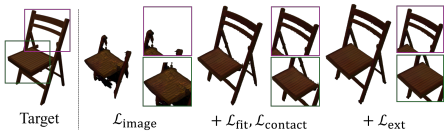
Fig. 7: Reconstruction of real-world articulated objects using 10 views. Blue/green Gaussian splats denote static/dynamic parts, respectively. Joint axes and estimated OBBs of our method are shown as red arrows and black wireframes, respectively.

5.3 Real-world evaluation

We further evaluate our method's generalization under sparse and noisy real-world conditions using only 10 views per state. As shown in Fig. 7, photometric-only baselines struggle to recover geometry and articulation, producing scattered Gaussians and inaccurate joint estimates. In contrast, our method reconstructs coherent part-level 3DGS and accurate kinematics by leveraging tightly fitted OBBs that properly bound each semantic part. These results demonstrate that our explicit structural guidance effectively generalizes to diverse real-world articulated objects. Additional animations are available in the supplementary material.

5.4 Discussion

Loss function ablation To validate our structural losses, we conduct an ablation study on the PARIS dataset (Fig. 8). Optimizing solely with the photometric loss ($\mathcal{L}_{\text{image}}$) yields severe geometry and kinematic errors due to unresolved shape-motion ambiguity. Introducing the fitting and contact losses ($\mathcal{L}_{\text{fit}}$, $\mathcal{L}_{\text{contact}}$) reduces both geometric distortion and joint inaccuracies, enabling coherent part decomposition and stable kinematics. Finally, adding the extent regularization ($\mathcal{L}_{\text{ext}}$) rectifies under-estimated bounding volumes, stabilizing part extents and further decreasing geometric error. Notably, the large CD-m reduction after



Loss	Ang	Pos	GD (r)	GD (p)	CD-s	CD-m	CD-w
$\mathcal{L}_{\text{image}}$	5.60	0.11	7.65	0.01	6.08	31.8	3.90
$+\mathcal{L}_{\text{fit, con}}$	1.71	0.00	0.06	0.00	1.89	31.4	0.98
$+\mathcal{L}_{\text{ext}}$	0.08	0.00	0.06	0.00	1.56	1.40	0.91

Fig. 8: Ablation on structural losses. Left: Visual comparison of the effect of each loss term on reconstruction quality. Right: Quantitative results for each loss configuration.

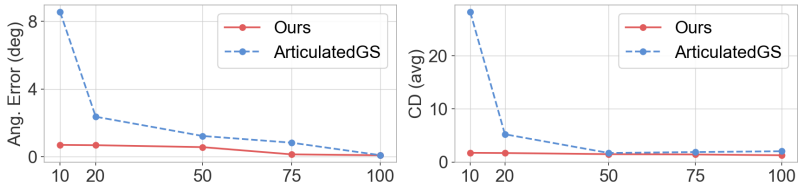
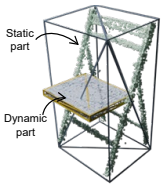


Fig. 9: Robustness to sparse observations. Angular error and Chamfer distance on the PARIS benchmark under varying numbers of training views. Our method remains stable as views become sparse, while ArticulatedGS [7] degrades sharply.

adding $\mathcal{L}_{\text{ext}}$ mainly comes from the Blade scene. There, the regularizer helps the OBB recover the full movable-part extent, rather than fitting only to the exposed tip where silhouette changes are most dominant.

Robustness to view sparsity We evaluate the robustness of our model to view sparsity on the PARIS benchmark dataset [18]. We optimize models using 10 to 100 uniformly subsampled training views and compare geometric and motion errors against ArticulatedGS [7]. Fig. 9 shows that, as observations become sparser, ArticulatedGS exhibits a sharp increase in both angular error and Chamfer distance, indicating severe shape-motion ambiguity under limited photometric supervision. In contrast, our method maintains stable performance across view counts, demonstrating that OBB-based structural guidance effectively constrains the solution space when visual evidence is limited.

Non-OBB-shaped objects Although our structure representation uses OBBs, StructureGS is not constrained to box-like shapes. By jointly optimizing each OBB with its enclosed 3D Gaussians, the OBB provides a compact proxy for the coarse part structure, while the Gaussians recover fine-grained geometry. For instance, the static part of the *foldchair* scene (PARIS benchmark) consists of extremely thin structures. To quantify this non-cuboidal nature, we define an empty space ratio $r = 1 - V_{\text{mesh}}/V_{\text{OBB}}$. For the *foldchair*, we obtain $r = 0.97$, meaning the bounding volume contains 97% empty space. As illustrated in our qualitative results, even in such extreme cases, the OBB-constrained Gaussians successfully recover intricate geometric details.



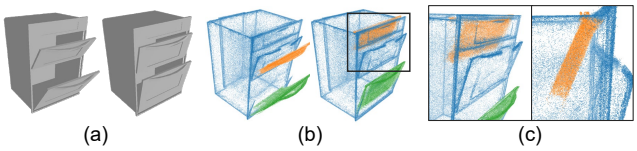


Fig. 10: Failure case. (a) Target object in the initial and articulated states. (b) Reconstructed 3DGS in the corresponding states, with parts color-coded. (c) Zoomed-in views showing part collapse in the articulated state.

6 Conclusion

In this work, we presented StructureGS, a framework for articulated object reconstruction that achieves clean part decomposition through structure-aware guidance. By representing parts with oriented bounding boxes and enforcing spatial coherence and structural connectivity constraints, our method produces well-defined part boundaries and geometrically coherent structures. Extensive experiments demonstrate that StructureGS outperforms existing methods in both geometric quality and articulation accuracy.

Limitation Although our OBB-based formulation is more robust than photometric-only optimization, it can still fail when articulated motion induces only weak visual changes. For example, in the oven case in Fig. 10, the upper moving part provides limited silhouette and appearance cues, causing its geometry to be absorbed by the nearby static region. As a result, the dynamic part can collapse into the static body, producing implausible geometry and motion. Incorporating higher-level semantic features such as DINO [25] could further improve robustness in such cases.

Acknowledgements

We thank anonymous reviewers for their valuable feedback. This work was supported by NRF grants (RS-2026-25485382, RS-2025-02216257), IITP grant (RS-2022-II220290), KOCCA grant (RS-2026-25508598) funded by the Korean government (MSIT & MCST).

References

1. Abbatematteo, B., Tellex, S., Konidaris, G.: Learning to generalize kinematic models to novel objects. In: Proceedings of the 3rd Conference on Robot Learning (2019)
2. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)

3. Deitke, M., VanderBilt, E., Herrasti, A., Weihs, L., Ehsani, K., Salvador, J., Han, W., Kolve, E., Kembhavi, A., Mottaghi, R.: Procthor: Large-scale embodied ai using procedural generation. Advances in Neural Information Processing Systems **35**, 5982–5994 (2022)
4. Ericson, C.: Real-time collision detection. Crc Press (2004)
5. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: Density-based spatial clustering of applications with noise. In: Int. Conf. knowledge discovery and data mining. vol. 240 (1996)
6. Geng, H., Xu, H., Zhao, C., Xu, C., Yi, L., Huang, S., Wang, H.: 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. pp. 7081–7091 (2023)
7. 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: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 27144–27153 (2025)
8. Heppert, N., Irshad, M.Z., Zakharov, S., Liu, K., Ambrus, R.A., Bohg, J., Valada, A., Kollar, T.: Carto: Category and joint agnostic reconstruction of articulated objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21201–21210 (2023)
9. Jiang, Z., Hsu, C.C., Zhu, Y.: Ditto: Building digital twins of articulated objects from interaction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5616–5626 (2022)
10. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
11. Kerr, J., Kim, C.M., Wu, M., Yi, B., Wang, Q., Goldberg, K., Kanazawa, A.: Robot see robot do: Imitating articulated object manipulation with monocular 4d reconstruction. arXiv preprint arXiv:2409.18121 (2024)
12. Kim, S., Ha, J., Kim, Y.H., Lee, Y., Park, F.C.: Screwsplat: An end-to-end method for articulated object recognition. arXiv preprint arXiv:2508.02146 (2025)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017), `https://arxiv.org/abs/1412.6980`
14. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything (2023), `https://arxiv.org/abs/2304.02643`
15. Li, X., Wang, H., Yi, L., Guibas, L.J., Abbott, A.L., Song, S.: Category-level articulated object pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3706–3715 (2020)
16. Lin, S., Fang, J., Irshad, M.Z., Guizilini, V.C., Ambrus, R.A., Shakhnarovich, G., Walter, M.R.: Splart: Articulation estimation and part-level reconstruction with 3d gaussian splatting. arXiv preprint arXiv:2506.03594 (2025)
17. Lin, Y., Dai, Z., Zhu, S., Yao, Y.: Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21136–21145 (2024)
18. Liu, J., Mahdavi-Amiri, A., Savva, M.: Paris: Part-level reconstruction and motion analysis for articulated objects. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 352–363 (2023)
19. Liu, X., Zhang, J., Hu, R., Huang, H., Wang, H., Yi, L.: Self-supervised category-level articulated object pose estimation with part-level se (3) equivariance. arXiv preprint arXiv:2302.14268 (2023)

20. Liu, Y., Jia, B., Lu, R., Ni, J., Zhu, S.C., Huang, S.: Artgs: Building interactable replicas of complex articulated objects via gaussian splatting. arXiv preprint arXiv:2502.19459 (2025)
21. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In: 2024 International Conference on 3D Vision (3DV). pp. 800–809. IEEE (2024)
22. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4460–4470 (2019)
23. Mu, J., Qiu, W., Kortylewski, A., Yuille, A., Vasconcelos, N., Wang, X.: A-sdf: Learning disentangled signed distance functions for articulated shape representation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13001–13011 (2021)
24. Noguchi, A., Iqbal, U., Tremblay, J., Harada, T., Gallo, O.: Watch it move: Unsupervised discovery of 3d joints for re-posing of articulated objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3677–3687 (2022)
25. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: Dinov2: Learning robust visual features without supervision (2024), https://arxiv.org/abs/2304.07193
26. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 165–174 (2019)
27. Qiu, X., Yang, J., Wang, Y., Chen, Z., Wang, Y., Wang, T.H., Xian, Z., Gan, C.: Articulate anymesh: Open-vocabulary 3d articulated objects modeling. arXiv preprint arXiv:2502.02590 (2025)
28. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
29. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06). vol. 1, pp. 519–528. IEEE (2006)
30. Shridhar, M., Thomason, J., Gordon, D., Bisk, Y., Han, W., Mottaghi, R., Zettlemoyer, L., Fox, D.: Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10740–10749 (2020)
31. Song, C., Wei, J., Foo, C.S., Lin, G., Liu, F.: Reacto: Reconstructing articulated objects from a single video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5384–5395 (2024)
32. Szot, A., Clegg, A., Undersander, E., Wijmans, E., Zhao, Y., Turner, J., Maestre, N., Mukadam, M., Chaplot, D.S., Maksymets, O., et al.: Habitat 2.0: Training home assistants to rearrange their habitat. Advances in neural information processing systems **34**, 251–266 (2021)
33. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)

34. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: Generating 3d mesh models from single rgb images. In: Proceedings of the European conference on computer vision (ECCV). pp. 52–67 (2018)
35. Wei, F., Chabra, R., Ma, L., Lassner, C., Zollhöfer, M., Rusinkiewicz, S., Sweeney, C., Newcombe, R., Slavcheva, M.: Self-supervised neural articulated shape and appearance models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15816–15826 (2022)
36. Weng, Y., Wen, B., Tremblay, J., Blukis, V., Fox, D., Guibas, L., Birchfield, S.: Neural implicit representation for building digital twins of unknown articulated objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3141–3150 (2024)
37. Xu, W., Wang, J., Yin, K., Zhou, K., Van De Panne, M., Chen, F., Guo, B.: Joint-aware manipulation of deformable models. ACM Transactions on Graphics (TOG) **28**(3), 1–9 (2009)